A Framework for realistic modeling and display of object surface appearance

Benjamin Darling

Follow this and additional works at: https://scholarworks.rit.edu/theses

Recommended Citation

This Dissertation is brought to you for free and open access by RIT Scholar Works. It has been accepted for inclusion in Theses by an authorized administrator of RIT Scholar Works. For more information, please contact ritescholarworks@rit.edu.
A Framework for Realistic Modeling and Display of Object Surface Appearance

by

Benjamin A. Darling

B.S. University of Virginia, 2002
M.S. University of Virginia, 2002

A Dissertation Submitted in Partial Fulfillment of the Requirements for the Degree of Doctor of Science in Color Science

Department of Color Science
College of Science
Rochester Institute of Technology
Rochester, NY
July 17, 2013

Signature of Author

Accepted by

Dr. Mark Fairchild, Graduate Program Director
The Ph.D. Degree Dissertation of Benjamin A. Darling has been examined and approved by the dissertation committee as satisfactory for the dissertation required for the Ph.D. degree in Color Science.

Dr. Matthew Marshall, Chair

Dr. James Ferwerda, Advisor

Dr. Roy Berns

Dr. Jinwei Gu

Date
ABSTRACT

With advances in screen and video hardware technology, the type of content presented on
computers has progressed from text and simple shapes to high-resolution photographs, photorealistic
renderings, and high-definition video. At the same time, there have been significant advances in the area
of content capture, with the development of devices and methods for creating rich digital representations
of real-world objects. Unlike photo or video capture, which provide a fixed record of the light in a scene,
these new technologies provide information on the underlying properties of the objects, allowing their
appearance to be simulated for novel lighting and viewing conditions. These capabilities provide an
opportunity to continue the computer display progression, from high-fidelity image presentations to
digital surrogates that recreate the experience of directly viewing objects in the real world.

In this dissertation, a framework was developed for representing objects with complex color,
gloss, and texture properties and displaying them onscreen to appear as if they are part of the real-world
environment. At its core, there is a conceptual shift from a traditional image-based display workflow to an
object-based one. Instead of presenting the stored patterns of light from a scene, the objective is to
reproduce the appearance attributes of a stored object by simulating its dynamic patterns of light for the
real viewing and lighting geometry. This is accomplished using a computational approach where the
physical light sources are modeled and the observer and display screen are actively tracked. Surface
colors are calculated for the real spectral composition of the illumination with a custom multispectral
rendering pipeline.

In a set of experiments, the accuracy of color and gloss reproduction was evaluated by measuring
the screen directly with a spectroradiometer. Gloss reproduction was assessed by comparing gonio
measurements of the screen output to measurements of the real samples in the same measurement
configuration. A chromatic adaptation experiment was performed to evaluate color appearance in the
framework and explore the factors that contribute to differences when viewing self-luminous displays as
opposed to reflective objects. A set of sample applications was developed to demonstrate the potential
utility of the object display technology for digital proofing, psychophysical testing, and artwork display.
ACKNOWLEDGEMENTS

First, I would like to thank my advisor, Dr. James Ferwerda, for his guidance, advice, and support over the years, for inspiring my interest in computer graphics, and for giving me the opportunity to work with him in exploring how display systems can be made to act like real things.

I would like to express my gratitude to Dr. Matthew Marshall, Dr. Roy Berns, and Dr. Jinwei Gu for serving on the dissertation committee, for their advice, and for their comments and recommendations on the dissertation document. I would also like to express my gratitude to Dr. Jeff Pelz for his participation in the dissertation defense.

I would like to thank Dr. Mark Fairchild for sharing his knowledge of color appearance, answering my questions, and providing advice. I would like to thank Dr. David Wyble for sharing his measurement expertise and for his assistance in constructing the measurement apparatuses used in the dissertation experiments. I would like to express my gratitude to Lawrence Taplin for all his advice and assistance in using a variety of technologies. I would like to thank Dr. Tongbo Chen and Dr. Roy Berns for providing painting surface models to present on the object display and for their collaboration in the work on multispectral illumination for rendering. I would like to thank Dr. Paul Debevec for providing the surface model of the illuminated manuscript.

I would like to thank MCSL staff member Valerie Hemink for providing me with so much help and all she does at MCSL to keep everything going smoothly for the color science students. I would like to express my gratitude to academic coordinator Susan Chan for all her help and advice over the years.

I would like to thank the sponsors who provided support for this work. This research was supported by NSF grants 0811680 and 1064412, a grant from the Andrew W. Mellon Foundation, an RIT CIS-Kodak Innovative Research Grant, and a Student Research & Creativity Grant from the RIT Office of Graduate Studies.

To all my fellow students over the years, and with special thanks to Erin, Jonathan, Stefan, Rod, Dan, and Marissa, thank you for all your help and friendship. Finally to my family, my sister Korey and my parents Ron and Diane Darling, thank you so much for all the encouragement and support.
# TABLE OF CONTENTS

1 INTRODUCTION......................................................................................................................... 1
  1.1 MOTIVATION.......................................................................................................................... 2
  1.2 OVERVIEW............................................................................................................................. 5

2 REVIEW OF LITERATURE ........................................................................................................ 14
  2.1 OVERVIEW.............................................................................................................................. 14
  2.2 MODELING OF MATERIAL APPEARANCE ........................................................................ 15
    2.2.1 Attributes of Appearance............................................................................................. 16
    2.2.2 Realistic Image Synthesis ......................................................................................... 21
    2.2.3 Surface Modeling......................................................................................................... 22
    2.2.4 Illumination and Rendering ..................................................................................... 37
  2.3 DISPLAY SCREEN MODELING ............................................................................................. 50
    2.3.1 Color Display Technologies .................................................................................... 51
    2.3.2 Display Properties ....................................................................................................... 52
    2.3.3 Display Modeling Methods ...................................................................................... 55
  2.4 INTERACTIVE DISPLAY SYSTEMS FOR COMPUTER GRAPHICS SIMULATIONS ............. 64
    2.4.1 Augmented and Virtual Reality .................................................................................. 64
    2.4.2 Components for Augmented Reality ......................................................................... 66
    2.4.3 Integrated Display Systems for Mixed Reality .......................................................... 69
  2.5 COLOR IMAGE REPRODUCTION AND APPEARANCE .................................................... 75
    2.5.1 Factors in Cross-media Color Reproduction .............................................................. 76
    2.5.2 Chromatic Adaptation ............................................................................................... 77
    2.5.3 Perceived Lightness and Tone Reproduction ............................................................ 82
  2.6 APPLICATION DOMAINS ...................................................................................................... 85
    2.6.1 Psychophysics of material appearance ....................................................................... 85
    2.6.2 Print Proofing .............................................................................................................. 86
    2.6.3 Computer-aided Appearance Design .......................................................................... 87
    2.6.4 Access to digital collections ...................................................................................... 88
  2.7 SUMMARY OF OBJECT DISPLAY AND RELATED WORK ................................................. 90

3 CONCEPTUAL DESIGN OF AN OBJECT DISPLAY SYSTEM ................................................... 94
  3.1 LEVELS OF THE FRAMEWORK ......................................................................................... 94
  3.2 PHYSICAL DESIGN CONSTRAINTS .................................................................................... 106

4 TANGIBLE DISPLAY SYSTEMS .............................................................................................. 109
  4.1 OVERVIEW ........................................................................................................................... 109
  4.2 TANGIBLE DISPLAY SYSTEM CAPABILITIES ................................................................. 111
    4.2.1 Physical Interactivity ................................................................................................. 111
    4.2.2 Dynamic Viewing ..................................................................................................... 113
    4.2.3 Dynamic Control of Material Properties ................................................................... 113
  4.3 TANGIBLE DISPLAY SYSTEM DESIGN ......................................................................... 114
    4.3.1 Coordinate Systems .................................................................................................... 115
    4.3.2 Tracking for Modes of Natural Interaction ............................................................... 116
    4.3.3 Surface modeling (geometry, texture, gloss, and color) ........................................... 119
FIGURES

Figure 1. Examples of object surfaces being presented on object display systems .............. 7
Figure 2. Overview of the object display system concept............................................. 9
Figure 3. Illustration of the spatial and directional parameters of the BSSRDF function. ....28
Figure 4. Illustration of the BRDF geometry ..........................................................30
Figure 5. Diagram of the Blinn half-vector geometry .............................................33
Figure 6. A real physical painting is tilted in physical space, in a standard computer graphics 
representation, and in the object display representation.......................................96
Figure 7. The viewing position of the observer determines the angle used to index the BRDF 
curve ....................................................................................................................98
Figure 8. The positions of the observer and lighting with respect to the surface normal determine 
the angles in the BRDF calculation.................................................................99
Figure 9. The local surface normals vary due to surface texture, resulting in a different set of 
BRDF angles ........................................................................................................100
Figure 10. The surface texture that results from small scale surface geometry are represented by 
normal vectors describing the orientation of the surface at each position ............. 101
Figure 11. Self-shadowing from the mesoscale geometry of the surface texture ..........102
Figure 12. The physical height of features on the surface occlude the observer’s view of another 
point.......................................................................................................................103
Figure 13. Light that is reflected toward another point on the surface produces an inter-reflection 
................................................................................................................................104
Figure 14. Tangible displays for two different architectures ........................................110
Figure 15. Image sequence showing a painting model on the first tangible display system, the 
“tangiBook” ............................................................................................................ 112
Figure 16. Image sequence showing a painting model on a customizable tangible display system 
...................................................................................................................................... 112
Figure 17. Image sequence illustrating the viewpoint-based tracking and rendering capabilities of 
a tangible display system.......................................................................................113
Figure 18. The two renderings of the painting illustrate the ability of a tangible display to 
dynamically change the material properties of a displayed surface ....................... 114
Figure 19. Coordinate systems for tangible displays ................................................115
Figure 20. The polygonal geometry of the virtual object surface is modeled as a rectangle that is 
coincident with the plane of the physical screen .................................................. 121
Figure 21. A normal map encoded in an RGB image and the textured surface rendered from the 
normal map .............................................................................................................122
Figure 22. The textured surface rendered from the normal map alone and the surface rendered 
with diffuse color and gloss, specified by a BRDF model in three color channels ...... 123
Figure 23. The virtual camera and rendering positions .............................................. 127
Figure 24. Light booth environment used for the implementation of an object display system.130
Figure 25. Diagram illustrating the interaction of the five main components in an object display 
system ....................................................................................................................131
Figure 26. A set of six horizon maps for a painting model ........................................135
Figure 27. Using a horizon map-based technique, virtual shadowing increases as the object 
display screen is rotated away from the light .....................................................136
Figure 53. Renderings using filtered importance sampling, the median cut approximation, and hybrid approach ................................................................. 209
Figure 54. The geometric coordinate system used for rendering is shown on an image of the light booth environment ................................................................. 210
Figure 55. Two-dimensional diagrams illustrating the coordinate systems used in the object display system ................................................................. 212
Figure 56. A diagram illustrating the four unit vectors required in the rendering calculation at a position \((s,t)\) on the surface virtual ................................................................. 214
Figure 57. Diagram illustrating the calculation of the light intersection point in physical space ................................................................. 215
Figure 58. Illustration of the factors used to calculate the unit \((a, b)\) coordinates for the lookup into the illumination map ................................................................. 217
Figure 59. Measurement configuration for the display in the landscape orientation ................................................................. 229
Figure 60. Chromaticity over a set of angles for the display ................................................................. 230
Figure 61. Relative luminance for gray ramps at viewing angles from -30 to 30 degrees off-axis ................................................................. 231
Figure 62. Maximum display luminance as a function of viewing angle ................................................................. 232
Figure 63. The relative viewing-angle luminance as a proportion of the on-axis luminance for the display ................................................................. 233
Figure 64. Results of the model fitting for the black level luminance \((DC=0)\) as a function of viewing angle ................................................................. 235
Figure 65. The viewing-angle luminance factor model results for the portrait orientation measurement data ................................................................. 236
Figure 66. The viewing-angle luminance factor model results for the landscape orientation measurement data ................................................................. 237
Figure 67. Measurement configuration used to test the interactive luminance correction model ................................................................. 241
Figure 68. Absolute luminance results for the interactive correction model ................................................................. 242
Figure 69. Normalized luminance results for the interactive model with the display in the portrait orientation ................................................................. 243
Figure 70. Normalized luminance results for the interactive model with the display in the landscape orientation ................................................................. 244
Figure 71. Normalized luminance results for the interactive model in the oblique orientation ................................................................. 245
Figure 72. Comparison of the viewing-angle dependent error with the corrective model applied and without any corrective modeling applied ................................................................. 247
Figure 73. Gonio-based measurement configuration for the screen front surface reflectance ................................................................. 249
Figure 74. The luminance reflected from the screen as a function of the spectroradiometer detector angle ................................................................. 251
Figure 75. Cropped image of the vertical line source reflected from the front surface of the display screen ................................................................. 252
Figure 76. The median data vector from the captured image of the screen surface reflecting a line source ................................................................. 252
Figure 77. Configuration for measuring a virtual model of the X-Rite Classic ColorChecker ................................................................. 256
Figure 78. ColorChecker patch \(u’v’\) coordinates for the tungsten light condition ................................................................. 259
Figure 79. Screen gamut limitations for reproducing the virtual ColorChecker under tungsten illumination considering the effects of the fixed black-level offset ............................................................. 261
Figure 80. The light source and camera configuration of the apparatus for image-based BRDF measurements of high gloss surfaces ............................................................. 265
Figure 81. Images of the line source reflected from the surface of the five highest-gloss samples .................................................................................................................. 266
Figure 82. BRDF model results for the samples estimated by imaging a line source ............ 269
Figure 83. BRDF modeling with the Murakami GSP .......................................................... 271
Figure 84. Controlled light source constructed from a lamp, fiber optic cable, integrating sphere, and rectangular aperture covered with a diffusing filter .................................. 273
Figure 85. Side view of the measurement configuration for the real physical samples in the gloss measurement experiment ................................................................. 275
Figure 86. Top view of the measurement geometry illustrating the physical positions of the light, detector, and sample used in the 3D model .......................................................... 275
Figure 87. Front view of the measurement configuration for the physical samples ............ 277
Figure 88. The virtual sample on the screen is located at the same physical location as the real sample and measured using the same measurement configuration ........................................ 278
Figure 89. Front view of the measurement configuration for the virtual gloss samples ........ 279
Figure 90. Goniometric measurements of the real physical samples and the screen output for the virtual samples measured with a spectroradiometer mounted to a goni-arm .......... 281
Figure 91. Root mean square error (RMSE), correlation, and R² statistics for the gloss sample measurement data ................................................................................. 285
Figure 92. Photographic images of the 5 glossiest samples (GL100 to GL85), showing the specular reflection pattern from a rectangular light source ........................................ 286
Figure 93. Configuration of the patch samples in the chromatic adaptation experiment ........ 293
Figure 94. The set of patch chromaticities presented in the first selection stage of each trial ... 294
Figure 95. Screen capture of the textured background and virtual patches .......................... 297
Figure 96. Appearance of the viewing booth with the daylight illumination and tungsten illumination .................................................................................................................. 300
Figure 97. Photographic images of the patch grids for the three lighting conditions ............ 301
Figure 98. Plot of the intra-observer contrasts for the effect of ambient lighting ............... 309
Figure 99. Plots of the intra-observer surface texture - flat shading contrast for the three lighting conditions .......................................................................................................................... 310
Figure 100. The achromatic points selected by all observers for each condition .................. 312
Figure 101. The mean and median of the set of observer responses for each condition ........ 313
Figure 102. Comparison of the object display results to the results of the original Gorzynski thesis [1992] soft copy/hard copy experiments ................................................................. 316
Figure 103. Images illustrating a simple interactive soft-proofing application on a tangible display system ............................................................................................................. 324
Figure 104. A simulated print on canvas-textured paper is presented on the object display .... 325
Figure 105. Cropped screen capture images illustrating different paper options in the printing example application ........................................................................................................ 326
Figure 106. Screen captures of a set of glossy samples modeled in the object display framework ....................................................................................................................... 329
Figure 107. One of the modeled gloss samples being presented on the object display system.

Figure 108. An illuminated manuscript is displayed on the “tangiBook” system.

Figure 109. A physical painting side-by-side with an object display system.

Figure 110. A real painting alongside the captured surface model presented on an object display system.
TABLES

Table 1. CIEDE2000 Color Error for Six Channel, Sharpened RGB, and XYZ Color Calculations ................................................................. 177
Table 2. Comparison of Six Channel and Sharpened RGB Rendering for Multispectral Input (Optimization Lighting) .............................. 180
Table 3. Comparison of Six Channel and Sharpened RGB Rendering for Multispectral Input (Measured Test Lighting) ......................... 180
Table 4. Mean and Standard Deviation of Chromaticity over the ±30 Degree Range .......... 230
Table 5. Off-axis Luminance as a Proportion of the On-axis Luminance at 20, 25, and 30 Degrees .............................................................. 240
Table 6. Ward-Dür Model Parameters for a 3-Specular-Lobe Estimate of the Screen Surface BRDF ....................................................................... 253
Table 7. CIEDE2000 Color Error for ColorChecker Reproduction ............................................................. 258
Table 8. CIEDE2000 Color Error for ColorChecker Reproduction (Tungsten Condition Comparisons) .............................................................. 260
Table 9. Estimated 2-Lobe BRDF Parameters for the Painted Glass Samples .................. 268
Table 10. Physical Positions of the Sample, Light and Detector ........................................ 274
Table 11. The Full Set of Counterbalanced Condition Sequences used in the Experiment ....... 292
Table 12. Radii of the Major and Minor Axes of the Sampling Patterns for the Six Stages of a Trial ................................................................. 295
Table 13. Comparison of Display Characterization Accuracy with the Day et al. and the Enhanced Approach .............................................. 303
Table 14. Measured Chromaticities of the Tungsten Background and Patch Stimuli across the Ambient Lighting Conditions .............................. 304
1 INTRODUCTION

With advances in display screen and video hardware technology, the type of content presented by computers has progressed from text and simple shapes to high-resolution photographs, photorealistic renderings, and high-definition video. In this dissertation, the goal is to explore another stage of the progression, moving from high-fidelity image presentations toward digital surrogates that recreate the experience of directly viewing objects in the real world.

Along with advances in display hardware, there have been significant advances in the area of content capture, with the development of devices and methods for creating rich digital representations of real-world objects. Unlike photographic or video content, which provide a fixed record of the light in a scene under one particular set of conditions, these new technologies provide information about the underlying properties of the objects, which allows for simulation of their appearance under novel lighting and viewing conditions. In the past, with the prevalence of photo and video as the form of input, traditional display methods have focused on how best to present image-type stimuli to recreate the captured record of a scene. With the availability of digital object content, there is an opportunity for new types of display methodologies that present objects in a manner that is more like viewing the real objects directly.

This dissertation presents a general framework for representing objects with complex material properties and displaying them on electronic screens, with a goal of making them appear as if they are part of the real-world environment. At its core, there is a conceptual shift from a traditional image-based display representation to an object-based one. Instead of presenting the stored patterns of light from a captured scene, the objective is to reproduce the surface attributes
of a stored object, by recreating the dynamic patterns of light that would be produced by a real object if it were at the screen location. Though there are limitations to the display system implemented in this dissertation, the ultimate goal of the framework is to present objects onscreen that are nearly indistinguishable from directly viewing objects in the real world.

1.1 Motivation

Computers, presenting content on display screens, are widely used in applications where the intention is to convey the appearance of object surfaces. Computer-based displays provide the means to rapidly evaluate the appearance of different options in proofing and material design applications, allow for the presentation of stimuli that would be difficult to construct for studies of material appearance, and support widespread dissemination of culturally significant objects, such as artwork, that otherwise may not be easily accessible. In these cases, a primary goal is for the appearance of the reproduction onscreen to represent the appearance of the physical object surface that is being portrayed. The principal objective of the dissertation was to develop a methodology to allow object reproductions on display screens to behave and appear, to the extent possible, like their real-world counterparts.

When viewing a real object surface, the patterns of light that ultimately reach the observer are a product of the light sources in the environment, the material properties of the surface, and the movements of the observer. The appearance of the object changes with variations in the spectral composition or geometry of the incident light and also with changes in the geometric relationship between the surface and observer. With a real physical sample, observers can evaluate the surface appearance within the context of the light sources illuminating
it and can directly observe the changing patterns of reflections across the surface as they physically interact with the object.

In contrast, display screens are typically used to present objects as part of an independent self-luminous image, often with a dim or dark real-world surround. The objects displayed in images, videos, or interactive simulations are usually located in a virtual space separated from the viewer, and the screen serves as a window into this world. This virtual space typically has its own illumination (e.g. the scene lighting where a photo was recorded or the lighting model in a computer graphics simulation), so the appearance of the object depicted onscreen is independent of the geometry, color, and intensity of the real-world lighting around the observer. With the object located in a separate virtual world, the surface appearance does not change with the types of natural interactions that observers may engage in to evaluate real ones (tilting the object, looking at it from different perspectives). Interaction with the virtual object is typically based on indirect manipulation with a device such as a control pad, computer mouse, or keyboard.

The ability to provide natural modes of interaction with virtual objects was the impetus for developing tangible display systems, the first stage of development toward the more comprehensive object display systems that were the final goal of the dissertation. By tracking the orientation of the display and the observer’s viewing position using sensors, the tangible display system was able to update the patterns of surface reflections as the user rotated the screen or looked at it from different positions, the types of interactions that observers often use when evaluating the appearance of real surfaces.

There were multiple factors motivating the extension of the tangible display platform to a more comprehensive object display framework. Though tangible displays behaved like physical objects in the sense of user interaction, the display screen was still clearly self-luminous and did
not have the appearance of a real object reflecting light. This limitation was addressed in the object display framework by incorporating models of the real world illumination, so that the virtual reflections produced by the screen would be consistent with the intensity and geometry of the real lighting present. With the display producing light output for the virtual surface that is plausible for a real object (and with the rest of the screen masked), it is more difficult to distinguish the virtual surface from a physical one reflecting light.

In addition to removing the self-luminous visual cue, another motivation for reproducing virtual objects consistent with the real lighting conditions was related to factors in color appearance modeling. A goal of the object display framework was to minimize, as much as possible, any viewing condition differences between the virtual object onscreen and a real object that would require the need for a cross-media adjustment with a color appearance model. Color appearances models are used to account for differences in the surround (typically dim for a screen vs. an average surround for an object), in the absolute luminance level of the stimuli, and in the chromatic adaptation state of the observer. With the object display framework, the screen is viewed in the same (illuminated) surround and the stimuli are displayed at the same absolute luminance intensity as a physical object at that location. To match the chromatic adaptation state, the virtual surface is rendered based on the real white point of the light in environment, so that the adapting white point is consistent with a real object. However, in the case of chromatic adaptation, there is still the potential for a cross-media difference. Past research has shown that while observers fully adapt to the white point of real reflective objects, they only partially adapt to self-luminous display screens for white points other than D65. A visual experiment was conducted as part of the dissertation to investigate the chromatic adaptation state of observers when viewing the object display screen and evaluate whether their adaptation is complete, as it
would be with a real object. If the object display can produce reproductions that appear enough like reflective objects that they invoke a similar degree of adaptation, then it could be used directly for cross-media proofing without the need for adjustments with a color appearance model.

The final motivating factor for creating an object display system that reproduces virtual objects consistent with real lighting is that it provides the means to directly compare a captured digital model to the physical original. Once the real-world lighting environment is captured and entered into the system, the object display framework automates the process of presenting objects in the world, so that any captured surface model converted to the object display format can be displayed and should appear as if it were really in the scene. To determine the accuracy of a model, the original can be placed side by side with the object display reproduction, under the same illumination, and both of them can be directly viewed to see if there are any visible differences. For a quantitative comparison, the digital model can be evaluated by taking physical measurements of the screen output and comparing these to measurements of the original at the same location. In practice, reproductions are subject to the limitations of the object display implementation, but for an ideal object display, the similarity between the original and the displayed model onscreen would provide a direct indication of the accuracy of the digital model.

1.2 Overview

The overall goal of this dissertation is to provide the means to display objects with complex surface properties (color, gloss, and texture) on a display screen in a way that recreates the experience of viewing a surface in the real world. A set of examples that illustrate an object
display simulating these surface properties is shown in Figure 1. The general methodology used to accomplish this goal is to model surfaces in an illumination-independent form that has been standardized in the object display system. The real-world lighting is also specified in a compatible standardized form and the two are combined in a real-time computer graphics rendering engine so that the reflections for the modeled surfaces can be updated to match the real-world lighting present. These calculated light patterns are displayed by the screen at the same absolute intensity and with the calculated colorimetry of a real physical surface at that position to give the impression of a physical surface reflecting light.

The simulation of a reflective surface is achieved with traditional self-luminous screen hardware using a computational approach where the observer and display are actively tracked so that surface reflections can be rendered to the screen in a manner consistent with the observer’s real viewing position and the real lighting present. For a virtual object to behave like a real surface, the virtual reflections should update to remain consistent with changes in the geometry or the spectral composition of the illumination. The system accounts for the position of the viewer relative to the screen and the angle of the screen with respect to the real lights, which allows the luminance of diffuse reflections and patterns of specular reflections to change as they would for a real physical surface. Finally, by interactively monitoring the spectral composition of the ambient light (or by pre-measuring the real lighting) and incorporating a real-time multispectral color rendering pipeline, the color of virtual surface reflections are able to maintain consistency with the real physical lighting present.
Figure 1. Examples of object surfaces being presented on object display systems. In the top row, (left) a virtual Classic ColorChecker is shown along with the real chart, (center) a glossy surface is shown simulating a reflection of the light in its environment, and (right) a textured surface is shown with shadowing based on its actual orientation with respect to lighting. In the bottom row, (left) a real painting is shown alongside a captured model [Berns & Chen 2012] presented on an object display, and (right) a softcopy photo is simulated to appear like a reflective print.

The steps in the general process of simulating a reflective object with a self-luminous display screen are illustrated in Figure 2 (the process is described in detail in the conceptual design of Chapter 3). A self-luminous image is formed with a spatial array of pixels that are emitting combinations of red, green, and blue light. With the addition of observer-tracking and
interactive updating, the pixel array can be used to simulate directionally-varying output, in addition to the spatially-varying output. In the next stage, a virtual object model is associated with the pixel array of the screen. Using tracking information on the observer position and screen orientation, the reflectance function (BRDF) for each point on the virtual surface is simulated interactively to produce dynamic virtual surface reflections that are based on the real angles between the observer and screen. To simulate mesoscale texture, each virtual object point is provided with surface orientation information for use in the BRDF calculation and height-based information for dynamic shadowing. In the final stage, the computer graphics lighting model used to render the virtual surface is matched to the real lighting in the environment around the display. Using virtual lighting matched to the real lighting, the calculated reflections produced by the screen simulate the color (in absolute luminance) and the spatial patterns of reflections that would be produced by a physical reflective object if it were placed at the screen location.
<table>
<thead>
<tr>
<th>Level 1. Pixel (trichromatic light)</th>
<th>Level 2. Pixel array</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1" alt="Pixel Image" /></td>
<td><img src="image2" alt="Pixel Image" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Level 3. Screen output modulated by viewing direction</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image3" alt="Screen Output" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Level 4. Physical mapping of screen to virtual object</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image4" alt="Virtual Object Geom" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Level 5. Surface BRDF based on real observer &amp; screen angles</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image5" alt="Virtual Light" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Level 6. Per-pixel virtual surface orientation</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image6" alt="Virtual Surface" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Level 7. Per-pixel virtual surface height (for shadowing)</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image7" alt="Surface Height" /></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Level 8. Virtual surface matched to physical lighting</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image8" alt="Mismatched Lighting" /></td>
</tr>
</tbody>
</table>

**Figure 2.** Overview of the object display system process for simulating a physical object reflecting light.
The principal goals of this dissertation are to:

1. Overall, create a display system capable of reproducing objects with gloss, texture, and color properties that behave and appear like real physical objects
2. Facilitate the process of displaying captured objects by specifying a common form for color, texture, and gloss properties that is compatible with typical captured data, and once converted to this form will allow the captured object to automatically be shown onscreen
3. Allow virtual objects shown onscreen to respond to the natural, physical types of interaction used with real objects
4. Provide a spectral representation for surface reflectance and illumination that can be used in a real-time rendering workflow and calculate accurate color for a range of real-world lighting types
5. Provide the capability to calculate photometrically-accurate gloss reflections, for real-world lighting, within a real-time rendering engine
6. Develop a display model capable of producing the absolute luminance and colorimetry of the rendering engine output and the ability to maintain this output for a range of viewing angles (out to 30°)
7. Evaluate the accuracy of the overall object reproduction system by taking direct physical measurements of the screen and demonstrate that an object display provides a way to use direct physical measurements to assess the accuracy of digital models
8. Assess whether an object display methodology allows observers to more fully adapt to a self-luminous display screen, and if so, determine which factors may contribute to the increase in adaptation
9. Demonstrate potential uses of the object display methodology for real-world applications.
These goals were addressed in the dissertation by designing a conceptual framework for object display systems, describing the stages in transitioning from an image-based to object-based display methodology. A functional prototype of an object display system was created by integrating a color processing pipeline, a gloss reproduction pipeline, real-world illumination modeling, observer and display tracking, and display screen modeling. To streamline the process of displaying captured data, a set of illumination-independent surface models for representing texture, color, and gloss properties in the system were specified. The surface models specified were ones commonly used for captured data or ones compatible with commonly-used methods, so that typical types of captured data can be readily converted to the input form used in the object display system. To support color calculations based on changing real-world illumination, an end-to-end workflow for color rendering was developed using an optimized six-channel multispectral representation. To support accurate glossy reflections with interactive rendering, a gloss reproduction pipeline was constructed using a hybrid approach that combines two approximate real-time rendering techniques to produce photometrically-accurate results for output to the screen. The capability to display the absolute results from the rendering engine was achieved by measuring and characterizing the properties of a screen in terms of absolute colorimetry and building a model to correct for luminance changes with viewing angle. A system evaluation based on direction physical measurements of the screen was performed to assess the overall accuracy of the color and gloss reproduction. A visual experiment was conducted to assess the state of chromatic adaptation when using the object display to evaluate its color appearance for cross-media proofing applications. The experiment included systematic variations of the lighting and material properties to investigate whether these factors contribute
to adaptation differences when viewing display screens instead of real objects. Finally, a set of sample applications and demonstrations were developed to illustrate the capabilities of an object display system and the potential utility in a range of areas.

The dissertation is organized in the following set of chapters. In Chapter 2, a literature review is provided that describes background material and research related to the work in this dissertation. The literature review includes sections on object material properties, digital modeling of material appearance attributes, methods for modeling display screens, cross-media color reproduction, research in the field of augmented reality, and an overview of previous work in potential applications domains. In Chapter 3, a high-level conceptual framework is described that outlines the stages in transitioning from a traditional pictorial display representation to an object-based one. In Chapter 4, tangible display systems are described. These systems, which allow for natural forms of interaction with virtual surfaces, were the first stage in the development of the type of object displays envisioned in the dissertation. In Chapter 5, an example of a more recent object display system is presented. This chapter includes a description of the various technologies involved in creating a working prototype and a section illustrating the capabilities. Chapter 6 provides a detailed description of the multispectral color processing pipeline developed for use in the object display framework. Chapter 7 provides this type of detailed description for the gloss reproduction workflow. In Chapter 8, the methods used to measure and model the colorimetric and directional properties of the display screen are described. In Chapter 9, the results are presented for a measurement-based system evaluation of the color and gloss reproduction. Chapter 10 describes a visual experiment performed to evaluate the state of chromatic adaptation when using the object display system. In Chapter 11, a set of sample applications is presented to demonstrate the potential utility of this work. Chapter
12 provides a discussion of final conclusions and areas for future work. Supplemental material is provided in a set of appendices. These appendices include the description of a display characterization method for improving accuracy when channel interaction is present and a projector-based lighting workflow for creating computer-controlled lights that can be used to illuminate the real world environment surrounding an object display.

The work described in this dissertation has a potential impact for a range of application domains where the goal is to preview, evaluate, or reproduce the appearance of object surfaces. It could allow for new types of computer-aided design systems where designers can interactively edit the properties of material surfaces, while directly viewing the results of their work in a real-world context. In the digital printing domain, it could provide the capability to conduct proofing using softcopies that behave like real physical prints. It could also be used to bring digital archives of artwork and cultural heritage into the physical world, allowing them to be shared in interactive virtual museum exhibits that simulate the experience of viewing the original. For researchers studying material perception, it could facilitate experimental testing by providing a platform where stimuli that behaves like a physical surface can be varied systematically under computer control.
2 REVIEW OF LITERATURE

2.1 Overview

A framework for reproducing the appearance attributes of object surfaces on display screens incorporates concepts from multiple disciplines. Ultimately, the appearance of a surface depends upon the properties of the material, the illumination, and the perceptual processes of the observer. The ability to reproduce that appearance depends upon the physical light production by the display device, the lighting environment in which it is viewed, and the perceptual processes of the observer. Accounting for the spectral dimension of light, as it pertains to surface reflectance factor and the spectral power distribution of illumination, and the resulting perception of color by the observer’s visual system has been studied extensively in the color science domain. The simulation of the directional dimension of light reflections, as governed by the geometry of the illumination environment, the geometry and texture of the surface, and the bidirectional reflectance distribution function (BRDF) of the material, has been studied extensively in the computer graphics domain.

Methods for modeling and simulating the spectral and directional dimensions of surface reflections provide the underlying data for reproducing the appearance of a surface. These data must then be conveyed through the display screen in such a way that it reproduces the actual appearance of a real reflective object. The perceptual phenomena that must be considered when attempting to match the appearance of self-luminant electronic displays to reflective objects have been studied in the context of cross-media color image reproduction. The process of determining the digital values needed to drive the display screen to produce the physical light output calculated from the surface simulation falls within the literature on colorimetric characterization.
and modeling of electronic displays. The underlying technologies needed to provide natural interactivity with the computer graphics simulation, by tracking the real position of the observer and the display and incorporating this information in real-time, draw on the fields of augmented and virtual reality.

The background literature relating to the properties of objects, their interactions with light, the related appearance attributes, and methods for digitally modeling them is discussed in Section 2.2. In Section 2.3, there is a focus on color display technologies, screen properties, and methods for characterizing and modeling the color and directional aspects of the light emitted and reflected from the display screen. Section 2.4 provides a description of the methods required for developing interactive displays systems and discusses related research from the field of augmented reality. Section 2.5 describes related concepts from traditional color image reproduction. The background literature related to application areas for the object display system is discussed in Section 2.6.

2.2 Modeling of Material Appearance

A critical element of the object display system is the ability to accurately model the properties of a real object surface and simulate its appearance in a realistic lighting environment, so that this data can ultimately be reproduced on the object display screen. This modeling and simulation falls into the category of digital modeling of material appearance [Dorsey et al. 2008], an interdisciplinary field that incorporates concepts from physics, computer science, vision science and color science.
2.2.1 Attributes of Appearance

The appearance of a surface depends on the way that the properties of the surface influence the light reaching the observer, and how the results of that interaction are perceived by the visual system. The attributes associated with surface appearance fall into two main categories: geometric attributes, which are related to the spatial and directional aspects of surface-light interaction, and color attributes, which relate to the spectral aspects of surface-light interaction [Hunter & Harold 1987].

2.2.1.1 Color Attributes

The color attribute of surface appearance relates to the spectral properties of the physical light-surface interaction as perceived by the human visual system. Surface color is modulated by the interaction of the spectral distribution of incident light and the wavelength-dependent absorption and scattering by surface colorants [Berns 2000]. The spectral reflectance of the surface, the proportion of incident light that is reflected at each wavelength over the visible spectrum, is the property of the surface that describes its contribution to diffuse color. More formally, the spectral properties of a surface are defined in terms of the spectral reflectance factor, which is the ratio of the reflected light from the surface relative to the reflected light from a similarly illuminated perfect reflecting diffuser [Berns 2000]. The product of the spectral power distribution of the illumination, $S_\lambda$, and the spectral reflectance factor of the surface, $R_\lambda$, determines the spectral distribution of the stimulus viewed by the observer [Berns 2000]:

$$\Phi_\lambda = S_\lambda R_\lambda.$$  \hspace{1cm} (1)
The spectral composition of the light reaching the observer from the surface, along with the perceptual processes of the human visual system, results in the surface appearance attribute of color. The human visual system has three types of receptors (cones), each with a different spectral sensitivity to light [Stockman et al. 1993], which results in a trichromatic response to the spectral stimuli reaching the retina. The response of the visual system to a spectral stimulus is dependent on the integrated product of stimulus spectral power distribution and the spectral sensitivity of the three cones classes. This results in the phenomenon known as metamerism, where two stimuli with different spectral power distributions will produce the same the visual response if the integrated values over the visual spectrum are identical for all three receptors [Fairchild 2005]:

\[
\int_{\lambda} \Phi_1(\lambda)L(\lambda)d\lambda = \int_{\lambda} \Phi_2(\lambda)L(\lambda)d\lambda \tag{2}
\]

\[
\int_{\lambda} \Phi_1(\lambda)M(\lambda)d\lambda = \int_{\lambda} \Phi_2(\lambda)M(\lambda)d\lambda \tag{3}
\]

\[
\int_{\lambda} \Phi_1(\lambda)S(\lambda)d\lambda = \int_{\lambda} \Phi_2(\lambda)S(\lambda)d\lambda \tag{4}
\]

where \(\Phi_1\) and \(\Phi_2\) represent stimuli with different spectral distributions and \(L, M, S\) represent the spectral responsivities of the long, medium, and short wavelength cones classes.

These three visual responses to the spectral stimuli along with perceptual and cognitive processes result in the appearance attributes of color. There are three perceptual dimensions associated with surface color: hue, lightness (or brightness), and chroma (or colorfulness) [Fairchild 2005]. Hue is defined as the attribute “according to which an area appears to be similar to one of the perceived colors: red, yellow, green, and blue, or to a combination of two of them,” brightness as the attribute “according to which an area appears to emit more or less light,”
and colorfulness as the attribute “according to which the perceived color appears to be more or
less chromatic” [from the *International Lighting Vocabulary*, quoted by Fairchild [2005], pgs. 85-87]. Along with hue, the three dimensions of color may also be described in terms of a relative version of brightness and of colorfulness, lightness and chroma, which are defined relative to a similarly illuminated area that is perceived to be white [Fairchild 2005].

### 2.2.1.2 Geometric Attributes

Geometric attributes of surface appearance relate to the directional and spatial aspects of light-surface interaction. The scale of the surface geometry is an important factor in the type of appearance attributes that result. Geometry at the microscale is associated with surface gloss, while geometry at the mesoscale is associated with surface texture, and large scale geometry (megascale) is associated with the shape of the object [Westin et al. 1992; Koenderink & van Doorn 1996].

#### 2.2.1.2.1 Gloss

Specular light reflection from an object’s front surface is associated with gloss, the attribute of appearance that relates to the shininess or luster of a surface [Hunter & Harold 1987]. The microstructure of the surface, variation in the surface topography at the microscopic level (microscale), is the principal physical factor responsible for the level and type of gloss produced by the surface [Westin et al. 1992]. A proportion of the light that strikes the first surface of an object is reflected in the specular (mirror) direction, producing the reflections associated with surface gloss. The smoothness of the surface, the index of refraction of the material, and the angle of incidence of the light affect the magnitude and distribution of the first surface reflection
Surfaces that are smooth at the microscale level produce sharp, mirror-like reflections in the specular direction, while surfaces with microscale roughness spread the reflected light over a wider cone of angles [Dorsey et al. 2008]. Rough surfaces may also produce off-specular peak reflection at large incidence angles due to shadowing and masking by the microscale surface facets [Torrance & Sparrow 1967].

Hunter and Harold [1987] identified several visual properties that are indicative of different types of gloss. High gloss surfaces produce sharp mirror reflections and exhibit distinctness-of-image gloss [Hunter & Harold 1987]. Medium gloss surface are identified by the appearance of shininess and the brilliance of the specular reflections, described by Hunter as specular gloss. Lower gloss surfaces may exhibit sheen, identified by Hunter and Harold as “shininess at grazing angles” [pg. 78] or contrast gloss (luster), characterized by the contrast between the regions of specular reflections and the rest of the surface. Hunter and Harold also identified the absence of haze around specular highlights, absence-of bloom, as a visual indicator of the degree of gloss of the surface.

The set of visual properties that Hunter associated with gloss perception were based on visual observations and not derived to be independent of one another (as noted by Ferwerda et al. [2001]). O’Donnell & Billmeyer [1986] and Ferwerda et al. [2001] conducted psychophysical experiments using multi-dimensional scaling techniques to better determine the set of independent perceptual attributes associated with gloss perception. For the samples tested, O’Donnell and Billmeyer found that measurements of distinctiveness-of-image were most highly correlated with the perception of gloss. Ferwerda et al. found that the perception of gloss could be described principally in terms of two dimensions: distinctness of reflections (distinctness-of-image gloss) and contrast between the highlights and diffuse regions (contrast gloss).

[Hunter & Harold 1987].
2.2.1.2.2 Texture

The texture attribute of surface appearance relates to the surface geometry at the mesoscale, the intermediate scale between the surface microstructure and the large scale geometry providing the object shape. [Westin et al. 1992; Koenderink & van Doorn 1996]. Though difficult to formally define the size of the mesoscale features, a general distinction is often made based on visual criteria [Koenderink & van Doorn 1996]. At the smaller end of the continuum, the surface geometry reaches the mesoscale when it is large enough that spatial variation is evident and the appearance effects are no longer adequately represented by an averaged level of surface reflectance [Koenderink & van Doorn 1996]. At the larger end of the continuum, the distinction is made between the higher-frequency spatial variation of the mesoscale and the smooth shading variation that results from large scale geometry [Koenderink & van Doorn 1996]. The appearance of texture is characterized by spatial irregularity or “bumpiness” [Ho et al. 2007b]. The irregularity in surface reflection is caused by the mesoscale geometry through two mechanisms: local vignetting (self-shadowing) and modulation of the surface orientation relative to light sources [Koenderink & van Doorn 1996].

2.2.1.2.3 Shape

At the largest scale, the geometric attributes of the object determine its shape. The large scale (megascale) geometry provides a geometric three-dimensional form (e.g. spherical, cubic, ellipsoidal) that can be characteristic of the object [Koenderink & van Doorn 1996]. The large scale geometry, and its relation to the illumination, impacts the shading of the surface. The relationship between shape and shading is an important aspect of conveying shape information in
artwork and has led to a field of study on the ability of the visual system to extract shape information from the appearance of shading [Kleffner & Ramachandran 1992].

2.2.2 Realistic Image Synthesis

As described in the preceding sections, there are a number of surface attributes, both spectral and geometric, that must be considered when attempting to model and reproduce the appearance of real objects. Greenberg et al. [1997] developed a framework for realistic image synthesis to describe the processes needed to fully model the appearance attributes of objects and create synthetic images with the fidelity of real photographs. Their objective was to define the capabilities for moving beyond visually-appealing renderings in computer graphics to physically accurate, high fidelity ones that would be predictive of real materials in real environments. They identified three main components that must be considered when attempting to produce accurate reconstructions: light reflection models for the surface, light transport simulations, and models of the perceptual processes of the observer. They noted that a comprehensive light reflection model would require physically-based modeling of the specular and diffuse (directional) reflectance of materials on a wavelength-dependent (spectral) basis, while accounting for the effects of surface variation and sub-surface scattering. As they described, a complete approach to the light transport simulation would require physically-based global illumination rendering methods that account for the interaction of lighting and material surfaces in complex, realistic environments. The final component of the framework described the need to model the perceptual processes of the human observer, to allow content to be optimally adapted to the display screen and to provide perceptual metrics for determining whether various methods
could produce results that were visually indistinguishable from photographs [Greenberg et al. 1997].

2.2.3 Surface Modeling

The surface modeling component describes how the surface interacts with the light that is incident upon it and is a critical aspect of modeling the appearance of a material. The surface model needs to account for the spectral dependency of light reflectance to simulate color properties and the directional dependency to simulate gloss properties. To support the reproduction of complex surfaces, the surface modeling must also account for the spatial variation at the mesoscale that contributes to the appearance of texture.

2.2.3.1 Spectral Reflectance Modeling (Color)

The color properties of a surface relate to its spectral reflectance factor at each wavelength over the visible spectrum, but a high-dimensional spectral representation is often not practical in computer graphics. Representations based on a small number of color channels are often used, but provide varying degrees of color accuracy depending on factors such as the number of channels, the spectral power distribution of light sources, and the selection of the primaries or basis functions for the channels.

2.2.3.1.1 RGB modeling

Color representations used in computer graphics are typically three dimensional, and in many cases, employ an RGB model where color is specified in terms of the relative amounts of the red, green, and blue channel for a particular device [as noted by Ward & Eydelberg-Vileshin
For an RGB color representation to be meaningful, the red, green, and blue primaries must be formally defined so that a given set of RGB values correspond to a particular three-dimensional color stimulus.

### 2.2.3.1.2 Modeling based on CIE XYZ Colorimetry

In 1931, the CIE developed a standardized system for colorimetry by defining a set of reference primaries, denoted \(X, Y, Z\), and deriving the spectral color matching functions for the 2° standard observer based on experimental data collected by Wright and Guild [Fairman et al. 1997]. The color matching functions (denoted \(x, y, z\)) define the set of tristimulus values, the amounts of the \(X, Y, Z\) primaries, needed to produce a visual match for each wavelength of monochromatic light. On the basis of Grassman’s law of additivity, visual matches between spectral stimuli can be determined using the color matching functions by taking the product of the stimuli and functions at each wavelength and summing the results over the visual spectrum [Fairman et al. 1997]. The integrated XYZ tristimulus values resulting from this type of calculation provide a trichromatic representation that can be used to specify whether two stimuli will produce a visual match. The XYZ values for a spectral stimulus are calculated from the spectral reflectance factor of the surface, the spectral power distribution of lighting and the color matching functions according to the equations [Berns 2000]:

\[
X = k \int \lambda S_\lambda R_\lambda \bar{x}_\lambda d\lambda \\
Y = k \int \lambda S_\lambda R_\lambda \bar{y}_\lambda d\lambda \\
Z = k \int \lambda S_\lambda R_\lambda \bar{z}_\lambda d\lambda
\]
where $S_{\lambda}$ represents the spectral power distribution of the light source, $R_{\lambda}$ represents the spectral reflectance factor of the surface, and $k$ is a normalizing constant given by:

$$
\frac{100}{\int_{\lambda} S_{\lambda} \frac{y_{\lambda}}{d\lambda}}
$$

(8).

Color management techniques based on CIE colorimetry have been integrated into computer graphics systems. Joblove and Greenberg [1978] described possible color spaces for rendering and suggested the use of a CIE-based color space. Borges [1991] evaluated the accuracy of performing lighting calculations by multiplying surfaces and illumination represented in terms of CIE XYZ primaries. Ward and Eydelberg-Vilesin [2002] developed an RGB-based rendering workflow for computer graphics based on a linear transformation of the CIE primaries to sharpened RGB primaries. They combined the use of the sharpened RGB primaries (developed by Süsstrunk et al. [2001]) with a method of spectral pre-filtering the stored reflectance factor of objects to improve the color accuracy of rendering in three channels.

When representing spectral data in a three-channel system, the reflectance factor of the surface is often determined by calculating tristimulus values under a selected illuminant and dividing by the tristimulus values of a white surface under the same illumination [Ward & Eydelberg-Vilesin 2002]. During relighting, these relative tristimulus for the surface reflectance are multiplied on a per-channel basis by the intensity of light sources, also specified in tristimulus values, to estimate surface reflections [Borges 1991]. Ward and Eydelberg-Vilesin found that highest color accuracy for three channel rendering could be achieved by performing the calculations in the sharpened RGB space [Süsstrunk et al. 2001], and when the dominant light source for rendering was known, spectral pre-filtering by using its spectral power distribution in the initial calculations where surface reflectance factor is reduced to three channels. However, as
Ward and Eydelberg-Vileshin noted, the calculation of the color of reflections during the rendering process is inherently a spectral problem, because it is dependent upon the interaction of the continuous spectral power distribution of the light sources and the spectral reflectance factor of the surface (as described by the terms on the right-hand side of Eqs. (5, 6, 7)). Relighting in only three channels may produce inaccurate results, in particular if the spectral power distribution of the lighting is not known a priori, or the rendering involves multiple light sources with different spectral power distributions [Ward & Eydelberg-Vileshin 2002].

2.2.3.1.3 Spectral Rendering

Higher dimensional methods for spectral estimation have been developed to overcome some of the color accuracy limitations of rendering in only three channels. Johnson and Fairchild [1998] developed a full spectral rendering workflow for OpenGL that performed rendering in 93 spectral channels and allowed for the simulation of wavelength-based color phenomena. The use of OpenGL running on the GPU provided the computational processing necessary to perform such high dimensional rendering, but was limited by OpenGL’s capabilities to provide only direct lighting calculations [Johnson & Fairchild 1999].

Meyer [1988] preformed an analysis to select an optimal set of wavelengths for rendering and concluded that at a minimum, a set of four wavelengths could produce reasonable color accuracy. Peercy et al. [1993, 1996] developed a computer graphics framework for spectral rendering based on linear sets of orthonormal basis functions. Peercy’s framework provided the flexibility to use a general set of basis functions if spectral distributions were sufficiently smooth or in the case of light source with spikes in the distribution, to tailor an optimal set of basis functions using a characteristic vector analysis of the spectral properties of
surfaces and lights in the scene. Drew and Finlayson [2003] developed an abridged spectral rendering approach based on the use of sharpened basis functions. They demonstrated that performing rendering calculations by multiplying the sharpened basis functions coefficients of the lighting and reflectance could produce results similar to full spectral calculations.

2.2.3.1.4 Multispectral Input

The capture methodologies used in multispectral color image reproduction can be used to provide spectral-based input data to a rendering pipeline. Spectral imaging methods provide the means to capture higher-dimensional spectral data than provided by typical trichromatic imaging systems and have been used as the input for end-to-end spectral color reproduction workflows in printing [Imai et al. 2003]. Hardeberg et al. [1999] developed a system for estimating the spectral reflectance of surfaces from multispectral images acquired using a monochrome CCD camera and a set of optimized spectral filters. The captured surface reflectance data from the camera system was used to more accurately predict the color of the surface under different illuminants. Additional methods for multispectral image capture have been developed that combine trichromatic color-filter array cameras with two additional colored filters to provide six spectral capture bands for estimating surface reflectance [Berns et al. 2005]. While multispectral capture is typically used to estimate surface reflectance data, recently Tominaga & Fukuda [2007] and Tominaga & Tanaka [2008] have applied multispectral imaging techniques to capture illumination maps of real-world environments. By imaging through multiple filters when capturing mirrored spheres or using panoramic cameras, they have been able to estimate the spectral distribution of illumination in real-world environments.
2.2.3.2 Geometric Reflectance Modeling

Surfaces are often modeled in computer graphics at three scales to account for the effects of geometry on surface reflectance [Westin et al. 1992]. The microscale geometry that impacts the directional distribution of reflections is often modeled using the bidirectional reflectance distribution function (BRDF) of the surface, while the mesoscale scale structure is modeled using bump and shadowing maps [Max 1988]. The large scale geometry corresponding to the shape of a surface is often modeled using a mesh of polygons [Watt 1993].

2.2.3.2.1 Directional Reflectance (Gloss)

The microstructure of the surface, responsible for the gloss attribute of appearance, impacts the directional distribution of light reflecting from the surface. Nicodemus et al. [1977] defined the nomenclature and mathematical formulas used to describe the directional distribution of light reflectance. In its most general form, the directional distribution of light reflection from a surface can be described in terms of an eight-dimensional function of the position and direction of incoming and outgoing light, defined by Nicodemus et al. as the bidirectional scattering-surface reflectance-distribution function (BSSRDF). The BSSRDF function, $S$, describes the ratio of the flux leaving the surface at position $(x_r, y_r)$ to the flux incident upon it at position $(x_i, y_i)$ [Nicodemus et al. 1997]:

$$S = S(\theta_i, \phi_i, x_i, y_i; \theta_r, \phi_r, x_r, y_r)$$  \hspace{1cm} (9)

This ratio is specified as a function of the direction of the light incident upon an area $dA_i$, where $(\theta_i, \phi_i)$ represent the incident zenith and azimuth angles, and the direction of light emerging from the surface, where $(\theta_r, \phi_r)$ represent the exitant zenith and azimuth angles, as shown in Figure 3 (which is based on Figure 1 from Nicodemus et al. [1977], pg. 4).
Figure 3. Illustration based on the diagram from Nicodemus et al. [1977, pg. 4] of the spatial and directional parameters of the BSSRDF function.

The spatial parameters \((x, y)\) of the BSSRDF allow it to account for the more general case of sub-surface scattering, but the distribution of reflectance is often represented in a simplified form to reduce the dimensionality. When considering the reflectance properties of a uniform isotropic surface over an area that is uniformly irradiated, the reflectance can be characterized in terms of only the directions of the incident irradiance and reflected flux [Nicodemus et al. 1977].
Nicodemus et al. defined the four-dimensional bidirectional reflectance-distribution function (BRDF) as the ratio of reflected luminance to the incident irradiance on the surface:

\[
f_r(\theta_i, \phi_i; \theta_r, \phi_r) = \frac{dL_r(\theta_i, \phi_i; \theta_r, \phi_r; E_i)}{dE_i(\theta_i, \phi_i)} \text{ [sr}^{-1}] \tag{10}\]

where \( dL_r \) is the elementary luminance and \( dE_i \) is the incident irradiance. The BRDF quantity represents a “concentration of reflectance” per solid angle and has the units of inverse steradians [Nicodemus et al. 1977, pg. 6]. The BRDF quantity is defined as a function of the two sets of polar angles, \((\theta_i, \phi_i)\) and \((\theta_r, \phi_r)\), corresponding to the direction of incident irradiance and reflected luminance, as shown in Figure 4 (which is based on the BRDF figure from [Nicodemus et al. 1977, pg. 6]). The BRDF relates to infinitely small quantities and is not physically-realizable, but can be integrated over a finite set of solid angles to describe the physical reflectance of a surface [Nicodemus et al. 1977]. The reflectance \((\rho)\) of a surface, the ratio of the incident to reflected flux, can be derived from the BRDF for finite beam angles \(\omega_i\) and \(\omega_r\) by integrating the product of the BRDF and incident luminance over the two solid angles, and comparing to the integrated radiance contained within the incident beam [Nicodemus et al. 1977]:

\[
\rho(\omega_i; \omega_r; L_i) = \frac{\int_{\omega_i} \int_{\omega_r} f_r(\theta_i, \phi_i; \theta_r, \phi_r) L_i(\theta_i, \phi_i) d\Omega_i d\Omega_r}{\int_{\omega_i} L_i(\theta_i, \phi_i) d\Omega_i} \tag{11}\]
where \(d\Omega_i\) is the element of the projected solid angle for the incident illumination and \(d\Omega_r\) is the element of projected solid angle for the reflected beam.

**Figure 4.** Illustration based on the Nicodemus et al. [1977, pg. 6] diagram defining the geometry of BRDF in terms of a beam of incident light with direction \((\theta_i, \phi_i)\) and an elementary solid angle \(d\omega_i\), and a reflected beam with direction \((\theta_r, \phi_r)\) and elementary solid angle \(d\omega_r\).

### 2.2.3.2.1.1 Parametric BRDF Models

The BRDF of a surface is a useful concept for describing its directional reflectance properties, but due to its high dimensionality can not be fully enumerated as a function of all possible angles. In the computer graphics domain, parametric models of the BRDF have been developed to represent BRDF in a more compact, tractable form.


2.2.3.2.1.2 Lambertian Model

The simplest BRDF model is based on Lambert’s model for perfectly diffuse reflection, where the radiance exiting the surface is equal in all directions [Dorsey et al. 2008]. Because the viewing direction does not need to be considered in reflectance calculations, the use of a Lambertian model minimizes the computational complexity of rendering [Dorsey et al. 2008]. Though convenient for rendering and widely used, the Lambertian model is limited in its ability to represent real physical surfaces [Oren & Nayar 1994]. It does not account for surface gloss, and as demonstrated by Oren and Nayar, does not describe the real behavior of diffuse surfaces with surface roughness, which instead may exhibit a backscattering reflectance phenomenon.

2.2.3.2.1.3 Phong Model

Phong [1975] developed a model for specular surface reflection able to account for varying degrees of surface gloss. The Phong model allows for varying levels of gloss by incorporating a cosine term raised to a power function $n$ to describe the width of the specular lobe and a multiplier $W(i)$ to adjust the peak specular reflectance as a function of the incident lighting angle. The total reflection from surface, including diffuse and specular components, is calculated from the formula [Phong 1975]:

$$S_p = C_p \left[ \cos(i)(1 - d) + d \right] + W(i) \left[ \cos(s) \right]^n$$

(12)

where $C_p$ is the reflection coefficient at a given wavelength, $i$ is the incident lighting angle, $s$ is the angle between specular reflection direction and viewing direction, and $d$ is the diffuse reflection coefficient. The $\left[ \cos(s) \right]^n$ term, a function of the angle between the specular direction and actual viewing direction, spreads the specular reflection over a cosine lobe, while the $n$
exponent allows the lobe to be narrowed with increasing $n$ values to produce higher gloss surfaces [Dorsey et al. 2008]. A limitation of the model, however, is that it is not physically-based. As Phong noted in the original paper, model parameters are empirically adjusted to produce a certain appearance.

2.2.3.2.1.4 Blinn-Phong Model

Blinn [1977] developed a modified form of the Phong model that replaced its specular term, which was based on the angle between the specular direction and viewing direction, with a specular term based on the angle between the normal and half-vector of the lighting and viewing directions. Blinn noted that specular (mirror) reflections occur when the half-vector between viewing and lighting direction is perfectly aligned with the surface normal and that the falloff over the specular lobe could be modeled based on the angle between these two vectors. Blinn’s geometry for the half-vector $H$, which lies halfway between the light vector $L$ and viewing vector $E$, and its relation to the surface normal $N$ for specular and off-specular viewing, is shown in Figure 5. The half-vector is calculated from $L$ and $E$ by the formula [Blinn 1977]:

$$H = \frac{L + E}{||L + E||}$$ (13)
Figure 5. Diagram of the Blinn [1977] half-vector geometry. Left (a), the eye-direction E is at the specular direction with respect to the light vector L and the surface normal N. The half vector H between L and E is coincident with the normal direction. Right (b), the eye-direction E is past the specular direction R by an angle of $\theta_s$, causing the half-vector H to deviate from the normal direction N by an angle of $\theta_h$.

The Blinn model uses a cosine term (often represented as a dot product) raised to an exponent, as in the Phong model, to adjust the width of the specular lobe. The intensity of reflection ($i$), considering ambient, diffuse, and specular components, is calculated with the formula [Blinn 1977]:

$$i = p_a + \max(0, N \cdot L)p_d + (N \cdot H)^{c_1}p_s$$  \hspace{1cm} (14)

where $N$ represents the surface normal, $L$ is the lighting direction vector, $H$ is the half-vector between the viewing and lighting directions, $c_1$ (the shininess) is the exponent narrowing the width of the specular lobe and where $p_a$, $p_d$, and $p_s$ represent the proportions of ambient, diffuse, and specular reflection, respectively.

2.2.3.2.1.5 Ward Model

Ward [1992] developed an additional BRDF model based on the half-vector concept from Blinn-Phong, with the objective of providing greater physical validity. Ward’s principal
objectives were to insure that the formula was symmetric with respect to the lighting and viewing directions (to maintain Helmholtz reciprocity) and to provide proper normalization such that the model would maintain conservation of energy. In the Ward model, the specular reflectance lobe is modeled as a Gaussian function of the angle between the half-vector and surface normal, instead of the cosine lobe raised to an exponent of the Phong model. The formula for representing the complete BRDF includes a Lambertian diffuse term and a normalized specular term [Ward 1992]:

$$\rho_{brdf}(\theta_i, \phi_i, \theta_r, \phi_r) = \frac{\rho_d}{\pi} + \rho_s \frac{1}{\cos \theta_i \cos \theta_r} \exp\left(-\frac{\tan^2 \delta}{\alpha^2}\right)$$

(15)

where $\rho_d$ is the diffuse reflectance parameter, $\rho_s$ is specular reflectance, $\alpha$ is the standard deviation of surface slope, and $\delta$ is the angle between the surface normal vector ($\mathbf{n}$) and the halfway vector ($\mathbf{h}$) defined by the bisector of the surface-to-light source and surface-to-viewing point vectors (shown as $\theta_b$ in Figure 5). $\theta_i$ represents the incident lighting zenith angle from the normal and $\theta_r$ represents the angle of the reflection from the normal (as defined by Nicodemus et al. [1977]). The $4\pi\alpha^2$ term of the model is used to normalize the area of the specular lobe so that as the width of the lobe is increased by the alpha parameter, the peak decreases to maintain energy conservation [Ward 1992].

Ward also developed an anistropic form of the model to account for varying degrees of surface roughness in different (azimuthal) surface directions. The formula provides roughness parameters for two perpendicular directions, $\alpha_x$ and $\alpha_y$, and incorporates the azimuthal angle of the half-vector, $\phi$, into the model to weight the two roughness parameters [Ward 1992]:
\[ \rho_{\text{brdf}}(\theta_i, \phi_i, \theta_r, \phi_r) = \frac{\rho_d}{\pi} + \rho_s \frac{1}{\sqrt{\cos \theta_i \cos \theta_r}} \frac{\exp\left(-\tan^2 \delta \left(\frac{\cos^2 \phi + \sin^2 \phi}{\alpha_x^2 + \alpha_y^2}\right)\right)}{4\pi \alpha_x \alpha_y}. \] (16)

### 2.2.3.2.1.6 Ward- Dür

Dür [2006] developed a modified form of the Ward [1992] model. An analysis by Dür revealed that for highly specular surfaces (small \( \alpha \) values and \( \rho_s = 1 \)) the specular term in the original version did not integrate to a value of 1, as is expected for physically-plausible BRDFs. The original Ward model instead was found to exhibit an angular dependence and integrate to \( \cos(\theta) \) [Dür 2006]. Dür removed the square root from the denominator of the specular term, but otherwise left the model the same as the original (anisotropic) Ward version:

\[ \rho_{\text{brdf}}(\theta_i, \phi_i, \theta_r, \phi_r) = \frac{\rho_d}{\pi} + \rho_s \frac{1}{\cos \theta_i \cos \theta_r} \frac{\exp\left(-\tan^2 \delta \left(\frac{\cos^2 \phi + \sin^2 \phi}{\alpha_x^2 + \alpha_y^2}\right)\right)}{4\pi \alpha_x \alpha_y}. \] (17)

The isotropic form of the Ward [1992] model with the Dür [2006] square root modification is:

\[ \rho_{\text{brdf}}(\theta_i, \phi_i, \theta_r, \phi_r) = \frac{\rho_d}{\pi} + \rho_s \frac{1}{\cos \theta_i \cos \theta_r} \frac{\exp\left(-\tan^2 \delta \left(\frac{\alpha^2}{\alpha_x^2 + \alpha_y^2}\right)\right)}{4\pi \alpha_x \alpha_y}. \] (18)

### 2.2.3.2.2 Modeling Mesoscale Texture

As the size of surface features increases to the mesoscale, surface features are large enough to be visually distinct and so can not be averaged into BRDF, but are too small to be efficiently modeled using polygonal geometry. Features at this scale are associated with the appearance of surface texture. Blinn [1978] developed a technique for perturbing the geometric
surface normals to add the appearance of wrinkles to otherwise smoothly-shaded geometry. Altering the normal vectors across the polygon introduces spatial variation into the results from the lighting calculation (which depends on the angle between the perturbed surface normals and lighting direction) to create the illusion of texture [Blinn 1978]. Typical implementations of Blinn’s technique represent the perturbation of the normal direction in terms of a one-dimensional height parameter, a method commonly referred to as bump mapping [Peercy et al. 1997]. Peercy et al. developed a more efficient representation for describing surface normals, referred to as normal mapping, in which the three-dimensional normal direction at each point on the surface is pre-calculated and stored as a texture.

Techniques such as bump-mapping and normal mapping model the effects of variation in surface orientation for use in the lighting calculation, but do not account for the surface shadowing that results from mesoscale geometry [Max 1988]. Max developed a mapping technique, referred to as horizon mapping, to also model information about the shadows cast by the mesoscale geometry associated with bump maps. Horizon maps store the minimum elevation angle, at each position on the surface, required to avoid occlusion by other bumps so that the horizon (or a light source) can be seen from that position [Max 1988]. The horizon must be considered for every azimuthal direction to fully model real shadowing, but in practice has been specified in eight directions to allow for the computational efficiency needed to achieve interactive rates [Sloan & Cohen 2000].

2.2.3.2.3 Modeling Shape

Several techniques have been used to model global shape in computer graphics, but traditionally the most common method has been polygonal meshes [Watt 1993]. Polygonal
meshes are typically described in terms of a list of three-dimensional (x, y, z) vertices that define a set of polygons, and in some cases, may also store information on facet or vertex normal directions [Watt 1993]. Gouraud [1971] introduced an initial method for interpolating vertex normals in rendering calculations that allowed for smooth shading of curved surfaces from polygonal geometry.

2.2.4 Illumination and Rendering

The rendering process combines illumination sources and object surface models to simulate light transport throughout a scene. A virtual image is formed from the light transport simulation by locating a virtual camera in the scene and calculating the incident radiance that will reach each pixel of the camera’s imaging plane.

2.2.4.1 The Rendering Equation

Kajiya [1986] formally defined a rendering equation to describe the general light transport problem that must be solved during the rendering process:

\begin{equation}
I(x, x') = g(x, x') \left[ \varepsilon(x, x') + \int_{S} \rho(x', x'') dx'' \right]. \tag{19}
\end{equation}

The equation is used to calculate the intensity of light, \(I(x, x')\), transported from a point \(x'\) to \(x\) during rendering (the wavelength dependency of the equation is implicit) [Kajiya 1986]. The intensity of light arriving at \(x\) from \(x'\) is dependent upon the geometry between the two points, described by the term \(g(x, x')\); the amount of light emitted at point \(x'\) that reaches \(x\), described by the emittance term \(\varepsilon(x, x')\); and the amount of light incident upon \(x'\), integrated from all
directions, that is redirected toward point \( x \) [Kajiya 1986]. The geometry term accounts for situations where there is occlusion blocking the path between the two points (i.e. \( g(x,x') = 0 \)) [Kajiya 1986]. The emittance term, \( \varepsilon(x, x') \), describes the light contribution if \( x' \) is a light source, while the \( \rho(x, x') \) term describes the light contribution if \( x' \) is a reflective or transmissive surface [Kajiya 1986]. For a reflective surface, the \( \rho(x', x'') \) term can be evaluated by integrating the BRDF function at the surface location \( x' \) over all incident directions of irradiance (all points \( x'' \)), for the outgoing reflection direction toward \( x \).

In radiometric terms, the rendering equation can be expressed as the radiance from a point \( x \) in the outgoing direction \( \Theta_o \), due to the radiance emitted by point \( x \) (\( L_e \)) or due to radiance scattered at point \( x \) [Dorsey et al. 2008]:

\[
L(x \rightarrow \Theta_o) = L_e(x \rightarrow \Theta_o) + \int_{\Theta_i \in \omega} f_r(x, \Theta_i \rightarrow \Theta_o) L(x \leftarrow \Theta_i) \cos(N_x, \Theta_i) d\omega_i
\]  

(20)

In the case of an opaque reflective surface, the \( f_r \) term represents a BRDF function that is evaluated in the outgoing direction \( \Theta_o \) for all incident lighting directions. The reflected radiance is determined by multiplying the BRDF by the incident radiance from sources in all possible directions (considering cosine falloff for the angle between lighting and surface normal, \( N_i \)) and integrating the results [Dorsey et al. 2008].

As Dorsey et al. [2008] noted, the integral of Eq. (20) is difficult to compute due to the global illumination problem. The complete solution requires that the incident radiance from all possible directions, at point \( x \), be considered. However, that incident radiance may be due to reflections from other surfaces, which in turn must be evaluated based on the radiance incident upon them (and may even include light reflected from point \( x \)) [Dorsey et al. 2008]. This
problem is simplified to the local illumination problem if only the incident radiance from direct light sources (emitters), and not other surfaces, is considered in the integral [Dorsey et al. 2008].

2.2.4.2 Rendering Methods

2.2.4.2.1 Ray-tracing for Global Illumination

Whitted [1980] developed the general ray-tracing method commonly used to account for the effects of global illumination during rendering. In Whitted’s method, light rays for each pixel of the rendered image are cast into the scene starting from the location of the virtual camera. The first intersection of each light ray with a surface is determined and the reflection (or transmission) from each of those surface points is calculated from its BRDF and the global incident illumination at that point, determined through a recursive process [Whitted 1980]. Each initial ray produces a tree that branches at each intersection with a surface until a light source is reached [Whitted 1980]. The light reaching the camera from the rendered image is estimated by traversing the tree, calculating the incident irradiance due to the preceding level, and using this information to evaluate the reflectance function at the level above until the entire tree is traversed [Whitted 1980].

The surface reflectance at each stage of the recursive tree is often estimated stochastically using Monte Carlo integration to evaluate the integral of the rendering equation on the right-hand side of Eq. (20) [Dorsey et al. 2008]. For each surface intersection, a set of $N$ incident lighting directions, $\Theta^{(k)}$, is randomly generated and the radiance leaving the surface at that point is estimated according to the formula [Dorsey et al. 2008]:
\[
\langle L(x \rightarrow \Theta_o) \rangle = \frac{1}{N} \sum_{k=1}^{N} \frac{L(x \rightarrow \Theta_{i}^{(k)}) f_r(x, \Theta_{i}^{(k)} \rightarrow \Theta_o) \cos(N_x, \Theta_{i}^{(k)})}{d(\Theta_{i}^{(k)})}
\]  

(21)

where \(d(\Theta_{i}^{(k)})\) is the probability distribution of the sampled directions.

### 2.2.4.2.2 Importance Sampling

Instead of randomly selecting the lighting directions in a Monte Carlo simulation, rendering performance can be improved by selecting the lighting directions based on their relative importance in the resulting surface reflection. Typically, the lighting directions are sampled based on the surface BRDF or on the distribution of light source intensities in the environment [Dorsey et al. 2008]. Veach and Guibas [1995] developed a multiple importance sampling method in which the results from both BRDF-based and illumination-based sampling could be combined according to a set of weighting heuristics.

### 2.2.4.2.3 BRDF-based Importance Sampling for the Ward Model

For BRDF-based sampling distributions, the expression in Eq. (21) can be simplified if a sampling distribution can be derived that approximates the BRDF of the surface, canceling the \(f_r\) in the numerator with the \(d(\Theta_{i}^{(k)})\) in term in the denominator. In the Ward [1992] model, the directional distribution of samples is dependent on the \(\alpha\) parameter, which specifies the width of the specular lobe. The importance sampling equations for the specular lobe of the isotropic Ward model, as described by Walter [2005], are:

\[
\theta_s = \arctan \left( \alpha \sqrt{-\log u} \right)
\]

(22)
where $u$ and $v$ are uniform random variables for the range $(0, 1]$. The resulting sampling directions are specified in terms of the zenith angle, $\theta_h$, and the azimuth angle, $\phi_h$, of the half-vector of the Ward model. The unit half-vector, $\mathbf{h}$, is determined from these angles according to:

$$\mathbf{h} = (h_x, h_y, h_z)$$

where

$$h_x = \cos(\phi_h) \sin(\theta_h)$$

$$h_y = \sin(\phi_h) \sin(\theta_h)$$

$$h_z = \cos(\theta_h)$$

These sampling distributions are intended to approximate the direct evaluation of the Ward specular lobe, so that ideally the only weighting function necessary would be the constant $\rho_s$. Walter demonstrated, however, that a more complex weighting function is necessary to fully match the Ward model BRDF. This weighting function is given by the ratio of the BRDF, weighted by the cosine of the incident illumination angle, to the sampling probability from importance sampling (as described by Walter [2005]):

$$w = \frac{f_r \cos(\theta_i)}{p_{ill}}.$$  \hfill (25)

As derived by Walter, the sampling probability of the illumination directions is:

$$p_{ill} = \frac{1}{4\pi \alpha^2 (\mathbf{h} \cdot \mathbf{r}) \cos^3(\theta_h) e^{-\tan^2(\theta_h)/\alpha^2}}.$$  \hfill (26)

After representing the BRDF and weight functioning in vector notation, the weighting function derived by Walter [2005] is:
\[ w = \rho_s \left( \mathbf{h} \cdot \mathbf{r} \right) \left( \mathbf{h} \cdot \mathbf{n} \right) \sqrt{\frac{\mathbf{n} \cdot \mathbf{i}}{\mathbf{n} \cdot \mathbf{r}}} \].

(27)

The importance sampling weighting function used in the object display framework is based on a similar derivation performed for the Ward-Dür BRDF model. The Ward-Dür model is a modified form of the original Ward [1992] model that has the square root removed from the specular lobe normalization [Dür 2006].

### 2.2.4.2.4 Real-time Local Rendering on the GPU

The recursion required to evaluate the rendering equation for global illumination makes ray-tracing a very computationally demanding process, so for interactive rendering, local methods that only consider the direct lighting component are often used instead [Dorsey et al. 2008]. Local rendering methods based on the projection of polygons to an imaging plane can be performed efficiently using specialized graphics hardware (the graphics processing unit, or GPU) and graphics libraries such as OpenGL [Neider et al. 1993]. The fixed OpenGL pipeline provides methods to specify polygons and light sources in a three-dimensional space; calculate the color of reflections due to direct lighting at polygon vertices; interpolate colors across surfaces; perform matrix operations to model the rotation, translation, and scaling of the polygons; and model the projective transform of a virtual camera [Neider et al. 1993]. In the final stage of rendering, the data associated with the projected polygons is transformed to the discrete pixels of an image through a rasterization process [Rost 2006]. While initially the surface reflectance calculations in the fixed pipeline were limited to a simple BRDF model similar to Blinn-Phong, the introduction of a programmable shader architecture for OpenGL
allowed for the use of more complex BRDF functions and rendering techniques [Rost 2006].
The programmable pipeline allows custom algorithms to be performed at each vertex, in a vertex
shader, or on a per-pixel basis across polygon surfaces, in a fragment shader [Rost 2006]. The
flexibility provided by the programmable pipelines allows for the use custom BRDF models in
lighting calculations and the use of specialized algorithms for per-pixel normal mapping or
shadowing [Rost 2006].

2.2.4.3 Illumination Methods

Several methods have been used to model sources of illumination in computer graphics
simulations. In traditional real-time rendering simulations, simple point lights, describing the
location and color of illumination, or directional (spot light) sources are often used [Neider et al.
1993]. In ray-tracing systems such as PBRT [Pharr & Humphreys 2010] and Radiance [Ward
1994, Ward-Larson & Shakespeare 1998], area light sources, along with the global illumination
of light reflected from other surfaces, have been used to provide more realistic renderings of
scenes. Other methods have developed to light surfaces with illumination captured from real-
world environments.

2.2.4.3.1 Environment-mapped Lighting

In many situations global illumination is impractical, so lighting techniques have been
developed to allow direct lighting calculations to produce complex, realistic surface reflections.
Blinn and Newell [1976] developed the technique of environment-mapped lighting, a method for
producing complex patterns of reflections on rendered objects using a spherical intensity map of
the lighting environment that is stored as an image texture. To render reflections at each surface
point, the specular reflection direction is used to index the image texture, parameterized in polar
coordinates, and extract the lighting intensity [Blinn & Newell 1976]. Blinn and Newell’s approach resulted in sharp, mirror-like reflections from the illumination map as would be produced by a high gloss surface. Miller and Hoffman [1984] extended the technique to allow for surfaces with other gloss levels by pre-convolving (blurring) the illumination maps with the kernel for a Phong BRDF function. Heidrich and Seidel [1998] developed a view-independent parameterization for environment maps, based on two hemispheres, to overcome limitations that made the initial environment maps only suitable for rendering from a single pre-specified viewing direction. Heidrich and Seidel [1999] combined this view-independent parameterization with a technique for pre-filtering environment maps using physically-based BRDF functions to create a real-time rendering system capable of producing realistic depictions of materials in complex lighting environments. A view-independent cube-based representation of environment maps was developed by Greene [1986], where the lighting environment was represented by six textures corresponding to the interior faces of a cube. Heidrich and Seidel [1999] did not perform real-time rendering with their approach because it was not feasible on the graphics hardware of the time, but with more recent hardware developments, real-time cubic environment map-based approaches (such as described by Rost [2006]) are now possible and commonly used.

2.2.4.3.2 Imaged-based Lighting

Debevec [1998] developed a rendering method for lighting with radiance maps captured from real-world scenes. Typical environment-mapped lighting techniques at the time were not able to represent the full range of luminances found in real-world settings. Using the method developed by Debevec and Malik [1997], Debevec was able to capture the wide range of radiance information in real-world environments by taking a series of photographs at different
exposure settings and merging them to create a high-dynamic range image. The series of photographs were taken of mirrored spheres to create an omni-directional radiance map, referred to as a lighting probe, of the entire environment [Debevec 1998].

2.2.4.3.3 Filtered Importance Sampling of Image-based Lighting

The filtered importance sampling method was developed by Colbert et al. [2006] and Colbert & Křivánek [2007] to support the use of image-based environmental illumination along with realistic BRDF models in real-time rendering applications. In real-time rendering applications, it is often only possible to randomly sample a small number of illumination directions, and if these are not well distributed, it can lead to error in the estimates of the reflection. Colbert and Křivánek addressed this issue by using a deterministic sampling technique. In Colbert and Křivánek’s sampling method, instead of randomly generating sampling directions, the \((u, v)\) uniform random variables in the importance sampling equations are generated using a fixed quasirandom sequence. This provides well-distributed and consistent pixel-to-pixel sampling with a limited sampling size. Using the Hammersley sequence, the \((u, v)\) pair is generated as [Colbert & Křivánek 2007]:

\[
\begin{align*}
u &= \frac{i}{N}, \\
\phi_{inv\_bin}(i) &= \Phi_{inv\_bin}(i) \quad (28)
\end{align*}
\]

for \(i = 1:N\), where the number of samples, \(N\), is a power of 2 and the corresponding variable \(v\) is generated by taking the inverse binary value of \(i\). The inverse binary values are calculated by taking the integer value of \(i\), expressing it as a binary number, reversing the digits around the decimal point, and then expressing the result as a base-10 number [Colbert & Křivánek 2007]. For example, to find the inverse binary of \(i = 3\), the value 3 is first converted to its binary value
The inverse binary value is found by reversing the digits $(0.11)_2$ and the final value is found by converting back to decimal: $(0.11)_2 = \frac{1}{2} + \frac{1}{4} = 0.75.$

Sampling with the Hammersley sequence creates a well-distributed set of directions, but introduces aliasing, which can result in a specular reflection that appears as a disjoint pattern instead of a single continuous blurred element. Colbert and Křivánek [2007] addressed this type of aliasing by filtering the illumination map based on the probability density function of the illumination sampling. They used OpenGL’s built-in mipmapping capabilities (creating a pyramid of lower resolution versions of an image by successively halving the pixels along an edge using a box filter), so that the filtering could be performed efficiently in hardware. In the method, the number of pixels to average with the filter is given by the ratio of the solid angle of the sample to the solid angle of a pixel of the full-resolution illumination map [Colbert & Křivánek’s 2007]:

$$N_{\text{ave}} = \frac{\Omega_s}{\Omega_p}$$

where the solid angle of a pixel of the full resolution map is calculated by:

$$\Omega_p = \frac{d(i)}{w \cdot h}$$

and the solid angle of a given sample is:

$$\Omega_s = \frac{1}{N} \frac{1}{pdf(u,v)}$$

In these formulas, $w \cdot h$ is the total number of pixels in a map of one hemisphere (the product of the width and height in pixels) and $d(u)$ is a distortion factor that corrects for any geometric distortion in the map in the sampling direction specified by $u$. Křivánek & Colbert [2008]
provide a formal mathematical analysis of this technique, where they describe how the distortion factor is the Jacobi determinant of the mapping from the image space of the illumination map to the unit spherical space used for lighting calculations in terms of solid angle.

When using a box-filter implemented in OpenGL, the mipmap level of the filtered pyramid (LOD) that corresponds to averaging $N_{\text{ave}}$ pixels is calculated by the formula [Colbert & Krivánek 2007]:

$$LOD = \max \left( \frac{1}{2} \log_2 \left( \frac{\Omega_s}{\Omega_p} \right), 0 \right).$$

(32)

2.2.4.3.4 Median Cut Illumination with Image-based Lighting

An alternative method for rendering with image-based illumination maps is to reduce the image map to a representative set of point lights. If surfaces are sufficiently diffuse, a point light approximation can produce results that are visually similar to more complex rendering processes. Debevec [2005] developed a median cut algorithm method for dividing complex illumination images into a number of discrete regions, which could then be represented by point lights (the method is described in detail in the HDR imaging textbook by Reinhard et al. [2010]). In Debevec’s method, all the pixels in the illumination image map are initially assigned to a single region. In an iterative process, each region from the previous step is subdivided into two new regions such that each has the same summed light energy, and the subdividing continues until a specified number of regions is reached [Debevec 2005]. For each final region, a point light with a light intensity equal to the sum of all pixels in the region is placed at the centroid location of that region [Debevec 2005]. Viriyothai and Debevec [2009] developed a modified version of the
algorithm where the regions are divided to minimize a lighting variance metric that also considers the distance from the centroid of the region to the pixels locations it would represent:

\[ V = \sqrt{\sum_{p \in \text{Region}} L_p d_p^2} \]  

(33)

where \( L_p \) is the \( p^{th} \) pixel of a region and \( d_p \) is the distance from that pixel to the centroid of the region.

2.2.4.3.5 Light Fields

As Unger et al. noted [2003], the radiance map probes from image-based lighting are limited in that they only provide accurate lighting for an object at a single position (the location of the mirror ball during capture), and do not account for local variations in the incident lighting on objects at different spatial locations across the scene. To overcome this limitation, techniques have been developed to capture more comprehensive light fields from real-world scenes for use as rendering light sources [Unger et al. 2003].

Levoy and Hanrahan [1996] defined the light field as a four-dimensional function describing radiance in terms of a two-dimensional direction and two-dimensional position. Levoy and Hanrahan developed their 4D light field representation as a simplification of the plenoptic function proposed by Adelson and Bergen [1991]. The full plenoptic function, \( P \), is a seven-dimensional comprehensive function for describing the structure of light in a scene [Adelson & Bergen 1991]:

\[ P = P(\theta, \phi, \lambda, t, V_x, V_y, V_z) \]  

(34)
It describes the intensity of light in terms of its two-dimensional direction of travel, \((\theta, \phi)\), at every possible three-dimensional viewing position in space, \((V_x, V_y, V_z)\), as a function of time \(t\) and wavelength \(\lambda\) [Adelson & Bergen 1991]. In free space, Levoy and Hanrahan demonstrated that this can be simplified to a four-dimensional function (for a given wavelength and time). The four dimensions correspond to a 2D position and 2D direction, but Levoy and Hanrahan found it more convenient to parameterize them as two sets of 2D points, \((x, y)\) and \((s, t)\), that describe the positions through which light rays intersect two planes. Gortler et al. [1996] developed the Lumigraph concept based on a similar 4D representation. In the full Lumigraph representation, 4D light fields of the enclosed scene are specified for each of the six outer faces of a cube [Gortler et al. 1996]. From a full Lumigraph, it is then possible to determine the image of the enclosed scene for any possible external viewing point in space [Gortler et al. 1996]. Levoy and Hanrahan [1996] and Gortler et al. [1996] used image-based rendering techniques to generate new images for each view, interpolating between the captured views of the scene stored in the light field representation. Unger et al. [2003] incorporated the light field concept into a global illumination framework using light fields instead as the scene illumination and allowing synthetic objects to be rendered into the scenes. Cossairt et al. [2008] developed a system capable of real-time capture and rendering with light fields.

2.2.4.4 Summary

Object surface models are used to represent the underlying physical mechanisms that are related to the appearance attributes of color, gloss, and texture. In the rendering process, surface models are combined with virtual illumination to simulate the surface reflections, as seen by a virtual camera. These reflections depend upon geometric properties of the surface, represented
by BRDF models that describe directional reflectance and surface texture models that describe spatially-varying surface orientation and shadowing effects. The reflections also depend on the spectral properties of the surface, represented by diffuse reflectance parameters specified in multiple color channels. Finally, the reflections depend on the illumination model and type of light transport simulation. The result of the rendering process is a 2D radiometric image of the light reflected from the virtual object surface, as observed at a certain point in space. The final stage in the process is to convert the radiometric image results to a form that can ultimately be presented on a display screen.

2.3 Display Screen Modeling

To effectively reproduce the appearance of real objects, the object display system needs both the ability to accurately simulate the reflections from a real surface and the ability to convey the simulation results through the display screen. In Section 2.2, methods were described for modeling materials and estimating their surface reflections in lighting simulations. In this section, the methods for modeling the properties of the display screen will be described. The appearance attributes of real objects depend upon both the color and geometric distribution of light reflected from the surface. To faithfully reproduce appearance attributes on a display screen, both the color and geometric distribution of light emitted (as well as any reflected) from the display screen must be considered.
2.3.1 *Color Display Technologies*

There are several common display technologies used to produce color images based on the additive mixing of trichromatic primaries [Hunt 2004]. Due to the trichromatic nature of color vision, a display device can produce a range of color stimuli by combining light in varying amounts from three colored primaries [Hunt 2004].

2.3.1.1 *Cathode-ray Tubes (CRTs)*

Cathode-ray tube displays (CRTs) combine an electron emitting source and a screen coated with red, green, and blue phosphors [Hunt 2004]. Using magnetic fields, an electron beam is directed across the screen in a sequential scan as the voltage is varied to modulate the amount of light emitted from each type of color phosphor [Hunt 2004].

2.3.1.2 *Liquid Crystal Displays (LCDs)*

Instead of color phosphors, liquid crystal display (LCD) screens produce trichromatic color output using a grid of red, green, and blue filters and an addressable array of liquid crystals that modulate the amount of light passing through the three color filters for each pixel [Hunt 2004]. The liquid crystals are placed between two crossed polarizers that block transmission of a backlight through the screen [Hunt 2004]. The liquid crystals have the ability to rotate the polarization of the light to allow it to pass through the second polarizer [Hunt 2004]. The orientation of the crystals can be varied as a function of the voltage applied, allowing the amount of light transmission through each of the red, green, and blue filters of a pixel to be controlled [Hunt 2004].
2.3.1.3 Digital Light Projection (DLP)

Digital Light Projection (DLP) devices produce color images by combining a digital micro-mirror array (DMA) and a spinning color filter wheel (as described in an article by [Wyble & Rosen 2006]). The color filter wheel contains sections with red, green, and blue filters (as well as unfiltered section in many cases) that act as the color primaries [Wyble & Rosen 2006]. As the wheel rotates through a cycle, the spatially-addressable DMA varies the amount of light directed toward the output optics during the time that each of the color filters is in place, modulating the amount of light for each primary at each pixel position [Wyble & Rosen 2006]. In systems with an unfiltered section, the unfiltered portion acts as a fourth (white) primary, requiring the use of more complex color management techniques [Wyble & Rosen 2006].

2.3.1.4 Organic Light-emitting Diode (OLED) displays

Organic light-emitting diodes (OLEDs) are being used to develop a new generation of display screens [Forrest et al. 2000]. OLEDs are organic materials that luminesce when a voltage is applied and can be formed into addressable grids to serve as spatially-varying light sources for use in displays [Forrest et al. 2000]. Trichromatic color reproduction can achieved by using three types of OLEDs, each tuned to emit red, green, or blue light, or by combining white OLEDs with the red, green, and blue filter arrays typically found in LCD screens [Forrest et al. 2000].

2.3.2 Display Properties

There are several display properties that must be considered in an object display system. These include typical factors considered in color reproduction, such as the color gamut and
dynamic (luminance) range. Additionally, geometric attributes of the display screen, such as changes in color and intensity as a function of viewing angle and the directional distribution of light reflecting from the front surface of the screen must be considered.

2.3.2.1 Gamut

The *gamut* of a display device refers to the range of colors that it can produce [Fairchild 2005]. Though often shown as a triangle in chromaticity space, Fairchild noted that a gamut should be expressed in a three dimensional color space, as two-dimensional chromaticity diagrams lack critical information about the luminance dimension. A three dimensional color gamut may be represented by a volume in the CIELAB space, or as Fairchild recommended, may be specified in a color appearance space that describes color in terms of lightness, chroma, and hue or the absolute dimensions of brightness, colorfulness, and hue.

2.3.2.2 Dynamic Range

Dynamic range refers to the minimum and maximum luminance that can be reproduced by a display system. The range of a display is often expressed as a contrast ratio determined by dividing the luminance of the white level by the luminance of the black level [Boynton & Kelley 1996]. As reported by Seetzen et al. [2004], a typical LCD display has a maximum luminance of approximately 300 cd/m² and a contrast ratio of 300:1. Using a custom-built high-dynamic range display that combined an LCD panel with a projector-based backlight, Seetzen et al. were able to achieve maximum luminances of up to 8500 cd/m² and contrast ratios over 50,000:1.
2.3.2.3 Viewing Angle

The color properties of a display screen can change as it is viewed from different directions. This can be a particular problem with certain types of LCD displays [Cheng 2007]. The viewing angle of a display is often defined as maximum angle at which the display contrast ratio maintains a value above a specified threshold (often 10) [Cheng 2007]:

$$VA = \left\{ \theta \mid \frac{L_{\max}(\theta)}{L_{\min}(\theta)} \geq 10 \right\}$$

(35)

As Cheng notes, this definition is quite limited. More comprehensive colorimetric measurement methods have been performed to fully characterize the changes in chromaticity and display luminance as a function of screen viewing angle [Li et al. 2004; Cheng 2007].

2.3.2.4 Screen Surface Reflection

The reflective properties of the screen surface are an important consideration when displays are viewed in environments with ambient lighting, as the reflected light will lead to glare superimposed on the displayed image [Schenkman et al. 1999]. CRT displays typically exhibit a combination of mirror-like specular reflection and diffuse reflection, while matte LCD screens typically exhibit haze-type reflection [Becker 2006]. A screen with a high degree of specular reflectance produces reflections that vary in intensity across the surface and results in sharp, clearly delineated virtual images of objects and light sources [Schenkman et al. 1999]. Diffuse reflectance leads to uniform scattering of light over the surface of the screen [Schenkman et al. 1999]. The haze reflectance associated with LCD screens is centered around the specular
direction, but is spread over a range of angles so that it does not produce distinct images of environmental lighting [Kelley et al. 1998].

2.3.3 Display Modeling Methods

2.3.3.1 Colorimetric Characterization

In order to produce a specified set of color stimuli on a display device, is it necessary to model the color reproduction of the display system through a colorimetric characterization process. This process provides the means to relate the digital input values, controlled by the user, to the physical luminance and chromaticity of the light emitted by the display screen.

2.3.3.1.1 Modeling Approach

Berns [1997] described the general two stage process used to model the color produced by a display system. The first stage models the conversion from user-controlled input values ($d_r$, $d_g$, $d_b$ digital counts) to a set of R, G, B radiometric scalars on the red, green, and blue primary channels. The linear stage of the model then provides a matrix transformation to convert from the R, G, B radiometric scalars to XYZ tristimulus values. The channels in a display system must meet a set of additivity and scalability requirements in order for a linear transformation from R, G, B radiometric scalars to XYZ tristimulus values to be valid [Berns 1997]. To meet the additivity requirement, the total radiometric output of the system must equal the sum of the scaled primaries [Berns 1997]:

$$L_{\lambda,mix} = RL_{\lambda,r, max} + BL_{\lambda,b, max} + GL_{\lambda,g, max}$$  \hspace{1cm} (36)
where $L_{\lambda,\text{mix}}$ represents the total radiance produced by the display and $L_{\lambda,r\text{ max}}, L_{\lambda,b\text{ max}}, L_{\lambda,b\text{ max}}$ represent the maximum radiant output of the red, green, and blue channels, respectively.

Channel scalability requires that the output of a channel, $L_{\lambda,\text{ch}}$, can be described by applying a multiplicative scaling factor to $L_{\lambda,\text{ch, max}}$, the maximum output of that channel [Berns 1997]:

$$L_{\lambda,r} = RL_{\lambda,r\text{ max}}$$
$$L_{\lambda,g} = GL_{\lambda,g\text{ max}}$$
$$L_{\lambda,b} = BL_{\lambda,b\text{ max}}$$

If the channel additivity and scalability requirements are met for a three primary display system, then the transformation from radiometric scalars to XYZ tristimulus values can be described by a linear 3x3 transformation matrix [Berns 1997]:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} X_{r,\text{max}} & X_{g,\text{max}} & X_{b,\text{max}} \\ Y_{r,\text{max}} & Y_{g,\text{max}} & Y_{b,\text{max}} \\ Z_{r,\text{max}} & Z_{g,\text{max}} & Z_{b,\text{max}} \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

where $\text{XYZ}_{r,\text{max}}, \text{XYZ}_{g,\text{max}}, \text{XYZ}_{b,\text{max}}$ represent the tristimulus values of the primary channels at their maximum output levels.

### 2.3.3.1.2 Non-linear Modeling of the OETF

The non-linear relationship between the input digital counts on the channels and the radiometric output from the display screen is described by the optoelectronic transfer function (OETF) of each display primary channels. This non-linear relationship was typically modeled for CRT displays using a gain, offset, gamma model for each channel (shown for the red channel) [Berns et al. 1993]:

$$\frac{R}{R_{\text{max}}} = \left[ k_{g,r} \left( \frac{d_r}{2^N - 1} \right) + k_{o,r} \right]^{\gamma_r}$$

\[39\]
Because the OETF of an LCD system is not guaranteed to follow any specific functional form, Fairchild and Wyble [1998] developed a characterization method for LCDs using one-dimensional look-up tables (LUTs) for each channel to provide adequate flexibility in modeling the shape of the transfer function. The approach of using three 1D LUTs was used by Day et al. [2004] in their characterization method. The relationship between the radiometric scalars R, G, B and the digital counts \( d_r, d_g, d_b \) for the red, green, and blue channels is described by [Day et al. 2004]:

\[
R = LUT(d_r) \\
G = LUT(d_g) \\
B = LUT(d_b) \\
0 \leq R,G,B \leq 1
\]  

(40)

2.3.3.1.3 Modeling the Linear Stage

The 3x3 matrix equation of Eq. (38) may not adequately describe the transformation from RBG scalars to XYZ tristimulus values if there is emission by the display at its black level [Berns et al. 2003]. This is commonly the case with LCD displays because the liquid crystals modulating the light emission of the display never completely block light transmission [Berns et al. 2003]. The black-level flare in an LCD system results in a violation of the scalability requirement for the individual channels and introduces error into the calculation of XYZ tristimulus values using a 3x3 transformation matrix [Berns et al. 2003]. Fairchild and Wyble [1998] subtracted the measured black-level from the measured primaries when calculating the 3x3 matrix in their LCD model. Katoh et al. [2001] developed an expanded form of Eq. (38) to separate the effects of flare and restore channel scalability when modeling CRTs. This method
was incorporated by Berns et al. [2003] and Day et al. [2004] into modeling approaches for LCD displays. The Day et al. approach uses an optimized 3x4 transformation matrix to account for the effects of the black level emission of the display. The first three columns contain the black level-corrected tristimulus values of the red, green, and blue channels at maximum output, and the fourth column is used to add the XYZ tristimulus values of the black level into the final XYZ result [Day et al. 2004]:

\[
\begin{bmatrix}
X \\
Y \\
Z
\end{bmatrix}
= \begin{bmatrix}
X_{r,\text{max}} - X_{k,\text{min}} & X_{g,\text{max}} - X_{k,\text{min}} & X_{b,\text{max}} - X_{k,\text{min}} & X_{k,\text{min}} \\
Y_{r,\text{max}} - Y_{k,\text{min}} & Y_{g,\text{max}} - Y_{k,\text{min}} & Y_{b,\text{max}} - Y_{k,\text{min}} & Y_{k,\text{min}} \\
Z_{r,\text{max}} - Z_{k,\text{min}} & Z_{g,\text{max}} - Z_{k,\text{min}} & Z_{b,\text{max}} - Z_{k,\text{min}} & Z_{k,\text{min}}
\end{bmatrix}
\begin{bmatrix}
R \\
G \\
B \\
1
\end{bmatrix}
\]  

(41)

where \(X_{k,\text{min}}, Y_{k,\text{min}}, \) and \(Z_{k,\text{min}}\) represent the tristimulus values of the black level. Day et al. also described a reversed matrix transformation to calculate a set of RGB scalars from a set of XYZ tristimulus values:

\[
\begin{bmatrix}
R \\
G \\
B
\end{bmatrix}
= \begin{bmatrix}
X_{r,\text{max}} - X_{k,\text{min}} & X_{g,\text{max}} - X_{k,\text{min}} & X_{b,\text{max}} - X_{k,\text{min}} \\
Y_{r,\text{max}} - Y_{k,\text{min}} & Y_{g,\text{max}} - Y_{k,\text{min}} & Y_{b,\text{max}} - Y_{k,\text{min}} \\
Z_{r,\text{max}} - Z_{k,\text{min}} & Z_{g,\text{max}} - Z_{k,\text{min}} & Z_{b,\text{max}} - Z_{k,\text{min}}
\end{bmatrix}^{-1}
\begin{bmatrix}
X - X_{k,\text{min}} \\
Y - Y_{k,\text{min}} \\
Z - Z_{k,\text{min}}
\end{bmatrix}
\]  

(42)

2.3.3.1.4 Optimizing the Model

In the Day et al. [2004] approach, the starting values are based on the physical measurements of the primaries at maximum output and the black level at the system’s minimum output level (as in the Fairchild and Wyble [1998] method) to estimate the 3x4 matrix transform of Eq. (41). To improve the model performance, Day et al. performed a constrained non-linear optimization to select an optimum black level and an optimal set of black level-corrected
tristimulus values for the 3x4 matrix to minimize the average CIEDE2000 color differences between the model predictions and a set of measured tristimulus values.

The overall characterization approach described by Day et al. [2004] was the primary method used for characterizing displays in this dissertation. As noted by Day et al., their approach is based on the general model form developed by Fairchild and Wyble [1998]. For conciseness, the characterization method will be referred to as the Day et al. approach throughout the dissertation.

### 2.3.3.2 Viewing-angle Characterization and Correction Modeling

Li et al. [2004] developed a method for modeling viewing-angle color shifts and applying corrections in real-time. They performed a set of display measurements to characterize the change in display gamma, white point chromaticity, and chromaticity of the RGB primaries at a range of viewing angles. Based on the characterization model, Li et al. proposed an active correction method using a video camera to track the location of the viewer and estimate the viewing direction in real-time. They described a simple tone correction method for applying a gamma compensation factor to correct the characterized gamma of the detected off-axis viewing angle back to the on-axis gamma of 2.2. Li et al. also proposed a more complex tone correction method based on the use of three look-up tables. In the first look-up table, the input digital counts were mapped to the luminances that would be produced on-axis. In the second, these luminance values were linearly rescaled to the luminance range possible at the off-axis viewing angle. In the final LUT, an inverse look-up was performed to determine the digital counts at the off-axis viewing angle associated with the rescaled luminance values. Li et al. also described a
compensation method for the chromaticity shift in the primaries. The method used the measured XYZ values of the display RGB primaries on-axis (subscripts \( r0, g0, b0 \)) and the measured XYZ of the primaries at the off-axis angle (subscripts \( rx, gx, bx \)) to calculate an adjusted set of radiometric scalars for the detected viewing angle:

\[
\begin{bmatrix}
R_x \\
G_x \\
B_x
\end{bmatrix} = \frac{Y_{rx} + Y_{gx} + Y_{bx}}{Y_{r0} + Y_{g0} + Y_{b0}} \begin{bmatrix}
X_{rx} & X_{gx} & X_{bx} \\
Y_{rx} & Y_{gx} & Y_{bx} \\
Z_{rx} & Z_{gx} & Z_{bx}
\end{bmatrix}^{-1} \begin{bmatrix}
X_{r0} & X_{g0} & X_{b0} \\
Y_{r0} & Y_{g0} & Y_{b0} \\
Z_{r0} & Z_{g0} & Z_{b0}
\end{bmatrix} \begin{bmatrix}
R_0 \\
G_0 \\
B_0
\end{bmatrix}
\]  

(43)

where \( R_0, G_0, \) and \( B_0 \) represent target radiometric scalars for on-axis viewing and \( R_x, G_x \) and \( B_x \) represent the corrected radiometric scalars at the off-axis viewing angle.

Cheng [2007] developed additional methods for correcting for viewing-angle based color dependencies. Cheng performed comprehensive colorimetric measurements of the display output as a function of viewing angle using a ConoScope™, an imaging device capable of measuring color from directions over the entire hemisphere simultaneously. Cheng then employed a camera-based tracking methodology similar to Li et al., but incorporated the use of an adjustable backlight in the compensation algorithm to maintain the absolute maximum luminance of the display as it was viewed from different directions.

### 2.3.3.3 Modeling Surface Reflectance of the Display Screen

Initial methods for characterizing the surface reflectance of displays were based on a set of three measurement intended to provide an estimate of the diffuse and specular reflectance properties of the screen (as described in a review by Becker [2006]). The measurement geometries, specified by ISO9241-7 and ISO13406-2, include a diffuse measurement \((R_d)\), a specular measurement with a 1° light source aperture \((R_s1)\), and a specular measurement with a
$15^\circ$ light source aperture ($R_{s15}$) [Becker 2006]. The diffuse coefficient ($R_d$) is estimated by lighting the screen with two area sources $30^\circ$ from normal and measuring the reflected luminance with a photometer normal to the screen, which is then compared to the luminance from a diffuse reflectance standard [Umezu et al. 1998]. The specular coefficients ($R_{s1}$ and $R_{s15}$) are estimated based on measurements taken with the photometer and light source each $15^\circ$ from the normal on opposite sides and are calculated according to the ratio [Umezu et al. 1998]:

$$R_s = \frac{L_S}{L_A} \quad (44)$$

where $L_S$ is the luminance of the reflection and $L_A$ is the luminance of the light source. Using Kubota’s variable aperture method, the coefficients for the $1^\circ$ aperture light source and $15^\circ$ aperture light source are then compared to provide an estimate of haze for LCD panels [Becker 2002]. A perfect specular mirror will produce the same reflected luminance regardless of aperture size, while the luminance reflected from a perfect diffuser will increase linearly with aperture due to the increased integrated irradiance on the surface from the larger source [Becker 2002]. Assuming each source has the same constant luminance, the degree to which the reflected luminance for the wider $15^\circ$ aperture light source exceeds the $1^\circ$ thus provides an indication of the spread of the specular lobe [Becker 2002]. As reported by Becker [2006], typical matte LCD displays exhibit specular $R_{s1}$ reflectance between 0.0003 and 0.001 (compared to 0.04 for glass), $R_{s15}$ reflectance of 0.02 to 0.08, and diffuse reflectance between 0.005 and 0.013.

For a more complete characterization of the surface reflectance properties, BRDF estimation methods have also been applied to display screens [Kelley et al. 1998]. This type of
representation is necessary to fully account for the complexity of the haze reflection associated with flat panel displays [Kelley et al. 1998]. Kelley et al. developed a BRDF model for display screens consisting of a diffuse component \( D \), specular component \( S \), and haze component \( H \):

\[
B = D + S + H
\]

(45)

where

\[
D = \frac{\rho_d}{\pi}
\]

(46)

\[
S = 2\rho_s \delta\left(\sin^2 \theta_r - \sin^2 \theta_i\right)\delta\left(\phi_r - \phi_i \pm \pi\right)
\]

(47)

and \( \rho_d \) is the diffuse reflectance parameter, \( \rho_s \) is the specular reflectance parameter, \( \theta_i \) and \( \theta_r \) are the incident and reflected zenith angles, and \( \phi_i \) and \( \phi_r \) are the azimuth angles. The diffuse component is expressed in a traditional Lambertian form and specular component expressed in terms of perfect mirror reflection (using delta functions). Kelley et al. made the simplifying assumption that displays are typically viewed in a narrow cone of angles around the surface normal, allowing for a reduced representation, \( B(\theta_i, \phi_i, 0, 0) \), parameterized only in terms of the incident illumination angles. Assuming isotropic reflection, the azimuth angle can also be discounted leading to a parameterization \( B(\theta) \) based on the zenith angle of incident illumination [Kelley et al. 1998]. Using this simplified parameterization, Kelley et al. proposed two functions to describe the haze component:

\[
H(\theta) = \frac{h}{1 + |\theta/w|^n + b|\theta/u|^n}
\]

(48)

\[
H(\theta) = h\left[\frac{1 - b}{1 + |\theta/w|^n} + \frac{b}{1 + |\theta/u|^n}\right]
\]

(49)
where $h$ is the haze peak value, $w$ and $u$ represent haze lobe widths, and $b, m, n$ are haze shape parameters.

Several measurement techniques have been proposed for estimating the BRDF of display surfaces. Kelley et al. [1998] suggested the use of a small aperture annular light source and an imaging detector, normal to the surface, to estimate the three components of the BRDF model described in Eq. (45). Kelley et al. proposed that the diffuse component could be estimated from measurements taken normal to the surface with the light source at a wide angle. With the light source placed normal to the surface, the annular shape of the light (specular center excluded) was intended to allow for the estimation of the haze peak reflectance and haze profile from the detector image [Kelley et al. 1998]. Becker [2004] noted that the detector image from the Kelley et al. technique could not be used to directly estimate the BRDF profile, but developed a method to relate the point spread function of the detector image to the surface BRDF. Becker [2002] proposed the use of the variable aperture light source method to estimate the BRDF of the screen surface. Instead of taking measurement at just two angles as in the $R_{s1}$ and $R_{s15}$ method, Becker suggested taking measurements for several aperture sizes between 1° and 15° and relating the curve shape of $R_s$ as a function of angle to the BRDF curve shape. Becker [2006] provided a review of additional methods for estimating screen BRDF, including the use of a conoscopic imaging device with a collimated light source through the lens, and mechanical gonio approaches where a small light source is physically moved relative to the screen to provide a range of illumination directions for reflectance measurement.
2.4 Interactive Display Systems for Computer Graphics Simulations

The goal of the object display system is to reproduce the appearance of a real object at the location of the display screen in a way that is consistent with the real lighting environment in which the display is situated. This goal is made more challenging by the fact that when a real object moves relative to lighting or the viewer moves relative to the object, the patterns of surface reflections are affected and so, in most cases, the appearance can not be adequately reproduced by a static surface representation. To meet the goals of the object display, the material appearance modeling component from Section 2.2 and the display modeling component from Section 2.3 must be used within an interactive system that can sense the relation of the display to the lighting environment, the relation of the viewer to the display, and incorporate this information into the rendering simulation and eventual display output. These types of capabilities have been explored in the computer graphics literature for applications in augmented and virtual reality. This section provides an overview of the tracking and display technologies that can be used to enable more natural forms of interaction with computer graphics simulations and will describe previous research on interactive and light-sensitive display systems that are related to the object display.

2.4.1 Augmented and Virtual Reality

Computer graphics simulations are often displayed on standard computer screens and manipulated using indirect input devices, like a mouse or joystick. However, there has been an interest in developing more natural ways to view and interact with virtual content since
Sutherland’s [1968] pioneering work on see-through head-mounted displays (HMDs) in the 1960’s. The HMD created by Sutherland was an optical device worn by the user that allowed virtual content to be overlaid on the user’s view of the real world and included orientation and position tracking capabilities to allow the correct perspective view of virtual objects to be calculated for movements of the user’s head. Sutherland’s work is considered to be the first example of augmented reality [Bimber & Raskar 2005]. Augmented reality refers to computer graphics simulations that combine virtual content with the real world, while the term virtual reality also includes simulations in which the rendered view is composed entirely of an artificial environment [Bimber & Raskar 2005]. The object display system attempts to present the virtual object shown onscreen as a part of the real environment in which it is situated and can be considered as a form of augmented reality.

Milgram et al. [1994] described virtual reality and augmented reality in terms of a mixed reality continuum, with complete immersion in a synthetic virtual environment at one extreme and viewing of the physical world at the other, and they developed a taxonomy to aid in classifying different types of mixed reality. When differentiating between mixed reality systems, Milgram et al. identified several key properties to be considered, such as the degree of reality, the directness of viewing, and the level of immersion. The degree of reality property relates to whether the view is primarily of the real world, with virtual content added, or is primarily a virtual environment, with some elements of the real world included [Milgram et al. 1994]. Milgram’s directness factor relates to whether the view of the real world component is direct (as in a see-through HMD) or is scanned and re-displayed (as in a video stream on a monitor). The immersion property relates to whether the user feels present in the depicted scene and depends
on factors such as whether the user is provided with an egocentric or exocentric view [Milgram et al. 1994].

At the time of Milgram’s classification scheme, technologies for mixed reality were based primarily on superimposing virtual content on the view through a see-through HMD or on a video stream displayed on a standard computer monitor. Raskar et al. [1998b] made a significant conceptual contribution and provided a new direction for the field with the conception of *spatially augmented reality*. Instead of overlaying virtual content on a view of the world, Raskar et al. suggested that virtual content be integrated directly into the real physical space of the observer by projecting virtual content onto real objects or by embedding display screens in the user’s environment.

### 2.4.2 Components for Augmented Reality

Bimber and Raskar [2005] identified three main components that are integral to developing an augmented reality system: tracking and registration capabilities, display devices, and real-time rendering. An overview of tracking and display technologies is provided in the following sections, and methods for real-time rendering are described in Section 2.2.4.

#### 2.4.2.1 Tracking and Registration

The tracking and registration component is responsible for accounting for movement by the user, the display, or objects in the environment so as to maintain the 3D spatial registration between the virtual and real components of the scene [Bimber & Raskar 2005]. Bimber and Raskar emphasized its importance for producing convincing augmented reality simulations but noted that robust, accurate real-time tracking is still a difficult problem. Several different approaches have been used for tracking the movement of observers and objects.
Rolland et al. [2001] reviewed tracking technologies and identified five principal types of systems: mechanical linkages, time/frequency (time-of-flight), direct field sensing (magnetic), spatial scanning (optical), and inertial sensing (gyroscopes, accelerometers). Early tracking techniques, such as those used by Sutherland [1968] were mechanically-based. In Sutherland’s system, a mechanical arm with multiple joints was attached to the HMD and as the user moved, a set of digital position encoders in the joints were used to provide six-degree-of-freedom information on the user’s position and orientation. Time-of-flight systems use a set of receivers to determine the position of ultrasonic or pulsed laser diode emitters that are attached to the tracking target [Rolland et al. 2001]. The emitters are activated sequentially, the time to reach each receiver is used to estimate its distance from emitter, and the location of the target is triangulated from the emitter-receiver distance pairs [Rolland et al. 2001]. Magnetic field (direct) sensing systems estimate the position and orientation of targets by measuring differences in flux induced in a receiver element as it is moved relative to the magnetic field generated by an emitter coil [Rolland et al. 2001]. The flux induced in the receiver depends on the distance and orientation relative to the emitter, and by using a set of three orthogonal emitters and a set of three sensors, the orientation of the target can be determined in six degrees-of-freedom [Rolland et al. 2001]. Optical spatial scanning systems use one or more video cameras to track a set of features and estimate the target position and orientation from the location of the features in the images [Rolland et al. 2001]. The features to be tracked may be identified with special markers, or in the case of marker-less techniques, machine-vision pattern-matching may be used to detect natural features of a target (e.g. facial features) [Bimber & Raskar 2005]. In multiple camera (multiscopy) systems, the corresponding feature points in the set of images can be used to triangulate positions, while in a single camera system, known size information about a pattern of
features can be used to estimate target distances [Rolland et al. 2001]. Inertial systems use accelerometers or gyroscopes to track the orientation or position of a target [Rolland et al. 2001]. In gyroscopic systems, the orientation of the target can be determined with sensors that measure the rotation in the external target frame relative to the constant reference axis provided by a spinning wheel [Rolland et al. 2001]. Accelerometers use a mass attached to a force sensor (such as a piezo-electric crystal) to estimate linear acceleration in one dimension and can combined to form an orthogonal tracking system [Rolland et al. 2001].

2.4.2.2 Display Modalities

Various types of technologies, including head-mounted displays (HMDs), standard computer monitors, and digital video projectors, have been used as the display component for augmented and virtual reality simulations. Head-mounted displays are worn by the user and use half-silvered mirrors or beam-splitters to allow the user to see both the real world and the virtual content from small display screens [Chung et al. 1989]. Standard computer monitors have been used for a type of “window-on-the world” augmented reality where video from a real-world scene is displayed onscreen along with superimposed virtual content [Milgram et al. 1994]. Digital video projectors projecting on screens or directly on real objects in the scene have also been used extensively in virtual and augmented reality applications [Cruz-Neira et al. 1993, Raskar et al. 2003]. In the object display system, a standard (flat-panel) computer monitor is used, but instead of adopting a window-on-the-world representation, the display itself is used to simulate a physical object in the real environment of the viewer.
2.4.3 Integrated Display Systems for Mixed Reality

A range of mixed reality display systems have been developed to allow users to interact more directly with virtual content or augment views of the real world. These systems represent a merging of concepts and technologies from the fields of human-computer interaction (HCI), computer graphics, and color image reproduction. From the HCI domain, mixed reality systems incorporate the concept of “tangible bits” [Ishii & Ulmer 1997] and tangible computing [Buxton 2008] allowing users to interact with computer-generated virtual content using natural, direct forms of interaction such as moving or grasping real-world objects and controls. These interfaces have been merged with techniques from computer graphics for simulating the appearance of virtual objects and combined with novel uses of color display technologies to produce systems that can augment the way a user experiences the real world.

2.4.3.1 Augmented Reality Systems

The earliest interactive systems for integrating virtual imagery into real world scenes were created by combining see-through head-mounted displays with six-degree-of-freedom trackers. Sutherland [1968] developed the first such system by combining a mechanical tracking device with a head-mounted display that used half-silvered mirrors to allow the user to see content from the real-world as well as virtual content from a set of CRT screens. Sutherland’s system superimposed simple 3D line drawing imagery onto the real-world environment using information from the tracking system to account for the changing perspective of the user. Clark [1976] extended the capabilities of Sutherland’s HMD system beyond just line imagery using a rendering approached based on B-splines to allow for the display of solid surfaces. In Clark’s system, control points could be edited using a 3D wand to allow the user to interactively design
surfaces in a 3D environment. In the 1980’s, Callahan developed a more advanced see-through HMD system for augmented reality using magnetic tracking technology instead of mechanical and using a beam-splitter to superimpose computer graphics imagery on a real-world view (described by Chung et al. [1989]).

Edwards et al. [1993] adopted a different approach to augmented reality, using fully immersive head-mounted display hardware instead of an optical see-through HMD. Closed HMD systems, where the user views display screens only showing virtual content, were developed to immerse the viewer in a virtual environment [Fisher et al. 1986]. In the system designed by Edwards et al., video cameras were attached to a closed HMD and the live video feed used to simulate the view of the real world. The video feed was then combined with virtual content generated on computer graphics hardware and displayed on stereo LCD screens in the HMD to provide an augmented reality experience [Edwards et al. 1993]. State et al. [1996] developed a method for increasing the registration accuracy between real and virtual content in a video-based HMD system by placing markers on real objects in the scene and using them as landmarks in the live video stream. The registration accuracy of the State et al. system allowed for appearance effects like casting shadows from the virtual object onto real ones and also the ability to incorporate complex textured objects, such as virtual playing cards, into the HMD view.

A video-based approach to augmented reality has also been used to create “window on the world” systems where live video is combined with computer-generated virtual imagery and displayed on conventional computer monitors [Milgram et al. 1994]. Using calibrated stereo video cameras, Milgram et al. [1991] demonstrated that it was possible to extract enough information about the real world geometry to embed stereoscopic computer graphics, such as a
3D tape measure, 3D pointer or wireframe outlines, into live video feeds that were then shown on a display screen. These concepts were further developed in the ARGOS project (Augmented Reality through Graphic Overlays on Stereovideo) for applications in telerobotic control [Drascic et al. 1993]. These types of monitor-based augmented reality systems have the advantage that they do not require the user to wear intrusive hardware, but this comes at the expense of a loss of immersion in the viewed scene.

2.4.3.2 Spatially Augmented Reality Systems

The development of projector-based spatially augmented reality by Raskar et al. [1998b] was an important conceptual advance, freeing the user from wearing a head-mounted display while still providing a highly immersive viewing experience by projecting virtual content directly on objects in the user’s physical space.

The projector-based approach that was adopted for use in spatially augmented reality was based on earlier projection systems used for virtual reality. The CAVE system developed by Cruz-Neira et al. [1993] projected virtual content onto multiple screens to fully immerse the user within a virtual environment. Users wore shutter glasses to provide stereoscopic depth cues and a magnetic tracking sensor to allow for parallax to convey the depth of virtual objects located beyond the projected walls of the CAVE [Cruz-Neira et al. 1993]. Work by Raskar et al. [1998a] on the Office of the Future system and Underkoffler et al. [1999] on the Luminous Room moved the projector-based approach in the direction of spatially augmented reality by projecting virtual content on walls, table-tops, and other surfaces throughout typical real-world environments, instead of specially-constructed CAVE enclosures. Raskar et al. [1998a] used the projectors in a dual role to enable the creation of immersive virtual content on the arbitrary
surfaces of an office environment, first projecting imperceptible structured light, detected by cameras, to estimate the geometry of real world surfaces and then applying computer generated imagery registered to the estimated physical surface geometry.

As noted by Raskar [1998b], it was work on the office of the future that led to the development of the spatially augmented reality concept, which was then advanced with implementations in the Shader Lamps and iLamps systems [Bandyopadhyay et al. 2001; Raskar et al. 2001, 2003]. In the Shader Lamps system, registered images were geometrically warped and projected onto physical three-dimensional models to allow the appearance attributes of real-world objects to be varied under computer control [Raskar et al. 2001]. A head-tracking system was incorporated to allow the position of virtual specular highlights on the model surface to be updated as the viewing position of the observer was changed [Raskar et al. 2001]. Bandyopadhyay et al. [2001] extended these capabilities in the Dynamic Shader Lamps project, adding tracking sensors on the objects to allow them to be moved and still retain their projected appearance content. The Dynamic Shader Lamps system also allowed users to interactively edit the material appearance, using a tracked stylus as a virtual paint brush [Bandyopadhyay et al. 2001].

The projection-based approach to augmenting the appearance of real physical surfaces has been incorporated into several types of systems. Bimber et al. [2002] applied the concept for use in paleontology, projecting computer-generated imagery onto real fossils to simulate the appearance of muscle and skin over the real skeletal structure. Yoshida et al. [2003] applied the approach in cultural heritage applications, developing techniques for correcting faded colors on paintings. Bimber et al. [2005] developed additional methods for enhancing the appearance of pictorial artwork and techniques for interactively displaying information directly on works of art.
2.4.3.3 Spatially-aware Display Systems

Several research projects on interactive display methods have focused on combining display screens with tracking devices to create a class of spatially-aware displays [Fitzmaurice 1993]. An early example, the Chameleon system developed by Fitzmaurice, coupled a six DOF tracking device with a handheld computer to create a display that could update its content based on its physical location in the environment. The Boom Chameleon system [Tsang et al. 2002] used a larger LCD display attached to a motion-tracked boom arm and allowed the user to change the view shown of a virtual 3D model by manipulating the physical position of the LCD screen. Konieczny et al. [2005] combined trackers with a flexible handheld rear-projection screen and custom-built projector to allow users to navigate and view slices through 3D medical data volumes by moving or bending the screen. With the Virtual Mirror System, Francois et al. [2003] took the concept of spatially-aware display systems in the direction of simulating the behavior of a physical object in the user’s environment. Using tracking devices to monitor the viewing position of the user and a camera pointed outward to capture video of the environment above the display surface, the system rendered images to the screen to give the impression that it was a real physical mirror [Francois et al. 2003]. This device was also used as part of an art exhibition to display virtual daguerreotypes, allowing video of the real-world scene to be superimposed to simulate a daguerreotype’s mirror reflection [Lazzari et al. 2002].

2.4.3.4 Light-Sensitive and Light Field Display Systems

An important recent development in interactive display technology is the creation of display systems that can interact with the real light from the scene. Though the use of this
approach is still limited, it represents a new direction in augmented reality that provides an alternative to spatial tracking and registration for combining virtual content and real-world environments.

Nayar et al. [2004] developed the “lighting sensitive display”, a system able to take into account the real ambient lighting when illuminating the virtual surfaces depicted in images shown on the screen. This system employed cameras to measure the 2D (directional) field of incident illumination and used it to relight the displayed image accordingly using real-time image-based rendering. This was an important step in using real-world illumination, though it did not measure the full 4D incident light field or take into account the dynamically changing viewing positions of the observer when calculating specular reflections for surfaces in the image.

Yang et al. [2008] developed a system capable of directionally modulating the light output from the screen to simulate a 4D exitant light field. Their system employed a cluster of projectors and a micro-lens array to produce different views of the onscreen images as a function of the viewing direction of the observer. Koike and Naemura [2008] developed a similar system, the “BRDF Display,” based on the lenticular techniques used for creating autostereoscopic 3D displays. Using a micro-lens array to control the direction of light emitted from different pixels, the image displayed could be varied as a function of viewing angle to emulate the viewing angle dependent reflectance aspects of a BRDF function [Koike & Naemura 2008]. With this type of system, there is a tradeoff between the spatial resolution of the displayed image and the directional resolution of the BRDF, as it is necessary to use a separate (spatial) pixel for each of the different directions of light modulation.

Fuchs et al. [2008] developed a passive display system that extended beyond 4D representations to simulate six dimensions of spatially varying reflectance over a 2D surface.
With this six dimensional display, the light output at each 2D spatial position was dependent on the 2D directional field of the incident light, and the light output varied as a function of the 2D viewing direction. Instead of employing active measurement-based techniques like in the lighting sensitive display [Nayar et al. 2004], the approach taken by Fuchs et al. was based on physical optical elements that allowed for passive modulation of surface transmittance as a function of both the lighting direction and viewing directions. This system modulated light from behind the screen, which allowed it to emulate real transmissive objects, but not real reflective ones.

There are similarities between research on light field displays and the object display system. However, the research on light field displays is primarily focused on developing the underlying hardware technologies needed to directionally modulate light with display screens (through optical means), while the object display framework seeks to address the broader question of how to reproduce the appearance of complex material surfaces (color, gloss and texture properties) with high-fidelity for the real geometric configuration of the display in the real-world lighting environment. In addition to the geometric aspects of light-surface interaction that are the primary concern of light field displays, the object display framework is also concerned with the spectral aspects of the interaction to allow for high-fidelity color reproduction.

2.5 Color Image Reproduction and Appearance

Color reproduction refers to the process by which color imaging systems are used to recreate the color of a real-world scene on a device or across various output devices and media
While basic CIE colorimetry serves an important role in color reproduction, matching tristimulus values alone is not sufficient to guarantee an appearance match [Fairchild 2005]. There are a variety of factors related to the viewing conditions and absolute luminance levels that must also be considered when trying to recreate the color appearance of a scene or match appearance across different media [Hunt 1982, 1996, 2004; Fairchild & Johnson 1999; Fairchild 2005]. In a traditional color reproduction workflow, color appearance models are used to account for the differences that arise when viewing different types of media under different viewing conditions. The goal in the object display framework is to minimize viewing condition differences between real-world objects and viewing the object display screen to allow for direct cross-media comparisons. In this section, the factors contributing to cross-media color appearance differences will be discussed, along with the methods that have been used to investigate these differences in previous research studies.

2.5.1 Factors in Cross-media Color Reproduction

Hunt [1996] reviewed the factors that result in differences in color appearance when reproducing images across different media. As Hunt noted, typical viewing conditions and display properties vary across media, which may result in reproductions being viewed at different absolute luminance levels, with a different color balance, with different surround color and luminance, or at a different scale of magnification. As Hunt described, softcopy versions onscreen are often viewed at lower absolute luminance levels and with darker surrounds than hardcopy reproductions, resulting in differences in colorfulness and perceived lightness contrast between the two media. An additional factor when reproducing images between self-luminous and reflective media, Hunt noted, is that a reflection print is itself a physical object, and thus
interpreted as a set of object colors. This is in contrast to self-luminant screens or projected displays, in particular in dim or dark surrounds, where the observer does not see the image as part of a colored object, but instead has an experience more similar to viewing a scene [Hunt 1996]. Hunt noted that an important difference between these two media is that chromatic adaptation is typically more complete in the reflection print case then for self-luminous displays in dim surrounds.

2.5.2 Chromatic Adaptation

There has been a body of work in the color science literature investigating differences in chromatic adaptation when viewing self-luminous displays as opposed to reflective prints. The adaptation differences between viewing reflective and self-luminous media are attributed to mechanisms for cognitive discounting of the illuminant that occur when viewing a reflective object, but that do not occur when viewing a self-luminous display [Fairchild 2005]. In the dissertation work, this phenomenon was studied on an object display system to evaluate whether the object display approach produces a state of adaptation more similar to a reflective object or more similar to a self-luminous screen.

2.5.2.1 Experiments on Chromatic Adaptation with Self-luminous Displays

Gorzynski and Berns [1990] performed a series of experiments to investigate chromatic adaptation differences between self-luminous CRT screens and reflection prints. In their first experiment, performed with a CRT display, observers were asked to select the patch appearing most achromatic from a grid of patches varying in chromaticity. To provide a reference for the
white-point, the background of the patch grid was set to either the chromaticity of tungsten illumination or of a daylight fluorescent light (simulating D65). The real ambient illumination surrounding the display was also varied, with conditions for tungsten, daylight fluorescent, or dark (no ambient) lighting. With the background set to the chromaticity of tungsten, Gorzynski and Berns found that observers did not select patches with the chromaticity of tungsten, but instead selected patches with chromaticities in the direction of D65. Similar results were found for all the ambient illumination conditions, indicating that regardless of the ambient surround, adaptation to tungsten on the CRT screen was incomplete.

Gorzynski and Berns’ second experiment was similar to the first, but instead of a CRT display, the grid of patches was presented with a hardcopy print illuminated by either tungsten or daylight fluorescent lighting. In contrast to the CRT results from experiment 1, under tungsten lighting, observers selected neutral patches with a chromaticity near that of the tungsten illumination as the most achromatic. The results of these two experiments indicate that discounting of the illuminant occurs for reflective hardcopy stimuli, but not in the case of computer-based softcopy images.

In Gorzynski and Berns’ third experiment, observers made cross media matches between a hardcopy reference, illuminated by either tungsten or daylight fluorescent, and a set of patches on a CRT screen with a background set to either a tungsten or daylight fluorescent chromaticity. When directly matching neutral patches of the tungsten-illuminated hardcopy to the tungsten-balanced CRT display, observers selected patches near the tungsten chromaticity, unlike the first experiment with the CRT alone. However, when the hardcopy was shown under daylight fluorescent illumination instead, observers again selected patches shifted in the direction of the daylight illumination (relative to the tungsten screen balance).
In the fourth experiment, Gorzynski and Berns presented an image of a complex scene on the CRT display, with a neutral patch embedded in the image. Similar to experiment three, observers were tasked with adjusting the patch in the scene to match a neutral hardcopy reference. However, with the complex image stimuli, they found that observers set the chromaticity of the patch to near tungsten, even in the mismatched case with the hardcopy under daylight illumination.

Fairchild [1992] described an additional series of experiments that were performed to assess chromatic adaption when viewing self-luminous displays. Fairchild performed an initial experiment using the method of adjustment to assess the chromaticity of a test patch perceived to be achromatic when surrounded by backgrounds of various chromaticities. Fairchild’s results were similar to those of experiment 1 from Gorzynski and Berns [1990]. For adapting backgrounds near D65, Fairchild found that observers specified the achromatic chromaticity to be close to D65, but for adapting backgrounds near illuminant A (tungsten) observers did not fully adapt and specified perceived achromatic chromaticities that were in the direction of D65. In additional experiments, Fairchild evaluated adapting field size, adaptation time, and cognitive factors to assess their role in chromatic adaptation to self-luminant displays. In the experiment on adapting field size, Fairchild found that the level of adaptation to illuminant A (though still incomplete) approached its maximum level (~58%) with a 10° adapting field. In the experiment to examine temporal aspects, Fairchild found that the maximum level of chromatic adaptation to illuminant A was approached after just 20 seconds. Fairchild also performed an experiment to assess order effects and found that the degree of adaptation to illuminant A was greater when that condition was tested first, before presenting observers with the D65-balanced condition. Finally, in an experiment that showed strong evidence of an effect from cognitive factors, Fairchild
repeated the first experiment, but overlaid the adapting background and test patch on an image of a print being held by a pair of hands. In the hands condition, where the stimulus appeared to belong to an illuminated print, Fairchild found observers set the achromatic chromaticity closer to the white point of illuminant A, indicating an increased level of chromatic adaptation (though not complete) to the illuminant A color-balance of the CRT softcopy.

### 2.5.2.2 Modeling of Incomplete Adaptation

Incomplete adaptation to self-luminous displays has led to the use of degree of adaptation factors in the chromatic adaptation transforms of color appearance models used for cross-media image reproduction. Fairchild [1991] developed a model that incorporated adaptation factors varying as a function of the adapting luminance and chromaticity of the display. Partial chromatic adaptation was incorporated into the CIECAM02 color appearance model with a degree of adaptation factor, $D$, that varies as a function of the adapting luminance, $L_A$, and a surround factor $F$ [Moroney et al. 2002]:

$$ D = F \left[ 1 - \left( \frac{1}{3.6} \right) \left( e^{-L_A - 42} \right) \right]. $$  \hspace{1cm} (50)

The $D$ factor is then used within the adaptation transform of the CIECAM02 model when calculating corresponding colors (shown here for the $R$ channel of the CAT02 space) [Moroney et al. 2002]:

$$ R_c = \left[ \left( Y_w \frac{D}{R_w} \right) + (1 - D) \right] R $$ \hspace{1cm} (51)

where $R$ represents the color under the initial viewing condition, $R_w$ represents the adapting white-point, and $R_c$ is the corresponding color under an equal energy illuminant. As the value of
$D$ increases to 1 in Eq. (51), the formula reduces to a simple von Kries-type transform with complete adaptation [Fairchild 2005]. As Fairchild noted, the $D$ factor in the CIECAM02 model is manually set to a value of 1 for reflective objects, as complete adaptation is assumed in situations where cognitive discounting of the illuminant is expected.

2.5.2.3 Mixed Adaptation

Color appearance models incorporating a degree of adaptation factor can be used to assess the chromatic adaptation state of an observer viewing a display alone, but the adaptation state becomes more complex if light sources of other chromaticities are present in the observer’s environment. Studies by Berns and Choh [1995], Choh et al. [1996], Katoh et al. [1998], Henley and Fairchild [2000], and Sueprasen and Luo [2001] investigated this mixed illumination case and found that the observers were partially adapted to the screen white point and partially adapted to the white point of the ambient illumination. Though there was general agreement among these studies that incorporating a partial adaptation ratio between the screen white point and ambient white point into color appearance models improved cross-media color accuracy, the exact value of the optimal ratio varied between the studies. Braun et al. [1996] recommended against the use of viewing configurations that induce mixed adaptation in cross-media comparisons because of the difficulty in characterizing the adaptation state of the viewer. This situation often occurs in practice, however, when observers simultaneously view an illuminated hardcopy print and a softcopy proof to directly compare the two [Braun et al. 1996]. Because of the practical importance of this issue, a CIE committee [CIE TC8-04 2004] was formed to study the problem and ultimately recommended the use of a modified form of the
CIECAM02 model incorporating a partial adaptation ratio. An impetus for developing the object display system is to allow observers to make direct side-by-side comparisons between electronic proofs and illuminated physical samples without introducing the type of mixed adaptation state complexity that can occur with traditional methods.

2.5.3 Perceived Lightness and Tone Reproduction

Differences in the typical viewing conditions of different media can also impact the perceived image contrast in cross-media image reproduction. The adapting luminance [Stevens & Stevens 1963] and the surround condition [Bartleson & Breneman 1967] influence the relationship between the luminance factor of an image element and its perceived brightness (or lightness). With these effects, cross-media tone reproduction becomes more complex because simply reproducing the relative luminances of image elements will not meet the criterion of maintaining the set of perceived relative brightnesses in the image [Bartleson 1968]. In the object display framework, the objective is to allow for direct one-to-one luminance reproduction by presenting the electronic proof at the same absolute luminance levels and with the same surround as an illuminated hardcopy.

The Stevens effect [1963] describes the phenomenon that as the adapting luminance increases, the relationship between the luminance of a surface and its perceived brightness changes. Stevens and Stevens studied the effect in a series of magnitude estimation experiments where observers, with one eye dark-adapted and the other light-adapted, rated the perceived brightness of a set of patches varying in luminance. Stevens and Stevens described the results in
terms of a power function relating the luminance of a stimulus \((L)\) to its perceived brightness \((\psi)\):

\[
\psi = k(L - L_0)^\beta
\]

with the \(\beta\) exponent taking on greater values for higher levels of light adaptation. The values of the luminance threshold, \(L_0\), and scalar parameter, \(k\), were also found to vary as a function of the adaptation level [Stevens & Stevens 1963].

Bartleson and Breneman [1967] demonstrated that in addition to the level of light adaptation, the luminance of the surround is also a factor in the relationship between luminance and perceived brightness. In a series of experiments, Bartleson and Breneman had observers view projected images with a dark surround and illuminated reflection prints with an average surround. Under each surround condition, observers were asked to rate the perceived brightness of different regions of the image relative to a provided standard. Based on the results, Bartleson and Breneman concluded that an illuminated surround acts as an inducing field, affecting the shape of the luminance-brightness function. Bartleson [1975] modeled the relative luminance-relative brightness curve shape as a power function with exponent of 0.33 for a dark surround, 0.41 for a dim surround, and 0.5 for a light surround.

In imaging applications, the practical implication of the Bartleson-Breneman effect is that an illuminated print, viewed under typical average surround conditions, will have a higher perceived contrast than a luminous reproduction with same relative luminances, viewed with a dim or dark surround [Fairchild 1995]. To recreate the perceived contrast across surround conditions, it is necessary to adjust the physical luminance contrast for the reproduction [Fairchild 1995]. Hunt [2004] described the need to apply a gamma of 1.25 for reproductions in
a dim surround and 1.5 for reproductions in a dark surround to match the perceived contrast for an average surround.

The effects of the adapting luminance and surround conditions are taken into account in color appearance models for cross-media image reproduction. In CIECAM02, the $F_L$ term, a function of the adapting field luminance ($L_A$), is used when calculating the non-linear response compression for the three channels and when calculating the brightness correlate, $Q$ [Moroney et al. 2002]. The selection of the surround condition as dim, dark, or average affects multiple parameters of the model, including the $c$ factor used in the calculation of the lightness correlate, $J$, and the brightness correlate, $Q$ [Moroney et al. 2002].
2.6 Application Domains

There are a variety of application domains where the ability to simulate the appearance of real objects using an object display system could be beneficial. These include the psychophysical study of material appearance, visualization and soft proofing of coatings and printed materials, and interactive access to digital library and museum collections. In the following section, an overview of related work in these applications areas is described.

2.6.1 Psychophysics of material appearance

Understanding the perception of material appearance has important implications for both basic science and industry [Adelson 2001]. Traditionally, an impediment to material appearance research has been the difficulty of creating physical sample stimuli that vary systematically in the parameters of interest. Recently, the study of material appearance has been facilitated by the ability to use 3D computer graphics techniques to create and display physically accurate simulations of objects with complex geometries and material properties in realistic illumination fields [Nishida & Shinya 1998; Pellacini et al. 2000; Fleming et al. 2003, 2004; te Pas & Pont 2005; Ho et al. 2007a]. Computer graphics-based methods were used by Nishida and Shinya [1998] to investigate specular and diffuse reflectance estimation for complex surface shapes. Pellacini et al. [2000] used computer graphics rendering to provide sets of stimuli systematically varying in physical gloss properties for a multi-dimensional scaling experiment on the perceptual dimensions of gloss perception. Fleming et al. [2003] conducted a computer-based study to investigate the role of the illumination in the ability of observers to estimate the gloss properties
of surfaces. Fleming et al. [2004] investigated the influence of surface properties on the estimation of the shape of complex computer-generated objects. te Pas and Pont [2005] used computer-based methods to study whether observers are able to discriminate between changes in material properties and illumination properties that produce similar changes in appearance. Ho et al. [2007a] used computer graphics-based methods to generate systematically-varying surface textures to study the perception of visual surface roughness.

Computer-based studies provide several benefits, including the relative ease and flexibility of generating computer-based stimuli, as opposed to real physical samples, and the ability to dynamically change properties for method of adjustment experiments. A limitation of computer-based studies on standard display screens is that they do not allow for the natural types of viewing and interaction that are possible with real physical samples. An object display system could provide the means to present computer-generated stimuli in the physical environment of the observer and create an interactive viewing experience more like viewing physical samples when studying material appearance.

2.6.2 Print Proofing

In photographic printing and desktop publishing, it is useful to be able to simulate the appearance of a hardcopy image before printing by soft proofing on a computer display. Traditionally, soft-proofing has been performed to simulate the color appearance of printed materials on display screens and has focused on methods to account for differences in color gamut and viewing conditions across media [Masia et al. 1985]. Laihanen [1994] developed an early system for exact soft proofing, attempting to produce an appearance match by reproducing
the exact colorimetry (absolute luminance) of prints on a CRT screen, though encountered issues related to the non-uniformity and limited spatial resolution of the CRT. Recently, Hill [2010] developed a display system for exact color soft-proofing to allow for direct comparisons of hardcopy and soft-proofed patches on an LCD screen in an illuminated light booth. Hill’s system was calibrated to reproduce the exact colorimetry of a physical ColorChecker, using spectral data on the light sources and by adjusting the luminance output level of screen regions until they matched a physical mask placed on the screen.

Recent research efforts have investigated incorporating print gloss properties into the soft proofing process. Gatt et al. [2006] performed goniometric measurements on combinations of printer inks and papers to develop a predictive BRDF model for the gloss properties of printed materials. Patil et al. [2004] developed a soft-proofing system that incorporated a gloss model for printing and used computer-graphics rendering to generate images of prints mapped to 3D planes in a virtual environment. In this system, Quicktime VR sequences of the images were created in an offline process to allow the viewpoint on the print to be updated in real-time, and results were presented to the screen using standard computer graphics display methods.

2.6.3 Computer-aided Appearance Design

The ability to render realistic simulations of surfaces and materials at interactive rates on modern graphics hardware has fostered the field of Computer-Aided Appearance Design (CAAD) [Meyer 2000]. Interactive CAAD tools have been developed for designing and modeling the appearance of complex materials such as gonio-chromatic paints [Shimizu et al. 2003] and automotive finishes [Meyer & Shimizu 2005]. Shimizu and Meyer [2005] and
Colbert et al. [2006] developed material-editing systems incorporating real-time image-based lighting techniques to allow materials with complex BRDF functions to be rendered under realistic types of illumination during the design process.

Though CAAD applications are often presented on 2D screens using traditional computer graphics display methods, there have been efforts to develop virtual reality and projection-based augmented reality systems for use with CAAD. Konieczny and Meyer [2006] and Kamimigaki et al. [2009] developed projection-based systems for CAAD, based on the Shader Lamp concept, that simulated the appearance of color and material properties by projecting onto physical objects. Konieczny et al. [2008] developed a virtual spray painting simulation allowing users holding a motion-tracked spray gun to virtually apply realistic paints to automotive panels viewed on a head-mounted VR display.

The capabilities of the object display framework, to recreate the appearance of color and material properties of real surfaces under natural illumination and with direct modes of interaction, may make it a useful platform for CAAD applications.

2.6.4 Access to digital collections

Digitization has had an enormous impact on libraries and museums. Manuscripts, paintings, and other collections that were only accessible by physical visit are now documented and accessible worldwide though digital images. Digital archiving efforts were initially focused on creating color-accurate images of collections [Berns 2001] and later toward spectral imaging to allow for improved color accuracy, scientific evaluation, and the ability to simulate color appearance under other illuminants [Berns 2005]. More recently, with advances in digital
imaging and computer graphics technology, digitization efforts are starting to include 3D measurement and modeling of surface geometry and BRDF to capture more comprehensive information about the surface properties of works of art [Gardner et al. 2003, Tominaga 2005, Chen et al. 2007, Tominaga & Tanaka 2008, Berns & Chen 2012, Chen & Berns 2012]. Gardner et al. [2003] developed a method to model the surface appearance properties of objects, including an illuminated manuscript and daguerreotype, using a linear light reflectometry system that acquired sequences of images as a linear light source moved across the surface. Tominaga [2005] developed a system for capturing the properties of oil paintings by incorporating a laser-range finder to estimate surface geometry and imaging the surface illuminated from a range of different directions with a six-channel camera to estimate bidirectional reflectance and surface orientation information. Tominaga and Tanaka [2008] developed an updated version of this system, incorporating a projector with six colored filters for use in estimating the spectral reflectance of the painting surface. Chen et al. [2007] developed a camera-based system for estimating BRDF parameters of painted surfaces and performed visual experiments to assess the accuracy of BRDF models of varying complexities in an effort to create a practical system for use in museums. In related research, a database was constructed of commonly-used artist materials, applied using a variety techniques, and measured with a gonio-imaging system to provide a digital library of BRDF and surface normal data for a range of artist materials [Ashbaugh et al. 2009]. In recent work, Chen and Berns [2012] have developed a system for total appearance imaging of art work using multiple light sources to capture surface orientation data, multispectral cameras to capture surface color properties, and a linear source to measure surface gloss properties (BRDF). The 3D models captured using these types of techniques
provide the capability to render works of art and simulate their appearance from different viewpoints or under different types of illumination.

2.7 Summary of Object Display and Related Work

The dissertation work on object display systems incorporates concepts from multiple disciplines and is related to previous work in areas including digital modeling of material appearance, color reproduction, interactive display systems, light-sensitive displays, and virtual and augmented reality. The relationship of the object display system to previous work is summarized in this section.

The object display framework is related to aspects of digital modeling of material appearance and realistic image synthesis. The methods from these areas provide the basis for computer-based modeling of the behavior of complex surfaces. The object display framework is designed to provide a way to interact with these types of computer-based models in the physical world. The simulation aspects of the object display framework share similarities with the realistic image synthesis framework developed by Greenberg et al. [1997]. In the realistic image synthesis framework, their objective was to develop radiometrically accurate simulations to produce synthetic images that were visually indistinguishable from real photographs. In the object display framework, the goal is to take this concept a step further and attempt to produce a result indistinguishable from directly viewing real objects, by using real-time rendering simulations and combining realistic surface models with interactive tracking information and captured models of the real illumination around the display.
The object display system is also related to work on color reproduction with exact color soft proofing systems. In these systems, the light output from the screen is matched in absolute colorimetry to the diffuse surface colors of a physical hardcopy [Laihanen 1994, Hill 2010]. In the object display framework, the goal is to allow the absolute proofing paradigm to support a range of material properties, such as gloss and texture, and allow for natural modes of interaction when viewing proofs matched in absolute colorimetry with the real world lighting.

The object display work incorporates aspects of interactive and spatially-aware display systems. The intention with an object display system is to allow the appearance of complex surface materials to update based on physical modes of interaction with them, as if manipulating a real object situated in the environment of the user. The spatially-aware Boom Chameleon [Tsang et al. 2002], provided a similar method of interaction, physically moving the screen, though was based on a different screen viewing paradigm. The screen acted as physical window that was moved by the user to give different views of a virtual object located behind it. More similar to an object display, the screen in the virtual mirror system [Francois et al. 2003, Lazzari et al. 2002] was designed to act like an object in the physical space of the observer. Using an outward facing camera, a head-tracker worn by the user, and an environment-mapped rendering technique, the system was able to simulate mirror reflections that updated based on the real position of the observer. The object display expands on the virtual mirror concept by allowing for a range of surface material properties and photometrically-matched rendering when simulating the presence of a real object located in the viewer’s space.

Object display systems can be considered as a form of spatially augmented reality and share similarities with other spatially augmented reality systems like Shader Lamps [Raskar et al. 2001; Bandyopadhyay et al. 2001]. The object display simulates objects with a real-world
presence on display screens, while these systems typically use a projector-based approach. In projector-based systems, an object is physically constructed from white diffusely reflecting material and calculated patterns of light are projected onto it to simulate different material properties. Though each new geometric surface has to be constructed, the ability to present objects that have significant 3D geometry and are not generally flat is an advantage of using a projector-based approach over screens. To meet the goals an object display system, display screens are used instead because they provide the ability to use natural types of real-world illumination. With the white projection surface, the constructed object will reflect any ambient light present in the room according to its physical reflectance, not the virtual one, and so the projectors need to be the only significant sources of light present in an otherwise darkened room. In the context of an object display system, this would produce a mismatch between the real lighting in the environment (directional projectors) and the illumination implied by the rendered patterns on the virtual objects. Screen-based approaches (with low reflectivity screens and appropriate lighting geometries) allow for the use of natural forms of ambient lighting. With screens, the virtual surfaces can be viewed in an illuminated surround and surfaces can be rendered in a manner consistent with the real lighting present in the environment.

The object display system is also related to work on display systems that incorporate light sensing and light modulating capabilities. The light-sensitive display [Nayar 2004] introduced the concept of illuminating virtual content on a display based on the real lighting surrounding the screen, using the captured light from a camera to re-light a 3D scene on the display screen. Though the viewing paradigm and modeling and rendering approaches are different, the underlying concept of a light-sensitive display is an important aspect of an object display system. Koike and Naemura [2008], Yang et al. [2008], and Fuchs et al. [2008] have been exploring
hardware technologies for creating light field-type displays that respond to incident light or directionally modulate light output. While light-field display technologies are still in the early stages, conceptually they present interesting opportunities for new implementations of the object display framework. The current method used for object display systems, based on a professional-grade 2D display screen and active observer tracking, was used because of the need for mature display screen technology that would provide a high level of image quality, luminance stability, and color accuracy. These screen properties are necessary to support the types of high-fidelity surface reproductions for color proofing, material design, and psychophysical testing that are the goal of the object display framework. In the future, if lenticular display systems become available that provide high-fidelity display of 4D light output, they could be used in place of active observer tracking and incorporated with the surface modeling and rendering components of the object display framework to create object display systems with enhanced capabilities.

Overall, the object display framework brings together concepts from realistic image synthesis, colorimetric-based proofing systems, interactive display systems, light sensitive displays, and augmented reality in an attempt to create virtual objects, with realistic material properties, that appear and behave like the real physical objects they are portraying.
3 CONCEPTUAL DESIGN OF AN OBJECT DISPLAY SYSTEM

In this chapter, a multi-level conceptual framework is described that outlines a set of stages from the glowing images presented by a typical display system to the emulation of a physical reflective object surface. Using this organization, it is possible to chart the progression of the object display technology from pictorial display representations, through tangible display systems, to the final object display system.

3.1 Levels of the Framework

Level 1. Pixel

Trichromatic Color Reproduction

At the most basic level, a standard electronic display screen is composed of individual light elements that emit a combination of red, green, and blue light of varying intensities. Though limited to spectral power distributions that are linear combinations of the three primaries, a pixel can produce metameric matches to lights with a variety of spectral distributions and luminance levels.

Level 2. Pixel array

Trichromatic Color Image or Video Reproduction

A standard electronic display screen consists of a two-dimensional spatial array of trichromatic pixels. The elements of the array are organized into complex spatial patterns of color to form discrete images. The image can be a pictorial representation of an object, though
physically the pixel array is emitting light and not reflecting light as a real object would. The pixel array patterns can be temporally modulated, allowing the images to change over time and the ability to convey the appearance of motion.

**Level 3. Viewing-direction modulated pixel array**

*Directional Color Image Reproduction*

Moving beyond the typical display capabilities of *Level 2*, the addition of viewing-direction dependent modulation increases the dimensionality of the display representation from a 2D spatial pattern of color reproduction to a 4D representation with both spatial and directional dependencies. This is an underlying capability that is necessary to simulate directionally dependent appearance phenomena such as gonio-apparent colors or surface gloss properties, where the intensity of reflections vary as a function of the observation angle. In a display system, directional dependence can be achieved with active observer-tracking or with the use of passive lenticular techniques that optically vary the image as a function of the viewing direction. In the active paradigm, sensors are used to detect the direction of the observer, and a new 2D spatial image is computed based on the estimated viewing location for each screen refresh. An active system allows directional modulation to be achieved using traditional display hardware by creating a form of “directional metamerism” for the observer. The display is not actually reproducing all the different directional views of the screen, only the one matching the current direction of the tracked observer, but from the perspective of the tracked observer, the screen appears to be directionally varying the emitted light.
Level 4. Mapping the directionally-dependent pixel array to points of a virtual surface

Virtual object, with color, situated in physical space with respect to the observer

The spatial and directional aspects of color reproduction from Level 3 provide the underlying display capabilities needed for an object display system, but ultimately the display is still producing dissociated images until used within a framework that associates a surface representation with the screen. In the object display representation, the principal geometry of the virtual object is modeled as a rectangle that occupies the physical space of the display screen. This allows the virtual object to appear to occupy a physical location in real space and provides a physically consistent presence with respect to the observer. The difference between this approach and the typical display of computer graphics simulations, where the screen acts as a window on a virtual world, is illustrated in Figure 6.

Figure 6. Left (a), a real physical painting is tilted in physical space. Center (b), in a standard computer graphics representation, the virtual painting is tilted by rendering an oblique view making it appear to extend into a virtual space behind the screen. Right (c), in the object-display representation, the virtual surface is directly mapped to the location of the screen. To show the virtual painting being tilted, the screen itself is tilted in physical space.
The co-location of the screen surface and the virtual object surface is an important concept in transforming the screen from a pictorial representation of a scene to a proxy for a real object surface. By co-locating the screen and virtual object, there is a direct mapping of points in the pixel array to positions on the virtual surface, which allows the virtual surface reflections to be directly associated with the real locations of physical light emission.

**Level 5. Per-pixel BRDF representation consistent with incident/observation angle changes**

*Virtual surface, with gloss and color properties, in physical space with respect to the observer*

In *Level 4*, the virtual surface of the object is geometrically modeled in such a way that the pixels are directly associated with surface points, and the surface appears to occupy a position in the real physical space of the viewer. To complete the representation of a surface in physical space, the display must be able to reproduce surface reflections in a way that is consistent with a real object at the physical location of the screen.

Reflection from a real object surface can be described by its BRDF, the four-dimensional function, $\rho_{brdf}(\theta_i, \phi_i, \theta_r, \phi_r)$, that describes how reflectance varies based on the incident lighting directions, $(\theta_i, \phi_i)$, and outgoing reflection directions, $(\theta_r, \phi_r)$. By adopting a BRDF representation for each pixel and using the real viewing angle and surface orientation to set the BRDF direction parameters, the object display will have the underlying mechanism necessary for producing surface reflections consistent with a real object at the screen location.

As an observer moves relative to a real object, the intensity of the reflection at a given surface position will change because the movement affects the direction of the viewing vector from the viewer to the surface point. For the given surface point, the observer sees the reflection governed by a different portion of its BRDF curve, the portion where the outgoing $(\theta_r, \phi_r)$
direction parameters are equal to the new viewing direction (as shown in Figure 7). With the object display, this can be simulated by evaluating the BRDF function for the outgoing direction equal to the real screen viewing direction and then modulating the light emitted accordingly. The necessary display capability corresponds to the directional color modulation described in Level 3.

Figure 7. Left (a), the viewing position of the observer, with respect to the surface normal, determines the $\theta_r$ that should be used to index the BRDF curve (shown in red). When the observer moves to a new position (b), the $\theta_r$ is changed and indexes a different portion of the BRDF curve, resulting in a change in the magnitude of the reflection directed toward the viewer.

When a physical object is rotated or tilted, both sets of BRDF directions are impacted (as shown in Figure 8). As the orientation of the surface changes, the viewing angle is changed with respect to a fixed observer, which has a similar effect on the ($\theta_r$, $\phi_r$) parameters to the movement of the observer relative to the screen. Rotation of the surface also results in a change in the ($\theta_r$, $\phi_r$) parameters.
φ) incident angles with respect to fixed lighting. This is simulated in the object display framework by detecting the orientation change of the display screen and updating the direction from the surface to each virtual light source for use in the (θ, φ) angles of the BRDF calculation. While this level provides the mechanism for detecting and responding to orientation changes of the surface in a realistic manner, the results at this stage are based on the locations of virtual lights. The addition of the matched physical lighting capabilities described in Level 8 is required to reproduce reflections that are physically consistent with the real lighting present.

![Diagram](image)

**Figure 8.** Left (a), the positions of the observer and lighting with respect to the surface normal determine the θ<sub>r</sub> reflection and θ<sub>i</sub> lighting directions to be used in the BRDF calculation. When the surface is rotated and the direction of the normal is changed (b), both the θ<sub>r</sub> reflection direction and θ<sub>i</sub> lighting direction are affected for fixed positions of the observer and lighting.
**Level 6. Per-pixel modulation of surface orientation**

*Simulated texture for a virtual surface with color and gloss properties*

The BRDF representation of *Level 5* provides the virtual object with spatially-varying gloss properties, but is limited in that it only accounts for perfectly smooth surfaces where each point (pixel) has exactly the same orientation. Many real surfaces have texture, mesoscale geometry that modulates the orientation of different points along the surface. Within the context of surface reflection, a main effect of the texture is that it introduces a source of variation into the BRDF \((\theta_i, \phi_i, \theta_r, \phi_r)\) angles used to calculate reflections at different points on the surface. The lighting and viewing direction are defined relative to the local surface normal, so variation in the surface orientation impacts the BRDF angles at each point along the surface (as shown in Figure 9).

![Diagram](image)

**Figure 9.** Left (a), with a smooth surface, the local surface normals are aligned with the principal geometric normal. Right (b), the local surface normals vary due to surface texture, resulting in a different set of BRDF angles, with respect to fixed lighting and viewing positions, at each position on the surface.
On a smooth surface, all the surface normals are aligned and the BRDF angles only vary gradually across the surface, producing coherent reflections. On a textured surface, points in neighboring regions, even with identical BRDF properties, may produce very different reflections due to the differences in the BRDF angles used when calculating the reflections. This spatially-varying texture can be simulated in the object display framework by assigning each pixel a vector that adjusts its orientation relative to the geometric normal of the display screen surface (i.e. using a normal map representation [Peercy et al. 1997], illustrated in Figure 10). When estimating the lighting and viewing directions for the BRDF calculation, the adjusted normal for the pixel is used instead of the geometric normal of the surface.

![Figure 10](image-url)  
**Figure 10.** The surface texture that results from small scale surface geometry (left) can be represented by the normal vectors describing the orientation of the surface at each position (right).

**Level 7. Per-pixel height used to determine shadowing (and other height effects)**

### 2.5 2.5D representation of a surface

The representation in Level 6 accounts for the main effect of surface texture by incorporating per-pixel variation in surface orientation into the surface reflectance calculation. In a real surface, however, the mesoscale geometry of the texture also has a physical height that
results in additional appearance effect that cannot be represented by a model of orientation alone. With shallow lighting angles, a relatively high region of the surface may block incident light from reaching the lower-lying regions that neighbor it, resulting in self-shadowing effects (as shown in Figure 11). Conversely, light may reflect from one region of the surface and be directed toward a neighboring region, causing inter-reflections that add to the incident illumination at the second point. Shadowing and inter-reflection result from the effect of the geometry with respect to lighting, but the geometry also has an influence with respect to viewing. If a relatively high region is between a lower-lying neighbor and the observer’s viewing position, parallax occlusion effects will result. As an observer views the surface from wider angles, high regions of the surface that are closer to the observer will obscure the view of lower lying regions that are further away.

Figure 11. Left (a), with incident lighting directly above the surface, the effects of self-shadowing from the mesoscale geometry of the surface texture are minimal. With incident lighting at wide angles (b), the physical height of the texture must be considered as portions of the surface block the incident light and cast shadows over other regions of the surface.

A physical height must be associated with each pixel in order to support shadowing, inter-reflection and parallax occlusion effects. The effects of shadowing can be simulated at each
given pixel location by determining whether any other pixel position, with its specified height, intersects the path from the starting pixel to a given light source. If any other pixel on the surface does intervene, then the incident light from that source should not be included in the BRDF reflectance calculation for the blocked pixel. The determination of shadowing only requires a binary decision as to whether or not the light toward a given pixel location is blocked by any intervening pixels. A more complex process is required to account for parallax occlusion effects, because it is necessary to identify exactly which of the occluding pixels will be the one viewed by the observer (as shown in Figure 12). To correctly identify which occluding pixel will be viewed, it is necessary to trace the entire path from the starting pixel location to the edge of the surface, determine all the occluding pixels, and select the one that is closest to the observer. The material properties of the occluding pixel can then used in place of the properties of the blocked pixel when calculating the reflectance for the surface point at the occluded location.

![Figure 12](image.png)

**Figure 12.** The physical heights of features on the surface occlude the observer’s view of point c. The view of c is occluded by the features at both points a and b, but to correctly account for occlusion parallax, the algorithm must select a, the point closest to the observer, as the replacement.

Inter-reflection is the most difficult of the three phenomena to simulate. While shadowing and occlusion for a given surface point depend upon the effect of one other surface point along a
known line (to the viewer or light), the light contribution from inter-reflection originates from multiple arbitrary points along the surface and is a recursive process, requiring that the reflections from each point on the surface be recalculated multiple times to converge toward a solution. The light contribution due to inter-reflection at a point \( x \), for the first level of recursive reflection, is illustrated in Figure 13.

**Figure 13.** Light from a source strikes the surface and can be reflected over a range of directions. Any light that is reflected toward another point on the surface (e.g., the point \( x \)) produces an inter-reflection that contributes to the incident illumination at point \( x \).

**Level 8. Matched Virtual and Real lighting environment.**

*Physically-consistent 2.5 D representation of an object surface*

The first 7 levels provide the capabilities to modulate light output from the display based on a representation of surface properties and to provide a physical presence with respect to the observer, but an additional capability is critical for creating reflections that are physically consistent with the real reflections for a surface at the position of the display. The surface representation up to this point is consistent with respect to the observer’s position, but to
complete the conversion must also be made physically consistent with the real lighting environment where the display is situated.

For a real object, the color and intensity of reflections from the surface depend upon the spectral irradiance of the illumination arriving from each direction. The spectral power distribution of the incident light affects the color of reflections, the incoming direction determines the \((\theta_i, \phi_i)\) angles for the BRDF calculation, and the magnitude of the irradiance reaching the surface determines the absolute scaling for the physical luminance of the reflections.

In the object display framework, the effect of real-world illumination can be accounted for by developing a virtual lighting model that is consistent with the absolute spectral radiance of the real lighting environment surrounding the display. In Level 5, a BRDF-based representation was introduced that could account for the effects of physical modes of interaction, but the changes with respect to incident lighting were calculated for arbitrary virtual lighting environments. If instead the virtual lighting model is designed to be physically consistent with the actual environment, then the \((\theta_i, \phi_i)\) angles determined by orientation tracking, when used in the BRDF calculation, will correspond to the real direction to physical lights. In this case, rotating the display surface relative to the real physical lights in the environment will result in changes to the virtual surface reflections that are consistent with a real object in that environment.

In addition to accounting for the real directions to lights in the environment, it is also necessary to account for the intensity and spectral power distribution of the real light sources so that the physical luminance and color of the reflections reproduced will be consistent with those of a real surface at that location. Matching the physical luminance emitted by the display screen to the physical luminance that would be reflected by a real object at the display’s position is a
critical step in transforming the display from a self-luminant source to a proxy for a reflective object. In this case, it is necessary for the lighting environment to be measured in absolute physical units, for the virtual lighting model to be represented and stored in physical units, and for the colorimetric characterization of the display screen to be conducted in absolute luminance units. Along with the intensity of reflections, the spectral properties of the real light sources must be considered so that the reflections also have the correct chromaticity (xyY colorimetry) for the given lighting environment.

In addition to modeling and physically producing light to create the reflections for the virtual surface, the unwanted light contribution from the real physical screen surface must also be taken into account to meet the Level 8 requirements. With the screen placed in an illuminated environment, the front surface of the screen has the potential to produce a conflicting set of reflections that will interfere with the perception of the virtual surface simulated by the light emitted from the display. These reflections can be addressed by a combination of physical minimization with optical coatings, the selection of favorable lighting geometries, and modeling and subtracting the unwanted light (from the light that would otherwise be emitted from the display as reflections from the virtual surface). An additional approach is to use computational light sources that can be configured to mask out the portion of the light that would reflect off the screen.

3.2 Physical Design Constraints

Given the use of a self-luminous, two-dimensional display screen, there are certain physical limitations in recreating the appearance of a real object surface. These limitations
relate to binocular disparity, the focal plane of reflections, grazing angle effects, and spatial resolution.

One limitation of the object display framework described in the conceptual design is that it does not account for the binocular disparity that results from the height of the mesoscale geometry. With a real physical surface, the height of the mesoscale texture would result in a slightly different view of the surface to each eye. Accounting for binocular disparity with current technologies, however, would require the use of 3DTV systems in which the user has to wear shutter glasses or polarized lenses. In the interests of providing natural viewing (not through specialized glasses) and maintaining display luminance, binocular disparity is not addressed in the current object display framework. As high quality auto-stereoscopic display technology becomes available, however, this aspect could be introduced into the object display framework.

A related limitation is that with the use of a 2D screen, the simulated specular reflections are reproduced in the plane of the screen surface. With real surfaces, however, the image of a mirror reflection appears to be behind the surface. A 3DTV system could be used to display the binocular disparity of reflections appearing behind the screen plane, but still would not account for the different focal planes of the object surface and mirror reflections. With real surfaces, a viewer can only focus on one of the planes at a given time, so that when focusing on the surface plane, specular reflections appear out of focus. With the current form of the object display, both the surface and simulated reflections are in focus concurrently when the user is focused on the plane of the 2D screen.

An additional limitation stems from the use a self-luminous display screen to simulate reflective surfaces. The magnitude of the specular reflection from a real surface typically increases at a wide viewing angle, while the maximum luminance of a display screen typically
decreases under these viewing conditions. Because the display must actively produce light to match the luminance of simulated reflections, it is not possible to recreate the types of grazing angle reflections that result when viewing real surfaces at angles near 90°degrees from the surface normal. With the performance of current LCD panels, the object display system prototype was designed for accurate use out to approximately 30° from the surface normal.

Finally, the finite pixel pitch of the screen produces a limitation on the spatial resolution of surface features that can be presented. When viewed at short distances, the appearance of the screen’s pixel grid becomes evident, which may interfere with the perception of the screen as a real physical surface. With the recent growth of high-resolution display technology in laptops, mobile devices, and 4K video screens, display resolution may not be a significant issue in the future.

Several of the limitations described in this section are primarily a concern when viewing the screen from short distances. The degree of binocular disparity is reduced as the observer moves further from the screen, small lateral movements by the user do not result in as extreme viewing angles when the screen is viewed from greater distances, and the visual angle subtended by the pixel pitch is reduced.
4 TANGIBLE DISPLAY SYSTEMS

Tangible displays were the first stage of development for what ultimately became the final object display systems in this dissertation. The tangible display systems discussed in this chapter were described in papers on the laptop-based tangiBook [Darling & Ferwerda 2009] and second-generation customizable system [Darling & Ferwerda 2010]. These systems were termed tangible displays because physical interactions with the display screen cause the reflections to change in a realistic manner. They provide natural forms of interaction, which is a key element of the conceptual framework (Level 5), but were not designed to produce renderings consistent with the real lighting surrounding the display. Though a user can interact with them like a real object, they appear self-luminous and not like a physical object reflecting light. The capabilities of tangible display systems, their design, and the progression toward current object display systems are described in this chapter.

4.1 Overview

When presented with a real object, observers often engage in complex behaviors involving active manipulation and dynamic viewpoint changes to better understand its surface properties. With these behaviors, the user is able to observe the changing pattern of reflections over the surface as the object is moved relative to lighting and observer’s own viewpoint. Tangible display systems were developed to provide these types of natural interactions in computer-based applications. These systems combine tracking sensors for the orientation of the display and the position of observer with an interactive computer graphics simulation that updates the virtual reflections shown on the display screen. The user experience is similar to
holding a physical surface in one’s hands and being able to actively tilt it and observe it from different directions to see the changing patterns and properties of surface reflections.

The first generation tangible display system (shown in Figure 14, left), the “tangiBook,” was created using an off-the-shelf laptop containing a triaxial accelerometer and webcam as standard components. The accelerometer is used to provide real-time information on the orientation of the screen and the webcam is used with computer vision-based headtracking to estimate the position of the observer. Custom software combines this tracking information with a real-time rendering engine to allow the appearance of a virtual object onscreen to update with changes in the real orientation of the display and position of the observer.

![Figure 14. Tangible displays were developed for two different architectures. Left, the tangiBook, was developed from an off-the-shelf laptop computer. Right, a customizable tangible display system was created from a desktop computer, sensors, and a display screen.](image)

The objective of the laptop-based system was to create a self-contained tangible display from an off-the-shelf computer so that a user could experience a tangible display without the need to acquire specialized hardware. The second generation system (shown in Figure 14, right) was developed from a set of selected components: a powerful desktop computer workstation, sensors (triaxial accelerometer, camera), and an IPS-based display screen able to better maintain
color accuracy over a range of viewing angles. The second-generation architecture, with its flexibility for customization and greater processing power, served as the development platform for the final object display.

4.2 Tangible Display System Capabilities

Tangible display systems provide a unique set of capabilities for interacting with virtual objects. They support direct manipulation of the virtual object’s orientation by rotating the screen and dynamic viewpoint changes by tracking the observer’s head position. With these capabilities, the tangible display responds to the two types of interaction that produce changing surface reflections when viewing a real object. Tangible display systems also provide a capability not possible with real world surface. Because the system supports real-time rendering, the material properties can be changed interactively. Users can specify the material properties of the surface in terms of diffuse reflectance, specular reflectance, and specular roughness, and change these properties dynamically while physically interacting with the system. These capabilities are illustrated in the following sections.

4.2.1 Physical Interactivity

As shown in Figure 15 and in Figure 16, tangible display systems support dynamic, natural interaction with virtual surfaces. As the screen is manipulated by the user, the changes in its physical orientation are tracked and used to dynamically update the rendering of the virtual object. The updated rendering displays the surface reflections for the virtual object’s new
orientation relative to the specified virtual lighting environment. As the laptop is tilted from its orientation in the left image (of Figure 15) to its orientation in the far right image, the painting’s surface catches the reflection from one of the virtual lights when the new direction of the surface normal causes the viewing position to near the new specular angle with respect to the light position. Similar capabilities are supported by the second generation system, shown in Figure 16, using a triaxial accelerometer attached the screen.

**Figure 15.** Image sequence showing a painting model on the first tangible display system, the “tangiBook.” As laptop is tilted, the change in orientation is sensed by the accelerometer and the rendered highlights are changed accordingly for the new relationship between the surface, the virtual lighting, and the observer position.

**Figure 16.** Image sequence showing a painting model on a customizable tangible display system. An accelerometer attached to the screen allows the orientation of the screen to be tracked in real-time so that tilting the display produces changes in surface reflections, as shown. Though this natural type of interaction was possible, the reflections were changed with respect to virtual lighting and not the real physical lighting environment surrounding the display (which as shown, was typically dark).
### 4.2.2 Dynamic Viewing

The tangiBook and workstation-based tangible display systems also provide the capability to dynamically track the observer’s viewing position relative to the display and update the rendered image accordingly. As shown in Figure 17, as the viewer’s head position moves relative to the display, the rendered reflections for the painting’s surface change to reflect the new relationship between the viewpoint, surface normal, and the virtual illumination environment.

![Figure 17. Image sequence illustrating the viewpoint-based tracking and rendering capabilities of a tangible display system. The insets show the observer’s position (lower right) and the output of the head-tracking software (lower left) as the observer moves in front of the screen. The large images show the corresponding movement of the rendered surface reflections.](image)

### 4.2.3 Dynamic Control of Material Properties

The interactive rendering capabilities of the tangible display systems allow for dynamic control of material properties specified in terms of the diffuse reflectance, specular reflectance, and roughness of the object’s surface. These parameters can be adjusted dynamically, while the user is interacting with the virtual surface, and will immediately produce changes to the surface’s
appearance. This is shown in Figure 18, where the material being rendered is changed from matte to glossy while the tangiBook is actively being used.

**Figure 18.** The two renderings of the painting illustrate the ability of a tangible display to dynamically change the material properties of a displayed surface. The surface on the left has a relatively low specular reflectance ($\rho_s$) and a high roughness ($\alpha$) giving a matte appearance. Using the sliders, these two parameters have been altered in the surface on the right to produce glossy reflections.

### 4.3 Tangible Display System Design

A tangible display system requires three primary components: tracking devices for the display and user, a display screen, and an interactive rendering module that generates a realistically shaded view of the virtual surface based on the tracking information. In a typical system, a triaxial accelerometer is used to provide data on the orientation of the display, and a webcam is used for computer-vision-based head-tracking to estimate the location of the observer’s viewpoint. Custom software integrates the tracking information with a custom OpenGL shader, executed on the computer’s GPU, to generate a realistically shaded view of the
virtual surface to the LCD display. By integrating these components in a software system, virtual surfaces can be observed and manipulated in the same manner as real surfaces.

### 4.3.1 Coordinate Systems

Two coordinate systems are used for performing calculations and representing interactions with a tangible display system (shown in Figure 19). The first is the world coordinate system, where the **x**, **y**, and **z** axes remain fixed relative to the physical direction of gravity. The second is the local screen-object coordinate system **uvw**, which is affixed to the display and has its axes defined by the directions: normal to the screen (**w**), from bottom-to-top of the screen (**v**), and from left-to-right on the screen (**u**).

![Coordinate Systems Diagram](image)

**Figure 19.** The **xyz** axes define a world coordinate system that is fixed with respect to gravity. The **uvw** axes define an object space coordinate system that is fixed with respect to directions on the screen. Left, the two coordinate systems are aligned in the screen’s initial reference state. Right, the screen coordinate system has been rotated relative to the world coordinate system.
In the system’s reference state, with the screen in a vertical orientation (Figure 19, left), the screen’s \(uvw\) axes are aligned with the \(xyz\) axes of the world coordinate system. As the display is manipulated, the \(uvw\) axes are rotated relative to the \(xyz\) axes (Figure 19, right). A common origin is maintained for the two systems and orientation changes are represented as rotations around that point. In the final object display system, a more complex model that also considers the physical location of the display, and not just its orientation relative to gravity, is necessary.

### 4.3.2 Tracking for Modes of Natural Interaction

#### 4.3.2.1 Orientation-tracking

The orientation-tracking component provides real-time data describing how the screen has been tilted or rotated by the user. This information is necessary for assessing the orientation of the surface relative to lighting when calculating reflections for the virtual surface. The use of real surface orientation information for the \((\theta_i, \phi_i)\) incident lighting angles in the BRDF calculations is part of the Level 5 capabilities from the conceptual design.

In the initial tangiBook system, tracking was implemented using the triaxial accelerometer in the laptop’s Sudden Motion Sensor (SMS) along with the SMSLib software library (Suitable Systems, www.suitable.com). With the customizable system, an ActionXL Wired Motion sensor FP100W containing a triaxial accelerometer was mounted to the back of the display screen. A triaxial accelerometer provides information for tracking the display’s orientation relative to the world by relating the three screen directions \((uvw)\) to the direction opposite gravity (the \(y\) axis of the world coordinate system). This information is used to define a rotation matrix, \(R\), for calculating the coordinates of the screen-affixed \(uvw\) vectors in the
world space. The axis and degree of rotation is determined by comparing the s vector data from
the sensor to the sensor reading, r, in the reference state. The sensor reports the direction
opposite gravity in terms of uvw components. In the reference state, the (negative) gravity vector
is aligned with the v-axis (Figure 19, left), so r = (0, 1, 0). The rotation necessary for specifying
the uvw frame in xyz world coordinates is given by the matrix that transforms the s vector back
to the reference state, where it is consistent with the world system. The axis of rotation (a) is
calculated from the normalized cross product of the reference state vector (r) and the current
state vector (s):

$$a = \frac{s \times r}{\|s \times r\|}$$  \hspace{1cm} (53)

The angle of rotation is calculated from the dot product of the r and s (unit) vectors:

$$\theta_{rotation} = \cos^{-1}(s \cdot r)$$  \hspace{1cm} (54)

The final 3 x 3 rotation matrix R is calculated from the axis of rotation and angle (details on
the mathematics for calculating rotation matrices can be found in the text by Strang [1993]). The
xyz space coordinates of the screen’s three axis vectors in their initial state, $u_0 = (1,0,0)$,
v_0=(0,1,0), and $w_0=(0,0,1)$, are multiplied by R to specify the screen’s current orientation in
world coordinates:

$$u = Ru_0$$
$$v = Rv_0$$
$$w = Rw_0$$  \hspace{1cm} (55)

It is convenient to use triaxial accelerometers because they are inexpensive and readily
available in a range of devices, but they have an important limitation. Because they detect
orientation relative to the direction of gravity, they are not able to detect rotations of the display
that are made around the axis aligned with gravity (the y-axis). The accelerometer can detect if
the display is tilted backward or rotated in plane so that the screen is upside-down, but does not detect when the screen is turned to face in a new compass direction. In the customizable system, this could be addressed by replacing the accelerometers with a six degree-of-freedom tracking system. Mathematically, it would be simpler to use one of these systems because they generally provide a direct estimate of the final rotation matrix or the vectors for the $uvw$ frame. However, these were not integrated for practical reasons, as these tracking systems generally exceed the cost of all the other components in a tangible display system combined.

4.3.2.2 Viewpoint-tracking

Dynamic viewpoint tracking is used in conjunction with orientation tracking to enhance the degree of interactivity. Real-time tracking of the observer’s viewing position is an important capability for conveying the appearance of surface gloss or directionally-dependent (gonio-apparent) color effects (Level 3). Accurate tracking of the user’s position relative to the display screen allows for the use of an “active” paradigm for representing view-dependent changes to the virtual surface depicted on screen. As the observer moves relative to the display, the change in the real viewing direction must be used to update the $(\theta_r, \phi_r)$ angles in the BRDF calculation to accurately convey specular surface reflections (Level 5).

In both versions of the tangible display system, the location of the user is estimated using computer-vision-based head-tracking. The built-in camera is used for the tangiBook system, and in the customizable system, a Logitech Quickcam Pro 9000 webcam was attached to the top of the display screen. The location of the head in the stream of camera images, along with information about the camera’s physical orientation and position determined from orientation tracking, is used to estimate the position of the eye-point in world coordinates. The position and
size of the head in each image is determined using the Haar cascade algorithm from OpenCV [Lienhart & Maydt 2003]. The location of the eye-point in the image is estimated by adding an offset to the head center position. The size of the head radius in the image is used to estimate the approximate distance of the viewer from the screen. The eye position in three-dimensional space is determined by relating directions in the image plane to the \( uvw \) coordinate system and using an ideal pinhole camera model to estimate physical distances. A detailed description of the mathematical modeling used to estimate the 3D position of the eye from the camera data is provided in Appendix C.

There are advantages to using a markerless head-tracking approach for determining the observer’s position, but also limitations that led to the use of different tracking technology in the final object display system. The primary advantages are that the observer does not need to wear any special equipment and it can be performed using commonly available camera systems. However, markerless tracking is less robust and may fail to maintain a consistent viewing position in complex environments or if multiple faces are within the field-of-view of the camera. This was addressed in the final system by incorporating an IR-based camera tracking system (NaturalPoint TrackIR, http://www.naturalpoint.com/trackir), which has the user wear a small apparatus with a known geometric pattern of IR reflective dots.

4.3.3 Surface modeling (geometry, texture, gloss, and color)

Tangible display systems are intended to reproduce complex virtual surfaces with color, gloss, and texture properties and simulate their changing appearance based on the physical interactions detected in the tracking module. The tangible display software incorporates three
types of modeling to create virtual objects: polygonal geometry to give the shape of the surface, normal maps to provide the small scale surface texture, and bidirectional reflectance distribution function (BRDF) models to represent the color and gloss properties of the material. The virtual surface is modeled in such a way that it appears to be at the physical location of the screen, and it is rendered from a camera position that makes it respond to orientation and viewpoint changes as a physical surface would.

4.3.3.1 Geometry

The principal geometry of the virtual object is modeled as a rectangular surface that coincides with the physical surface of the display screen (Level 4). The tangent-bitangent-normal (tbn) frame of the virtual surface, used for BRDF shading calculations, is matched to the uvw coordinate frame describing the physical orientation of the display screen (as shown in Figure 20). By aligning the tbn frame of the virtual surface with the physical orientation of the display screen, reflections calculated based on the BRDF of the virtual surface will be consistent with an object at the current orientation of the display screen (Level 5).
The polygonal geometry of the virtual object surface is modeled as a rectangle that is coincident with the plane of the physical screen. The tbn frame used for surface shading calculations is aligned with the uvw coordinate system used to describe the orientation of the display screen.

4.3.3.2 Mesoscale Geometry (Texture)

With planar geometry alone, the virtual surfaces would appear smooth. Additional modeling has been incorporated to simulate the texture that results from the mesoscale geometry of the surface (Level 6). A principal effect of this small scale geometry is that it modulates the orientation of surface facets such that they are no longer perfectly aligned with the orientation of the principal planar geometry. This orientation change is modeled in tangible display systems using spatially varying object-space normal maps that describe the adjustment to the geometric surface normal orientation at each position [Peercy et al. 1997]. At each \((ix, iy)\) image position, the object space normal map provides a set of scalars describing the mesoscale normal in uvw coordinates \((u_{map}(ix,iy), v_{map}(ix,iy), w_{map}(ix,iy))\). After applying the map, the spatially varying normal vector for the object’s surface \((n_{(ix,iy)})\), in world coordinates, becomes:
The three directional components of spatially-varying normal maps are typically encoded in the channels of an RGB image. An example normal map and the corresponding rendered image of the surface texture are shown in Figure 21. Though this type of modeling accounts for the change in surface orientation, it does not account for the effects produced by the physical height of the mesoscale geometry. In the final object display system, the shadowing effects of the mesoscale geometry are also simulated.

\[
\mathbf{n}_{(ix, iy)} = u_{\text{map}(ix, iy)} \mathbf{u} + v_{\text{map}(ix, iy)} \mathbf{v} + w_{\text{map}(ix, iy)} \mathbf{w}.
\]  

(56)

**Figure 21.** Left, a normal map encoded in an RGB image, with the \( \mathbf{u} \) direction represented by the red channel, the \( \mathbf{v} \) direction represented by the green channel, and the \( \mathbf{w} \)-direction by the blue channel. Right, the textured surface rendered from the normal map.

### 4.3.3.3 Material properties: Color and Gloss

In tangible display systems, the color and gloss properties of the surface are modeled using a trichromatic BRDF representation. The diffuse, specular, and roughness properties of the material are specified by a set of maps corresponding to the three components of the Ward
BRDF model [Ward 1992]: $\rho_d$, $\rho_s$, and $\alpha$. In the final object display system, the Ward-Dür model was used in place of the original Ward [1992] model. The same parameters are used in the Ward-Dür model, but the specular lobe normalization is modified [Dür 2006]. The $\rho_d$ and $\rho_s$ parameters are specified in three color channels to provide diffuse color information and allow for colored specular highlights when simulating metals. In Figure 22, a trichromatic BRDF model is used to add the appearance of color and gloss to the textured surface from Figure 21.

![Figure 22](image)

**Figure 22.** Left, the textured surface rendered from the normal map alone. Right, the surface rendered with diffuse color and gloss, specified by a BRDF model in three color channels.

With tangible displays, the color representation was limited to three channels, but ideally, the diffuse color of a surface would be specified by a $\rho_d$ parameter value at each wavelength, providing the full spectral reflectance factor data needed to calculate accurate colorimetry. The limitations of trichromatic color processing were less of a concern for tangible displays, which were intended to render with virtual illumination and did not attempt to calculate color for
changing real-world illumination condition. This virtual illumination was typically represented by simple point lights, modeled as equal energy, or with standard environmental illumination maps that were only captured in three channels. In the final object display system, a multispectral workflow was incorporated to overcome some of the limitations of a trichromatic rendering pipeline. The multispectral rendering pipeline is described in detail in Chapter 6.

4.3.4 Rendering

The rendering component of a tangible display uses the information collected by the tracking module along with the geometry and BRDF models of the surface to calculate realistic reflections for a virtual illumination environment. It includes custom software to integrate the tracking information into the rendering process. Rendering calculations are implemented using custom OpenGL shaders and performed on the GPU to allow them to be completed at interactive rates.

The BRDF shading methods used for tangible display systems are based on the Ward model shader described by Rost [2006] and implement an isotropic form of the original Ward model (described by Eq. (15)). The orientation information from the tracking module is used within the custom vertex shader to update the position of the virtual object as the display is rotated. The user’s estimated viewing position from the tracking module and the interpolated surface positions from the vertex shader are used in the fragment shader to determine the surface-to-viewing point vector (providing the $\theta_r$ BRDF angle) in the Ward model’s reflectance calculations. The surface normal at each fragment position is specified by the spatially-varying normal map. The properties of the material are provided to the shader using either single values
of $\rho_d$, $\rho_s$ and $\alpha$ per channel (in three channels) to describe a surface with uniform properties or specified with maps to describe materials with spatially-varying BRDF.

### 4.3.4.1 Illumination Methods

Two methods are available for illuminating surfaces on tangible displays. For applications where complex patterns of illumination are necessary, the surface is illuminated with image-based cubic environment maps [Debevec 1998]. (The maps used were of real-world environments, but were ones captured by Debevec and were not the physical environment surrounding the display screen). The surface normal and the direction from the surface to viewpoint are used to calculate the direction of the specular reflection. A small number of angles around the specular reflection direction are then uniformly sampled, used to index the cubic environment map to find the illumination color at that location, and the contribution of each to the overall sum, based on its direction, is calculated with the Ward model. A diffuse cubic map of the environment is also used and indexed by the surface normal. The amount of diffuse light reflection is weighted relative to specular reflectance and the light from the two is summed to determine the fragment color. Though the reflections appear realistic, the calculations using the environment map technique may not accurately reproduce the properties specified by the BRDF parameters, due to the sparse uniform sampling of the specular lobe. The second technique, point lighting, more accurately represents the BRDF properties, but produces much simpler reflection patterns. The lights are specified by the position and color of a set of points. The reflections due to each of the virtual lights are calculated from the light color and intensity, the incident lighting and viewing direction, and the Ward model parameters. More advanced
illumination methods are used in the object display framework to reproduce the patterns of illumination for the real light sources that surround the display screen. The rendering pipeline for the object display is described in Chapter 7.

4.3.4.2 Viewing Positions for Rendering

When rendering to the display, the objective is to treat the screen as a physical object with virtual material properties and not as a window into a virtual scene (Level 4). This effect is produced by using the real viewing position of the observer when calculating the fragment colors (the surface reflections) in the shader, but using a fixed viewpoint when rendering the surface to the screen’s viewport. In traditional rendering workflows, the shading viewpoint and virtual camera point are typically the same, producing the windowed effect. In tangible displays, the dynamically-tracked viewing position and its true relationship to the screen’s current orientation are used to calculate the light reflecting toward the viewer from each position on the virtual object’s surface (as shown in Figure 23). The virtual camera, however, is positioned at fixed difference from the virtual surface, along a normal ray intersecting the center of the surface, so that the rendered image stays fixed in the viewport onscreen.
Figure 23. The virtual camera used to render to the viewport stays normal to the surface at a constant distance. The tracked position of the observer is used when calculating surface reflections in the shader.

4.4 Tangible Display vs. Object Display Systems

A complete object display system requires five primary components: a characterized lighting environment, a characterized display screen, a representation for material properties of the surface, devices for tracking the display and user, and an interactive rendering module that can calculate accurate virtual surface reflections based on the observer’s viewpoint and the properties of the real lighting environment. The initial tangible display systems included screen, tracking, surface property, and rendering components, but did not incorporate a characterized lighting environment. Additionally, they did not include a rendering workflow capable of producing accurate color and luminance patterns based on the spectral composition and geometry of real-world lighting. With respect to the conceptual framework, tangible display systems provided a BRDF representation consistent with incident and observation changes (Level 5) as
well as the per-pixel surface orientation (*Level 6*), but did not provide shadowing capabilities (*Level 7*), or more importantly, matched physical lighting. To meet the *Level 8* objectives of the object display system, the virtual lighting environment used to render the virtual surface must be consistent with the real-world illumination surrounding the display screen.
5 OBJECT DISPLAY SYSTEMS

Tangible display systems were designed to allow users to physically interact with virtual objects shown onscreen like they would with real objects. Object display systems extend this concept in an effort to make the virtual objects appear like interactive physical objects reflecting the light around them. The object display system is designed to recreate colorimetrically the patterns of reflected light that a physical surface, positioned at the screen’s location in the real lighting environment, would produce in the direction of the observer.

Toward this goal, an object display system incorporates sensing and modeling of the real-world lighting along with the tracking methods of a tangible display. In addition, it includes a multispectral real-time rendering engine to simulate the types of light-surface interactions that contribute to the appearance attributes of a surface. The results are displayed through a photometrically-calibrated screen so that the luminance and chromaticity of the emitted light can be matched to the light that would be reflected by an object at the screen’s position.

In this chapter, the implementation of an object display system is described for a specific real-world lighting environment, in this case a viewing booth with multiple overhead lighting options (shown in Figure 24). This system and its capabilities were described in the paper [Darling & Ferwerda 2012] and are discussed in greater detail in this chapter. The light booth environment was selected because it is representative of the viewing environments for some intended applications of object displays and it is well-suited to modeling in the object display framework. This environment is used to illustrate capabilities of an object display system and the technologies involved; however, the general object display framework can be applied to a range of lighting configurations and is not limited to this specific one.
Figure 24. Light booth environment used for the implementation of an object display system. The illumination in the booth is an overhead luminaire with multiple lighting options. Virtual objects are rendered to the screen, which has an orientation sensor attached (not shown). The IR-based system for tracking the observer position is shown at the front of the booth. A spectral sensor at the rear of the booth is directed at the light to estimate the spectral composition of the illumination.

A flowchart illustrating the interaction of the main components of an object display system is shown in Figure 25. Surfaces are represented with models of the color, gloss, and texture properties. The interactive tracking component collects information on the physical viewing position of the observer and the physical position and orientation of the screen surface. The real lighting environment is modeled in terms of the spectral properties of the illumination, a geometric model of its position in space, and spatial maps of the luminance levels across the light source. This information is entered into a real-time rendering engine that uses it in a 3D computer graphics simulation of the light interaction with the surface. The calculations are
performed in a set of six multispectral channels and the results are converted to XYZ (specified in terms of absolute luminance). The colorimetric results from the rendering engine are sent to the display component, which uses an inverse display characterization model to output the results through the screen. The display component also uses information from the tracking module to estimate the viewing direction of the observer and compensate for viewing angle dependencies in the LCD panel.

**Figure 25.** Diagram illustrating the interaction of the five main components in an object display system.
5.1 Object Surface Modeling

A key element of the object display framework is that the properties of the virtual surfaces, including diffuse color, are represented in a manner that is independent of illumination. With this type of illumination-independent representation, the virtual surface can be interactively updated for changes in the real-world lighting, in the same manner that a physical sample would change.

5.1.1 Diffuse Color

With the object display, color calculations are typically performed in an expanded set of multispectral channels instead of the simpler trichromatic color workflow used with tangible displays. This increased number of channels allows more spectral information to be maintained about the surface reflectance and allows for more accurate lighting when applying a range of illumination. These multi-channel rendering calculations are based on the multispectral factor methodology developed by Drew & Finlayson [2003], where results are calculated by multiplying the per-channel coefficients of reflectance and illumination in a larger number of channels than an RGB workflow. For the object display system, an optimized set of multispectral rendering channels were selected and an end-to-end color pipeline was developed to bring captured data into the system, perform rendering calculations, and convert the final results to XYZ for display. This multispectral color pipeline is described in detail in Chapter 6.

The diffuse color data for virtual surfaces are represented in a multispectral form similar to reflectance factor, where the reflectance coefficient for each wavelength band varies between 0 and 1. To support spatially varying color for objects such as paintings or digital print proofs,
six-channel reflectance data can be encoded in two three-channel (RGB) floating point images. Each image channel stores the Ward-Dür $\rho_d$ parameter for one of the multispectral primaries.

### 5.1.2 Gloss

The gloss properties of the surface are represented using the specular reflectance parameters of the Ward-Dür BRDF model (Dür [2006] modified the specular lobe normalization from the original Ward [1992] model). The $\alpha$ (specular roughness) parameter is used to describe the width of the specular lobe. The parameter describing the magnitude of the specular reflectance, $\rho_s$, may be specified for each of the six multispectral channels to allow for spectrally-selective front surface reflection. The gloss parameters can be specified in floating point image maps to support spatially-varying surface gloss.

### 5.1.3 Small Scale Geometry, Texture, and Shadowing

The system is intended to display surfaces with principally planar geometry, like the real screen has, so that the locations of virtual reflections are consistent with the physical location of light emitted from the screen. To support object surfaces with some texture or relief, small scale geometry (surface height $\ll$ viewing distance) is modeled as a dense height field relative to the physical plane of the screen. The effect of this geometry on the orientation of surface facets, relative to the planar screen surface, is represented with image-based normal maps [Peercy et al. 1997]. The height of each surface point above the plane of the screen (along the screen normal) is stored in a floating point height map. When shading, this small height is added (in the
direction of the screen normal) to the physical 3D position associated with its point on the screen.

The self-shadowing that results from higher surface points blocking light from reaching lower surface points is represented by image-based horizon shadow maps [Max 1988; Sloan & Cohen 2000]. The determination of shadowing from a surface height map is a computationally intensive process, but can be simplified before rendering by pre-calculating this intermediate set of maps. The horizon maps store information, for each position of the surface, on the minimum elevation angle required so that a light source will not be blocked by some intervening surface point. The minimum elevation angle may differ for each surface direction (azimuth angle), so it is necessary to calculate multiple horizon maps corresponding to different azimuth angle ranges. Ideally, there would be a separate horizon map for each small change in azimuth angle, but it is not feasible to store or render with a large number of high-resolution floating point horizon map images. Past real-time implementations have divided the azimuth angle into the eight compass directions [Sloan & Cohen 2000]. To obtain smoother shadowing in the object display system, the azimuth angle is divided into 18 segments (of 20 degrees each). Using three channels per image, the 18 horizon angle maps are stored in six floating-point horizon map images (shown in Figure 26). For computational efficiency when rendering, the horizon maps store the tangent of the minimum elevation angle, which directly provides the ratio of elevation above the surface to distance along the surface. The off-line generation of these horizon maps is a computationally intensive process that can require a significant amount of processing time on the CPU. A GPU-based approach was developed to allow these maps to be generated more rapidly and is described in Appendix B.
Figure 26. A set of six horizon maps for a painting model. The maps are storing the tangent of the minimum elevation angle above the surface necessary to insure that a light source will not be occluded by an intervening surface point. The six RGB maps encode this minimum clearance angle for 18 azimuthal angle bands, with one angle band stored in each color channel.
During rendering, the pre-calculated horizon maps allow the occlusion of a lighting sample to be checked quickly for every position on the surface. In the fragment shader, the azimuth direction to the light source is calculated for each surface position. A conditional statement is used to select the 20° degree band in the horizon maps that contains the calculated azimuth angle. If the angle to the light source is below the elevation angle stored in the horizon map channel, the light is not included in the rendering calculation or is considered to a small degree (e.g. 20%) to produce lighter shadowing. To reduce computation, the tangent of the elevation angle for the light is compared to a stored tangent value. The self-shadowing effect, calculated from a set of horizon maps, is illustrated in Figure 27 with a set of cropped screen-capture images that were rendered on the object display. In the series of images, the length of shadows increases as the screen normal is tilted further away from the light.

**Figure 27.** Using a horizon map-based technique, virtual shadowing increases as the object display screen is rotated away from the light (left to right in the screen capture image sequence).
Unlike shadowing, the modeling of the two other height-based effects, occlusion parallax and inter-reflection, can not be easily pre-calculated to reduce the computational load during the rendering process. Though there are not fundamental constraints on reproducing these effects in the object display framework, there were practical limitations for implementing these effects using current graphics processing hardware. With future advancements in graphics hardware, implementing these effects on an object display may become more practical. Additionally, with more powerful hardware, it may be possible to calculate shadowing directly from the height map at interactive rates and remove the extra step of calculating intermediate horizon maps.

5.2 Modeling the Real-World Light Booth Illumination

The system uses a model of the real-world illumination in the screen’s environment to allow the virtual surface to be rendered in a manner consistent with a physical surface at the same position. For the characterized lighting environment, a light booth (shown in Figure 24) was selected that has a tungsten source, a fluorescent D50-simulating source, and a fluorescent D65-simulating light source. The spectral power distributions of the sources, measured with a PhotoResearch PR-655 spectroradiometer, are shown in Figure 28. For each of the sources, the spatial configuration of the lighting elements also differs. Images of the three light sources (originally captured with an HDR series) are shown in Figure 29.
Figure 28. Measured spectral power distributions of the three light sources (shown at 5 nm intervals).

Figure 29. Images captured of the spatial luminance variation for the three lighting options: the tungsten light (illuminant A), the fluorescent D50-simulating light, and the fluorescent D65-simulating light.
5.2.1 Interactive Spectral Sensing

To account for the spectral composition of the illumination, the ambient light in the booth is continuously monitored using an Ocean Optics USB2000+ spectrometer, which samples the entire visible spectrum at a refresh rate of approximately five spectra per second. The sensor is attached to a pole located behind the screen and directed toward the top of the light booth, as shown in Figure 30. The sub-nm spectral data from the sensor are smoothed and resampled at 10 nm. The data are then converted to the six-channel multispectral representation with the [36 x 6] matrix representing the spectral sensitivity curves of the six optimized rendering primaries (described in Chapter 6).

Figure 30. Ocean Optics USB2000+ sensor used to interactively estimate the spectral composition of the illumination.
The system has two modes of spectral operation. In the first mode, the direct spectral sensor mode, the continuously-updating six-channel illumination values calculated from the sensor data are used directly when rendering the color of virtual surfaces. The six-channel values are uploaded to the GPU after each update from the sensor and are used as the illumination representation in the rendering shader. The second mode is a classifier mode, where spectral data for each light source is pre-measured and the calculated six-channel illumination values are stored. At rendering time, the spectral sensor is used to select between the three possible sources. To select the light source, the sensed six-channel values are compared to the six-channel values of each of the pre-measured sources. The sum squared error for each is calculated, and the one with the smallest SSE value is used. The classifier mode could potentially allow a simpler type of sensor to select between pre-measured light sources, though currently only the full spectral sensor has been used.

5.2.2 Geometric and Spatial Luminance Modeling

In addition to spectral information, the geometric configuration and luminance of the light sources are also modeled. This information is used to estimate the incident irradiance levels at the virtual surface so that diffuse reflections can be rendered at the appropriate luminance levels and specular reflections can be rendered with correct spatial patterns.

The planar geometry for the ceiling of the light booth is specified in terms of a global coordinate system to describe its physical size and location (described in Section 7.5 of the gloss workflow chapter). Though the ceiling has a diffuser, there is still significant luminance variation across different regions of the booth light and this varies with the selection of the light source.
The spatial luminance patterns for each of the light sources were captured in an offline process by taking a high-dynamic range image series of each light with a camera calibrated to estimate absolute luminance. These images (shown in Figure 29) are geometrically mapped to the planar geometry of the light booth ceiling to place the luminance patterns at their physical locations in the environment (shown in Figure 31). The luminance pattern for the active light is chosen interactively from the three options using a classifier that selects the most likely light given the sensed spectral data.

![Image](image-url)

**Figure 31.** Spatial luminance images for the tungsten light (left) and D65 light (right) are mapped to the geometry of the light booth ceiling.

To simplify rendering of diffuse reflections, a variance-minimizing median cut algorithm [Viriyothai & Debevec 2009] is applied to the mapped luminance images to generate a set of representative point lights. These points store physical XYZ positions along with summed luminance values that represent different regions of the surface. The set of 32 point lights that
were identified for the tungsten light and for D65 light using the median cut algorithm are shown in Figure 32. In the second row of the figure, the points are shown at their positions in the environment.

Figure 32. Left, the 32 point median cut approximation of the tungsten light. Right, the median cut approximation of the D65 light source. In the second row, the point lights are shown at their position in the 3D environment.
5.3 Display and Observer Tracking

5.3.1 Tracking the Observer Viewing Position

In the object display system, the webcam-based observer tracking was replaced with a more accurate off-the-shelf IR tracking system (shown in Figure 33), the NaturalPoint TrackIR (www.naturalpoint.com). The user wears a small apparatus with a known geometric pattern of IR reflective dots (typically attached to a hat). The dots are illuminated by a set of IR lights in the camera module and the set of reflections are detected by the IR sensitive camera. A software API included with the tracker calculates the position of the IR dots relative to the camera position. The use of an IR tracking system provides increased robustness over head-tracking with a standard camera because the points of IR light can easily be detected in the specialized camera image, the known geometric pattern allows for a more precise estimate of distance and pose, and the ability to detect the intended viewer is not affected by the presence of additional faces in the camera FOV.

Figure 33. IR-based tracking system (NaturalPoint TrackIR). The camera module detects the position of the set of three IR reflective dots, which are attached to a hat or visor worn by the observer.
Unlike the tangible display systems, where the camera moves with the display screen, the tracking camera in the object display is usually kept in a static position in front of the screen. The camera provides the physical position (in cm) of the detected markers relative to its location. The geometric configuration for determining the position of the observer in the world space is illustrated in Figure 34. If the camera is aligned with the world axes, such that the camera is pointed along the +Z axis and the up-vector is pointed at the +Y axis, the physical position of the observer in the world coordinate system is simply the sum of the camera’s physical position and the reported relative position from the tracking system. In general, the camera is either left completely aligned or tilted upward in the Y-Z plane by a small angle $\theta$. In this case, the position of the observer in the world space is:

$$p_{\text{observer}} = O_{\text{IR cam}} + m_U \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} + m_V \begin{bmatrix} 0 \\ \cos \theta_{\text{cam}} \\ \sin \theta_{\text{cam}} \end{bmatrix} + m_W \begin{bmatrix} 0 \\ \sin \theta_{\text{cam}} \\ \cos \theta_{\text{cam}} \end{bmatrix}$$

(57)

where $O_{\text{IR cam}}$ is the position of the camera in the world (the origin of the relative camera coordinate system), and $m_U, m_V, m_W$ are the components of the detected physical position of the marker apparatus in the camera’s coordinate system (where V is the up-vector direction and W is direction of the principal ray).
Figure 34. Side view diagram for calculating the position of the observer using the IR camera tracking system (the $+X$ axis is directed into the page). The camera system provides the position of the observer relative to the camera position. The fixed position and tilt angle of the IR camera (in the Y-Z plane) are used to calculate the absolute observer position in same world coordinate system as the screen surface and light.

5.3.2 Tracking the Screen/Virtual Surface

In addition to tracking the physical position of the observer, the display is also tracked to estimate its orientation and the physical positions of points on the screen. In the current object display system implementation, screen tracking is still performed using the triaxial accelerometer from the customizable tangible display system. However, because the absolute physical position of the screen is needed and not just its rotation about the origin, it was necessary to incorporate a model of how the screen rotates on its stand. A diagram illustrating the screen rotation model is shown in Figure 35. The screen model accounts for two types of manipulations of the screen on its stand. The screen can be tilted up or down (toward or away from the overhead light) or it can be rotated in plane (changing the screen from a landscape to portrait viewing orientation).
Figure 35. Diagram illustrating the estimation of the physical position of the screen. The diagram is a side view with the +X axis directed into the screen. Left, the measured position of the screen center in its initial state (vertical) is specified. Right, when the screen is rotated, the position of the screen origin in the world is calculated based on the \([r u n]\) frame orientation and the fixed point of rotation.

As the display is manipulated, the physical locations of points on the screen are estimated based on the UVW rotation frame from the tracking sensor and a model of how the rotation changes the physical position of the screen origin (the top left corner, when the screen is vertical). (In the diagram, the UVW frame is referred to by the vectors \(r_{\text{screen}}\) \(u_{\text{screen}}\) \(n_{\text{screen}}\), specifying the directions to the right, up, and normal to the screen.)

When tilting the screen on its stand, the fixed center of rotation is a point at a distance of \(d_{arm}\) (19 cm) behind the center of the screen, in the direction opposite the screen normal. To determine the center of rotation position, the physical world position of the screen center \((p_{\text{Init,center}})\) is first specified in the reference state from physical measurements (in the reference state, the \(r_{\text{screen}}\) \(u_{\text{screen}}\) \(n_{\text{screen}}\) directions are aligned with the \(XYZ\) world frame directions). The center of rotation is then calculated (in the reference state) by moving a distance \(d_{arm}\) in the \(-Z\) direction (equal to \(-n_{\text{screen}}\):
\[
\mathbf{p}_{\text{rotation}} = \mathbf{p}_{\text{init.center}} + \begin{bmatrix} 0 \\ 0 \\ -1 \end{bmatrix}.
\] (58)

This fixed center of rotation point is stored for use when the display is rotated. As the display is tilted, the \( \mathbf{r}_{\text{screen}} \) \( \mathbf{u}_{\text{screen}} \) \( \mathbf{n}_{\text{screen}} \) frame is updated by the tracking sensor and these directions are used to estimate the new position of the screen origin point. This is determined by starting at the fixed center of rotation point (\( \mathbf{p}_{\text{rotation}} \)) and moving along the \( \mathbf{n}_{\text{screen}} \) vector a distance \( d_{\text{arm}} \) to the current position of the screen center:

\[
\mathbf{p}_{\text{center}} = \mathbf{p}_{\text{rotation}} + d_{\text{arm}} \mathbf{n}_{\text{screen}}
\] (59)

The current position of the screen origin (top left) is calculated by moving along the screen up vector a distance equal to half the screen height (\( h \)) and along the negative right vector a distance of half the screen width (\( w \)):

\[
\mathbf{o}_{\text{screen,TL}} = \mathbf{p}_{\text{center}} + \frac{h}{2} \mathbf{u}_{\text{screen}} - \frac{w}{2} \mathbf{r}_{\text{screen}}.
\] (60)

This formula also accounts for in-plane rotation. The calculated center point remains the same for in-plane rotation, as there is no change in the normal, only the \( \mathbf{u}_{\text{screen}} \) and \( \mathbf{r}_{\text{screen}} \) directions. The new origin point is calculated by moving from the center point along the new \( \mathbf{u}_{\text{screen}} \) and \( \mathbf{r}_{\text{screen}} \) directions in Eq. (60).

Physical locations on the virtual surface are determined based on the screen origin point and the \( \mathbf{r}_{\text{screen}} \) and \( -\mathbf{u}_{\text{screen}} \) directions along the screen. A diagram illustrating the geometry for calculating positions on a virtual surface is shown in Figure 36.
Figure 36. Diagram illustrating the geometry for determining the physical position of a point \((s, t)\) on a virtual surface. The origin of the surface is specified by two offsets \((d_r, d_u)\) from the screen origin along the \(r_{\text{screen}}\) and \(u_{\text{screen}}\) directions. The physical size of the surface is specified by \(l_s\) and \(l_t\).

The origin of the virtual surface, \(o_{\text{surf,TL}}\), is specified by a physical offset from the screen origin in the plane of the screen:

\[
o_{\text{surf,TL}} = o_{\text{screen,TL}} + d_r r_{\text{screen}} - d_u u_{\text{screen}} \tag{61}
\]

where \(d_r\) is the offset (in cm) in the \(r_{\text{screen}}\) direction (toward the right) and \(d_u\) is the offset in the negative \(u_{\text{screen}}\) direction (downward). Starting from the surface origin point, the physical position of a surface pixel \((s, t)\) is determined using the formula:
\[ \mathbf{p}_{\text{surf}(s,t)} = \mathbf{o}_{\text{surf},TL} + l_s \left( \frac{s}{s_{\text{total}}} \right) \mathbf{r}_{\text{screen}} + l_t \left( \frac{t}{t_{\text{total}}} \right) (-\mathbf{u}_{\text{screen}}) + H_{\text{map}(s,t)} \mathbf{n}_{\text{screen}} \]  

(62)

where \( s_{\text{total}} \) is the total number of map pixels in the horizontal direction, \( t_{\text{total}} \) is the total number of map pixels in the vertical direction, \( l_s \) is the physical surface length (in cm) in the horizontal direction, \( l_t \) is the physical width in the vertical direction, and \( H_{\text{map}} \) is the surface height specified in the height map (converted from mm to cm) for pixel \((s, t)\).

In future work, the display tracking process could be simplified by attaching a six-degree of freedom (6DOF) tracker to the display screen. A 6DOF tracking system would directly provide the \( \mathbf{r}_{\text{screen}}, \mathbf{u}_{\text{screen}}, \mathbf{n}_{\text{screen}} \) frame as well as the absolute position of the origin surface point, if the tracking base station were placed at the world origin and aligned with the XYZ axes. With this type of tracker, it would be possible to account for all types of rotation and translations and not just ones supported by the model of rotation of the specific display stand. In practice, however, there are potential issues with using six-degree of freedom trackers. Beyond the high cost, there can also be accuracy issues if magnetic trackers are used around certain types of electronics equipment. It may be necessary to evaluate multiple systems to determine which one can maintain accuracy while attached to a display screen.

### 5.4 Rendering Overview

The rendering component of the system uses the information on the screen and observer position along with the BRDF properties of the virtual surface to calculate the surface reflections for the modeled real-world lighting. These calculations are performed on the GPU and
implemented using custom OpenGL shaders to allow them to be completed at interactive rates. Diffuse reflections are calculated by iterating over the set of 32 pre-generated median cut lights for the currently active booth light source. The magnitude of the illuminance $E$ at each screen location is calculated based on the physical light-to-screen-pixel distance and the orientation of the normal-mapped pixel relative to each light source:

$$E = \sum_{n=1}^{32} \frac{L_n (N_{surf} \cdot I_n) (N_{light} \cdot -I_n)}{d_n^2} \left( \frac{A}{P} \right)$$

(63)

where $I_n$ is the surface-point to light-point unit vector for the $n^{th}$ light point, $d_n$ is the distance between these points in meters, $L_n$ is the summed luminance stored in the $n^{th}$ light point, $N_{surf}$ is the surface normal at the point on the virtual object, $N_{light}$ is the normal to the plane of the area light, and the area term $A/P$ is the physical area of the captured light surface (in m$^2$) divided by the number of pixels in the light image (the geometry is illustrated in Section 7.5, in the detailed chapter on rendering). This physical illuminance value is used to scale the product of the normalized six-channel multispectral illumination power distribution ($S_j$) and the multispectral diffuse reflectance ($\rho_{d,j}$) to determine the per-channel diffuse reflections on a luminance-based scale:

$$L_{out,j} = \left( \frac{\rho_{d,j}}{\pi} \right) \left( \frac{S_j}{E} \right), \text{ for } j = 1 \text{ to } 6.$$  

(64)

A similar calculation is performed to estimate the real diffuse reflection from the front surface of the display screen ($\rho_d = 0.002$), but with the real normal of the screen used instead of the normal-map modified virtual surface orientation. This estimated flare is subtracted from the virtual diffuse reflection to help mitigate the screen surface reflection present. The specular reflections
at each image point are estimated by real-time filtered importance sampling of the light source luminance image. The rendering algorithm is based on the real-time importance sampling method by Colbert & Křivánek [2007] and the Ward model sampling equations described by Walter [2005]. To improve the accuracy of specular reflections in the most recent version of the object display, a hybrid approach was developed to selectively switch between filtered importance sampling and a median cut method for evaluating specular reflections (the gloss reproduction pipeline based on these methods is described in detail in Chapter 7). The six channel results from the diffuse and specular reflection calculations are summed and multiplied by a [6 x 3] matrix transform to CIE XYZ for display to the screen.

5.5 Display Overview

The colorimetric results from the rendering engine provide the spatial pattern of light necessary to simulate the reflections from the virtual surface. The final stage is to translate these colorimetric values to the screen with a display characterization specified in absolute luminance levels. The screen used in the system is a high-luminance display (EIZO Radiforce RX220) capable of output levels as high as 900 cd/m² in a customizable mode and 400 cd/m² in the luminance-stabilized mode. The screen response was characterized with the Day et al. [2004] approach (based on the Fairchild & Wyble [1998] model form) using absolute luminance measurements made with a spectroradiometer. The determination of RGB digital counts for each pixel from the CIE XYZ output of the rendering engine is achieved at interactive rates using a GPU shader-based implementation for the inverse direction of the Day et al. characterization. An IPS-based screen was selected in an effort to minimize the impact of viewing angle on the
screen output, however, even with an IPS panel there was still a luminance decrease at wider viewing angles. A characterization model that includes compensation for screen luminance changes at the tracked viewing position of the observer was developed to address the viewing-angle dependent luminance falloff. The measurement and modeling of the display screen properties are described in Chapter 8.

5.6 System Capabilities

The system provides the capabilities to simulate diffuse color, surface texture, and gloss properties in a manner consistent with a real surface in the environment of the display.

5.6.1 Texture and Shadowing with Interactive Updating

In Figure 37, a model of a virtual painting is shown with the diffuse color removed to illustrate the appearance of shading and shadowing from the surface texture. In the image, the black border and white mask are real and the gray texture surface is rendered to the screen. Using the orientation information provided by the accelerometer attached to the screen, the rendered shading and shadowing automatically updates when the surface is physically tilted or rotated. This is illustrated in Figure 38, in a series of frames from a video of a user interacting with an object display. In the sequence, the screen begins tilted upward toward the overhead light and displays glossy highlights on the virtual surface. In the next two frames, the screen is slowly tilted downward, reducing the highlights and increasing the shadowing. Over the next three frames, the screen is tilted back up toward vertical and then is rotated in plane from a portrait to a landscape orientation (of the screen). With the virtual painting now in a new orientation, the
screen is again tilted downward and the shadows cast by the virtual surface still increase as the virtual surfaces is tilted away from the real overhead light.

Figure 37. The surface of a virtual painting shown without its diffuse color to illustrate the shading and shadowing from the mesoscale texture. The rendered gray surface is surrounded by a real white mat, black mat, and black frame.
Figure 38. Frames from a video illustrating the changes in shading and shadowing as the display is rotated.
5.6.2 Gloss

The system generates virtual reflections on surfaces that are consistent with the real spatial pattern of the booth lights. The spatial luminance images depicted in Figure 29 are sampled according to the surface BRDF to create realistic specular reflections (shown in Figure 39). Using a spectral classifier to identify which lighting option is active, the system automatically switches between the spatial luminance pattern models so that the reflections change when the real light in the booth is changed.

Figure 39. Left, the virtual reflection of the tungsten light source on a virtual curved surface. Right, the virtual reflection of the D65 source. Based on a spectral classifier, the system automatically identifies which light is present and switches the spatial light map used to calculate the specular reflections.
Using the tracking information from the IR camera system, the virtual reflections update with changes in the observer’s viewing position. This is illustrated in the sequence of images in Figure 40, which were captured from the tracked viewing position. As the observer moves from the left of the screen to the right of the screen, the virtual reflection of the tungsten light shifts from left to right correspondingly.

![Sequence of images](image)

**Figure 40.** Sequence of images captured as a camera with the tracking sensor attached is moved from left to right. The virtual reflection of the light updates its position with the change in the tracked viewing position.

### 5.6.3 Color

The display system uses interactive sensing of the spectral composition of the light and the multispectral rendering pipeline to automatically update the color of virtual surfaces for changes in the illumination. This capability is illustrated in Figure 41, where a real ColorChecker is shown side-by-side with a virtual model on the display system. The displayed color onscreen updates as the lighting changes between the D65, D50, and illuminant A booth lights and maintains consistency with the physical ColorChecker. Additionally, with spectral
sensing and a multispectral pipeline, the system has the capability to simulate metamerism, which would not be possible with a three-channel pipeline. A virtual target was created where each row contains a metameric pair of spectral reflectance curves. Each spectral pair was selected to create a metameric match under the tungsten booth light and to produce a color mismatch for the D65 simulator light. As shown in Figure 42, the samples appear to match under the tungsten light, but exhibit large color differences under the D65 light.

**Figure 41.** A sequence of images illustrating the system automatically updating the color of a virtual ColorChecker (right side) to continue to match the color of the real ColorChecker (left side) as the light changes between the D65 light (left image), D50 light (center image) and tungsten light (right image). The images were purposely not color balanced (all were set to a CCT of 5500) in order to clearly show the changing illumination.
5.7 Discussion

In this chapter, the implementation of an object display system was presented to demonstrate the technologies and capabilities of the larger framework. This prototype system supports color, gloss, and texture properties and provides methods for updating these attributes to maintain consistency with changes in the real-world illumination and the geometric relationship between the screen, observer, and lighting. Though there are limitations in the implemented system, this system takes steps toward the goal of presenting electronic objects on displays screens in a way that recreates the experience of directly viewing real surfaces.

One of the limitations in the example system relates to the current need to capture the spatial luminance patterns and create a geometric model of the real lighting in an offline process.
With the light booth used in the current prototype, the light sources have different spectral distributions so it is possible to automatically switch between the pre-generated spatial models with a spectral classifier approach based on the spectral sensor. The rendering engine is not limited to the light booth configuration and can handle a range of lighting options that can be represented as spatial luminance patterns mapped to geometric surfaces. However, if the lighting options do not have different spectral distributions, then the spectral classifier approach would not be able to automatically switch between them, and a different detection approach would be necessary to automatically update the spatial lighting model. An additional limitation of the prototype is that only the direct illumination from the light sources in the viewing booth is considered in the lighting calculations. Indirect light reflected from booth surfaces is not included during rendering. Currently there are dark coverings on the side walls and part of the floor to minimize the amount of unmodeled light present. In future work, the luminance patterns from these surfaces of the booth could be captured and included as light sources in the lighting model. Another limitation relates to the real physical reflection from the screen surface. In the light booth environment, where the light is primarily from above, the reflection from the screen toward the observer is relatively small and diffuse, so it can be modeled by diffuse reflectance parameters and subtracted from the virtual image. Lighting geometries where significant light is directly in front of the screen would require other methods to account for unwanted physical screen reflections.

Despite these limitations, the capabilities implemented in the example system demonstrate the potential for object display systems to produce virtual surfaces with color, gloss and texture properties that behave and appear like reflective objects in the real lighting present around the display.
6 MULTISPECTRAL COLOR REPRODUCTION FRAMEWORK

6.1 Overview

In the object display framework, as with real physical surfaces, the colors of virtual reflections are determined by the reflectance properties of the virtual object and the illumination in its environment. To act like a real surface, when the spectral composition of the lighting changes, the color of the rendered surface needs to change accordingly. This capability is supported in the object display framework using a multispectral representation for surface reflectance and for the spectral distribution of the ambient illumination. A real-time multispectral rendering framework was developed that allows color calculations to be performed interactively, so that the surface color can be automatically updated for sensed changes in the real world illumination. The multispectral color framework discussed in this chapter was described for use in general interactive rendering in the paper [Darling et al. 2011]. To facilitate color calculations in the object display system, the color framework was designed with the principal requirements:

- Accurate colorimetry for a range of typical illuminants
- A maximum of six channels, so that processing can be performed with two passes of a standard RGB workflow
- A single set of rendering primaries, so surface and lighting data can be maintained in one standardized form
- Support for illumination and reflectance data captured using a range of spectral measurement and imaging systems.
6.2 Workflows for Color Processing

Typically in interactive computer graphics simulations, colors are calculated with the three-channel RGB representation commonly used in traditional imaging systems. Illumination and surface reflectance are specified in three channels and multiplied together to simulate the color of the reflection (as illustrated in the left panel of Figure 43). A trichromatic representation can produce accurate results if only one lighting spectral distribution must be considered, but does not provide enough spectral information to accurately determine the color of a surface over a range of lighting distributions, as is required for an object display system.

Physically, the process simulated by computer graphics rendering, applying illumination to surfaces and calculating reflections, is a spectrally-based process. The colors of surface reflections are a product of the spectral power distributions of the illumination and the (bidirectional) spectral reflectance factors of the surfaces. After this multiplication is performed, color information can then be reduced to a trichromatic form using the CIE standard observer (as illustrated in Figure 44). Though they begin with surface data captured in an abridged set of channels, multispectral imaging workflows (e.g. Hardeberg et al. [1999], Berns et al. [2005]) also typically use the full-spectral calculation so that full-spectral illumination can be applied to an estimated spectral reflectance for the surface (as illustrated in the right panel of Figure 44).

A full-spectral rendering approach [Johnson & Fairchild 1998] would allow for the highest color accuracy, but would be too data and computationally intensive for the interactive color pipeline of the object display system. Abridged ( multispectral) rendering workflows [Peercy et al. 1993, Peercy et al. 1996, Drew & Finlayson 2003] provide a compromise between trichromatic and full-spectral color calculations. With these methods, it is possible to retain
enough spectral information to avoid major color errors for new lighting conditions, while only requiring a few additional color channels beyond a trichromatic workflow. The multispectral factor approach developed by Drew & Finlayson [2003] was used as the basis for color calculations in the object display. As shown in Figure 43 (right panel), this approach is similar to the workflow for trichromatic rendering, but increases the number of channels to maintain additional spectral information. To support the object display, a set of optimized multispectral channels were designed and methods were developed to incorporate both full spectral and multispectral input data into a multispectral factor-based rendering workflow.

![Figure 43](image)

**Figure 43.** Left, the typical RGB workflow used in interactive computer graphics simulations. Reflectance and illumination data are represented in three channels and are multiplied together to estimate the rendered color. The multispectral factor method uses a similar process, but increases the number of channels to maintain additional spectral information.
Figure 44. Left, a flowchart for the standard calculation of tristimulus values by multiplying the spectral reflectance factor $R_\lambda$ and the spectral power distribution of a source $S_\lambda$ on a per-wavelength basis. Right, in a typical multispectral imaging workflow, surface reflectance data captured in six channels is transformed to an estimate of full spectral reflectance, then calculated in a similar full-spectral manner using color matching functions.

6.3 Objectives

In developing the object display color framework, the principal objectives were to maintain a high level of colorimetric accuracy for a range of illumination types, while still performing lighting calculations in a small enough number of spectral channels to be suitable for real-time rendering. In particular, by limiting the number of rendering channels to six, the rendering workflow can closely follow a typical trichromatic RGB process and merely repeat the rendering calculations for a second set of RGB data. The input data, typically stored as RGB images, can still be represented in this form by maintaining just one additional image per reflectance map or illumination map.
An additional design objective was to base all the rendering calculations on a single set of rendering primaries. Certain methods for color-accurate rendering, such as point sampled wavelengths or basis vectors derived from characteristic vectors analysis, customize the rendering primaries to the set of lights in each virtual environment. While this can be advantageous for color accuracy, it requires that a full spectral representation of surface reflectance factor be maintained so that the abridged surface representation can be recalculated for each new lighting environment. With one fixed set of rendering primaries, once the data is converted to this representation, it can stay in this final form and does not require any additional data to be maintained or re-processing to be performed.

The final objective was to allow input data captured or specified with different methods, both full-spectral and multispectral, to be used interchangeably in the rendering system. Data on the reflectance of object surfaces and spectral distribution of lighting, which serve as the input for computer graphics rendering systems, may be measured with a range of different devices. For example, reflectance data measured with a spectrophotometer may be specified in 31 or more spectral bands. The reflectance of a complex, spatially-varying surface, such as a painting, however, will typically be captured with a multispectral-imaging system in a limited number of spectral bands. Illumination data may also be captured for a range of different devices and spectral channels. Regardless of how data is initially captured, the objective of the color framework is to allow that data to be incorporated into the rendering pipeline and used interchangeably with other lighting and reflectance data.
6.4 Multispectral Framework Design

The multispectral color processing pipeline, illustrated in Figure 45, has two main components. There is an offline pre-processing portion (marked by the dotted lines), where surface and lighting data are converted to the set of standard six-channel rendering primaries, and a real-time rendering portion where the surface shading calculations are performed by multiplying illumination and surface reflectance.

![Diagram of Multispectral Framework Design]

**Figure 45.** Calculation of tristimulus values using a six-channel rendering workflow. In a pre-processing stage (dashed boxes), full spectral or abridged multispectral data for the surface reflectance and illumination are converted to the six rendering primaries. Once all data are in this common form, lighting calculations are performed in the six channels and the results are converted to XYZ with a matrix transform.

By converting all the data to a common form, the pre-processing stage allows the input data captured with different measurement devices to be used interchangeably within the rendering system. The two principal input forms considered are full spectral data and abridged multispectral imaging data. To support full spectral data input, a spectral sensitivity curve was
created for each of the six primaries. Full spectral data are converted by summing the product of
the data and the derived curves:

\[
R_{i,OPT} = \sum_{\lambda=380}^{730 \text{ nm}} SS_{\lambda,i} R_{\lambda}, \text{ for } i = 1 \text{ to } 6 \tag{65}
\]

where \(SS_{\lambda,i}\) is the normalized spectral sensitivity of the \(i^{th}\) primary, \(R_{\lambda}\) is the spectral reflectance
factor, and \(R_{i,OPT}\) is the resulting scalar for the \(i^{th}\) primary. There is no explicit wavelength
interval term (\(\Delta \lambda\)) in the formula because each spectral sensitivity curve is normalized such that
it always sums to 1 over the specified set of \(N\) wavelengths:

\[
\sum_{j=1}^{N} SS_{\lambda(j)} = 1 \tag{66}
\]

where \(SS_{\lambda(j)}\) is the normalized spectral sensitivity at the \(j^{th}\) wavelength of the set of \(N\) wavelengths
(changing the wavelength interval requires the spectral sensitivity curves to be renormalized). With the spectral response curves normalized in this manner, the reflectance factor of a PRD \((R_{\lambda}
= 1 \text{ for all wavelengths})\) will produce a six-primary surface representation that is also equal to 1
for all channels \((R_{i,OPT} = 1, \text{ for } i = 1 \text{ to } 6)\).

A calculation similar to Eq. (65) is performed to convert light source spectra, \(S_{\lambda}\), to a
lighting representation specified in terms of the rendering primaries:

\[
S_{i,OPT} = \sum_{\lambda=380}^{730 \text{ nm}} SS_{\lambda,j} S_{\lambda}, \text{ for } i = 1 \text{ to } 6 \tag{67}
\]

where \(S_{i,OPT}\) is the scalar for the \(i^{th}\) primary. The absolute scaling of the six-channel illumination
representation depends on the manner in which the light source spectral data is specified, as well
as the final form that is required for the XYZ output. The scaling types are discussed in Section 6.5.4, where the six-channel to XYZ matrices are given and the impact of different illumination scalings on the final output is described.

The rendering pipeline is also intended for use with surface reflectance and lighting data captured with abridged multispectral imaging systems. The multispectral input example shown in Figure 45 is based on a six-channel multispectral imaging system [Berns et al. 2005] that combines an RGB camera with two additional filters, cyan and yellow, to provide a total of six color channels (identified with the subscript \{RGBcRGBy\}). The reflectance data, $R_{\{RGBcRGBy\}}$, is represented in a form similar to spectral reflectance factor and determined by dividing captured surface data by a similarly illuminated white diffusing standard. Captured illumination data, $S_{\{RGBcRGBy\}}$, is represented in an absolute form (proportional to luminance). Surface reflectance data and lighting data are converted to the rendering primaries using $[6 \times 6]$ matrix transforms that are optimized for the specific camera system. Separate matrices are optimized for reflectance data (matrix $M_R$) and for the lighting data (matrix $M_s$). Using the $M_s$ matrix, multispectral-captured illumination data is transformed to the six-channel primary representation by:

$$
egin{bmatrix}
S_{1,Dpt} \\
S_{2,Dpt} \\
S_{3,Dpt} \\
S_{4,Dpt} \\
S_{5,Dpt} \\
S_{6,Dpt}
\end{bmatrix}
= \begin{bmatrix}
m_{1,Re} & m_{1,Ge} & m_{1,Be} & m_{1,Re} & m_{1,Gy} & m_{1,By} \\
m_{2,Re} & m_{2,Ge} & m_{2,Be} & m_{2,Re} & m_{2,Gy} & m_{2,By} \\
m_{3,Re} & m_{3,Ge} & m_{3,Be} & m_{3,Re} & m_{3,Gy} & m_{3,By} \\
m_{4,Re} & m_{4,Ge} & m_{4,Be} & m_{4,Re} & m_{4,Gy} & m_{4,By} \\
m_{5,Re} & m_{5,Ge} & m_{5,Be} & m_{5,Re} & m_{5,Gy} & m_{5,By} \\
m_{6,Re} & m_{6,Ge} & m_{6,Be} & m_{6,Re} & m_{6,Gy} & m_{6,By}
\end{bmatrix}
\begin{bmatrix}
S_{Re} \\
S_{Ge} \\
S_{Be} \\
S_{Ry} \\
S_{Gy} \\
S_{By}
\end{bmatrix}.
$$

(68)
The \( M_R \) matrix is applied to captured reflectance data \( R_{i[RGB\cdotRGBy]} \) in the same manner to calculate the six channels for \( R_{i,OPT} \). Other spectral imaging systems, with different spectral sensitivities or a different number of channels (\( N \)) can be incorporated by providing the set of \([6 \times N] M_R \) and \( M_s \) matrices optimized for that particular system. These transform matrices assume the camera response is linear and there is no zero offset. To use these with a real-world imaging system, it may be necessary to first apply a gamma correction and subtract a constant offset from each channel before using the \([6 \times N] \) transform matrices.

Once all the data are represented in the set of six rendering primaries, surface lighting calculations are performed by multiplying surface reflectance and lighting on a per-channel basis:

\[
\Phi_{i,OPT} = (R_{i,OPT})(S_{i,OPT}) \text{, for } i = 1 \text{ to } 6 \tag{69}
\]

where \( \Phi_{i,OPT} \) represents the calculated light reflection in the \( i^{th} \) rendering channel. After summing over all the light sources, the final reflection, specified in the six rendering primaries, is multiplied by an optimized \([3 \times 6] \) matrix \( (M_{Ch6,XYZ}) \) to convert to CIE XYZ tristimulus values for display:

\[
\begin{bmatrix}
X \\
Y \\
Z
\end{bmatrix} =
\begin{bmatrix}
m_{x,1} & m_{x,2} & m_{x,3} & m_{x,4} & m_{x,5} & m_{x,6} \\
m_{y,1} & m_{y,2} & m_{y,3} & m_{y,4} & m_{y,5} & m_{y,6} \\
m_{z,1} & m_{z,2} & m_{z,3} & m_{z,4} & m_{z,5} & m_{z,6}
\end{bmatrix}
\begin{bmatrix}
\Phi_{1,OPT} \\
\Phi_{2,OPT} \\
\vdots \\
\Phi_{6,OPT}
\end{bmatrix} \tag{70}
\]
To maintain capability with trichromatic rendering pipelines, the six channel calculations can be performed for two sets of separate trichromatic data and the final XYZ results combined:

\[
\begin{bmatrix}
X \\
Y \\
Z
\end{bmatrix}
= 
\begin{bmatrix}
m_{X,1} & m_{X,2} & m_{X,3} \\
m_{Y,1} & m_{Y,2} & m_{Y,3} \\
m_{Z,1} & m_{Z,2} & m_{Z,3}
\end{bmatrix}
\begin{bmatrix}
\Phi_{1,OPT} \\
\Phi_{2,OPT} \\
\Phi_{3,OPT}
\end{bmatrix}
+ 
\begin{bmatrix}
m_{X,4} & m_{X,5} & m_{X,6} \\
m_{Y,4} & m_{Y,5} & m_{Y,6} \\
m_{Z,4} & m_{Z,5} & m_{Z,6}
\end{bmatrix}
\begin{bmatrix}
\Phi_{4,OPT} \\
\Phi_{5,OPT} \\
\Phi_{6,OPT}
\end{bmatrix}
\] (71)

6.5 Selection of Rendering Primaries

A key requirement for the rendering pipeline was that it be based on a fixed set of six rendering primaries and not ones customized for each particular lighting environment. A non-linear optimization was performed to select a general set of rendering primaries that could provide an acceptable level of accuracy for a range of different illuminants.

6.5.1 Training and Validation Data

The optimization was performed using 100 surface spectral reflectance factor curves from a database of artist materials [Okumura 2005] and a range of spectral power distributions for lighting: illuminant A; daylight illuminants: D40, D50, D65, D75, D90; fluorescent illuminants: F1 through F12; and the measured SPD of a D50 light booth. All full-spectral data were specified in 36 (10 nm) bands from 380 to 730 nm.

For validation of the workflow, surface reflectance and illumination data not used in the optimization were also tested. The spectral reflectance factor curves from two targets, the 24-patch X-Rite Classic ColorChecker and the 240-patch ColorCheckerDC, were used for the
surface data. Spectral illumination data captured from five real-world environments were used as the test lighting data. This illumination included a cloudy outdoor environment (x=0.317, y=0.329), an incandescent lamp on a desk (x=0.456, y=0.404), a light booth simulating horizon light (x=0.500, y=0.417), an office environment (x=0.405, y=0.404), and a classroom with fluorescent lighting (x=0.393, y=0.391).

The XYZ tristimulus values for each surface-light pair were calculated spectrally using the color matching functions for the CIE 1931 standard observer, to serve as the baseline for comparison to the abridged multispectral rendering workflow. These baseline XYZ values were calculated with the scaling factor for surface objects, such that a PRD has $Y = 100$. To provide a consistent scaling in the multispectral workflow, all the light sources that were entered as input had their spectral power distributions scaled to $Y = 100$ with a factor $K$:

$$S_{\lambda (\text{input})} = K \cdot S_{\lambda}$$

(72)

where $K$ is calculated with the formula [Berns 2000]:

$$K = \frac{100}{\sum_{\lambda = 380}^{730} S_{\lambda} y_{\lambda} \Delta \lambda},$$

$\Delta \lambda$ is the wavelength interval (10 nm), and $y_{\lambda}$ is the Y color matching function.

### 6.5.2 Optimization Method

The optimization was performed with the full-spectral data input path, as depicted on the left side of Figure 45. The spectral sensitivity curves for the six rendering primaries and the $[3 \times 6]$ matrix transform from the rendering primaries to XYZ were simultaneously optimized with the constrained non-linear optimization routine in MATLAB (fmincon). To limit the number of parameters involved in optimizing the spectral curves for the primaries, each was represented as
a Gaussian function with two parameters: peak wavelength and width (σ). Each Gaussian curve was normalized so that its sum over all (N=36) wavelengths was equal to 1.

As the Gaussian parameters were updated, the $R_{\text{ch},\text{OPT}}$ values for the 100 surface reflectance curves and $S_{\text{ch},\text{OPT}}$ values for the 19 light sources were recalculated and the results multiplied to determine $\Phi_{\text{ch},\text{OPT}}$ for each light-surface pair. These $\Phi_{\text{ch},\text{OPT}}$ values were used to estimate XYZ with the current version of the [3 x 6] matrix. These estimated XYZ values were transformed to D65 with the chromatic adaptation transform from CIECAM02 (MCAT02) before calculating CIELAB values (all the baseline data were similarly transformed to D65). The sum squared error of the set of CIEDE2000 color differences, calculated between the six-channel results and the baseline full-spectral results was minimized in the optimization.

### 6.5.3 Optimized Spectral Response Curves

The optimization resulted in the selection of six rendering primaries corresponding to Gaussians functions with peak wavelengths of (446.98, 481.53, 519.59, 543.11, 572.91, 622.38 nm) and width parameters of (16.92, 4.25, 7.78, 9.35, 15.51, 17.97 nm). The resulting set of spectral curves is shown in Figure 46 for both 1 nm intervals and 10 nm intervals. The 10 nm spectral curves were the ones actually generated in the optimization and used in data analysis. A table of the numerical values for 10 nm intervals is provided in Appendix E. At 10 nm intervals, the six channels have peaks at 450 nm (channel 1), 480 nm (channel 2), 520 nm (channel 3), 540 nm (channel 4), 570 nm (channel 5), and 620 nm (channel 6).
Figure 46. Spectral curves for the six Gaussian rendering primaries determined in the optimization. In the left panel, the primaries are shown at 1 nm intervals. In the right panel, the primaries are shown at 10 nm intervals. The change in the y-axis scale is due to the renormalization for different wavelength intervals.

In Figure 47, the set of six spectral curves (at 10 nm) are used to reconstruct the color matching functions for the 1931 CIE standard observer to compare the spectral similarities between the two sets of curves. The linear combinations of the six channel primaries that provide the best least squares spectral fit to the color matching functions are plotted in the figure.

The $\bar{z}$ color matching can be partially reconstructed with the first six-channel primary (peak near 450 nm), which has a similar peak wavelength and shape for the shorter-wavelength edge, along with the second primary (peak near 480 nm), which can be used to reconstruct a portion of $\bar{z}$ at longer wavelengths. With the narrow width of the second primary, the reconstruction decreases more sharply along the longer-wavelength edge than the actual $\bar{z}$ color matching function. The third primary (peak near 520 nm) with a small weighting can be used to reconstruct the end portion of the $\bar{z}$ tail.
The center portion of the $\bar{y}$ color matching function can be partially reconstructed with a combination of the third (peak near 520 nm), fourth (peak near 540 nm), and fifth (peak near 570 nm) primaries. The left (shorter-wavelength) portion of the $\bar{y}$ tail can be approximated with small weightings on the first and second primaries. The end portion of the longer-wavelength tail can be partially reconstructed with the sixth primary (peak near 620 nm).

In the $\bar{x}$ color matching function, there are two lobes to reconstruct, a smaller lobe in the shorter-wavelength band and a larger lobe in the longer-wavelength band. The first primary can be used to reconstruct the general shape of the shorter-wavelength lobe. The longer wavelength lobe can be partially reconstructed with a combination of the fourth, fifth and sixth primaries, though the reconstruction does not match the central peak region of this lobe (near 600 nm), which falls between the peaks of the fifth (570 nm) and sixth (620 nm) primaries.

![Figure 47](image)

**Figure 47.** The least squares fit of the six channel curves to the CIE 1931 color matching functions. The actual color matching functions (at 10 nm intervals) are shown as the dotted lines and the least squares fit of the six channel primaries are shown as the solid lines.
6.5.4 Optimized Matrix from the Six-Channel Representation to CIE XYZ

In addition to the spectral curves for the primaries, the [3 x 6] matrix for converting from the six channel representation to CIE XYZ was also determined in the optimization to minimize the colorimetric error in the overall workflow. The matrix calculated directly in the optimization was in a relative form (such that Y = 100 for a PRD) and was later rescaled to the final luminance-based matrix for the object display. The initial matrix resulting from the optimization workflow was:

\[
M_{Ch6,XYZ(\text{relative})} = \begin{bmatrix}
0.1571 & 0.0279 & -0.0139 & 0.0534 & 0.3855 & 0.4522 \\
-0.0014 & 0.0800 & 0.1705 & 0.2134 & 0.4080 & 0.1931 \\
0.8607 & 0.1507 & 0.0474 & 0.0028 & -0.0034 & -0.0001
\end{bmatrix}.
\]  \( (73) \)

6.5.4.1 Scaling for the Object Display

The object display has a different scaling mechanism (to produce absolute luminance) than was used for the optimization simulation, so it was necessary to rescale the initial matrix from the optimization. The absolute scaling of the object display matrix is defined such that the six-channel output vector \( \Phi_{6ch,OPT} \) with values \([1 \ 1 \ 1 \ 1 \ 1 \ 1] \) will produce an XYZ with the luminance \( Y = 1 \ \text{cd/m}^2 \). The initial matrix was rescaled by taking the sum of the six elements in the second row and then dividing every element in the matrix by this value. The final matrix used in the object display system, scaled for output in absolute luminance is:
\[
\mathbf{M}_{Ch6,XYZ(\text{object display})} = \begin{bmatrix}
0.1477 & 0.0262 & -0.0131 & 0.0502 & 0.3624 & 0.4251 \\
-0.0013 & 0.0752 & 0.1603 & 0.2006 & 0.3836 & 0.1816 \\
0.8092 & 0.1417 & 0.0446 & 0.0026 & -0.0032 & -0.0001
\end{bmatrix} \text{[cd/m}^2\text{]. (74)}
\]

For the final object display matrix in Eq. (74) to produce correct luminance output values, the captured illumination must be scaled properly with respect to luminance when entered into the rendering framework. The typical case in the object display framework is to have a relative spectral power distribution of the illumination (arbitrarily scaled) along with a separate set of luminance estimates that are used to specify the physical scale. To set the proper scaling in this case, the unscaled spectral power distribution \( S_i \) is converted directly to an unscaled six channel representation \( S_{6ch,\text{opt}(\text{unscaled})} \) with Eq. (67). The unscaled six channel representation is normalized to 1 cd/m\(^2\) by dividing by its (arbitrary) luminance value, as calculated with the second row of the final \( \mathbf{M}_{Ch6,XYZ(\text{object display})} \) matrix:

\[
S_{i,\text{Opt}(1 \text{ cd/m}^2)} = \frac{S_{i,\text{Opt}(\text{unscaled})}}{\sum_{j=1}^{6} m_{Y,j} S_{j,\text{Opt}(\text{unscaled})}} \quad \text{(75)}
\]

where \( m_{Y,j} \) is the value in the \( j^{th} \) column of the second (Y) row of \( \mathbf{M}_{Ch6,XYZ} \) in Eq. (74). The result is stored in this intermediate form for later use. The absolute light source representation \( S_{6ch,\text{opt}(\text{absolute})} \) is calculated from the intermediate 1 cd/m\(^2\) form by scaling the intermediate result with the physical luminance value \( Y_{[\text{cd/m}^2]} \) from a lookup into a spatial luminance map or the luminance value of a concentrated median cut point light:

\[
S_{i,\text{Opt}(\text{absolute})} = Y_{[\text{cd/m}^2]} \cdot S_{i,\text{Opt}(1 \text{ cd/m}^2)} \quad \text{for} \ i = 1 \text{ to } 6 \ . \quad \text{(76)}
\]
6.5.5 Model Results

The colorimetric accuracy when performing lighting calculations with the optimized rendering primaries and the optimized XYZ matrix (from Eq. (73)) was evaluated for the 100 patch target used in the optimization (S100) and two additional targets, the Classic ColorChecker and the ColorCheckerDC. The three targets were evaluated for the 19 illuminants used during the optimization and also for the five measured real-world lighting environments.

For comparison to the six-channel rendering method, results were also calculated for more typical three-channel rendering methods, either directly in XYZ [Borges 1991] or with the sharpened RGB primaries developed by Süsstrunk et al. [2001] and used for rendering by Ward & Eydelberg-Vileshin [2002]. In XYZ relighting, the three-channel representation of surface reflectance was calculated by assuming a taking illuminant of D65, calculating XYZ values under D65 for each of the surface reflectance factor curves, and then dividing by the XYZ values of the D65 lighting. Surface re-lighting was simulated by multiplying the surface representation by the XYZ values of the light source. A similar calculation method was used for the sharpened RGB space. For all methods, the final calculated XYZ values were transformed to corresponding colors under D65 and a CIEDE2000 color difference was calculated between the abridged rendering result and a full spectral calculation. Summary statistics (mean, standard deviation, maximum error, and 90th percentile) for the three different methods are shown in Table 1.
Table 1. CIEDE2000 Color Error for Six Channel, Sharpened RGB, and XYZ Color Calculations

<table>
<thead>
<tr>
<th>Surface Data</th>
<th>Method</th>
<th>Optimization Lighting CIEDE2000</th>
<th>Measured Test Lighting CIEDE2000</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Mean</td>
<td>SD</td>
</tr>
<tr>
<td>S100 (optimization)</td>
<td>Six Channel</td>
<td>0.5</td>
<td>0.5</td>
</tr>
<tr>
<td></td>
<td>Sharp RGB</td>
<td>1.9</td>
<td>2.1</td>
</tr>
<tr>
<td></td>
<td>XYZ</td>
<td>2.4</td>
<td>2.1</td>
</tr>
<tr>
<td>Classic ColorChecker (testing)</td>
<td>Six Channel</td>
<td>0.6</td>
<td>0.6</td>
</tr>
<tr>
<td></td>
<td>Sharp RGB</td>
<td>1.7</td>
<td>2.0</td>
</tr>
<tr>
<td></td>
<td>XYZ</td>
<td>2.3</td>
<td>2.2</td>
</tr>
<tr>
<td>ColorCheckerDC (testing)</td>
<td>Six Channel</td>
<td>0.5</td>
<td>0.5</td>
</tr>
<tr>
<td></td>
<td>Sharp RGB</td>
<td>1.6</td>
<td>2.1</td>
</tr>
<tr>
<td></td>
<td>XYZ</td>
<td>1.9</td>
<td>2.1</td>
</tr>
</tbody>
</table>

Under the 19-light optimization training set, the six channel workflow was found to provide greater colorimetric accuracy than either of the three channel methods for both the S100 data and for the two test set targets. With this lighting data, the mean CIEDE2000 color difference for the six channel workflow was in the 0.5 to 0.6 range. On the measured test illumination data, there was a moderate increase in the mean target error for the six channel workflow to a range of 0.8 to 0.9. With the measured testing illumination, the advantage relative to the sharpened RGB rendering was not as large as in the training light case, though it did still produce mean color differences that were generally only half as large as those produced by the sharpened RGB method.

### 6.6 Multispectral Input Path

The six channel rendering workflow is intended for use with data from abridged multispectral capture systems, in addition to full spectral input data. To evaluate the rendering workflow when starting from this type of input (as depicted on the right side of Figure 45), a
simulation was performed based on the set of multispectral channels of a real-world imaging system. The system, developed for use in artwork conservation, is a combination of a Sinarback 54H color-filter-array camera and additional cyan and yellow filters that are applied sequentially to provide multispectral capture in six channels [Berns et al. 2005].

In the simulation, the multispectral input data were generated using the spectral response curves for the six channels of the imaging system (Rc, Gc, Bc, Ry, Gy, By) on the full spectral data described in Section 6.5.1. Simulated surface reflectance signals were calculated assuming a D65 taking illuminant, calculating the six channel response to the product of the spectral reflectance curve and D65 illumination, and then dividing out the per-channel response for a similarly illuminated PRD. The result was the multispectral surface input, \( R_{[Rc,Gc,Bc,Ry,Gy,By]} \), in a form similar to reflectance factor. The idealized illumination input, \( S_{[Rc,Gc,Bc,Ry,Gy,By]} \), was calculated directly from the full spectral illumination data and the multispectral response curves.

These sets of simulated multispectral signals were used to evaluate the color rendering workflow for three input combinations:

1. Multispectral input data for reflectance only (with lighting data from the Gaussian full spectral input workflow)
2. Multispectral input data for lighting only (with reflectance data from the Gaussian full spectral input workflow)
3. Multispectral input for both the reflectance and lighting data.

Case 1 corresponds to the typical use of a multispectral-imaging system, to capture spatially-varying information over a surface (such as a painting), which is then re-lit using the full spectral power distribution of a standard illuminant (or, in the case of the object display, the sensed
ambient light SPD in a viewing booth). Case 2 corresponds to capturing a spatially-varying illumination map of an environment (for example, by multispectral imaging of a mirrored sphere) and using it to re-light an object of known full spectral reflectance (such as a ColorChecker patch measured with a spectrophotometer). Case 3 represents a combination of the two types of multispectral capture data.

For case 1, the \([6 \times 6]\) matrix transform \((M_R)\) required to convert from multispectral reflectance input data \((R_{[RGBcRGBy]})\) to the six rendering primaries \((R_{6ch,OPT})\) was estimated with a non-linear optimization. The objective function was to minimize the CIEDE2000 color difference between results calculated by following the entire rendering workflow and results calculated with a standard colorimetric workflow for spectral data. The optimization was performed for surface data from the S100 target (converted to the \(R_{[RGBcRGBy]}\) form) and illumination data for the set of 19 training illuminants (entered through the Gaussian full spectral input method). A similar type of optimization was performed to calculate the \(M_S\) matrix transform used to convert from multispectral lighting data, \(S_{[RGBcRGBy]}\), to lighting represented in the six rendering primaries \(S_{6ch,OPT}\). This optimization was performed using S100 target surface data entered through the Gaussian spectral input method and simulated multispectral lighting data \(S_{[RGBcRGBy]}\) generated for the set of training lights.

The color accuracy results for the three multispectral input cases are shown in Table 2 and Table 3. For comparison, baseline results are provided for the case where full-spectral input data are processed through the optimized Gaussian curves for both illumination and reflectance (Full Spectral-Both). CIEDE2000 mean, standard deviation, maximum, and 90th percentile error statistics on the S100 training data, the ColorChecker, and ColorCheckerDC with the initial training illumination are shown in Table 2 and with the measured test illumination are shown in
Table 3. In the data tables, the results for rendering in the six optimized channels are compared to a trichromatic rendering workflow, where multispectral input data were converted to sharpened RGB primaries [Ward & Eydelberg-Vileshin 2002] with an optimized [3 x 6] matrix before lighting calculations were performed. The results for the six channel workflow are plotted in Figure 48.

Table 2. Comparison of Six-Channel and Sharpened RGB Rendering for Multispectral Input (Optimization Lighting)

<table>
<thead>
<tr>
<th>Surface Data</th>
<th>Input Type</th>
<th>Optimization Lighting</th>
<th>CIEDE2000</th>
<th>Sharpened RGB</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Optimized Six Channels</td>
<td>Mean  SD  Max 90th</td>
<td>Mean  SD  Max 90th</td>
</tr>
<tr>
<td>$S100$ (optimization)</td>
<td>Full Spectral-Both</td>
<td>0.5 0.5 3.6 0.9</td>
<td>1.9 2.1 14.9 4.8</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(1) Multispectral-Reflectance</td>
<td>0.7 0.4 3.4 1.2</td>
<td>1.7 1.4 9.9 3.7</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(2) Multispectral-Lighting</td>
<td>0.8 0.6 3.4 1.6</td>
<td>2.0 1.9 13.4 4.8</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(3) Multispectral-Both</td>
<td>0.9 0.6 3.7 1.7</td>
<td>2.6 1.2 10.3 4.2</td>
<td></td>
</tr>
<tr>
<td>Classic ColorChecker (testing)</td>
<td>Full Spectral-Both</td>
<td>0.6 0.6 3.1 1.2</td>
<td>1.7 2.0 13.3 4.4</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(1) Multispectral-Reflectance</td>
<td>0.6 0.4 2.3 1.1</td>
<td>1.6 1.3 9.1 3.6</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(2) Multispectral-Lighting</td>
<td>0.9 0.7 3.4 1.9</td>
<td>2.0 1.8 12.8 4.3</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(3) Multispectral-Both</td>
<td>1.0 0.6 3.5 1.7</td>
<td>2.6 1.2 8.6 3.9</td>
<td></td>
</tr>
<tr>
<td>ColorCheckerDC (testing)</td>
<td>Full Spectral-Both</td>
<td>0.5 0.5 4.9 1.0</td>
<td>1.6 2.1 15.9 4.5</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(1) Multispectral-Reflectance</td>
<td>0.7 0.5 6.0 1.2</td>
<td>1.6 1.4 10.6 3.4</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(2) Multispectral-Lighting</td>
<td>0.9 0.7 4.4 1.9</td>
<td>2.0 1.9 15.4 4.5</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(3) Multispectral-Both</td>
<td>1.0 0.7 5.3 2.0</td>
<td>2.6 1.2 9.9 4.0</td>
<td></td>
</tr>
</tbody>
</table>

Table 3. Comparison of Six-Channel and Sharpened RGB Rendering for Multispectral Input (Measured Test Lighting)

<table>
<thead>
<tr>
<th>Surface Data</th>
<th>Input Type</th>
<th>Measured Test Lighting</th>
<th>CIEDE2000</th>
<th>Sharpened RGB</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>Optimized Six Channels</td>
<td>Mean  SD  Max 90th</td>
<td>Mean  SD  Max 90th</td>
</tr>
<tr>
<td>$S100$ (optimization)</td>
<td>Full Spectral-Both</td>
<td>0.9 0.7 5.0 1.7</td>
<td>2.0 1.9 9.8 4.7</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(1) Multispectral-Reflectance</td>
<td>0.9 0.5 2.9 1.7</td>
<td>1.9 1.5 9.2 4.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(2) Multispectral-Lighting</td>
<td>1.4 0.9 5.2 2.7</td>
<td>2.1 1.7 8.7 4.6</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(3) Multispectral-Both</td>
<td>1.4 0.9 5.7 2.7</td>
<td>2.6 1.2 8.9 4.1</td>
<td></td>
</tr>
<tr>
<td>Classic ColorChecker (testing)</td>
<td>Full Spectral-Both</td>
<td>0.8 0.6 2.8 1.5</td>
<td>1.7 1.6 7.5 3.9</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(1) Multispectral-Reflectance</td>
<td>0.9 0.5 2.7 1.5</td>
<td>1.8 1.4 6.8 4.0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(2) Multispectral-Lighting</td>
<td>1.5 1.0 5.1 2.7</td>
<td>2.1 1.4 7.5 3.6</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(3) Multispectral-Both</td>
<td>1.5 1.1 5.5 3.0</td>
<td>2.8 1.0 6.6 4.2</td>
<td></td>
</tr>
<tr>
<td>ColorCheckerDC (testing)</td>
<td>Full Spectral-Both</td>
<td>0.8 0.5 4.4 1.4</td>
<td>1.6 1.8 12.5 4.1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(1) Multispectral-Reflectance</td>
<td>0.8 0.5 4.3 1.5</td>
<td>1.6 1.3 11.9 3.4</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(2) Multispectral-Lighting</td>
<td>1.7 1.1 5.2 3.2</td>
<td>2.1 1.6 12.5 4.1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>(3) Multispectral-Both</td>
<td>1.7 1.2 5.7 3.5</td>
<td>2.6 1.2 12.0 4.1</td>
<td></td>
</tr>
</tbody>
</table>
Figure 48. Comparison of the six-channel rendering workflow accuracy for the four input method combinations. The solid bars represent the mean CIEDE2000 color error. The tops of the lines represent the 90th percentile of the CIEDE2000 color error for the target.

In the simulations with multispectral inputs, for the optimization lighting, the six-channel rendering workflow produced mean, max, and 90th percentile color errors that were less than 50% of the CIEDE2000 values found for the RGB workflow. Under the measured test lighting, the six-channel workflow still produced higher accuracy than the three channel RGB calculations, though its benefit was not as large, in particular for the multispectral lighting input case.

When comparing the accuracy of the different input types for the six-channel workflow, case 1 (only the reflectance data is entered through the multispectral path) produced the highest overall accuracy of the three multispectral cases. The accuracy results for case 1 were similar to
the baseline case of full-spectral input to the six-channel pipeline (in particular, on the verification illumination). In practice however, with real captured data from a multispectral camera system, noise and quantization error may reduce the performance to a greater degree relative to the full-spectral input method. Multispectral input of the light sources (case 2) introduced a greater degree of color error into the rendering pipeline, and multispectral input of both gave the lowest accuracy. The mean color difference results followed this pattern for nearly all the conditions. This was also generally the case for 90th percentile, though there were some exceptions (e.g. the ColorChecker for optimization illumination). There was not a consistent pattern for the maximum error results.

6.7 Six-Channel Color in the Object Display Framework

In the previous sections, the six-channel workflow is typically described in colorimetric terms, with surfaces initially specified in terms of spectral reflectance factor \( R_\lambda \) and illumination specified in terms of the spectral power distribution \( S_\lambda \). This was an efficient representation to use when performing the optimizations, as it allowed the six-channel workflow output to be directly compared to spectral color calculations using the CIE 1931 standard observer. In the object display framework, surface representations are specified in terms of BRDF model parameters. Because the Ward-Dür BRDF model has a Lambertian diffuse lobe, the six-channel reflectance factor \( R_{6ch,OPT} \) can be directly represented with the Ward-Dür model as a set of six-channel \( \rho_d \) parameters. In the case of spectrally-selective front surface reflection (e.g. for metals), the color properties of the specular reflection can also be specified in the BRDF model using a set of six-channel \( \rho_s \) parameters.
7 GLOSS REPRODUCTION FRAMEWORK

7.1 Overview

The object display framework was developed to reproduce the gloss properties of object surfaces in addition to their diffuse color properties. Working in conjunction with the color framework, the gloss reproduction framework provides support for rendering glossy surface reflections that are based on the real physical lights in the environment of the display. The gloss framework encompasses methods for capturing and representing real world light sources, methods for representing the gloss properties of physical surfaces, and methods for interactively rendering specular reflections based on the geometrical relationship between the observer, the virtual surface onscreen, and the illumination. The ultimate objective of the gloss framework is to not only produce renderings that look realistic, but also to produce quantitatively-accurate results that could be verified by comparing spectroradiometer measurements of the object display screen to measurements of real objects in the same lighting configuration.

One of the principal challenges in the gloss reproduction framework was the development of a rendering workflow that could produce a high level of radiometric accuracy within the constraints of real-time rendering engine. For offline rendering, where several minutes or hours may be spent on a single frame, physically-based ray-tracing methods can be used to achieve a high level of radiometric accuracy, and these methods are readily available in software packages like Radiance [Ward Larson & Shakespeare 1998] and PBRT [Pharr & Humphreys 2004]. However, it is necessary to use real-time rendering methods for the object display framework to allow for interactive updating with changes in the screen orientation or observer position.
The gloss framework for the object display system is designed to provide an end-to-end solution for obtaining an acceptable level of radiometric accuracy within a real-time rendering workflow. In this chapter, a specific implementation of the gloss framework is described to provide an example of how this can be achieved in practice and of all the necessary steps and components in the process. In the current implementation, surface gloss is based on the Ward-Dür BRDF model and captured illumination is represented with spatially-varying diffuse area light sources. Though these methods have properties that make them more practical computationally on current video processing hardware, there are no fundamental reasons why additional BRDF or lighting models could not be incorporated for an object display system in the future.

### 7.2 Representing Light Sources

There are a variety of methods used to capture and represent real-world environmental lighting for use in rendering. One of the most common is to use image-based lighting probes captured by photographing a reflective sphere in a high-dynamic-range series of images [Debevec & Malik 1997, Debevec 1998]. These captured images are geometrically mapped to the inside of a cube or sphere to create a virtual lighting environment. In tangible displays systems like the tangiBook and some early prototypes of the object display system, these types of cubic environmental illumination maps were used to provide realistic reflections for the virtual surfaces. However, this technique is based on an assumption that the light sources are relatively far from the sample and does not provide an accurate model of near field sources. In the current object display system, the rendering illumination is represented by a set of planar luminance
maps that are mapped to their real 3D physical location in the environment around the display screen.

7.3 Surface Model

In the implementation of the object display framework, the gloss properties of surfaces are represented using the specular reflectance parameters of the Ward-Dür BRDF model (the Ward [1992] model with a modification made by Dür [2006]). The Ward-Dür model $\alpha$ parameter (specular roughness) is used to describe the width of the specular lobe. The parameter describing the magnitude of the specular reflectance peak, $\rho_s$, may be specified for each of the six multispectral channels to allow for spectrally-selective front surface reflection.

7.4 Rendering Methods for Specular Reflections

When rendering specular reflections for complex lighting, such as extended spatially-varying sources, it is not feasible to evaluate the surface BRDF directly for every possible incident lighting direction within the illumination area. Instead, it is necessary to estimate the effect of light sources based on a limited sampling of either the surface BRDF directions or the light source positions. For offline rendering, the number of samples can be increased until noise levels are sufficiently low, at the expense of increased rendering time. In the case of real-time simulations, with the need to maintain interactive frame rates, it is necessary to use approximate techniques that can produce reasonable accuracy with a small, fixed number of samples.

In the objective display framework, specular reflections are evaluated using a hybrid approach that interactively selects between two approximate lighting techniques: median cut-based point lighting [Viriyothai & Debevec 2009] and filtered importance sampling of image-
based lighting sources [Colbert et al. 2006, Colbert & Křivánek 2007, Křivánek & Colbert 2008, Colbert et al. 2010]. In the importance sampling technique, the magnitude of the reflection is evaluated stochastically by sending out a set of random samples toward illumination sources with a directional distribution based on the BRDF. With the median-cut method, the extended light source is divided into regions and each region is represented by a point light source, which is evaluated directly. The median-cut point lighting technique is better suited to relatively low gloss surfaces and small light sources, while the filtered importance sampling technique is appropriate for high gloss surfaces and larger light sources. With only a single technique, limitations in each method can result in significant error under certain lighting conditions and gloss levels. In developing the gloss reproduction pipeline, a new interactive method for switching between these techniques during the rendering process, on a per-pixel basis, was developed to obtain an acceptable level of accuracy over a range of surface gloss levels and area lighting geometries.

7.4.1 Filtered Importance Sampling in the Object Display System

Colbert and Křivánek [2007] developed the filtered importance sampling rendering method to allow for the use of complex, realistic environmental illumination maps and BRDF models within a real-time rendering workflow. Their original method [Colbert et al. 2006, Colbert & Křivánek 2007, Křivánek & Colbert 2008] was based on hemispheric lighting maps that globally represent all directions of the incident light in a scene. For the object display framework, a local version of the technique based on spatially-varying area light sources was developed. The method used in the object display was derived from the original Colbert & Křivánek technique [2007], but was found to be similar to a modification that Colbert described
in the SIGGRAPH 2010 course notes on accelerated area-lighting for movie production pipelines [Colbert et al. 2010].

7.4.1.1 Sampling Equations for the Ward-Dür Model

In BRDF-based importance sampling, the magnitude of the reflection is evaluated stochastically by sending out a set of random samples toward illumination sources with a directional distribution based on the BRDF. For the Ward [1992] isotropic BRDF model, this distribution is based on the width of the Gaussian lobe specified by the $\alpha$ parameter. The sampling equations for the distribution, as described by Walter [2005], are:

$$\theta_h = \arctan \left( \alpha \sqrt{-\log u} \right) \quad (77)$$

$$\phi_h = 2\pi v \quad (78)$$

where $u$ and $v$ are uniformly distributed random variables between 0 and 1, and the resulting angles are the zenith angle, $\theta_h$, and the azimuth angle, $\phi_h$, of the half-vector of the Ward model.

The unit half-vector, $h$, is calculated from these angles by:

$$h = (h_x, h_y, h_z) \quad \text{where}$$

$$h_x = \cos(\phi_h) \sin(\theta_h)$$

$$h_y = \sin(\phi_h) \sin(\theta_h) \quad (79)$$

$$h_z = \cos(\theta_h)$$

A weighting function is used to correct for any differences between the function implied by the sampling distribution and the explicit BRDF. Walter [2005] described the method for calculating a weighting function of the general form:

$$w = \frac{f_r \cos(\theta_i)}{P_{\text{sampling}}} \quad (80)$$
(where \( f_r \) represents the BRDF and \( p_{\text{sampling}} \) the sampling probability) and derived the weighting formula for the original Ward [1992] BRDF model. In the object display system, the Dür-modified version of the Ward BRDF model is used instead, so the appropriate weighting function was derived using Walter’s method on the Ward-Dür BRDF formula. The ratio of the isotropic Ward-Dür model to the sampling probability is:

\[
  w = \frac{\rho_s}{4\pi \alpha^2 (\hat{n} \cdot \hat{i})(\hat{n} \cdot \hat{r})} e^{-\tan^2(\theta_s) \frac{(\hat{n} \cdot \hat{i})}{\alpha^2}}
\]

and after cancelling terms the resulting weighting function is:

\[
  w = \rho_s \frac{(\hat{h} \cdot \hat{r})(\hat{h} \cdot \hat{n})^3}{(\hat{n} \cdot \hat{r})^3}
\]

where \( \rho_s \) is the specular peak parameter, \( \hat{h} \) is the half-vector, \( \hat{r} \) is the direction to the viewing position and \( \hat{n} \) is the surface normal.

### 7.4.1.2 Quasirandom Sampling and Image Filtering

The object display system uses the quasirandom sampling and image filtering method from Colbert and Křivánek [2007] to overcome the limitations on stochastic sampling of image-based illumination in a real-time rendering engine. In interactive rendering systems, it is not possible to randomly sample a large number of incident BRDF directions while still maintaining interactive frame rates. This results in image noise, because the small random sets may not be well distributed. With Colbert and Křivánek’s sampling method, instead of randomly generating
sampling directions, the \((u, v)\) uniform random variables in Eqs. (77, 78) are generated using a fixed quasirandom Hammersley sequence (see Eq. (28) in the literature review). The use of a quasirandom process reduces noise, but can produce aliasing that results in discontinuous specular highlights. To reduce this effect, the illumination map is filtered based on the BRDF sampling probability density function, using the formula from Colbert and Krivánek [2007]:

\[
N_{\text{ave}} = \frac{\Omega_s}{\Omega_p}
\]

where \(N_{\text{ave}}\) is the number of pixels to average, \(\Omega_p\) is the solid angle subtended by a pixel, and \(\Omega_s\) is the solid angle of a sample implied by its probability density function. For computational efficiency, the averaging is implemented as a box filter using the mipmapping level-of-detail capabilities in OpenGL. The mipmap pyramid level (LOD) that corresponds to averaging the \(N_{\text{ave}}\) pixels from Eq. (83), as given by Colbert & Krivánek [2007] is:

\[
LOD = \max \left( \frac{1}{2} \log_2 \left( \frac{\Omega_s}{\Omega_p} \right), 0 \right)
\]

7.4.1.3 Approximation for Area Sources

In the object display framework, the lighting consists of near-field bounded area sources instead of distant environment maps. A modified form of the filtered importance sampling method used with hemispherical environment maps was implemented for use with area sources (a similar modification for area sources was also implemented by Colbert, as reported in the SIGGRAPH 2010 Course Notes [Colbert et al. 2010]). The form of the area source approximation used in the object display system is described in this section.
For the hemispherical environment maps in the original filtered importance sampling method, the solid angle subtended by a pixel of the light source ($\Omega_p$) is calculated based on a hemispherical distortion factor and can be determined from the illumination direction alone. In the case of planar maps located in geometric space, more information is necessary to determine $\Omega_p$. In the local case, $\Omega_p$ is dependent on the physical distance to the point on the light source plane, the orientation of the plane, and the illumination direction. Treating each map pixel as a small area element, this solid angle can be approximated as:

$$\Omega_p = \frac{A}{w_{pix} \cdot h_{pix} \cdot r_{(a,b)}^2} \left( n_{light} \cdot -i_{(a,b)} \right)$$  \hspace{1cm} (85)

where $A$ is the area of the light source in square meters, $w_{pix} \cdot h_{pix}$ is the total number of pixels in the light source image, $r_{(a,b)}$ is the distance from the surface to the $(a,b)$ pixel of the light source, $n_{light}$ is the surface normal of the light source, and $i_{(a,b)}$ is the unit vector from the surface position to the $(a,b)$ light source position. As in the hemispherical Colbert and Křivánek [2007] method, the solid angle for each sample is still estimated by:

$$\Omega_s = \frac{1}{N} \frac{1}{pdf(u,v)}$$  \hspace{1cm} (86)

where $N$ is the total number of samples and $pdf$ is the sampling probability density function used to approximate the BRDF. Note that by taking the ratio of these two solid angles, there is an assumption that on average, the pixels within the sampling area have approximately the same solid angle as the central pixel at the intersection point $(a, b)$. The accuracy of this approximation decreases when the solid angles of the samples are larger and include pixels that are distant from the intersection point.
### 7.4.1.4 Calculating an Area Source Reflection with the Ward-Dür BRDF Model

The reflected luminance for an illuminated surface is estimated from the intersection of the $N$ directional samples with the area light source. For each sample, the intersection point with the area light is calculated and the corresponding pixel of spatial luminance image is read after filtering to the appropriate level-of-detail. The estimated luminance of the reflection is the mean of the weighted luminance image samples:

$$L_r = \frac{1}{N} \sum_{j=1}^{N} w_j \cdot L_{LODj}(i_j). \quad (87)$$

Substituting the formula for the derived Ward-Dür model weighting factor (Eq. 82), the complete formula for calculating the reflected luminance is:

$$L_r = \rho_s \frac{1}{N} \sum_{j=1}^{N} (h_j \cdot r)(h_j \cdot n)^3 \cdot \frac{\left( n \cdot r \right)}{(n \cdot r)} \cdot L_{LODj}(i_j) \quad (88)$$

where $L_{LODj}$ is the spatial image filtered to the level-of-detail for the $j^{th}$ sample, at the position of the image that is intersected by the lighting vector $i$ of the $j^{th}$ sampling direction.

The level-of-detail formula for the Ward-Dür model is found by first substituting the probability density function from Eq. (26) into the sampling solid angle formula (Eq. 86), which simplifies to:

$$\Omega_s = \frac{1}{N} \frac{4 \pi \alpha^2 (h \cdot r)(h \cdot n)^3}{\left( -\tan^2(\theta_n) \right)^{\frac{1}{2}}} \quad (89)$$

Combining this result with the approximation of the solid angle of a pixel in Eq. (85), the complete formula for determining the LOD for filtering each sample is:
\[ \text{LOD} = \max \left( 1 - \frac{1}{2} \log_2 \left( \frac{1}{N} \cdot \frac{4\pi\alpha^2(h \cdot r)(h \cdot n_{surf})^3}{\frac{-\tan^2(\theta_i)}{e^{\frac{-\tan^2(\theta_i)}{\alpha^2}}} \cdot \frac{w_{\text{pix}} \cdot h_{\text{pix}} \cdot r_{(a,b)}^2}{A} \cdot \left( n_{\text{light}} \cdot -i_{(a,b)} \right) + 0} \right) \right) \] (90)

7.4.1.5 Padding the Luminance Image for Edge Effects

When using bounded area sources instead of hemispherical maps of an entire illumination environment, artifacts can result when sampling near the source edges. In the object display rendering engine, this was mitigated to some degree by zero-padding the light source image with a black border to increase the physical area subtended by the light source model. Though the luminance and physical size of the real source in the model is not changed, the padding allows the low-pass filtered versions of the light source image to extend beyond the bounds of the physical light source. This provides a larger sampling area (increasing the number of intersections with the filtered source) and avoids an abrupt transition at the light source edge when calculating semi-gloss surface reflections. However, padding the luminance image does have disadvantages for rendering efficiency, as it increases the image size (in pixels) and the amount of data storage and processing required. The image can be rescaled back to its original size after padding to maintain the same amount of processing, but this results in a reduced resolution for the portion actually encoding the light source. In practice, the light sources used with the object display are generally given only a narrow black border (10% of the light source edge size).
7.4.2 Median Cut Lighting for Gloss in the Object Display System

The filtered importance sampling method described in the previous section is effective when the surface is highly specular and the source is large enough that a large number of samples are likely to intersect the source. However, if the specular lobe is broad (a semi-matte or matte surface, with a large value of the $\alpha$ parameter) and the area source is relatively small, then the rendered result may have significant noise because the estimate will be based on only a small number of samples that intersect the light source. Conversely, as the specular lobe becomes wider and the area sources become smaller, the median cut point light representation of the area illumination source becomes an increasingly better approximation. A theoretical point light subtends an infinitely small solid angle, so as the solid angle of an area source approaches zero, it behaves more similarly to a point light. Additionally, as the BRDF lobe becomes wider and the surface becomes more diffuse, small changes in the illumination direction have a smaller impact on the outgoing reflection. In the BRDF calculation, the point light approximation primarily introduces error into the set of illumination directions, which results from transferring illumination from all the positions over the area source to a subset of concentrated point light positions.

The object display rendering system already uses the median cut point light approximation to evaluate the (Lambertian) diffuse lobe of the Ward-Dür model, and so it is possible to evaluate the specular lobe approximation without much additional interactive computation. The incident illuminance on the surface for each of the median cut points is calculated for use with the diffuse lobe evaluation by:

$$E_{(s,t)} = \sum_{k=1}^{M} L_k \left( \mathbf{n}_{\text{surf}(s,t)} \cdot \mathbf{i}_k \right) \frac{\left( -\mathbf{n}_{\text{light}} \cdot \mathbf{i}_k \right)}{d_k^2} \left( \frac{A}{w_{\text{pix}} \cdot h_{\text{pix}}} \right)$$

(91)
where $M$ is the number of median cut points, $A$ is the physical area of the region in the light image (in m$^2$), $d_k$ is distance between the surface point and the current median cut point (in m), and $L_k$ is the summed luminance stored in the $k^{th}$ light point. In the diffuse lobe evaluation, each light point illuminance is multiplied by the diffuse BRDF term ($\rho_d / \pi$). Within the same loop, the specular lobe can also be evaluated directly over the set of calculated illuminances using the specular BRDF portion of the Ward-Dür model. The luminance of the specular reflection is calculated by:

$$L_r = \sum_{k=1}^{M} \left( E_{k(s,t)} \right) \frac{\rho_s}{4\pi\alpha^2 \left( n_{surf(s,t)} \cdot i_k \right) \left( n_{surf(s,t)} \cdot r_{(s,t)} \right)} \exp \left( \frac{1 - 1}{\left( n_{surf(s,t)} \cdot h_k \right)^2} \frac{1}{\alpha^2} \right)$$  \hspace{1cm} (92)

where $E_{k(s,t)}$ is the illuminance due to the $k^{th}$ light point, $r_{(s,t)}$ is the unit vector from the surface point to the observer position, and the exponential term is a computationally efficient vector form of the $\tan^2(\theta_h)$ term in the Gaussian portion of the Ward-Dür model.

### 7.4.3 Method for Hybrid Rendering Selection

To take advantage of the relative strengths of the two approximate rendering techniques, it is necessary to have a method to quickly assess which technique will provide better accuracy during the rendering process. In the case of the materials with spatially-varying BRDF, the more accurate technique may vary on a pixel-by-pixel basis over the virtual surface. To address this issue, a selection method was developed that can be applied in real-time for each pixel of the rendered surface.
7.4.3.1 Deriving an Error Approximation Metric

As there is a substantial difference between the type of approximation made by the two techniques, one of the challenges was to derive a single metric that could be applied to either. In the case of the median cut technique, the error stems from discretizing a continuous extended light source and concentrating its energy into a small number of lighting points, without consideration to whether the resulting gaps may correspond to important directions in the surface BRDF. With filtered importance sampling, a discrete set of the most important directions, based on the BRDF, are evaluated. However, if the sampling is too sparse, many of the samples may fail to intersect the light source, resulting in a poor estimate of the specular reflection. The selection criteria was developed to consider how much error is likely to result for each of the techniques given the density of lighting samples and the BRDF distribution.

In the variance-minimizing median cut method [Viriyothai & Debevec 2009], the set of point lights are selected to minimize a spatial distance-based metric that describes how far each pixel in a given cluster is from the new concentrated point light position. Viriyothai and Debevec’s clustering metric uses the squared distance error for each pixel and weights the error by the light intensity:

\[
Var = \sum_j L_j d_j^2
\]

(93)

where \(d_j\) is the distance (in pixels) between the \(j^{th}\) pixel of the image and the centroid of the cluster to which it has been assigned, and \(L_j\) is the light intensity of the \(j^{th}\) pixel in the cluster.

The variance metric in Eq. (93) serves as a starting point for the hybrid selection criteria. To determine the relative importance of this lighting-weighted distance error to the overall rendering calculation, the BRDF of the surface also needs to be taken into account. If there is a
relatively large distance error for a given point light, but the specular lobe is not pointed in that light direction or the specular lobe is broad, the light distance error will not have a major impact on the specular reflection. However, if the specular lobe is narrow and its peak direction is near a given light point, then the light distance error may lead to significant error in the estimated reflection. In the selection criteria, the distance variance term for each point light is weighted by an estimate of the surface BRDF in that lighting direction:

\[ \sum_{i=1}^{N} f_i \cdot Var_i \]  

(94)

where \( f_i \) is the BRDF evaluated in the direction of the \( i^{th} \) median cut light point and \( Var_i \) is the summed variance term from Eq. (93) for the \( i^{th} \) light.

The weighted metric is normalized by dividing out the two weighting factors (BRDF and light intensity) that were applied to the distance error:

\[ \frac{\sum_{i=1}^{N} f_i \cdot Var_i}{\sum_{i=1}^{N} f_i \cdot L_{total,i}} \]  

(95)

where \( L_{total,i} \) is the luminance of the \( i^{th} \) point light, found by summing the luminances of the \( M \) pixels contained in the cluster it represents:

\[ L_{total,i} = \sum_{j=1}^{M_i} L_j \]  

(96)

These two weighting factors are necessary for determining the relative importance of the distance error from different samples (i.e. considering whether light point clusters with large errors are the ones that also have a relatively large impact on the reflection), but their overall sum should not increase the error metric. An increase in the global sum of the \( f_i \) weights and \( L_{total,i} \)
weights does not imply that there is more error present; only that overall more BRDF directions and lighting positions with a high importance were considered in the rendering estimate.

Finally, this normalized variance metric is converted to a standard deviation by taking its square root, and then it is divided by the square root of the number of samples used in calculating the metric. In this form, it resembles the calculation of the standard error of the mean used in statistics to describe how close a point estimate is likely to be to the true population mean.

By including the sample size in the denominator, the value of the error metric will increase when smaller numbers of samples are used in the lighting calculation. This is important for the case of a highly glossy surface with a narrow specular lobe, where the median cut algorithm is likely to perform poorly. In this case, nearly all the point light locations may fall outside the cone of the specular lobe, leading to a highly unstable lighting calculation with the potential for large errors. For the purposes of determining the number of samples \( N \) to use in the calculation, a small BRDF threshold value is set (e.g. a threshold value of 0.0001, for the product of the BRDF and luminance). The use of this threshold is necessary because a specular BRDF value away from the specular peak just approaches zero and will still have a non-zero value, even if its magnitude is so small as to be insignificant in the rendering calculation. The final metric in its general form is:

\[
Error = \frac{1}{\sqrt{N_{\text{thresh}}}} \, \left( \frac{\sum_{i=1}^{N_{\text{thresh}}} f_i \cdot \text{Var}_i}{\sum_{i=1}^{N_{\text{thresh}}} f_i \cdot L_{\text{total},i}} \right)
\]  

(97)

where \( N_{\text{thresh}} \) is the number of point lights that pass the minimum threshold criteria. The complete formula for calculating the metric from the clustered pixels of a luminance map image is:
\[
\text{Error} = \frac{1}{\sqrt{N_{\text{thresh}}}} \sqrt{\sum_{i=1}^{N_{\text{thresh}}} \left( \sum_{j=1}^{M} f_i \sum_{j=1}^{M} L_j d_j^2 \right)}
\]

This error metric can be evaluated in real-time in the fragment shader for each pixel of the virtual surface. The summed value of the luminance for each cluster, \( L_{\text{total},i} \), will already be available in the fragment shader because these are the point light luminances used to perform the rendering. The variance distance metric for a cluster is calculated during the clustering process. Though not typically required for rendering with median cut lighting, these cluster variance values can be stored along with the luminance values for each point light to facilitate interactive calculation of the error metric.

### 7.4.3.2 Calculating the Error Metric for Filtered Importance Sampling

The selection metric follows closely from the form of the error in the median cut method, but an equivalent calculation for the filtered importance sampling method, which is not inherently based on a distance on the light source surface, must also be determined. Though the filtered importance sampling technique uses a set of BRDF-based angular samples, it is possible to calculate an approximate light source distance error for each of these samples under certain assumptions.

From the median cut derivation, the starting point for the estimate is:
To approximate the variance term for the filtered importance sampling case, the luminance value is assumed to be constant within a sample region. With a constant value, it can be brought outside the summation:

\[
\text{Var}_i = \sum_{j=1}^{M_i} L_j d_j^2 \\
\approx L_i \left( \sum_{j=1}^{M_i} d_j^2 \right)
\]

(100)

The remaining squared distance portion can be approximated based purely on the area of a sample region if an assumption is made about its geometric shape. For an assumption that the averaged regions are square (on the basis of using a box filter to average regions), the sum of the squared distance is approximately:

\[
\sum_{j=1}^{M_i} d_j^2 \approx \frac{A_i^2}{6}
\]

(101)

where \(A_i\) is the area of the region in pixels. The derivation of the approximation in Eq. (101) is provided in Appendix F. In filtered importance sampling, the area of a sample (in image pixels) is estimated by the ratio of the solid angle of the sampling region to the solid angle of a pixel [Colbert & Křivánek 2007]:
where $\Omega_p$ is the solid angle of a pixel and $\Omega_s$ is the solid angle of the sample region to average based on the BRDF probability. Substituting this solid angle ratio provides a formula for a light distance-based variance metric that can be estimated from the BRDF sampling solid angle:

$$
Var_i \approx \frac{\Omega_s}{\Omega_p} \left( \frac{\Omega_s}{\Omega_p} \right)^2 \frac{1}{6} \quad (103)
$$

The mean luminance over the area of the $i^{th}$ sample can be approximated by performing a lookup into the luminance map at the appropriate level-of-detail, $\bar{L}_i \approx L_{i,LOD}$, giving the final variance term approximation:

$$
Var_i \approx \frac{\left( L_{i,LOD} \right) \left( \frac{\Omega_s}{\Omega_p} \right)^2}{6} \quad (104)
$$

The $L_{Total,i}$ term from the denominator of the error metric corresponds to the sum of all luminances in the image map for the $i^{th}$ sample region, which is equivalent to multiplying the mean luminance by the number pixels in the region. This can be approximated as the product of the area of the region (in pixels) and the lookup into the luminance map at the appropriate level-of-detail:
The final term that needs to be estimated for the filtered importance sampling case is the $f_i$ term used to weight the relative importance of each sample based on the BRDF. In the median cut case, it was the explicit BRDF function, but in filtered importance sampling, the BRDF is implicit as the general shape of the BRDF is largely captured by the choice of sampling directions. The $f_i$ term is replaced by the weighting function that relates the sampling pattern probability density function to the BRDF. For the Ward-Dür model, this weighting function was derived to be:

$$w_i = \rho_s \frac{(\mathbf{h} \cdot \mathbf{r}) (\mathbf{h} \cdot \mathbf{n})^3}{(\mathbf{n} \cdot \mathbf{r})}$$

where $\mathbf{h}$ is the half-vector, $\mathbf{n}$ is the surface normal, and $\mathbf{r}$ is the direction to the light source. Combining all these terms yields an equivalent error metric to the one used for the median cut method:

$$Error = \frac{1}{\sqrt{N_{\text{thresh}}}} \left( \sum_{i=1}^{N_{\text{thresh}}} w_i \left( L_{i,LOD} \cdot \frac{1}{6} \left( \frac{\Omega_s}{\Omega_p} \right)^2 \right) \right) \left( \sum_{i=1}^{N_{\text{thresh}}} w_i \left( L_{i,LOD} \cdot \left( \frac{\Omega_s}{\Omega_p} \right) \right) \right)$$ (106)
where again $N_{\text{thresh}}$ is the number of sample points that contribute to the calculation. Samples that do not intersect a light source (and therefore have a value of 0) or those with a magnitude below a small threshold value are not included in the $N_{\text{thresh}}$ number.

7.4.4 Simulation Results

Simulations were developed to evaluate the accuracy of the median cut method, the filtered importance sampling method, and the hybrid approach for rendering gloss highlights. In the simulations, the positions of a virtual light source, virtual sample, and virtual detector (luminance meter) were specified. Each approximate method was used to simulate the amount of light reflected from the surface in the direction of the detector. The position of the virtual detector was moved across a range of angles to provide a set of data for each method similar to a goniometric measurement of surface BRDF. For comparison to the approximate methods, the expected reflected luminance in the direction of the detector was calculated explicitly from the surface BRDF at every element of the [512 x 512] spatial luminance image representing the light source. For each element of the light source, the surface BRDF was evaluated based on the incident lighting and detector directions, the surface normals of the light and surface, the luminance stored for that element of the light source, and the area of each element. The results for these 262144 elements were summed to provide the total value for the luminance reflected toward the detector.

In the first simulation, the illumination was a small rectangular area source (2.3 cm x 3.5 cm) with a uniform luminance of 4200 cd/m$^2$, located 61 cm from the samples surface. (The lighting configuration in this simulation is based on the real physical configuration in the gloss
reproduction experiment described in Section 9.2). The results comparing the two individual methods to the baseline calculation are shown in Figure 49. The results for the hybrid approach, where the more appropriate method is selected based on the interactive error metric, are shown in Figure 50. In the first simulation the light source was relatively small, which generally favors the median cut method. A second simulation was performed by increasing the size of the lighting source to 20 x 20 cm. The results for the individual methods with this larger light source are shown in Figure 51 and are shown for the hybrid approach in Figure 52.
Figure 49. Simulation results comparing the filtered importance sampling and median cut methods for different Ward-Dür model α values. The illumination in the simulation is a 2.3 x 3.5 cm rectangular area source. The filtered importance sampling method (blue line) is using 64 samples and the median cut method (red line) is using 64 points. Both are compared to a baseline BRDF calculation that is performed for all 512 x 512 pixels in the spatial luminance image (black dots).
Figure 50. Simulation results comparing the hybrid method to an explicit BRDF calculation at every point of the spatial luminance image for different Ward-Dür model $\alpha$ values. The illumination in the simulation is a 2.3 x 3.5 cm rectangular area source. The hybrid method (black line) uses 64 samples for filtered importance sampling and 64 median cut points. The explicit BRDF calculation is performed for all 512 x 512 pixels in the spatial luminance image (red dots).
Figure 51. Simulation results comparing the filtered importance sampling and the median cut methods for a 20 x 20 cm area source. The filtered importance sampling method (blue line) is using 64 samples and the median cut method (red line) is using 64 points. Both are compared to a BRDF calculation that is performed for all 512 x 512 pixels in the spatial luminance image (black dots).
Figure 52. Simulation results comparing the hybrid method to an explicit BRDF calculation at every point of the spatial luminance image for a 20 x 20 cm area source. The hybrid method (black line) uses 64 samples for filtered importance sampling and 64 median cut points. The explicit BRDF calculation is performed for all 512 x 512 pixels in the spatial luminance image (red dots).
With the small light source, the median cut method had low accuracy for the narrowest specular lobe ($\alpha=0.001$) and a small degree of error for $\alpha=0.002$, but was generally accurate for all the wider specular lobes shown in the plots. The filtered importance sampling method exhibited the opposite trend. It had high accuracy for the narrowest specular lobes, but began to differ from the baseline calculation in the plot for $\alpha=0.02$. As the surface became more diffuse ($\alpha > 0.1$), there were major errors in the plotted filtered importance sampling results. The hybrid approach produced results consistent with baseline calculation over the entire $\alpha$ range.

With the larger 20 x 20 cm light source, the general accuracy of the median cut method declined, while the general performance of the filtered importance sampling method improved. For narrow specular lobes, the median cut method significantly underestimated the reflected luminance. It became relatively accurate for specular widths of $\alpha = 0.015$ and greater. The filtered importance method performed well for much of the $\alpha$ range, only exhibiting major differences from the baseline for the most diffuse sample. The hybrid again produced results that were highly consistent with the full image baseline calculations.

The types of visual artifacts that result from these errors are illustrated in Figure 53. The surface shown in the set of images was created with the digital model of a glass sample painted with a high gloss paint (sample GL100, described in Section 9.2). The glass sample is reflecting a 2.3 x 3.5 cm rectangular area source. The sample is modeled with a Ward-Dür BRDF model with two specular lobes. One lobe has a narrow specular producing the distinct reflection of the rectangular light source and the second lobe is broader and produces the haze around the sharp reflection. With filtered importance sampling, shown in the left image, the sharp reflection of the rectangular source is produced correctly, but the more diffuse haze lobe is produced with blocky artifacts around the edges. The median cut approach (shown in the center image)
reproduces the broader haze lobe accurately, but the individual point lights are visible in the sharp reflection from the narrow specular lobe. Using the hybrid approach (shown in the right image), the interactive selection method is able to choose the more appropriate method and produce both the sharp lobe and the broader lobe without visual artifacts.

![Importance Sampling](image1.png) ![Median Cut](image2.png) ![Hybrid](image3.png)

**Figure 53.** Left image, the rendering using the filtered importance sampling method has blocky artifacts around the edges of the specular lobe. Center image, the point lights used in the median cut approximation are visible in the specular reflection. Right image, these artifacts are avoided in the rendering with the hybrid approach.

### 7.5 Rendering in the Real-World Light Display Observer Coordinate System

In traditional image-based lighting, the environmental lighting is assumed to be infinitely far away from the surface and therefore can be parameterized and indexed directly by the unit illumination-direction vector. In the object display system, the real geometric relationship between the captured light source, object surface, and observer are considered and the rendering equation is evaluated in physical space, as opposed to a simplified shading coordinate system. The real-world coordinate system for the object display is illustrated in Figure 54.
Figure 54. The geometric coordinate system used for rendering is shown on an image of the light booth environment. The bottom left corner of the booth serves as the global origin. The light source and screen surface polygons are specified in the global coordinate system by the position of the top-left corner, their dimensions, and an orthogonal set of unit vectors specifying their orientation.

Importance sampling equations specify the directional distribution of samples in a local shading coordinate system, defined by the surface normal $[0 \ 0 \ 1]$ and two additional orthogonal vectors, $t$ and $b$. With the Ward-Dür model, the directional distribution is specified by a set of half-vectors ($h$ in Eq. 79), halfway between the illumination sample direction and the viewing direction in the local coordinate system. To calculate reflections for the physical configuration of the display screen, the half vector is transformed from the standard shading coordinate system.
to a vector in physical space using the geometric parameterization of the real screen surface and the orientation adjustment given by the normal map. In the first stage, a \textbf{tbn} frame is established for each point on the virtual surface from the surface orientation specified by the normal map (shown in Figure 55). For a flat region on the surface ($\mathbf{n}_{\text{map}} = [0 0 1]$) this frame is:

$$\begin{bmatrix} t & b & n \end{bmatrix}_{\text{map}=[001]} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$ (107)

In the case of other normal map values, the $\begin{bmatrix} t & b & n \end{bmatrix}_{\text{map}(s,t)}$ frame at the $(s,t)$ position on the surface is constructed by determining $t$ and $b$ vectors that are orthogonal to the normal vector specified in the map. An initial bitangent vector is found by taking the cross product of the flat surface tangent vector ([1 0 0]) and the map normal:

$$\mathbf{b}_{\text{temp}} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \times \mathbf{n}_{\text{map}(s,t)}.$$ (108)

If the map normal is equal to [1 0 0] or [-1 0 0], the cross product produces a null vector. The magnitude of $\mathbf{b}_{\text{temp}}$ is checked and if found to be 0, the null vector is replaced with [0 1 0]. The tangent vector is calculated as the cross product of the temporary bitangent and the map normal:

$$\mathbf{t}_{\text{map}(s,t)} = \mathbf{b}_{\text{temp}} \times \mathbf{n}_{\text{map}(s,t)}.$$ (109)

and the final bitangent vector is calculated as:

$$\mathbf{b}_{\text{map}(s,t)} = \mathbf{t}_{\text{map}(s,t)} \times \mathbf{n}_{\text{map}(s,t)}.$$ (110)

This map-based \textbf{tbn} frame is then converted to a physical \textbf{tbn} frame specified in the world space. This transformation is based on information from the tracking sensors and a model of the
display’s rotation on its stand. The display rotation model provides the screen’s surface normal \( \mathbf{n}_{\text{screen}} \), left-to-right direction vector \( \mathbf{r}_{\text{screen}} \), and bottom-to-top direction vector \( \mathbf{u}_{\text{screen}} \) in the physical world space. The \textbf{tbn} frame for each surface location, in world coordinates, is calculated by:

\[
\begin{align*}
\mathbf{t}_{\text{world}(s,t)} &= \begin{bmatrix} \mathbf{r}_{\text{screen}} & \mathbf{u}_{\text{screen}} & \mathbf{n}_{\text{screen}} \end{bmatrix} \mathbf{t}_{\text{map}(s,t)} \\
\mathbf{b}_{\text{world}(s,t)} &= \begin{bmatrix} \mathbf{r}_{\text{screen}} & \mathbf{u}_{\text{screen}} & \mathbf{n}_{\text{screen}} \end{bmatrix} \mathbf{b}_{\text{map}(s,t)} \\
\mathbf{n}_{\text{world}(s,t)} &= \begin{bmatrix} \mathbf{r}_{\text{screen}} & \mathbf{u}_{\text{screen}} & \mathbf{n}_{\text{screen}} \end{bmatrix} \mathbf{n}_{\text{map}(s,t)}
\end{align*}
\]  

(111)

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{diagram.png}
\caption{Two-dimensional diagrams illustrating the coordinate systems used in the object display system (the \( \mathbf{t} \) direction is orthogonal to the \( \mathbf{b} \) \( \mathbf{n} \) vectors and points into the page). In the left panel, the \( \mathbf{h} \) vector for sampling is initially specified relative to a fixed \textbf{tbn} frame in the local shading coordinate system. The map coordinate system, shown in the center panel, defines a \textbf{tbn} frame relative to the normal vector encoded in a normal map. These vectors are transformed to the world coordinate system, shown in the right panel, based on the physical orientation of the screen specified by \( \mathbf{u}_{\text{screen}}, \mathbf{n}_{\text{screen}}, \) and \( \mathbf{r}_{\text{screen}} \) (which points into the page and is not shown).

The half vector \( \mathbf{h} \) generated by the importance sampling equations (Eq. 79) in the local shading coordinate system is converted directly to the physical world space using the world \textbf{tbn} frame calculated in Eq. (111):}
\end{figure}
Note that this transform matrix differs for each position \((s, t)\) on the virtual object surface, because the transform is dependent on the orientation specified in the surface normal map.

Next, the illumination direction in the world space is calculated from the world space half-vector and the real viewing vector of the observer, using the standard formula for converting from a unit half-vector to a unit illumination vector (as described by Walter [2005]):

\[
i_{\text{world}(s,t)} = 2(r_{\text{world}(s,t)} \cdot h_{\text{world}(s,t)})h_{\text{world}(s,t)} - r_{\text{world}(s,t)} \quad (113)
\]

where \(r_{\text{world}(s,t)}\) is the unit vector from the physical position of surface point \((s, t)\) to the observer in the world space. After this calculation, all four direction vectors required to evaluate the magnitude of specular reflection \((r_{\text{world}}, h_{\text{world}}, n_{\text{world}}\text{ and } i_{\text{world}})\) for Eq. (88) are specified in the world coordinate system. This set of vectors is illustrated in Figure 56. Using these vectors, it is possible to calculate physical position on the light source that will be intersected by the illumination vector, so that the corresponding lookup into the light source luminance image can be performed.
Figure 56. A diagram illustrating the four unit vectors required in the rendering calculation at a position \((s, t)\) on the virtual surface. The surface normal \(\mathbf{n}\), in world coordinates, is determined from the normal map at \((s, t)\) and the screen orientation. The half-vector generated from the sampling equations is transformed to world coordinates \((\mathbf{h}_{\text{world}(s,t)})\) based on the frame derived from the surface normal. The unit eye-vector \(\mathbf{r}\) is the direction from surface position \((s, t)\) to the tracked position of the viewer. The illumination vector \(\mathbf{i}\) is calculated from \(\mathbf{h}\) and \(\mathbf{r}\) such that the angle between \(\mathbf{h}\) and \(\mathbf{i}\) and the angle between \(\mathbf{h}\) and \(\mathbf{r}\) are the same.

The physical position on the light source plane is determined with a ray-polygon intersection algorithm. The general formula for ray-polygon intersection (from a computer graphics text, e.g. Shirley [2005]) gives the distance, \(d\), from a point on the ray to its intersection with a polygon according to:

\[
d = \frac{(\mathbf{p}_{\text{poly}} - \mathbf{p}_{\text{ray}}) \cdot \mathbf{n}}{\mathbf{v} \cdot \mathbf{n}}
\]  

(114)

where \(\mathbf{p}_{\text{poly}}\) is a point on the polygon, \(\mathbf{p}_{\text{ray}}\) is a point on the ray, \(\mathbf{n}\) is the normal to the polygon and \(\mathbf{v}\) is the direction vector for the ray. In the rendering system, the distance along the illumination vector from the object surface position \((s, t)\) to the light plane is calculated by:
\[
d = \frac{(\mathbf{o}_{\text{TL,light}} - \mathbf{p}_{\text{surf}(s,t)}) \cdot \mathbf{n}_{\text{light}}}{(\mathbf{i}_{\text{world}(s,t)}) \cdot \mathbf{n}_{\text{light}}}
\]  

(115)

where \( \mathbf{p}_{\text{surf}(s,t)} \) is the physical position of the object surface point, and the light plane is specified by the plane normal \( \mathbf{n}_{\text{light}} \) and a point on the plane \( \mathbf{o}_{\text{TL,light}} \) (in this case the top left corner of the light rectangle, which is considered the origin). It is necessary to perform checks on the denominator as well as the sign of \( d \), as the actual illumination vector will only intersect the plane if \( d \) is positive and if the denominator of Eq. (115) is non-zero. If these conditions are met, then the light intersection point in space is found by moving from the object surface point a distance \( d \) along the illumination unit vector (as shown in Figure 57):

\[
\mathbf{p}_{\text{light}} = d(\mathbf{i}_{\text{world}(s,t)}) + \mathbf{p}_{\text{surf}(s,t)} \cdot
\]  

(116)

**Figure 57.** Diagram illustrating the calculation of the light intersection point, \( \mathbf{p}_{\text{light}} \), in physical space. The distance \( d \) from the surface point, \( \mathbf{p}_{\text{surf}(s,t)} \), to the light plane is calculated from a ray-plane intersection algorithm. The light intersection point is found by moving this distance in the direction given by the illumination unit vector, \( \mathbf{i}_{\text{world}(s,t)} \).
This calculated light position is specified in the world coordinate system. To determine the corresponding pixel in the spatial luminance image, the 2D parameterized position on the light plane is calculated by first determining the 3D position of this point relative to the origin of the light coordinate system:

\[ \mathbf{p}_{\text{relative}} = (\mathbf{p}_{\text{light}} - \mathbf{o}_{\text{TL,light}}). \]  

(117)

A pseudo-inverse calculation is then used to determine the unit pixel coordinates \((a, b)\) for the intersection:

\[
\begin{bmatrix}
  a \\
  b
\end{bmatrix} = \text{pinv}
\begin{bmatrix}
  l_A \cdot A_x & l_B \cdot B_x \\
  l_A \cdot A_y & l_B \cdot B_y \\
  l_A \cdot A_z & l_B \cdot B_z
\end{bmatrix}
\begin{bmatrix}
  \mathbf{p}_{\text{relative}}
\end{bmatrix}
\]  

(118)

where \(A=(A_x, A_y, A_z)\) is the unit left-to-right direction of the light source rectangle (when facing the light) in the world space, \(B=(B_x, B_y, B_z)\) is the unit top-to-bottom direction of the light source rectangle in the world space, and \(l_A\) and \(l_B\) are the physical dimensions of the light source rectangle in the \(A\) and \(B\) directions, respectively. The geometry is illustrated in Figure 58.

If both the \((a, b)\) unit pixel coordinates fall within the range [0 1], then the illumination vector intersects the actual bounded light source (as opposed to intersecting an area outside the light source, but still on the infinite plane containing the light source). These \((a, b)\) values serve as the texture lookup coordinates to index the spatial luminance image \(L_{LODj}\) when evaluating the reflectance in Eq. (88).
Figure 58. Illustration of the factors used to calculate the unit \((a, b)\) coordinates for the lookup into the illumination map. The relative light source position, \(p_{!relative}\), is determined by subtracting the light source origin, \(O_{light,TL}\), from the illumination vector intersection point. The corresponding 2D unit coordinates are determined based on the vector parameterization of the surface \((A_{light} and B_{light})\) and its physical dimensions \(l_A\) and \(l_B\).
8 DISPLAY SCREEN MODELING

8.1 Overview

The display screen is one of the most critical components in an object display system. The other components are necessary to calculate accurate surface reflections, but ultimately the display screen must be able to physically produce the calculated per-pixel luminance and chromaticity values if the object display is to successfully simulate the light patterns of a real surface. Several properties of the display screen impact the capability of the object display system to accurately reproduce the appearance of an object. There are four main considerations for selecting a screen to use in an object display system:

1. Color accuracy
2. Gamut (absolute luminance and chromaticity range)
3. Viewing-angle properties
4. Front surface reflection

The color accuracy of the display screen is an important consideration for several of the proposed uses of the object display framework. To be appropriate for use in the system, the colorimetry of the display screen must be adequately modeled by a characterization model that can be processed in real-time. The absolute luminance range of the display screen, as well as the color gamut that is maintained over that luminance range, is another important consideration for the capabilities of the object display. In standard display systems, often only relative colorimetry of the display is considered, but for an object display, the absolute luminance levels and the range of chromaticities that can be produced at a given luminance level are also critical.
absolute levels determine whether the display can simulate gloss highlights or can reproduce the appearance of a highly chromatic material at the same luminance level as a real object under similar ambient illumination. The directional, viewing-angle dependent properties of the display are also an important consideration for how well an object display can simulate the reflections from a real surface. The object display is intended to be viewed dynamically from different directions to simulate the viewing-angle dependent nature of gloss highlights, and so the stability of the display output with these viewing angle changes is important. Finally, the reflectance properties of the front surface of the display screen are another important consideration for an object display. It is intended to be viewed in an illuminated environment, so there will be physical light incident on the display screen. A screen with low front surface reflectivity is necessary to minimize unwanted physical reflections and reduce the amount of computational correction that is required.

### 8.2 Colorimetric Modeling

In the object display framework, the display characterization is used to model the screen in terms of its absolute colorimetric output. In a typical object display characterization, measurements are taken of the RGB ramps, the gray ramp, and a factorial combination of the RGB levels. These measurements are made using a spectroradiometer (e.g. PhotoResearch PR-655) instead of an illuminance colorimeter so that the display output can be modeled as a function of absolute luminance. Typically the characterization method used is the Day et al. [2004] approach, which is based on the Fairchild and Wyble [1998] model form.
In the typical object display workflow, the determination of RGB digital counts for the XYZ rendering results is achieved at real-time rates using a GPU shader-based implementation for the inverse direction of the Day et al. characterization. The three 1D inverse LUTs that describe the transfer functions of the RGB channels (linear RGB scalars to digital counts) are encoded as a three-channel [2048 x 1] pixel floating point image that is uploaded to the GPU. The [3 x 3] matrix, $M_{inv}$, describing the inverse transform from XYZ to RGB radiometric scalars and the [3 x 1] vector, $XYZ_K$, for black point subtraction are stored as constants in the fragment shader. When the XYZ value of the surface reflection at a pixel is determined, the black level is subtracted, and then the XYZ to RGB matrix transform is applied to calculate radiometric scalars:

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = M_{inv} \begin{bmatrix} X - X_K \\ Y - Y_K \\ Z - Z_K \end{bmatrix}.$$  

(119)

These radiometric scalars are used to index the 3 x 1D inverse LUTs stored as the 3 channels of a [2048 x 1] floating point texture to determine the digital count control values for the display:

$$DC_R = \text{LUT}^{-1}(R)$$

$$DC_G = \text{LUT}^{-1}(G)$$

(120)

$$DC_B = \text{LUT}^{-1}(B).$$

This serves as the basis for display modeling in the object display system. Additional processing has been incorporated to address out of gamut colors and luminance adjustments for off-axis viewing. Additionally, a more complex model was developed to provide a higher level of
colorimetric accuracy for the chromatic adaptation visual experiment. This enhanced model is described in detail in Appendix A, along with data comparing the accuracy of the enhanced model to the standard Day et al. approach for the EIZO Radiforce RX220 display screen used in the object display system.

8.2.1 Mapping Out-of-Gamut Colors

Given the physical limitations of the light output from the screen, it will not always be possible to generate the absolute CIE XYZ values calculated by the rendering engine. In general, the real-world lighting surrounding the display should be set at an appropriate luminance level to produce reflections for typical materials that will be within the display gamut. However, certain elements like dark shadows or specular reflections of lighting filaments will still likely result in regions of the virtual surface that are out of the absolute luminance range of the display. Additionally, highly chromatic surface colors, or some surface colors when combined with certain light sources (e.g. an orange patch under tungsten lighting), may also produce areas of the virtual surface that are out of the display gamut. In the object display framework, the goal is to handle these extreme cases while still displaying as much of the surface as possible at the absolute XYZ values calculated by the rendering engine.

There are three main cases that need to be addressed:

1. The target luminance level is below the minimum that can be produced by the screen (lower than the display black point at zero digital counts).
2. The target color has an absolute luminance level that is higher than the output capabilities of the screen at the target chromaticity, but the target chromaticity is within the triangle formed by the RGB primaries.

3. The target chromaticity is out of gamut for any luminance level (outside the triangle formed by the RGB primaries).

The procedure for mapping out-of-gamut stimuli needs to be performed at interactive frame rates on the GPU and for every pixel of the virtual surface, so it is important that it require only minimal computation and data storage. In general, it would be expected that the best mapping results could be obtained using a multiple step process with a conversion to a color appearance space (e.g. CIECAM02) and then back to CIE XYZ, but that would introduce a significant amount of computation. In the following section, a set of computationally-simple methods to meet the objectives of the object display framework are described.

8.2.1.1 Case 1 – Target Luminance below the Display Minimum

In case 1, the presence of target XYZ values below the black point will result in negative black-level corrected XYZ values ($XYZ_{kCor}$) in the initial stage of the Day et al. [2004] approach:

\[
\begin{bmatrix}
X \\
Y \\
Z_{kCor}
\end{bmatrix}
= \begin{bmatrix}
X \\
Y \\
Z_{target}
\end{bmatrix} - \begin{bmatrix}
X \\
Y \\
Z_K
\end{bmatrix}
\]  \hspace{1cm} (121)

Negative values in the calculated $XYZ_{kCor}$ will result in negative values for one or more of the RGB radiometric scalars on the display primaries. The simplest solution is to clip any negative radiometric scalars to zero, generating the minimum possible output from any negative target values.
channels. This is the approach generally used in the object display system. In this case, the error introduced by the mapping will typically result in a change in hue from the target color and will also introduce a small luminance increase (equal to the luminance output that would be subtracted, if it were possible, by the negative radiometric scalar). Though a hue shift is undesirable, alternatives that better maintain the hue will result in larger increases in the luminance level. For example, it would be possible to maintain the chromaticity by multiplicative scaling of the target XYZ by a factor $S$ until the XYZ values were above the minimum values for the screen:

$$
\begin{bmatrix}
X \\
Y \\
Z_{\text{mapped}}
\end{bmatrix} = S \cdot 
\begin{bmatrix}
X \\
Y \\
Z_{\text{target}}
\end{bmatrix} \quad (122)
$$

However, this has the potential to increase the radiometric scalars on all three channels (even non-negative ones that do not require clipping), and could increase the lightness of dark target colors to an unacceptable level.

### 8.2.1.2 Case 2 – Target Luminance Exceeds Display Capabilities

In case 2, the chromaticity of the target XYZ is within the triangle formed by the primaries, but the absolute luminance level is higher than the screen can produce at that chromaticity. Within the object display framework, the situation where the highest luminance values are expected to occur is for sharp reflections of light sources from high gloss surfaces. Given that incorrect chromaticities of these highlights could negatively impact the chromatic adaptation state, the gamut mapping procedure was selected to maintain chromaticity at the
expense of luminance. A scaling factor (less than one) is applied to the target XYZ until it is within the display gamut. The method used to determine the scaling factor is described below.

The factor $S$ for scaling the target XYZ (to bring it within gamut) is determined so that the radiometric scalars for controlling the display ($R_{control}$, $G_{control}$, and $B_{control}$) are all less than one in the formula:

$$
\begin{bmatrix}
R \\
G \\
B
\end{bmatrix}_{control} = \mathbf{M}_{inv} \left( \begin{bmatrix}
X \\
Y \\
Z
\end{bmatrix}_{target} - \begin{bmatrix}
X \\
Y \\
Z
\end{bmatrix}_K \right)
$$

(123)

where $\mathbf{M}_{inv}$ is the matrix used to convert from XYZ to the display RGB radiometric scalars (as used in the inverse direction of the Day et al. characterization):

$$
\mathbf{M}_{inv} = \begin{bmatrix}
X_{R_{max}} - X_K & X_{G_{max}} - X_K & X_{B_{max}} - X_K \\
Y_{R_{max}} - Y_K & Y_{G_{max}} - Y_K & Y_{B_{max}} - Y_K \\
Z_{R_{max}} - Z_K & Z_{G_{max}} - Z_K & Z_{B_{max}} - Z_K
\end{bmatrix}^{-1}
$$

(124)

The $S$ factor is calculated by distributing the $\mathbf{M}_{inv}$ matrix:

$$
\begin{bmatrix}
R \\
G \\
B
\end{bmatrix}_{control} = S \cdot \mathbf{M}_{inv} \begin{bmatrix}
X \\
Y \\
Z
\end{bmatrix}_{target} - \mathbf{M}_{inv} \begin{bmatrix}
X \\
Y \\
Z
\end{bmatrix}_K
$$

(125)

and converting to radiometric scalar RGB values:

$$
\begin{bmatrix}
R \\
G \\
B
\end{bmatrix}_{control} = S \cdot \begin{bmatrix}
R \\
G \\
B
\end{bmatrix}_{target} - \begin{bmatrix}
R \\
G \\
B
\end{bmatrix}_K
$$

(126)
This expression for the RGB control values is then set to be less than 1 for each channel:

\[
S \begin{bmatrix} R \\ G \\ B_{\text{target}} \end{bmatrix} - \begin{bmatrix} R \\ G \\ B \end{bmatrix}_K < \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}_{\text{control}}.
\]  

(127)

To meet these criteria, the value S must satisfy the three inequalities:

\[
S < \frac{1 + R_K}{R_{\text{target}}}, \quad S < \frac{1 + G_K}{G_{\text{target}}}, \quad S < \frac{1 + B_K}{B_{\text{target}}}. \]

(128)

All three inequalities are satisfied by taking the minimum of the channels, giving the formula for selecting S:

\[
S = \min \left( 1, \frac{1 + R_K}{R_{\text{target}}}, \frac{1 + G_K}{G_{\text{target}}}, \frac{1 + B_K}{B_{\text{target}}} \right).
\]  

(129)

The one is included in the minimum expression to insure the target values are not increased in the case where all three channels are already within gamut.

### 8.2.1.3 Case 3 - Target XYZ outside the Chromaticity Triangle

In case 3, the target XYZ has a chromaticity that is outside the triangle formed by the three primaries. Attempting to achieve the target chromaticity for the set of display primaries results in a negative radiometric scalar on at least one of the RGB channels. The simplest
method for addressing this situation is to clip the negative radiometric scalar to 0, but this has
the potential to produce an undesirable hue shift. An alternative is to maintain the hue angle, but
reduce the chroma until the color is within the display gamut.

A computationally-simple method was used to reduce the chroma by adding a portion of
the rendered white point XYZ to the target color until it is brought within the display gamut.
This shifts the color along a line between the target color and the rendering white point in a
chromaticity diagram. This maintains the hue angle in the CIELUV space, though will not
maintain a line of constant hue in other color spaces, such as CIELAB, or appearance spaces that
include modeling of perceptual hue effects (e.g. the Abney effect).

The final radiometric scalars for controlling the display (RGB\textsubscript{control}) are determined by
adding an amount of white, XYZ\textsubscript{W}, to the original target XYZ. The unit XYZ\textsubscript{W} term being
scaled by the factor $S$ has the chromaticity of the rendering white point and a luminance of 1
cd/m$^2$. An additional scaling factor $\frac{Y_0}{Y_0 + S}$ is applied after the white addition to return to the
original luminance $Y_0$ of the target color. The formula for the RGB control values with the white
addition is:

$$
\begin{bmatrix}
  R \\
  G \\
  B_{\text{control}}
\end{bmatrix}
= M^{-1}\begin{bmatrix}
  Y_0 \\
  Y \\
  Z_{\text{target}}
\end{bmatrix}
+ S \cdot \begin{bmatrix}
  X \\
  Y \\
  Z_{\text{W}}
\end{bmatrix}
- \begin{bmatrix}
  X \\
  Y \\
  Z_{\text{K}}
\end{bmatrix}.
$$

To calculate the appropriate value for $S$, it is necessary to find the minimum $S$ value that will
insure that all the RGB radiometric scalars are non-negative:
\[
\mathbf{M}_{inv} \left( \frac{Y_0}{Y_0 + S} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}_{target} + S \cdot \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}_W \right) \left( \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}_R - \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}_K \right) \approx \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}.
\] (131)

Multiplying each term by the inverse matrix from XYZ to RGB gives:

\[
\frac{Y_0}{Y_0 + S} \begin{bmatrix} R \\ G \\ B \end{bmatrix}_{target} + S \cdot \begin{bmatrix} R \\ G \\ B \end{bmatrix}_W \left( \begin{bmatrix} R \\ G \\ B \end{bmatrix}_R - \begin{bmatrix} R \\ G \\ B \end{bmatrix}_K \right) \approx \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}.
\] (132)

Solving for \(S\) (shown here for the R channel, with same process applied for G and B):

\[
\frac{Y_0}{Y_0 + S} R_{target} + \frac{Y_0}{Y_0 + S} S \cdot R_w - R_K \geq 0
\]

\[
\Rightarrow S \geq \frac{Y_0 \left( R_K - R_{target} \right)}{Y_0 \cdot R_w - R_K}.
\] (133)

The final \(S\) value is the largest factor required by any of the channels to insure that all three channels are non-negative:

\[
S = \max \left( 0, \frac{Y_0 \left( R_K - R_{target} \right)}{Y_0 \cdot R_w - R_K}, \frac{Y_0 \left( G_K - G_{target} \right)}{Y_0 \cdot G_w - G_K}, \frac{Y_0 \left( B_K - B_{target} \right)}{Y_0 \cdot B_w - B_K} \right).
\] (134)

The absolute value is used in the denominator to insure that the sign of \(S\) is positive when an RGB target value is negative (a negative RGB value is out of gamut and requires the addition of \(S\) units of white to move the color into the gamut, so \(S\) and RGB have opposite signs). If all the expressions in Eq. (134) are negative (indicating that the color is in gamut), the formula evaluates to 0, and no white is added.
8.2.2 Directional Display Properties

The light emitted from a LCD panel typically varies as function of the viewing angle. In the case of the object display, the observer is expected to move relative to the screen and so there are likely to be changes in viewing angle that can affect the display output. The angle from the viewer to the display is interactively tracked, so it is possible to estimate the real viewing angle of the observer. As demonstrated by Li et al. [2004] and Cheng [2007], an interactive correction model can be used to compensate for viewing angle dependencies when real-time tracking information is available. Though these types of interactive corrections can be applied, a high-quality IPS-based LCD panel was still selected in an effort to minimize the viewing-angle dependency and keep the amount of corrective modeling required to a minimum.

8.2.2.1 Viewing-angle Measurements

A set of measurements was taken to evaluate the degree of viewing dependency and assess the changes in chromaticity, absolute luminance, and screen gamma as a function of angle. The measurement data was evaluated to determine what types of corrective modeling are necessary to provide acceptable colorimetric accuracy for the object display system at viewing angles up to 30 degrees. The white point, black level, nine gray levels, and the RGB primaries at their maximum output were measured in 5 degree intervals from -30 to +30 degrees off-axis in the horizontal direction with a spectroradiometer (PhotoResearch PR655) mounted to a goniometer. Measurements were taken with the display in a landscape orientation and with the display rotated to a portrait orientation. The measurement configuration for the display in a landscape orientation is shown in Figure 59.
The chromaticities of the white point and the RGB primaries at their maximum output for a horizontal sweep in the landscape orientation are shown in Figure 60 (left panel) and for the portrait orientation (right panel). The mean and standard deviation of the chromaticity measurements for each over the ±30 degree range are shown in Table 4.
Figure 60. Chromaticity over a set of angles for the display in landscape orientation (left) and portrait orientation (right). The chromaticities of the display white point (gray dots) and the red, green, and blue channels at their maximum output are plotted for a horizontal sweep from -30° to +30° with measurements taken every 5°. The change with viewing angle is small enough that the points over the +/-30 degree range are nearly coincident.

Table 4. Mean and Standard Deviation of Chromaticity over the ±30 Degree Range

<table>
<thead>
<tr>
<th>Orientation</th>
<th>Color</th>
<th>Mean</th>
<th>Standard Deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>$u'$</td>
<td>$v'$</td>
</tr>
<tr>
<td></td>
<td></td>
<td>$u'$</td>
<td>$v'$</td>
</tr>
<tr>
<td>Portrait</td>
<td>Red</td>
<td>0.4617</td>
<td>0.5239</td>
</tr>
<tr>
<td></td>
<td>Green</td>
<td>0.1309</td>
<td>0.5642</td>
</tr>
<tr>
<td></td>
<td>Blue</td>
<td>0.1585</td>
<td>0.1757</td>
</tr>
<tr>
<td></td>
<td>White</td>
<td>0.1966</td>
<td>0.4689</td>
</tr>
<tr>
<td>Landscape</td>
<td>Red</td>
<td>0.4614</td>
<td>0.5236</td>
</tr>
<tr>
<td></td>
<td>Green</td>
<td>0.1309</td>
<td>0.5638</td>
</tr>
<tr>
<td></td>
<td>Blue</td>
<td>0.1590</td>
<td>0.1741</td>
</tr>
<tr>
<td></td>
<td>White</td>
<td>0.1966</td>
<td>0.4667</td>
</tr>
</tbody>
</table>

The viewing-angle dependence of the screen gamma was evaluated by measuring an 11-step gray ramp (nine intermediate gray levels, plus the white and black levels). It was expected that the absolute luminance of each level would change as a function of the viewing angle, so the ramps were normalized to compare whether each had a consistent shape. In the normalization,
the black level at each viewing angle was subtracted from the other ramp measurements. These black-level-corrected values were then peak normalized to a [0, 1] scale by dividing each by the black-level-corrected white point of the corresponding viewing angle. These normalized gamma curves for the viewing angles from -30 degrees to +30 degrees (in 5 degree intervals) are shown in Figure 61 for both the portrait and landscape orientations.

![Figure 61. Relative luminance for gray ramps at viewing angles from -30 to 30 degrees off-axis. The relative data shown is black-level corrected and normalized by the peak luminance for the viewing angle.](image)

The change in the absolute luminance output of the display as a function of viewing angle was also evaluated. The maximum display luminance values at viewing angles from -30 to +30 degrees for the portrait and landscape orientations are shown in Figure 62. (Note: the portrait and landscape measurements were taken as separate sequences and were not at the exact same spatial location on the screen, which may explain the absolute luminance difference at the 0
degree measurement angle.) The viewing-angle results were also calculated in terms of the proportion of the on-axis luminance, by dividing the measurement at each angle by the 0-degree measurement. These relative luminance results are shown in Figure 63.

**Figure 62.** Maximum display luminance as a function of viewing angle. The horizontal sweep from -30 to +30 degrees for the portrait orientation is plotted in blue. The horizontal sweep for the landscape orientation is shown in red.
Figure 63. The relative viewing-angle luminance as a proportion of the on-axis luminance for the display. The portrait orientation is shown in blue and the landscape orientations in red.

The measurement results indicate that a change in viewing angle primarily affected the absolute luminance, and that it did not produce major changes in the gamma and chromaticity within the +/-30 degree range. Based on these results, a luminance correction model was incorporated into the object display system, but a chromaticity correction was not included. In general, an effort was made to minimize the amount of viewing-angle modeling incorporated, because it had the potential to introduce global errors that could be distracting to a tracked observer and also required that stricter limitations be placed on the output capabilities of the system for on-axis viewing. In the luminance correction model, for example, there is a tradeoff between the amount of correction possible for the off-axis luminance reduction and the
maximum on-axis luminance that can be produced. In the typical object display settings, the luminance model is set to maintain a constant white point luminance level out to +/- 20 degrees, which requires limiting the on-axis luminance to approximately 80% of its maximum.

8.2.2.2 Modeling the Viewing-angle Luminance Dependence

The off-axis luminance changes are addressed by applying a viewing-angle based adjustment model to the on-axis Day et al. [2004] characterization method. The model incorporates a variable term for the luminance of the zero-offset (black level) and a viewing-angle-based multiplicative scaling factor. Both are specified in terms of the zenith and azimuth \((\theta, \phi)\) angles of the viewing vector. The zenith angle \(\theta\) is the angular deviation from the screen normal (the degree off-axis, in the -30 to +30 degree range). The azimuth angle \(\phi\) describes the direction along the screen surface, where 0 degrees is the direction toward the right of the screen and 90 degrees is the direction toward the top of the screen (for the horizontal sweep in portrait orientation, \(\phi = 0\) degrees and for the landscape orientation, \(\phi = -90\) degrees).

In the standard Day et al. approach, there is a single fixed \(XYZ_K\) term that is used to compensate for the non-zero black level of the display. In the viewing-angle-based model, the black level luminance value, \(Y_K\), is specified as a function of \((\theta, \phi)\). This black level luminance is modeled empirically by fitting a multiple regression model for an elliptical paraboloid shape:

\[
\hat{Y}_{k(\theta, \phi)} = \beta_0 + \beta_1 \cdot \theta \cdot \cos(\phi) + \beta_2 \cdot \theta \cdot \sin(\phi) + \beta_3 \cdot (\theta \cdot \cos(\phi))^2 + \beta_4 \cdot (\theta \cdot \sin(\phi))^2 + \epsilon
\] (135)
where $\theta$ and $\phi$ are specified in degrees. The measured black-level values are shown with the model results in Figure 64.

\[ L_{\text{factor}}(\theta, \phi) = \beta_0 + \beta_1 \cdot \theta \cos(\phi) + \beta_2 \cdot \theta \sin(\phi) + \beta_3 \cdot (\theta \cos(\phi))^2 + \beta_4 \cdot (\theta \sin(\phi))^2 + \epsilon \quad (136) \]

**Figure 64.** Results of the model fitting for the black level luminance (DC=0) as a function of viewing angle. The measured values are shown as dots and the model predictions as solid lines.

A similar second-order regression model was developed to estimate the factor describing the luminance reduction of the screen as a function of viewing angle:
The luminance factor quantity predicted in the regression is the ratio of the luminance at the angle \((\theta, \phi)\) to the luminance at normal viewing \((\theta = 0)\) after black level correction for each:

\[
L_{\text{factor}(\theta, \phi)} = \frac{Y_{dc, \theta, \phi} - Y_{K, \theta, \phi}}{Y_{dc, \theta=0} - Y_{K, \theta=0}}. \tag{137}
\]

The fitting was performed on the gray ramp data for the unit digital count levels: 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0. The fitting results for the portrait orientation data \((\phi = 0)\) are shown in Figure 65 and for the landscape orientation data \((\phi = -90)\) are shown in Figure 66.

**Figure 65.** The viewing-angle luminance factor model results for the portrait orientation measurement data. The measured points are represented by dots and the model fitting by the solid lines.
Figure 66. The viewing-angle luminance factor model results for the landscape orientation measurement data. The measured points are represented by dots and the model fitting by the solid lines.
The \( L_{\text{factor}(\theta, \varphi)} \) term is used as a scaling factor on the (black-level corrected) display primaries in the forward direction of the Day et al. characterization and the \( Y_{K(\theta, \varphi)} \) term is used to adjust the luminance of the black-level correction. With these two modifications, a viewing-angle dependent version of the forward calculation for the Day et al. approach is given by the formula:

\[
\begin{bmatrix}
X \\
Y \\
Z
\end{bmatrix} = \left( L_{\text{factor}(\theta, \varphi)} \right) \begin{bmatrix}
X_{R\text{max}} - X_K & X_{G\text{max}} - X_K & X_{B\text{max}} - X_K \\
Y_{R\text{max}} - Y_K & Y_{G\text{max}} - Y_K & Y_{B\text{max}} - Y_K \\
Z_{R\text{max}} - Z_K & Z_{G\text{max}} - Z_K & Z_{B\text{max}} - Z_K
\end{bmatrix} \begin{bmatrix}
R \\
G \\
B
\end{bmatrix} + \begin{bmatrix}
X \\
Y \\
Z
\end{bmatrix}
\]

(138)

where the [3 x 3] matrix multiplied by \( L_{\text{factor}(\theta, \varphi)} \) is the standard black-level-corrected on-axis matrix for the Day et al. approach. This calculation is inverted to determine the RGB values needed to control the display screen. The inverse of the adjusted XYZ-to-RGB matrix:

\[
M_{\text{inv}(\theta, \varphi)} = \left( L_{\text{factor}(\theta, \varphi)} \right) ^{-1}
\]

(139)

corresponds to the product of the reciprocal of the angular luminance factor and the standard on-axis inverse matrix used in the Day et al. approach:

\[
M_{\text{inv}(\theta, \varphi)} = \frac{1}{L_{\text{factor}(\theta, \varphi)}} M_{\text{inv} \theta=0}.
\]

(140)

Using this matrix, radiometric scalars are calculated as:
\[
\begin{bmatrix}
R \\
G \\
B
\end{bmatrix} = \frac{1}{L_{\text{factor}(\theta, \phi)}} \mathbf{M}_{\text{inv}, \theta = 0} \begin{bmatrix}
X \\
Y \\
Z
\end{bmatrix}_{\text{target}} - \begin{bmatrix}
X \\
Y \\
Z
\end{bmatrix}_{K(\theta, \phi)}.
\]

(141)

With this model formulation, colors viewed off-axis are adjusted to the correct target luminance by increasing the radiometric scalars to higher values than would be used if the screen were viewed on-axis. Attempting to correct high luminance target colors will result in RGB radiometric scalars that are greater than 1. To insure a constant display output out to a specified viewing angle, it is necessary to artificially limit the on-axis luminance by a set amount. This constraint is enforced within the gamut mapping portion of the system with a modification to Eq. (129). Instead of the 1’s in the original equation (shown below):

\[
S = \min \left( 1, \frac{1 + R_K}{R_{\text{target}}}, \frac{1 + G_K}{G_{\text{target}}}, \frac{1 + B_K}{B_{\text{target}}} \right)
\]

a \( C_{\text{max}} \) term, less than 1, is used to set the proportion of the display’s maximum channel output that can be used on-axis:

\[
S = \min \left( 1, \frac{C_{\text{max}}}{L_{\text{factor}(\theta, \phi)}} + \frac{R_K}{R_{\text{target}}}, \frac{C_{\text{max}}}{L_{\text{factor}(\theta, \phi)}} + \frac{G_K}{G_{\text{target}}}, \frac{C_{\text{max}}}{L_{\text{factor}(\theta, \phi)}} + \frac{B_K}{B_{\text{target}}} \right).
\]

(142)
By dividing the $C_{\text{max}}$ value by the viewing-angle luminance factor, the display output is limited to $C_{\text{max}}$ for on-axis viewing (where the $L_{\text{factor}}(\theta, \phi) = 1$), but can be increased up to the maximum output of the display when the off-axis viewing reduces the factor to $L_{\text{factor}}(\theta, \phi) = C_{\text{max}}$.

In the default implementation of the system, the $C_{\text{max}}$ value is set to 0.8. For the typical display mode of the screen (calibrated/luminance stabilized), which has a maximum luminance of approximately 400 cd/m$^2$, this artificially constrains the maximum output to 320 cd/m$^2$. The $C_{\text{max}}$ value of 0.8 allows for a constant white-point output for viewing angles up to 20 degrees for both landscape and portrait orientations. At 25 degrees, there will be a small degree of the white point luminance falloff that is not corrected, and a larger proportion at 30 degrees and above. The proportions of the white point luminances (before correction) produced for these viewing angles are shown in Table 5.

<table>
<thead>
<tr>
<th></th>
<th>20 degrees</th>
<th></th>
<th>25 degrees</th>
<th></th>
<th>30 degrees</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(negative)</td>
<td>(positive)</td>
<td>(negative)</td>
<td>(positive)</td>
<td>(negative)</td>
<td>(positive)</td>
</tr>
<tr>
<td>Portrait</td>
<td>0.88</td>
<td>0.85</td>
<td>0.82</td>
<td>0.78</td>
<td>0.76</td>
<td>0.71</td>
</tr>
<tr>
<td>Landscape</td>
<td>0.90</td>
<td>0.86</td>
<td>0.81</td>
<td>0.76</td>
<td>0.70</td>
<td>0.64</td>
</tr>
</tbody>
</table>

8.2.2.3 Testing the Interactive Correction Model on the Object Display

The performance of the interactive model was evaluated by measuring the output from the object display screen as a function viewing angle, with and without the model applied. The display output was measured at five degree intervals from -30° to +30° with a spectroradiometer mounted to a gonio-arm, for three different orientations of the screen (portrait, landscape, and an
oblique angle 45° between landscape and portrait). The measurement configuration for the display in the oblique orientation is shown in Figure 67. The display was measured at five different target luminance output levels (25, 50, 100, 200, 300 cd/m²) at the chromaticity of D65.

![Image](image_url)

**Figure 67.** Measurement configuration used to test the interactive luminance correction model. The screen, shown in the oblique orientation, was measured with a spectroradiometer mounted to a gonio-arm.

The absolute luminance results are shown in Figure 68, where the blue line represents the measurements with the correction model active, and the red line represents the measurements without the correction model applied. The objective of the model is to reduce the angular falloff so that a constant luminance level is achieved across the angle range. The target constant value is shown as a dotted gray line, which was calculated as the average of the on-axis (0°) measured
luminance for the “Angle Model” and “No Model” cases. Normalized results were calculated by dividing the absolute measured values by the on-axis luminance for each target level. The results for the portrait orientation are shown in Figure 69, for the landscape orientation in Figure 70, and for the oblique orientation in Figure 71.

**Figure 68.** Absolute luminance results for the interactive correction model. The set of constant target values are shown by the gray dotted lines. The luminance falloff with no model applied is represented by the red lines. The measured results with the interactive corrective model are shown in blue.
Figure 69. Normalized luminance results for the interactive model with the display in the portrait orientation, shown for the five different target levels (25, 50, 100, 200, 300 cd/m²). The luminance falloff with no model applied is shown in red. The results with the interactive model are shown in blue.
Figure 70. Normalized luminance results for the interactive model with the display in the landscape orientation. The luminance falloff with no model applied is shown in red. The results with the interactive model are shown in blue.
Figure 71. Normalized luminance results for the interactive model in the oblique orientation. The results with no model applied are shown in red. The results with the interactive model are shown in blue.

To assess the amount of error reduction provided by the corrective model, root mean squared error (RMSE) values were calculated for the angle model and no angle model datasets. For each target level, the RMSE statistic was calculated as:

\[
RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (Y_{\theta(i)} - Y_{\theta=\theta})^2}
\]  

(143)

where \(N\) is the number of angles tested (\(N=13\)), \(Y_{\theta(i)}\) is the measured luminance for the \(i^{th}\) angle, and \(Y_{\theta=\theta}\) is the luminance of the corresponding on-axis measurement. Normalized RMSE results
were calculated by dividing the RMSE results by the $Y_{\theta=0}$ luminance for each case. The absolute RMSE results are shown in the left column of Figure 72 and the normalized results are shown in the right column. In the portrait orientation, the use of the corrective model produced an 89 to 94% reduction in the luminance error for off-axis viewing. In the landscape orientation, the corrective model produced a 70 to 90% reduction, and in the oblique orientation, the model produced a 77 to 83% reduction.
Figure 72. Comparison of the viewing-angle dependent error with the corrective model applied (blue bars) and without any corrective modeling applied (red bars). In the left column, the root mean squared error results are shown in terms of absolute luminance. In the right column, the normalized RMSE results are shown.
8.2.3 Screen Surface BRDF Measurements

The object display is intended for use in an illuminated environment, so there will be some ambient light incident on the screen that can create unwanted physical reflections. The surface reflectance properties of the object display screen were evaluated with goniometric measurements and an imaging-based BRDF measurement system.

8.2.3.1 Goniometric Evaluation

The apparatus used in the gonio-based screen surface measurements consisted of a controlled light source and a spectroradiometer (PhotoResearch PR-655) mounted to a gonio-arm. The light was a diffuse area source constructed by covering the output port of an integrating sphere with a diffusing filter. The output port was circular with a 5.0 cm diameter. The center of the spectroradiometer lens, the center of the light source, and the measurement point on the screen surface were co-planar, and the center of the light source was located approximately 8 degrees from the screen surface normal. The measurement configuration and the reflection of the light source from the screen are shown in Figure 73.
Figure 73. Gonio-based measurement configuration for the screen front surface reflectance. The physical reflection of the 5 cm circular light source can be seen on the surface of the screen.

A white fluorilon (sintered PTFE) patch was measured at the approximate screen location to estimate the incident illuminance of the light source on the screen. The measured luminance from the patch was 7.36 cd/m², which corresponds to an estimated incident illuminance of 23.1 lux under an approximation that the fluorilon behaves similarly to a PRD. The light source was also measured directly with a spectroradiometer at the time of the screen surface measurements. The average luminance of three light source measurements was 4119 cd/m².

The screen surface reflection was measured at 17 angles ranging from 6 degrees to 22.5 degrees away from the surface normal. From 6 degrees to 10 degrees off-axis (near the specular angle of ∼8 degrees), measurements were taken at 0.5 degree intervals. The absolute luminance of the reflections measured over the range is shown in the left panel of Figure 74. The relative reflectance distribution, calculated by dividing the measured luminance by the
estimated illuminance (23.1 lux), is shown in the right panel. This relative distribution is similar to a BRDF, which is also calculated as the ratio of outgoing luminance to incoming illuminance. However, this is not a pure estimate of the BRDF because an area source with a finite size was used (instead of a smaller point-like source) and so the measured curve will be wider than the true BRDF. The peak absolute luminance of the specular reflection measured from the screen was 45.3 cd/m². With the measured source luminance of 4119 cd/m², this peak reflectance was approximately 1.1% of the light source luminance.

For the 22.5 degree measurement (~15 degrees off-specular), the reflected luminance declined to 0.14 cd/m², which was the largest off-specular angle that still produced a result measurable by the spectroradiometer. The ratio of this luminance to the fluorilon (14 / 7.36) would correspond to a diffuse reflectance factor of $\rho_d < 0.02$, but the actual diffuse reflectance factor is likely to be considerably lower because measurements further from the specular peak fell below the spectroradiometer measurement threshold. A separate measurement with an X-Rite Eye-One Pro spectrophotometer produced a diffuse factor result of $\rho_d = 0.002$ (0.2% diffuse reflectance), but this value may be artificially low because the device had to be placed just above the screen surface to avoid damaging the screen coating.
Figure 74. The luminance reflected from the screen as a function of the spectroradiometer detector angle. In the left panel, the absolute luminance is shown. In the right panel, the relative distribution of the luminance divided by the incident illuminance is shown.

8.2.3.2 Image-based Gloss Measurement with a Line source

In addition to the measurements with the spectroradiometer, the front surface gloss of the screen was also estimated by imaging the reflection of a narrow line source with a camera calibrated to absolute luminance. A detailed description of the measurement apparatus is provided in Section 0, where it was used to estimate the BRDF of the physical samples in the gloss reproduction experiment. The captured luminance image, cropped to a narrow window around the reflectance peak, is shown in Figure 75. The BRDF of the surface was estimated by taking the median value of each column and fitting the resulting data vector with a 3-lobe Ward-Dür model using a simulation of the line source and capture geometry. The estimated parameters
for the 3 specular lobes are given in Table 6. The measurement data vector and model results are shown in Figure 76.

Figure 75. Cropped image of the vertical line source reflected from the front surface of the display screen.

Figure 76. The median data vector from the captured image of the screen surface reflecting a line source. The captured data points are represented by blue dots and the BRDF model results are shown by the red solid line.
Table 6. Ward-Dür Model Parameters for a 3-Specular-Lobe Estimate of the Screen Surface BRDF

| ρ₃¹ | 0.00025 |
| ρ₃² | 0.00542 |
| ρ₃³ | 0.03700 |
| α₃  | 0.04435 |

In the line source image and associated data, the screen gloss had both a narrow specular peak, indicating some distinctiveness of image gloss, and a broader specular lobe indicative of screen haze. To model the shape of this measured surface reflection, it was necessary to use a multiple lobe BRDF model to account for both of these features.
9 SYSTEM EVALUATION: PHYSICAL MEASUREMENTS OF OBJECT DISPLAY REPRODUCTIONS

To evaluate the accuracy of the object display system, two measurement experiments were performed. In the first experiment, the color reproduction of the object display was evaluated by reproducing a Classic ColorChecker chart under three different light sources and measuring the set of rendered patches with a spectroradiometer. In the second experiment, the gloss reproduction of the object display system was evaluated by modeling a set of gloss samples and then comparing physical measurements of the virtual samples onscreen to measurements of the real samples.

9.1 Color Reproduction Experiment

A measurement experiment was performed to evaluate the relative colorimetric accuracy of the six-channel rendering workflow and output through the display characterization. This experiment was described in the paper by Darling & Ferwerda [2012]. In the experiment, a simulated 24-patch Classic ColorChecker was rendered under three different light sources and the screen output was measured for each patch. The accuracy of the reproduction was evaluated by comparing the screen measurements to the patch colors calculated spectrally from the light source spectral distribution and the surface reflectance factor of each patch.

9.1.1 Materials

A synthetic model of the 24-patch Classic ColorChecker was constructed in the object display framework with the same physical dimensions as the real ColorChecker chart. The spectral reflectance factor of each patch from a physical chart was measured with an X-Rite Eye-
One Pro spectrophotometer (380 to 730 nm range, 10 nm intervals). The spectral reflectance factor for each was converted to the six-channel representation using the spectral input path for surface reflectance. The six-channel $\rho_d$ values for the 24-patches were used to construct a spatially-varying diffuse reflectance map of the type used to specify surface color properties in the object display framework.

The spectral power distributions of three light sources in a GTI ColorMatcher lighting system (simulated D65, simulated D50, and tungsten) were measured with a PhotoResearch PR-655 spectroradiometer. These spectral measurements were converted to the six-channel representation using the spectral input path for illumination.

9.1.2 Procedure

For each of the lighting conditions, the light emitted by the display for each patch was measured with a PhotoResearch PR-655 spectroradiometer. The display measurement configuration is shown in Figure 77. All patches were measured at the same screen location by shifting the virtual model position on the screen. During the measurements, the ambient light modeling system was operated in the light classifier mode, in which the interactive spectral sensor (Ocean Optics USB2000+) was used to automatically select between the pre-measured light sources (as opposed to continually updating the spectral distribution of the ambient illumination). The use of the classifier mode insured that the same virtual lighting input data was applied during all the patch measurements for a given light source.
Figure 77. A virtual model of the X-Rite Classic ColorChecker is rendered to the screen. The position of the spectroradiometer remains fixed and the patches are shifted to a common position for measurement.

9.1.3 Data Analysis

The colorimetric values for each of the patches were calculated spectrally with the CIE 1931 standard observer to serve as a standard for comparison to the patch data measured from the screen. The XYZ of each was calculated at 10 nm intervals using the measured patch spectral reflectance (from the X-Rite Eye-One Pro) and the measured spectral power distribution of each of the three light sources, which were resampled to 10 nm from the PR-655 measurements. In addition to these full-spectral color calculations, simulated results were calculated based on the six-channel multispectral workflow. The multispectral XYZ results were calculated by first converting the measured spectral light distributions and measured surface
reflectance curves to the six-channel representation. The light and surface data were then multiplied on a per-channel basis and the result was converted to XYZ with a [6 x 3] transform matrix.

The baseline full-spectral calculation, the six-channel multispectral calculation, and the values from real screen measurements were converted to CIELAB and compared using the CIEDE2000 color difference formula. Prior to the CIELAB conversion, the XYZ data from the different workflows were normalized to a relative scale. For each dataset, all the patch data were divided by the Y value of the white ColorChecker patch multiplied by a scaling factor of 1.13:

\[
X_{\text{patch \ i, \ norm}} = \frac{X_{\text{patch \ i}}}{1.13 Y_{\text{white patch}}} \quad Y_{\text{patch \ i, \ norm}} = \frac{Y_{\text{patch \ i}}}{1.13 Y_{\text{white patch}}} \quad Z_{\text{patch \ i, \ norm}} = \frac{Z_{\text{patch \ i}}}{1.13 Y_{\text{white patch}}}
\]  \hspace{1cm} (144)

where \(X_{\text{patch \ i, \ norm}}\), \(Y_{\text{patch \ i, \ norm}}\), and \(Z_{\text{patch \ i, \ norm}}\) represent the normalized XYZ value for the \(i^{th}\) patch (for \(i = 1\) to 24). The 1.13 scaling factor was selected so that a PRD would have a \(Y_{\text{norm}}\) value of approximately 1 in the normalized scale. All the normalized patch XYZ data were transformed to corresponding colors under D65 using the chromatic adaptation transform from CIECAM02. For all three workflows (spectral calculation, multispectral calculation, measured screen), the chromaticity of the actual measured light source for the associated lighting condition was used as the source white point. The chromatic adaptation transformation to D65 was incorporated so that the magnitude of color error could be compared in a common reference condition.

The mean, standard deviation, 90\textsuperscript{th} percentile, and maximum CIEDE2000 color error between the baseline full-spectral calculation and the multispectral calculation for the ColorChecker are shown in the left portion of Table 7. The statistics for the color error between
the full-spectral calculation and the measured color output from the display screen are shown in
the right portion of the table. The CIELAB values for all the individual patches in the different
workflows are provided in Appendix G in the Tables G1 (D65 condition), G2 (D50 condition)
and G3 (tungsten condition).

Table 7. CIEDE2000 Color Error for ColorChecker Reproduction

<table>
<thead>
<tr>
<th></th>
<th>Six Channel Simulation</th>
<th>Screen Output</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>CIEDE2000</td>
<td>CIEDE2000</td>
</tr>
<tr>
<td></td>
<td>mean</td>
<td>std</td>
</tr>
<tr>
<td>D65-Booth Light</td>
<td>0.4</td>
<td>0.3</td>
</tr>
<tr>
<td>D50-Booth Light</td>
<td>0.4</td>
<td>0.3</td>
</tr>
<tr>
<td>IIIA-Booth Light</td>
<td>0.8</td>
<td>0.4</td>
</tr>
</tbody>
</table>

The multispectral workflow provided a generally acceptable level of color accuracy
across lighting conditions. The mean CIEDE2000 for each of the daylight light sources was 0.4.
Though there was increased error in the illuminant A (tungsten) condition, the mean CIEDE2000
value was still 0.8.

The measured screen output had slightly higher error than the simulated workflow for the
D65 and D50 light booth conditions, but the CIEDE2000 values in the tungsten light increased
dramatically for certain patches. The maximum error increased to as high as 14.0, with a 90th
percentile error of 11.0. Several of the highest errors can be attributed to screen gamut
limitations. All of the patches that exceeded a CIEDE2000 value of 6.0 were out of the screen
gamut and so even if correctly calculated by the rendering engine, could not be physically
produced. Five patches (orange, yellow, orange-yellow, yellow-green, red) fell outside the
chromaticity triangle formed by the display primaries (shown in Figure 78). Additionally, the
black patch was out of gamut because the screen’s minimum possible Z value (all channels at 0
digital counts) exceeded the small Z value needed to match a black patch under the tungsten
illumination. With these six out-of-gamut patches removed, the mean error for the tungsten
condition decreased to 1.8, and maximum error decreased to 5.9. The summary results for the 18
remaining patches in the tungsten condition are shown in Table 8 along with the results for all 24
of the patches.

Figure 78. ColorChecker patch u’v’ coordinates for the tungsten light condition. The solid gray lines represent the
dege of the display gamut. The dots represent the target patch colors. The tips of the arrows mark the color
measured from the display screen.
Additionally, with the effect of the non-zero black-level output of the display, there were three additional patches (shown in Figure 79 as patches 1, 4, and 14) that could not be produced by the display at the target chromaticity for the luminance levels in the experiment. These three patches accounted for the three remaining largest color errors (CIEDE2000 values of 5.9, 5.3, 4.1). The black level output of the display was $\text{XYZ} = (0.70, 0.73, 0.98)$ with $\text{Y}$ in cd/m$^2$. This fixed offset was subtracted from each of the target patch XYZ values at their estimated luminance (in cd/m$^2$) to determine whether the screen would be able to produce the remaining XYZ with a combination of the black-level corrected primaries. The $u’v’$ coordinates of the target patch XYZ values after the display black-level subtraction are plotted in Figure 79. In the plot, the first six patches removed from the analysis (7, 11, 12, 15, 16, 24) fall outside the chromaticity triangle of the primaries, as well as the additional patches that were not initially removed (1, 4, 14). In all nine cases, with the effect of the fixed black-level offset, the screen would require a negative scalar on the blue primary to produce the target chromaticity at the necessary luminance. With these nine patches removed, the mean CIEDE2000 of the remaining 15 patches is 1.1, the 90th percentile error is 1.8, and the maximum error is 2.1. For comparison,
these results with the nine patches removed are shown along with the results for all 24 patches in Table 8.

![Color gamut limitations for reproducing the virtual ColorChecker](image)

**Figure 79.** Screen gamut limitations for reproducing the virtual ColorChecker under tungsten illumination considering the effects of the fixed black-level offset. The patch u’v’ coordinates are plotted after subtracting the display black-level XYZ from the calculated absolute target XYZ values of the patches. After black-level subtraction, patches 1, 4, 7, 11, 12, 14, 15, 16 and 24 fall outside the range of chromaticities that can be produced by a combination of the display primaries.
In the version of the object display system used in the color measurement experiment, simple clipping methods were used for out-of-gamut colors. The gamut limitations and large color errors in the tungsten condition demonstrated the need to incorporate additional methods into the display workflow for mapping out-of-gamut colors (described in Section 8.2.1). As found in the additional analysis of the tungsten condition, the non-zero black level of the screen had an impact on whether certain target patch colors in the experiment could be produced by the display. The screen black level is considered in the methods that were incorporated to map out-of-gamut colors.
9.2 Gloss Reproduction Measurement Experiment

The ability of the object display system to physically reproduce directional reflectance properties (gloss) was evaluated in a measurement experiment. In a controlled lighting environment, the luminance reflected from a set of physical samples was measured as a function of angle. The samples and the real lighting were modeled in the object display framework and the virtual samples were reproduced on the display screen in the same physical location as the real samples had occupied. The luminance output from the screen was measured with the same apparatus as was used on the real samples to allow for direct comparison between the two sets of measurements.

9.2.1 Gloss Sample Set

The measurements were performed on a set of 4 x 4 cm painted glass samples with varying levels of surface gloss. The samples had been previously prepared at MCSL for use in assessing the gloss measurement capabilities of various instruments. The gloss levels for the different samples were varied by mixing a high gloss paint and a matte paint in different ratios. The set of nine samples used in the experiment had high gloss paint percentages of 100%, 98%, 95%, 90%, 85%, 80%, 75%, 70%, and 60%. The samples were without significant texture and were generally flat. All the painted samples had a similar diffuse gray color with a reflectance factor of approximately 0.3. As a baseline, a white fluorilon patch (sintered PTFE) was also included in the measurement experiment.
9.2.2 Modeling the Samples

The BRDF parameters for the samples were estimated using two methods, an image-based method for the higher gloss samples and measurements with a Murakami GCMS-10X goniospectrophotometer for the lower gloss samples. The imaging-based method provided high angular precision for a small range of angles around the specular peak and was used to estimate parameters for high gloss samples with narrow specular lobes. The goniospectrophotometer allowed for a wider range of measurement angles, though with less angular precision, and was used to estimate the parameters of low gloss samples with wide specular lobes.

The diffuse reflectance for each of the samples was estimated with an X-Rite Eye-One Pro spectrophotometer, which uses a 45-degree-illumination/0-degree-detection measurement geometry. The spectral measurements from the Eye-One were converted to a set of $\rho_d$ parameters using the spectral response curves for the six-channel primary system. The painted samples, as dielectrics, were treated as spectrally non-selective and the specular lobe was modeled by assigning all six channels of $\rho_s$ parameters the same value.

9.2.2.1 Modeling High Gloss Samples with a Line Source

The high gloss samples were modeled by imaging the reflection of a line source (1.2 mm wide by 35 mm in length) from the front surface of the samples. The measurement apparatus is shown in Figure 80.
Figure 80. The light source and camera configuration of the apparatus for image-based BRDF measurements of high gloss surfaces.

Images were captured in a 9-exposure high-dynamic range series using a Canon EOS 20D digital SLR, which was characterized to measure absolute luminance (the luminance scaling model was established by imaging a set of patches in HDR and then measuring them from the same location with a spectroradiometer). A narrow rectangular window (510 x 60 pixels, 3.2 x 0.4 cm) was extracted from each image. The full angular extent of the window in the horizontal directional was approximately 3.7 degrees, based on a distance of 50 cm from the center of the
sample to the camera. Images of the line source reflected from the surfaces of five gloss samples are shown in Figure 81.

![Figure 81](image)

**Figure 81.** Images of the line source reflected in the surface of the five highest-gloss samples (gloss levels: 100%, 98%, 95%, 90%, 85%).

To fit BRDF parameters using the images, the median value was calculated for each column, resulting in a single 510-element array of luminances for each sample (represented by the dotted lines in Figure 82). The elements in the array were mapped to their physical positions in space, and a rendering simulation was developed that estimated the reflected luminance at each position for a given set of BRDF parameters, based on a geometric model of the line source and the positions of the sample and camera. The total illuminance at the center of the sample surface from the line source was estimated by measuring the luminance reflected from a white
fluorilon sample (sintered PTFE). The illuminance was estimated to be 0.41 lux by multiplying the measured luminance of 0.13 cd/m$^2$ by $\pi$, under the assumption that the fluorilon was approximately Lambertian.

Using the rendering simulation, the BRDF parameters for a Ward-Dür model with two specular lobes ($\rho_s$, $\alpha$) were fitted for each sample using a non-linear optimization routine in Matlab (lsqnonlin) with upper and lower bounds on the parameters. The $\rho_d$ parameter was fixed to the value from the spectrophotometer measurements. An alignment parameter was also optimized, allowing for a small constant shift in the physical (horizontal) position assigned to the data elements. This was to insure that a small misalignment in the measurement apparatus did not prevent the location of the specular peak from aligning with the nominal specular angle.

The objective function for the optimization routine was the minimization of the sum squared error between the median measurement data and the reflected luminance estimate from the rendering simulation. To aid in matching the area under the curves, the objective function also included an additional term, with a small weighting ($w_k = 0.1$), for the difference in the overall mean of the two curves:

$$\min \sum_{i=1}^{N} \left( |L_{i,\text{measured}} - L_{i,\text{modeled}}| + w_k \frac{\sum_{j=1}^{N} L_{j,\text{measured}}}{N} - \frac{\sum_{j=1}^{N} L_{j,\text{modeled}}}{N} \right)^2$$  \hspace{1cm} (145)
where \( L_{i,\text{measured}} \) is the luminance at the \( i^{th} \) data location in the median image vector and \( L_{i,\text{modeled}} \) is the BRDF model estimate at that location. The absolute values are necessary to insure that the difference in the overall mean and the per-element error always add together and do not subtract in cases where their signs differ.

The BRDF parameters for the five samples estimated with the line source fitting are given in Table 9. The BRDF model results are plotted along with the median image data in Figure 82. Overall, the fitted BRDF models are able to provide a close approximation to much of the captured data, but even with two specular lobes, are not able to recreate the absolute peak values of the two high gloss samples (GV100, GV98) with the sharpest specular peaks.

<table>
<thead>
<tr>
<th>Fitting Type</th>
<th>Glossy Paint Percentage</th>
<th>RhoD</th>
<th>RhoS</th>
<th>Alpha</th>
<th>RhoS2</th>
<th>Alpha2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Line Source</td>
<td>100</td>
<td>0.3122</td>
<td>0.0060</td>
<td>0.0016</td>
<td>0.0185</td>
<td>0.0109</td>
</tr>
<tr>
<td>Line Source</td>
<td>98</td>
<td>0.3156</td>
<td>0.0041</td>
<td>0.0038</td>
<td>0.0179</td>
<td>0.0171</td>
</tr>
<tr>
<td>Line Source</td>
<td>95</td>
<td>0.3175</td>
<td>0.0021</td>
<td>0.0075</td>
<td>0.0164</td>
<td>0.0259</td>
</tr>
<tr>
<td>Line Source</td>
<td>90</td>
<td>0.3205</td>
<td>0.0012</td>
<td>0.0111</td>
<td>0.0155</td>
<td>0.0357</td>
</tr>
<tr>
<td>Line Source</td>
<td>85</td>
<td>0.3219</td>
<td>0.0034</td>
<td>0.0239</td>
<td>0.0562</td>
<td>0.1195</td>
</tr>
<tr>
<td>Murakami</td>
<td>80</td>
<td>0.3126</td>
<td>0.0183</td>
<td>0.1190</td>
<td>0.0064</td>
<td>0.0502</td>
</tr>
<tr>
<td>Murakami</td>
<td>75</td>
<td>0.3153</td>
<td>0.0202</td>
<td>0.1153</td>
<td>0.0035</td>
<td>0.0459</td>
</tr>
<tr>
<td>Murakami</td>
<td>70</td>
<td>0.3189</td>
<td>0.0188</td>
<td>0.1459</td>
<td>0.0028</td>
<td>0.0663</td>
</tr>
<tr>
<td>Murakami</td>
<td>60</td>
<td>0.3229</td>
<td>0.0160</td>
<td>0.1713</td>
<td>0.0018</td>
<td>0.0902</td>
</tr>
</tbody>
</table>
Figure 82. BRDF model results for the samples estimated by imaging a line source. The measured data points are represented by blue dots and the model predictions by solid red lines.

9.2.2.2 Modeling Low Gloss Samples with a Goniospectrophotometer

The BRDF parameters for the more diffuse samples (GL80 to GL60) were estimated from measurements made with a Murakami GCMS-10X goniospectrophotometer. The samples were measured at a range of detector angles for three incident lighting angles: 10°, 15° and 30°. For the 10° incident lighting case, there were a total of 27 detector angles ranging from 7° to 45° (away from the surface normal direction). There were 33 detector angles ranging from 5° to 45°
for the 15° illumination case and 33 detector angles from 0° to 45° in the 30° case. The interval between detector angles was set to 0.5° near the specular peak and increasingly wider intervals were used further from the specular angle.

The parameters for a Ward-Dür model with two specular lobes were fitted using a constrained non-linear optimization routine (fmincon) in Matlab. The fitting was performed in two stages. In a simpler first stage, BRDF parameters were fitted for the data assuming a single detection point, without considering the effect of the detector averaging over an extended surface region (the detected region is 1.6 cm wide on-axis and 9.4 cm wide when rotated to 80° detection, according to the goniospectrophotometer documentation). The BRDF parameters from the first stage served as the starting values for a more complex fitting that calculated values at multiple surface positions and averaged the results to estimate the mean reflection for the extended region. The extended region was approximated by projecting a 1.5° (radius) circular detection area onto a geometric model of the rotated sample holder plane. In both stages, the objective function was to minimize the mean square error between the measured BRDF values and the model results:

$$\min \left( \frac{1}{N} \sum_{i=1}^{N} (f_{i,\text{meas}} - f_{i,\text{model}})^2 \right).$$

(146)

The resulting BRDF parameters for the four lowest gloss samples (80%, 75%, 70%, 60% gloss percentages) are provided in Table 9. The model results (solid lines) are shown along with the measured data from the goniospectrophotometer (dots) in Figure 83.
9.2.3 Modeling the Lighting and Measurement Configuration

A controlled lighting environment and goniometric measurement apparatus were constructed for use in evaluating the capabilities of the object display to reproduce the BRDF of real physical samples. The measurement apparatus was designed to allow spectroradiometer
measurements to be taken as a function of angle from either the real physical samples or the object display screen.

The lighting was designed with the objective of creating a physical light source that could be precisely modeled in the object display framework. The required properties of the controlled light source were that it be a relatively small, uniform, diffuse area source of a known size, shape, location, and intensity. The source (shown in Figure 84) was constructed on an optical bench using a Mille Luce M1000 lamp, a fiber optic cable, an integrating sphere, and two filters. One end of the fiber optic cable was attached to the lamp and the other end placed into the entry port of the integrating sphere. The integrating sphere was used in an effort to create a spatially uniform light source. The circular exit port of the sphere was covered with a 35 mm slide holder to create a rectangular aperture of 2.3 cm x 3.5 cm. The rectangular aperture was covered with a blue filter to raise the color correlated temperature of the tungsten illumination (4700K with the filter in place) and also an additional diffusing filter to reduce any angular dependency in the light exiting from the sphere.
A total of 12 measurements of the light aperture were taken with a PhotoResearch PR655 spectroradiometer at the start and end of the experiment. These values were averaged (Mean = 4197, SD= 17.1) and used as the uniform luminance value over the aperture. The light was modeled in the object display framework as a diffuse, uniform area source image with a luminance of 4197 cd/m² and size of 2.3 cm x 3.5 cm. In the lighting model, this luminance map image was mapped to a polygon at its physical location in the measurement environment.

Along with the controlled light source, the measurement apparatus consisted of a PhotoResearch PR655 spectroradiometer mounted to a goni-arm and a platform with a sample holder. The vertical center line of the sample holder was aligned with the center of rotation of the goni arm. The center of the spectroradiometer, light source aperture, and sample holder were aligned to have the same physical height above the optical bench surface. A side view of

Figure 84. Controlled light source constructed from a lamp, fiber optic cable, integrating sphere, and rectangular aperture covered with a diffusing filter.
the sample, detector, and light source configuration is shown in Figure 85. To create a geometric model in the object display framework, the positions of the sample holder, light source and detector were all defined in a physical 3D coordinate system with the +X axis toward the right of the sample (when facing it), the +Y axis upward, and the +Z-axis in the direction of the sample normal. A diagram illustrating their physical positions is shown in Figure 86 and their XYZ coordinates are provided in Table 10. As shown in the diagram, the light source center was at an angle of approximately 7.5° with respect to the sample normal. Additionally, the plane of the light area source was rotated by a similar amount (~8°) so that light source normal was directed at the sample center. The position of the detector varied as a function of the gonio-arm angle $\theta$ and moved relative to a center of rotation at the sample center. This varying detector position was used to specify the observer eye-point in the object display framework.

Table 10. Physical Positions of the Sample, Light and Detector

<table>
<thead>
<tr>
<th>Position</th>
<th>X (cm)</th>
<th>Y (cm)</th>
<th>Z (cm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sample Center</td>
<td>15.25</td>
<td>41.6</td>
<td>-15.25</td>
</tr>
<tr>
<td>Light Center</td>
<td>7.2</td>
<td>41.6</td>
<td>45.5</td>
</tr>
<tr>
<td>Detector Position</td>
<td>$42 \sin(\theta) + 15.25$</td>
<td>41.6</td>
<td>$42 \cos(\theta) - 15.25$</td>
</tr>
</tbody>
</table>
Figure 85. Side view of the measurement configuration for the real physical samples in the gloss measurement experiment. The glass sample (at the left of the image) is illuminated by a rectangular area source and measured with a spectroradiometer mounted on a gonio-arm.

Figure 86. Top view of the measurement geometry illustrating the physical positions of the light, detector, and sample used in the 3D model. The detector, which changes position as a function of the gonio-arm angle $\theta$, is shown at the specular angle $\theta = 7.5^\circ$. 
9.2.4 Measurement of Physical Samples

The nine physical gloss samples and the white fluorilon patch were measured at a series of 19 angles: 6, 6.5, 7, 7.5, 8, 8.5, 9, 9.5, 10, 11, 12, 13, 14, 15, 17.5, 20, 22.5, 25, and 30 degrees. Measurements were performed at 0.5° intervals near the specular angle (7.5°) with increasingly larger intervals further from the specular lobe. The gonio-arm was manually adjusted to each of the angles and the luminance of the reflection was measured with the spectroradiometer. With the gonio-arm center of rotation aligned with the sample center, the spectroradiometer remained directed at the center of the sample as the detector angle was varied. The measurement set-up for one of the high gloss samples is shown in Figure 87, where the reflection of the light source is clearly visible on the sample surface.
Figure 87. Front view of the measurement configuration for the physical samples. The reflection of the rectangular area source can be seen in the sample. (Note that the room lights are turned on to illuminate the measurement apparatus, so the diffuse reflection from the sample is higher than it would be for the small area source alone.)

9.2.5 Measurement of Virtual Samples

The virtual samples were measured in the same geometric configuration as the physical samples. A side view of the configuration of the light source, detector, and screen (similar to Figure 85 for the real samples) is shown in Figure 88. A front view, similar to Figure 87, is shown in Figure 89 with the virtual reflection clearly visible. Both the overhead lights in the room and the small area source were turned off during the actual measurements.
In the object display condition, the physical sample holder was replaced by the screen and the luminance for each of the virtual samples was measured with the spectroradiometer at the same 19 detector angles used with the real samples. In the lighting simulation, the hybrid approach was used with 64 median cut points and 64 rays for filtered importance sampling. The measurement process was partially automated. For each measurement angle, a computer controlled script iterated through all the virtual samples and initiated measurements with the spectroradiometer. When the set of measurements for one angle was complete, an instruction was issued to manually advance the gonio-arm to the next angle. The script used the detector angle specified in the prompt to update the observer eye-point in the object display framework to the appropriate location for the detector.

Figure 88. The virtual sample on the screen is located at the same physical location as the real sample and measured using the same measurement configuration. The virtual sample is rendered with a model of the light source shown here, but the real light source was turned off during the measurements.
9.2.6 Results and Analysis

The absolute luminance values from the spectroradiometer measurements, as a function of the detector angle, are shown in Figure 90 for the real physical samples (red dotted lines) and
for the object display screen reproductions (blue dotted lines). Though there are some deviations between the physical sample measurements and the screen measurements, overall the two sets of results are similar in their scale (in absolute units of physical luminance) and match in their general shapes.

In general, the results for virtual samples 100GL-85GL, which were modeled with the line source, had peak values slightly higher than the measured values for the real physical samples. Samples 80GL-60GL, which were modeled from the Murakami data, had peak values that were generally lower than the physical samples, with the most pronounced difference in the highest gloss sample (GL80) that was measured with the Murakami. The screen output for the two mid-range gloss samples, GL85 and GL80, exhibited the clearest differences from the physical measurements in the data plots. These two were the most difficult for measuring and fitting the input BRDF models, because the widths of their specular lobes fell between the measurement capabilities of the line source and goniospectrophotometer methods. The narrow angular range of the line source method did not provide sufficient data to fit the tail of the GL85 sample, while the goniospectrophotometer tended to significantly underestimate the peak height of any high gloss samples due to the angular averaging from its relatively large measurement area (this was the primary reason that all samples with higher gloss levels than GL80 required fitting with the other method).
Figure 90. Goniometric measurements of the real physical samples (red line) and the screen output for the virtual samples (blue line) measured with a spectroradiometer mounted to a gonio-arm. Results are shown for the nine painted gloss samples (levels of 100% to 60% glossy paint) and a virtual perfect reflecting diffuser (PRD) compared to the real fluorilon sample.
To assess and compare the degree of overall error for each sample, a set of summary statistics were calculated from the angular luminance measurements. In the measurement data, the angle intervals between measurements varied over the angle range (with super-sampling near the specular lobe), so the measurement data were resampled by linear interpolation to 0.1° before calculating the error statistics. This was necessary to compare the real data to the screen data over the entire angle range and not just a heavily-weighted region around the specular peak.

The summary statistics included the absolute root mean square error over the angles of a sample (using the RMSE formula in Eq. 143) and also two normalized RMSE metrics. The first normalized statistic was calculated by dividing the RMSE by the mean luminance (of the combined physical and screen samples) for the set of angles. The second normalized statistic was calculated by dividing the RMSE by the luminance data range (over the physical and screen samples) for the \( N = 241 \) resampled angles:

\[
NRMSE_{\text{range}} = \frac{\text{RMSE}}{\max(Y_{\theta(1)}, \ldots, Y_{\theta(N)})_{\text{screen,real}} - \min(Y_{\theta(1)}, \ldots, Y_{\theta(N)})_{\text{screen,real}}}. \tag{147}
\]

Additional metrics that were calculated were the Pearson’s \( r \) correlation (using the formula from [Devore 1995]):

\[
r = \frac{\sum_{i=1}^{N} (Y_{\text{real, } i(\theta)} - \bar{Y}_{\text{real}})(Y_{\text{screen, } i(\theta)} - \bar{Y}_{\text{screen}})}{\sqrt{\sum_{i=1}^{N} (Y_{\text{real, } i(\theta)} - \bar{Y}_{\text{real}})^2} \sqrt{\sum_{i=1}^{N} (Y_{\text{screen, } i(\theta)} - \bar{Y}_{\text{screen}})^2}} \tag{148}
\]

and a coefficient of determination metric [Devore 1995]:
\[ R^2 = 1 - \frac{\text{SSE}}{\text{SST}} = 1 - \frac{\sum_{i=1}^{N} \left( Y_{\text{real},i(\theta)} - Y_{\text{screen},i(\theta)} \right)^2}{\sum_{i=1}^{N} \left( Y_{\text{real},i(\theta)} - \bar{Y}_{\text{real}} \right)^2}. \] (149)

In this case, the SSE term is the error sum of squares between the real and screen measurements and the SST term is the total sum of squares, which describes the baseline variation in the real physical measurements relative to their mean over the set of angles. The plots of these metrics for the nine gloss samples are shown in Figure 91.

The absolute error, as shown in the RMSE plot, was highest for the sample with the highest gloss (4.3 cd/m² for sample GL100) and decreased with decreasing gloss level. However, the highest gloss sample also had the largest magnitude luminance values (physical sample peak: 76.4 cd/m²) and overall the pattern of RMSE results followed the absolute luminance magnitudes for the samples. In this case, the normalized metrics may provide a better indicator of the accuracy of the sample reproduction relative to each other. The mean normalization only partially addressed the magnitude dependence, as the general pattern was still present, with the exception of the GL85 sample. The GL85 sample was the intermediate gloss sample that was difficult to model with either of the measurement systems. It was the only sample that had a higher NRMSE (mean normalized) than a glossier sample (GL90). With the range normalization for NRMSE, the pattern of higher error for samples with larger absolute magnitude values was no longer present, though it may have produced an overcorrection. The lowest gloss sample (GL60) had the highest NRMSE (range normalized), likely due to the small range of luminance values in its measurements. With the exception of GL60, the GL85 sample had the highest range-normalized RMSE (9.2%). The GL95 sample had the lowest range-normalized error with a NRMSE of 4.4%.
With the Pearson’s $r$ correlation statistic, the mid-to-low gloss samples GL80, GL75, GL70 had the best results, with values greater than 0.998. The GL98 sample had the lowest $r$ value (0.983). The $r$ correlation statistic, however, may overestimate how well the screen data matches the physical samples, because it calculated on data that has the mean value subtracted. It describes the similarity in shape between the screen and physical data, but does not take into account a global shift of one relative to the other. The coefficient of determination $R^2$ does take into account these types of global shifts. The GL95 sample had the highest $R^2$ value (0.975), followed by the GL75 sample (0.967). The GL60 had the lowest $R^2$ value (0.857), but this effect may be driven by its small SST in the denominator, because there was very little variation due to its diffuseness. The GL85 sample had the next lowest $R^2$ value (0.905).

In addition to the spectroradiometer measurements, the physical samples and the screen samples were imaged in high-dynamic-range with a camera characterized to an absolute luminance scale. In both cases, the camera was mounted to the same gonio-arm used for the spectroradiometer measurements and the gonio-arm was set to the specular angle (7.5°). The images of the five highest gloss samples (GL100, GL98, GL95, GL90, GL85) are shown in Figure 92. The photographed images of the real physical samples are shown in the left column and the photographed images of the screen are shown in the right column.
Figure 91. Root mean square error (RMSE), correlation, and $R^2$ statistics for the gloss sample measurement data.
Figure 92. Photographic images of the 5 glossiest samples (GL100 to GL85), showing the specular reflection pattern from a rectangular light source. Images of the real physical samples (left column) and the screen reproductions (right column) were captured from a luminance-calibrated DSLR camera, mounted on the gonio arm at the specular angle.
9.2.7 Discussion

The physical measurement results, along with the simulation results from Section 7.4.4, provide evidence that the object display system is capable of reproducing the directional patterns of surface BRDF for a captured light source, as long as the properties of the material can be described by a parametric BRDF model. The painted glass samples could be modeled reasonably well by a Ward-Dür model, though multiple specular lobes were required to achieve a close match to the shape of the measured BRDF curves. Other materials may require more complex BRDF models (or other types of models) to obtain similar levels of accuracy. Though only the Ward-Dür model has so far been used in the object display, there is no fundamental reason why other models could not be incorporated as graphics-processing hardware continues to improve.

By comparing the screen output measurements directly to measurements of the real physical samples in the gloss experiment, the results depend upon the entire end-to-end process of gloss capture, geometric modeling, rendering, and output through the display screen. In some cases, such as for sample GL85, the error in the initial modeling of the physical sample may have been a primary factor contributing to the final error in the measurements from the screen. The line-source imaging technique performed well for high gloss samples and the goniospectrophotometer approach performed well for low-gloss samples, but it was difficult to accurately measure samples that fell between the measurement capabilities of these two devices. A Fourier lens-based measurement system (e.g. the Eldim EZContrast), which captures a high-resolution image of the hemisphere of detector angles in a single measurement, may be a better option. In future studies, it would be useful to incorporate measurement devices like these that could simplify the process of creating accurate sample BRDF models for input. This could help
to minimize the input model as a source of error and allow the final results of experiments comparing the display screen and real samples to depend primarily on the output performance of the object display system.
10 COLOR APPEARANCE WITH AN OBJECT DISPLAY SYSTEM

10.1 Overview

The capabilities of the object display system make it a potentially useful platform for studying aspects of color appearance related to viewing self-luminous display screens vs. viewing real-world reflective objects. Because it has the ability to simulate the behavior of both types of media, and it can be varied under computer control, it provides the opportunity to test sets of experimental conditions that may otherwise be difficult to evaluate. In the chromatic adaptation experiment described in this chapter, it allowed variables like surface properties and real-world lighting consistency to be systematically varied, to test which factors may contribute to the differences in chromatic adaptation between self-luminous displays and reflecting objects. The object display platform provides the means to test combinations that are physically implausible, such as an object situated in an environment with daylight-balanced illumination, but with the surface reflecting tungsten-balanced light.

From an applied perspective, the color appearance properties of object display systems could have potential benefits for cross media color proofing and material design applications. If a reproduction on an electronic display can sufficiently imitate the behavior of a real-world surface that it creates an adaptation state consistent with the one for an illuminated reflective object, it could reduce the amount of color appearance modeling required to recreate the perceived color of an object on a display screen. One of the purposes for conducting the chromatic adaptation experiment was to determine to what extent this may be true, and whether recreating the behavior of a reflective object, while still using a self-luminous screen, will
produce a more complete adaptation state. If it does, it provides evidence that color proofing on an object display system can be simplified to absolute colorimetric matching if the system is viewed for the same ambient lighting as the reflective object. If the object display representation does not aid in adaptation, then it raises interesting questions about what is missing and if there is something inherent about self-luminous screens that limits the degree of adaptation to them.

10.2 Chromatic Adaptation Experiment

An experiment was performed to evaluate the chromatic adaptation state of observers when viewing the object display screen. In previous research, Gorzynski and Berns [1990] and Fairchild [1992] found that observers did not fully adapt to self-luminous displays when the color balance was moved away from the chromaticity of D65. In contrast, observers were able to discount the illuminant and fully adapt to reflective prints, even under tungsten lighting. This difference has been attributed to cognitive discounting of the illuminant, which occurs for reflective surfaces but not for self-luminous ones. The primary goal of the experiment was to assess whether observers are able to adapt to the object display representation in a manner more similar to a reflective object. In case observers did more fully adapt, the experiment was designed to evaluate which factor, realistic surface properties or lighting consistency, contributed to the adaptation increase relative to a typical display representation.

10.2.1 Experiment Design

The design of the experiment is based on experiments 1 and 2 performed by Gorzynski and Berns [1990] (described in Section 2.5.2.1) used to evaluate adaptation to a display screen
and to a physical illuminated print. In the current experiment, the object display system has been substituted in place of the illuminated print condition to assess whether, like a reflection print, it also produces more complete adaptation than a traditional screen display.

There were two main experimental manipulations in the study design. One factor that was varied was the ambient light surrounding the display, which had three conditions: a tungsten source, a D65-simulating source, or none (light off). These are generally the same ambient lighting conditions (tungsten, daylight, none) that were used in the display experiment by Gorzynski and Berns. However, the sources are not exactly the same chromaticities as in the previous experiment, because they are based on the daylight and tungsten illumination from the specific light booth used. The second factor in the current experiment was the type of texture and shading used to convey the surface properties of the patches. In one case, the patches and background were flat (without texture or shadowing) and were uniformly shaded as they would traditionally be for presentation on a display screen. In the second case, the surface was rendered with realistic texture and shadowing that corresponded to the geometry of the lighting in the booth environment. The overall design of the experiment was a two-factor (2 x 3 level) full-factorial design with all within-subject factors.

The trials were separated into blocks by ambient light condition. The six possible orders of the ambient light conditions (Off-Tungsten-Daylight, Daylight-Off-Tungsten, etc.) were counterbalanced so that an equal number of participants viewed each order of conditions. The order of the surface texture condition within a block was also counterbalanced, with half of the participants shown the flat-uniform shaded stimuli first within the block and half the participants shown the textured condition first. The set of orders for the ambient lighting conditions and the surface type are shown in Table 11.
<table>
<thead>
<tr>
<th>Sequence</th>
<th>Overhead Light Order</th>
<th>Surface Order</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Tungsten Daylight Off</td>
<td>Flat Tex</td>
</tr>
<tr>
<td>2</td>
<td>Daylight Tungsten Off</td>
<td>Tex Flat</td>
</tr>
<tr>
<td>3</td>
<td>Off Tungsten Daylight</td>
<td>Flat Tex</td>
</tr>
<tr>
<td>4</td>
<td>Tungsten Off Daylight</td>
<td>Tex Flat</td>
</tr>
<tr>
<td>5</td>
<td>Daylight Off Tungsten</td>
<td>Flat Tex</td>
</tr>
<tr>
<td>6</td>
<td>Off Daylight Tungsten</td>
<td>Tex Flat</td>
</tr>
<tr>
<td>7</td>
<td>Tungsten Daylight Off</td>
<td>Tex Flat</td>
</tr>
<tr>
<td>8</td>
<td>Daylight Tungsten Off</td>
<td>Flat Tex</td>
</tr>
<tr>
<td>9</td>
<td>Off Tungsten Daylight</td>
<td>Tex Flat</td>
</tr>
<tr>
<td>10</td>
<td>Tungsten Off Daylight</td>
<td>Flat Tex</td>
</tr>
<tr>
<td>11</td>
<td>Daylight Off Tungsten</td>
<td>Tex Flat</td>
</tr>
<tr>
<td>12</td>
<td>Off Daylight Tungsten</td>
<td>Flat Tex</td>
</tr>
</tbody>
</table>

### 10.2.2 Experimental Setup and Stimuli

The stimuli in the experiment were arrays of colored patches presented on a white background (as used by Gorzynski & Berns [1990]). The patches were arranged in a 5 x 5 grid as in the Gorzynski and Berns experiment, but in the current experiment four locations of the grid were left open to increase the area of the white background displayed (as shown in Figure 93). The patches were 2 cm squares and arranged over a 22 cm by 22 cm square region. The luminance factor of the background was set to $Y = 100\%$ and luminance factor of the patches was set to $Y = 35\%$. The background luminance for the flat/uniform trials was approximately 70 cd/m$^2$. The background luminance in the textured/rendered trials varied over the surface, due to the spatially varying difference in the distance to the light source and the varying orientation from the mesoscale geometry. The luminance was approximately 80 cd/m$^2$ for the top row of the surface and 55 cd/m$^2$ for the bottom row of the surface.
10.2.2.1 Color Sampling for the Patches

The chromaticities of the 21 patches displayed were arranged at three levels around a central u’v’ coordinate. There was 1 central patch, 8 patches at level one, 8 patches at level two, and 4 patches at level three. In the first level, the 8 patches were arranged in an ellipse around the central coordinates in the u’v’ space. The major axis of the ellipse was specified by the line between the tungsten and daylight chromaticities. The minor axis of the ellipse was perpendicular and had an extent equal to half the length of the major axis. The second level also had 8 patches in a similar pattern to level one, but with axes two times the extent of the corresponding axes of level one. The third level had axes three times the extent of the corresponding axes of level one, but included only 4 patches (forming a diamond) corresponding
to the end points of the major and minor axes of an ellipse. At the start of each new trial, the central u’v’ coordinate was set to the halfway point between the daylight and tungsten chromaticities (0.2336, 0.5034). The initial sampling pattern for a trial is shown in Figure 94.

For each trial, there was a series of six total selection stages. After a patch was selected for the first stage, a new set of patches was generated and participants made their next selection. For each stage after the first, the set of patches was re-centered on the chromaticity of the most recent selection. The range covered by the set of patches was successively narrowed by reducing the length of the ellipse axes. The radii of the major and minor axes of the sampling ellipses at each stage of the trial are listed in Table 12. The patch selected from the final narrowed set was recorded as the participant’s response for a trial.

![Figure 94](image-url)  
*Figure 94.* The set of patch chromaticities presented in the first selection stage of each trial. The sampling pattern is centered between the tungsten light chromaticity (orange triangle) and the daylight chromaticity (blue triangle).
It was necessary to modify the way that the object display rendering pipeline processes color to accommodate u’v’ input data specified in this manner. In the typical object display workflow, surface reflectance and illumination are specified independently and the final color is calculated with the multispectral pipeline. For the experiment, the target color for display is specified instead, and so it is necessary to work backwards to set the proper surface and illumination values to produce this u’v’ output. The u’v’ target chromaticity is first converted to a relative XYZ (normalized to Y = 1). The six channel workflow was modified to use only half its channels to work directly in XYZ. The illumination chromaticity (tungsten or daylight) was specified in terms of a relative XYZ and stored in the first three channels of the intermediate illumination variable \( S_{6ch, opt(1 \text{ cd/m}^2)} \) described in Section 6.5.4.1. Given the illumination XYZ, the three channel diffuse reflectance needed to produce the target output was calculated and stored in the first three channels of the six channel \( \rho_d \) representation:

\[
\rho_{d,1} = k \frac{X_{\text{out(relative)}}}{S_{1(\text{relative})}}, \quad \rho_{d,2} = k \frac{Y_{\text{out(relative)}}}{S_{2(\text{relative})}}, \quad \rho_{d,3} = k \frac{Z_{\text{out(relative)}}}{S_{3(\text{relative})}}
\]  

where \( XYZ_{\text{out(relative)}} \) is the XYZ value for target u’v’ normalized to Y = 1 and \( k \) is a reflectance constant set to 0.35 for patches and 1 for the background. The final change to form the XYZ

---

<table>
<thead>
<tr>
<th>Selection Stage</th>
<th>First Ring (in u’v’ units)</th>
<th>Second Ring (in u’v’ units)</th>
<th>Third Ring (in u’v’ units)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Major Radius</td>
<td>Minor Radius</td>
<td>Major Radius</td>
</tr>
<tr>
<td>1</td>
<td>0.0100</td>
<td>0.0050</td>
<td>0.0200</td>
</tr>
<tr>
<td>2</td>
<td>0.0080</td>
<td>0.0040</td>
<td>0.0160</td>
</tr>
<tr>
<td>3</td>
<td>0.0060</td>
<td>0.0030</td>
<td>0.0120</td>
</tr>
<tr>
<td>4</td>
<td>0.0040</td>
<td>0.0020</td>
<td>0.0080</td>
</tr>
<tr>
<td>5</td>
<td>0.0020</td>
<td>0.0010</td>
<td>0.0040</td>
</tr>
<tr>
<td>6</td>
<td>0.0010</td>
<td>0.0005</td>
<td>0.0020</td>
</tr>
</tbody>
</table>
process was to remove the matrix transformation ($\mathbf{M}_{\text{Ch6,XYZ}}$) from the six-channel primaries to XYZ described by Eq. (70) and use the first three channels directly for the XYZ output.

10.2.2.2 Sample Surface Properties

The virtual patches were displayed for two different types of surface conditions, a flat uniform surface condition and a more complex textured/shaded condition. In the uniformly-shaded flat surface condition, the stimuli were still generated using the object display rendering engine, but features were disabled or modified to produce the simpler more traditional display output. The normal map was set to [0 0 1] for all surface locations, the height of the surface was set to 0 at all locations, and shadowing was disabled. The surface was rendered at a uniform luminance level by using the illuminance at the center of the surface in the rendering calculations for every location on the surface.

In the textured/shaded surface condition, the stimuli were rendered using surface height maps, normal maps, and horizon shadow maps with the typical per-pixel shading of the object display rendering engine. The surface model was derived from the surface orientation data for a real painting that had been captured using a linear light reflectometry system ([Chen & Berns 2012], normal map data obtained from www.art-si.org). The normal map of the painting was converted to a height map using the Harker & O’Leary [2011] surface reconstruction method with Tikhonov regularization. With the data in a height field form, virtual patches, which were intended to resemble slightly-raised pieces of paper, were added to the height of the painting. Additionally, a 0.5 cm wide border with a 0.75 cm height and a 0.5 cm wide angled bevel were added at the edges. The narrow border was covered by the physical mask, but was included so that the real mask would appear to cast a shadow on the surface. A final normal map was
generated from the combined painting-patch height map by calculating the cross product of the height map gradient at each position. The height map was also used to construct a set of horizon maps for self-shadowing (using the method described in Appendix B). A screen capture of the surface rendered with the tungsten illumination is shown in Figure 95.

![Figure 95](image)

**Figure 95.** Screen capture of the textured background and virtual patches (shown with uniform color). The virtual patches are arranged in a 5 x 5 grid with four patches removed.

In both conditions, the patches and backgrounds were modeled as diffuse materials and rendered without specular gloss highlights ($\rho_s = 0$). In assessing chromatic adaptation, gloss could be a potential confound, because the highlights rendered on the patches would typically be
a different chromaticity than the base (diffuse) color of the patches and the amount of specularly-reflected light would be spatially varying over the surface, impacting each patch by a different amount.

10.2.2.3 Illumination

A light booth with sources simulating D65 (daylight) and illuminant A (tungsten) was used to provide ambient illumination. The lighting environment was characterized and modeled in the object display framework. A model of the spatial luminance for each light was created by taking an HDR image sequence of each light source with a calibrated digital SLR camera. The captured spatial luminance image was mapped to the plane of the light booth ceiling in a 3D model. The chromaticity of each light source was measured using a Minolta CL200 illuminance colorimeter. The daylight source had u’v’ coordinates of (.1991, .4752) and the tungsten source had u’v’ coordinates of (.2681, .5316).

10.2.2.4 Physical Configuration

The screen was mounted in a fixed position, at angle of 12° from the horizontal, throughout the duration of the experiment. Study participants were seated approximately 140 cm from the screen with an eye-level height approximately 30 cm above the center of the object displayed on the screen, but were not constrained in their head movements to allow for natural viewing. The visible area of the screen was 23 cm by 23 cm. A physical white border 4 cm on each edge was placed in front of the screen masking out the 23 cm square region. The white border was constructed from a white baryta-based paper (Ilford Galerie Gold Fibre Silk). In the
object-display rendering condition, a 0.5 cm virtual bevel was rendered to connect between the real border and the 22 cm x 22 cm virtual paper rendered to the screen. The back wall and floor of the viewing booth were covered with Color-aid® matte white paper. The side walls and a front portion of the floor were covered with black fabric. The appearance of the full viewing booth environment for the two types of ambient lighting is shown in Figure 96. The textured background, simulated for tungsten lighting, is shown on the screen in each case. In Figure 97, there are photographs of the masked screen displaying stimuli for the six different experiment conditions (textured vs. flat surfaces, tungsten vs. daylight vs. no ambient).
Figure 96. Appearance of the viewing booth with the daylight illumination (above) and tungsten illumination (below).
Figure 97. Photographic images of the patch grids for the three lighting conditions (Daylight, Off, Tungsten) and the two surface conditions (Flat, Textured). The color balance for all the images was set to a CCT of 5000K. The color distribution for the patches is representative of the first selection stage of a trial. The outer black border and inner white border are real. The inner square regions are displayed by the screen.
10.2.2.5 Display Screen and Surface Reflectance Characterization

An enhanced display characterization model was incorporated into the object display system to provide increased color accuracy for the chromatic adaptation experiment. The object display typically uses the Day et al. [2004] model, which provides a high enough level of color accuracy for general usage of the system (CIEDE2000 mean = 0.5, max = 1.3). However, the display was found to exhibit some deviation from the assumptions of the Day et al. approach that led to relatively small, but systematic error that made it difficult to maintain a constant chromaticity over a luminance ramp (in particular at the chromaticity of the tungsten background). This was addressed with an enhanced characterization method that incorporated modeling of the cross-channel interaction and chromaticity changes over the primary ramps. The inverse model also included methods to reduce the effects of quantization error and an adjustment method to correct day-to-day temporal drift. Summary results comparing the CIEDE2000 accuracy of the enhanced model to the standard Day et al. approach are given in Table 13. The table includes the accuracy results for the forward model results on the data used in the characterization as well as 100 random colors not used in the characterization. The experiment was run over the course of two weeks and re-characterized for the second week of testing. The results for both characterizations are shown. A detailed description of the model and additional results comparing its accuracy to the Day et al. approach are provided in Appendix A.
In addition to characterizing the output of the display, it was also necessary to estimate and correct for the ambient illumination reflecting from the front surface of the screen. In the typical object display framework, this is accomplished by estimating the screen reflection in the rendering pipeline with a model of the screen’s diffuse surface reflectance. However, because the experiment included some mismatched lighting conditions, a measurement-based approach was used instead. For each of the real ambient lights, the screen was turned off and the front surface was measured at three vertical positions (corresponding to the top row, middle row, and bottom row) with a spectroradiometer. In both cases, the absolute magnitude was relatively small. For the daylight booth light, the reflection luminance ranged from 0.58 cd/m² at the top row to 0.42 cd/m² at the bottom row. For the tungsten light, it ranged from 0.61 to 0.49 cd/m². For each light source, a linear model of the reflected luminance was fit to the data and was used along with the average chromaticity of the reflection to calculate an estimated XYZ for each vertical position. The estimated XYZ was subtracted from the target XYZ in the rendering engine as a correction.

With the corrective model in place, a set of measurements were taken of the chromaticities of the experiment stimuli to confirm there were no major differences between the ambient lighting conditions. Measurements were taken with the PR-655 of the rendered tungsten
background (luminance factor, 100%) and rendered tungsten patches (luminance factor, 35%) at three screen locations for each lighting and surface condition. The mean results for each condition are shown in Table 13. With the surface model active, the variation between the ambient lighting conditions was relatively small. The average differences in u’ and v’ due to ambient lighting were less than or equal to 0.0006 for each of the luminance factor/surface texture combinations in the table.

Table 14. Measured Chromaticities of the Tungsten Background and Patch Stimuli across the Ambient Lighting Conditions [Target u’v’ = (.2681, 0.5316)].

<table>
<thead>
<tr>
<th>Luminance Factor Level</th>
<th>Ambient</th>
<th>u'</th>
<th>v'</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Flat</td>
<td>Textured</td>
<td>Flat</td>
</tr>
<tr>
<td>Background (Y=1)</td>
<td>No Light</td>
<td>0.2680</td>
<td>0.2675</td>
</tr>
<tr>
<td></td>
<td>Tungsten</td>
<td>0.2679</td>
<td>0.2674</td>
</tr>
<tr>
<td></td>
<td>Daylight</td>
<td>0.2680</td>
<td>0.2676</td>
</tr>
<tr>
<td>Patch (Y=0.35)</td>
<td>No Light</td>
<td>0.2678</td>
<td>0.2674</td>
</tr>
<tr>
<td></td>
<td>Tungsten</td>
<td>0.2676</td>
<td>0.2672</td>
</tr>
<tr>
<td></td>
<td>Daylight</td>
<td>0.2682</td>
<td>0.2678</td>
</tr>
</tbody>
</table>

10.2.3 Procedure

Participants were given instructions that they would see a grid of patches on a background and were asked to select the patch that appeared the most achromatic (the one having the least color content). These were similar to the instructions in the Gorzynski and Berns [1990] experiment. Participants were asked to scan over the whole set of patches before making a selection and to not stare at any one patch for an extended period of time. Patches were selected by navigating to the selected patch, starting from the center, by pressing the up/down/left/right arrow keys. When participants reached their selection location, they pressed
the space bar key, and a text-to-speech engine voice said aloud the current location (e.g. “Up 2, right 2” for the top right corner) to allow for confirmation they were on the correct location, since no visual cues were given. After hearing the location, observers were able to continue navigating or confirm their choice and advance to the next selection stage by pressing the enter key.

Each experiment began with a set of two baseline trials where the background was set to the daylight chromaticity, the light booth was turned off, and the surface was untextured. After completing these two trials, the main experiment was initiated. The background color was set to the tungsten chromaticity for all trials of the main experiment. For each set of conditions, two trials of a given type were presented before advancing to the next set of conditions. Each time the conditions were changed, there was a 60 second adaptation period where the entire visible screen area was rendered as a white surface (being lit by a tungsten light).

The timeline of a representative experiment is shown on the following page (note that the block order and surface texture order were both counterbalanced, so the exact order of conditions in the main experiment was different for every participant).
• Baseline (Daylight Background)
  o BLOCK 0: Light Off
    ▪ Surface Flat
      • Adaptation 60s
      • Trial 0a
      • Trial 0b

• ----- - Adaptation 60s – (Tungsten Background) Light Off, Surface Flat

• Main Experiment (Tungsten Background)
  o BLOCK 1: Light Off
    ▪ Surface Flat
      • Adaptation 60s
      • Trial 1
      • Trial 2
    ▪ Surface Textured
      • Adaptation 60s
      • Trial 3
      • Trial 4
  o BLOCK 2: Daylight Ambient
    ▪ Surface Flat
      • Adaptation 60s
      • Trial 5
      • Trial 6
    ▪ Surface Textured
      • Adaptation 60s
      • Trial 7
      • Trial 8
  o BLOCK 3: Tungsten Ambient
    ▪ Surface Flat
      • Adaptation 60s
      • Trial 9
      • Trial 10
    ▪ Surface Textured
      • Adaptation 60s
      • Trial 11
      • Trial 12
In total, 12 observers (4 female, 8 male) with ages ranging from 24 to 51 (Mean = 36.2, SD = 10.2), participated in the experiment. The majority were experienced observers who had participated in previous vision and color experiments.

### 10.2.4 Analysis Methods

The experiment used a two-factor within-subject design, where every observer provided responses for every combination of the ambient illumination and surface texture factors. With this type of design, the data analysis was performed based on within-subject contrasts where the intra-observer differences between sets of conditions were calculated. Statistical testing over the group was performed based on these contrasts, instead of the absolute mean and variance of the group (as in a between-subject design). This was necessary because with the within-subject design, the data from different conditions cannot be considered independent due to correlation in the set of responses for the same observer. Before calculating contrasts, the observer’s two responses for a condition were averaged.

The statistical significance of the results was assessed by conducting a two-factor within-subjects (repeated measures) multivariate analysis of variance (MANOVA). The use of the multivariate version of ANOVA allowed for hypothesis testing to be performed simultaneously on both the u’ and v’ dependent variables. The MANOVA test was used to assess the main effects of the lighting condition factor (Off, Daylight, Tungsten) and the main effects of the texture factor (Flat, Textured). Additionally, the interaction of Lighting x Texture was also evaluated. The MANOVA statistical tests were performed using SAS version 9.3 with the PROC GLM procedure for repeated measures and the identity specifier for the multivariate u’v’.
response variables. Post-hoc paired Hotelling $T^2$ tests were applied to assess the significance of individual conditions in cases where a significant main effect was found.

### 10.2.5 Results

The set of intra-observer differences for the ambient lighting factor (Tungsten – No Light, Daylight – No Light) are plotted in Figure 98 and the differences for the surface texture factor (Textured – Flat) are plotted in Figure 99. As shown in Figure 98, when the real-world illumination in the light booth was set to the tungsten light, the achromatic point selected by observers shifted in the positive $u'$, positive $v'$ direction (toward the white point of the Tungsten background shown onscreen). When the light booth was set to daylight, the achromatic point selected by observers shifted in the negative $u'$, negative $v'$ direction (toward the ambient daylight chromaticity and away from the tungsten background shown onscreen). The use of more realistic textured surfaces, as opposed to simpler flat-shaded patches, was not found to have a clear effect on the achromatic point selected by observers (contrasts shown in Figure 99). In a MANOVA analysis, the main effect of the lighting factor was statistically significant at a $p < 0.05$ level [$F(4, 8)=12.12$, $p=.0018$]. There was not a significant main effect of the surface texture [$F(2,10)=2.30$, $p=.1505$]. There was also not a significant interaction between the ambient lighting and surface texture factors [$F(4,8)=0.66$, $p =.6371$].
Figure 98. Plot of the intra-observer contrasts for the effect of ambient lighting. The within-subject differences for the tungsten condition minus the light-off condition are shown as red squares. The within-subject differences for the daylight condition minus the light-off condition are shown as blue triangles.
Figure 99. Plots of the intra-observer surface texture – flat shading contrast for the three lighting conditions (light booth off, the booth set to tungsten, and the booth set to daylight). The top left plot includes the data for all three lighting conditions combined.

A set of post-hoc tests were performed for the Tungsten > No Light comparisons and for the No Light > Daylight comparisons using paired Hotelling $T^2$ tests on the bivariate $u’v’$ response [Jackson 1959]. As four post hoc tests were performed, a conservative Bonferroni criterion for the tests requires p-values below 0.0125 to meet a 0.05 level of significance. The test of Tungsten > No Light was statistically significant in both the textured case $[F(2,10)=11.06$,
p=.0029] and the flat surface case [F(2,10)=17.01, p=0.0006]. The effect No Light > Daylight was also statistically significant in both the textured case [F(2,10)=7.72, p=.0094] and the flat case [F(2,10)=84.91, p<.0001]. The main effect of surface factor was not significant (nor was the interaction with the light condition), so there was not sufficient statistical evidence to support additional testing for the Textured vs. Flat factor.

The achromatic points selected by observers for each of the six main experiment conditions are shown in Figure 100. The data for the baseline condition, which has a daylight instead of tungsten background, is also shown. The spread of the points in each plot represent the inter-observer variance for a particular condition. In some cases, the between-subject variance in a group is relatively large. However, for within-subject factors, this type of variance does not necessarily indicate whether or not there are significant differences between conditions. The mean and median values of the 12 data points in each condition are plotted in Figure 101. In the mean and median plots, there is a pattern of increased adaptation from the daylight ambient illumination, to no ambient, to the matched tungsten illumination. Based on the statistical analysis, this pattern was statistically significant. There are small differences between the texture vs. flat conditions in the mean and median plots, but based on the statistical analysis the surface texture factor was not significant.
Figure 100. The achromatic points selected by all observers for each condition. The blue triangle in the bottom left corner represents the daylight chromaticity and the one in top right corner represents the tungsten chromaticity.
Figure 101. The mean and median of the set of observer responses for each condition. The textured surface condition is shown in red and the flat surface condition in black. The shapes represent the ambient illumination with circles for no light, squares for tungsten light, and triangles for daylight. The baseline condition is represented by the green X.

10.2.5.1 Comparison of Object Display Results to the Original Screen/Print Experiments

The results from the object display experiments (left column) are shown in Figure 102 along with results from the original Gorzynski and Berns experiments performed on physical prints and on display screens. The plots of the original experiments (right column of Figure 102) were constructed from the data tables found in Gorzynski’s M.S. thesis [1992], by taking the mean and median statistics for the three observers that participated in both the screen and
print conditions (this thesis dataset has the same three points for the soft copy conditions as the Gorzynski & Berns [1990] paper, and three of the five data points that were plotted for the print experiment). It is not possible to compare u’v’ coordinate data directly between the current experiment and the original experiments because the adapting tungsten and daylight chromaticities (represented by the large blue triangles) were not identical in the two sets of experiments. However, the pattern of achromatic selections in relation to their respective tungsten and daylight adapting points can be still be informative.

In both sets of experiments, simple flat shaded patches were shown onscreen for a tungsten background, with the lights off (black circles), with tungsten ambient light (black squares), and with daylight-balanced ambient light (small black triangles). In the original soft copy experiment, there was not a clear effect of the ambient light on observers’ adaptation to the tungsten background. In all three conditions, the adapted white point was approximately halfway between the tungsten and daylight points. In the experiments with the object display, there was an effect of ambient illumination found for the flat-shaded conditions, with increasing adaption to the tungsten background chromaticity when the tungsten ambient lighting was present and decreased adaptation to the tungsten background with the simulated-daylight ambient lighting present.

In addition to the flat-shaded conditions, the object display experiments included patches that were rendered and shaded to resemble a real-world painted surface. In the case of the tungsten ambient condition (red squares) the virtual surface was rendered using a model of the actual physical lighting surrounding the display. For comparison, the achromatic point selected in the Gorzynski experiment (#2) for the real physical samples under tungsten illumination is plotted in the right column of Figure 102. In the case of the real physical sample, there is nearly
complete adaptation to the tungsten white point. In object display experiment, there is a significant shift in adaptation toward the tungsten white point when using the matched tungsten lighting (relative to no lighting), but it does not completely reach the full adaptation level found in the original experiment for real samples.
Figure 102. Comparison of the object display results (left column) to the results of the original Gorzynski thesis [1992] soft copy/hard copy experiments (right column). The large blue triangle in the bottom left corner represents the daylight chromaticity and the blue triangle in top right corner represents the tungsten chromaticity.
10.2.6 Discussion

Though the mean results for the object display did not reach the level of full adaptation to a tungsten background, overall the pattern of results across the experimental conditions indicate that an object display approach does partially increase chromatic adaptation beyond a typical display representation. Additionally, the findings provide information about which factors may contribute to the differences in adaptation that have been found for displays screens and real-world samples.

While the use of more realistic surfaces with simulated texture did not generally increase the level of adaptation, the use of real-world lighting consistent with the lighting of the virtual samples did increase the level of adaptation. It was hypothesized that both the matched real-world lighting and textured surfaces could aid in adaptation, because both factors had the potential to make the stimuli onscreen appear more like an illuminated real-world object (where adaptation is typically complete). Based on the results, it appears that matched ambient lighting is a more important factor than texture for achieving a level of chromatic adaptation close to the level found with real-world objects. These quantitative results are consistent with the qualitative experience (of the author) when viewing the stimuli. Even with texture present, the background appeared to have a distinctly orange hue when the daylight-balanced booth light was used. With or without texture, the background appeared to be nearly white when the light booth was switched to the tungsten light. The speed at which the apparent change occurred is also worth noting. The qualitative effect seemed to occur almost immediately with a change in the ambient lighting. However, it is not clear whether the effect measured quantitatively in the experiment occurred as quickly, since a 60 second adaptation period was always provided at the start of each condition.
In the analysis, there was no main effect of texture found. In the case of the mismatched daylight ambient lighting, there was a trend toward increased adaptation to the tungsten background for the texture condition. In the matched tungsten ambient lighting, there was a slight trend in the opposite direction. However, given the lack of a statistically significant result and the small magnitude of the differences relative to the variance of responses, there is not sufficient evidence to draw any conclusions about a texture effect. Further experiments would be necessary to determine if a small texture effect may be present for a specific lighting condition. In the current experiment, each trial began with a wide range of patches (in $u'v'$ space) centered around the point halfway between the tungsten and daylight chromaticities. In a future experiment, with tighter sampling patterns that begin closer to the typical adaptation point for a specific lighting condition and a large number of trials and observers, there may be a better chance of detecting a texture effect if a small one is present.

There was a main effect of ambient lighting, and increased adaptation for the tungsten lighting was present for both the texture and flat conditions. Though observers showed the greatest level of adaptation in the tungsten lighting case, the mean (as well as median) of the group response did not fully reach the tungsten background chromaticity. In the Gorzynski and Berns [1990] experiments with physical prints (experiment #2), observers were found to fully adapt for tungsten lighting. It is possible that there are visual differences between the object display screen and a real-world object that account for the lesser degree of adaptation. However, there are other possible reasons, beyond purely visual cues, that could have contributed to the lesser degree of adaptation. In the object display study, participants viewed all the conditions, including ones with no ambient lighting and no virtual texture. In these cases it would be clear that they were viewing a typical self-luminous display screen. The display was not hidden from
view when changing the lighting or texture conditions or when the patch colors were updated, and so it is likely that observers were consciously aware they were viewing a display screen, even if it was visually difficult to distinguish from a real object in certain conditions. In future work, it would be interesting to test a naïve set of observers for just the matched lighting condition, while taking precautions to physically hide the display from view any time a change to the stimuli onscreen was made.

Another potential reason for the lesser degree of adaptation is related to the patch sampling method used for the object display experiment as opposed to the typical sampling of patches for a real-world object. To remain consistent over all the conditions in the object display experiment (including the daylight baseline), the initial sampling pattern was always centered halfway between the daylight chromaticity and tungsten chromaticity points. In the tungsten ambient light condition, this starting point had a far lower CCT than the final average achromatic point selected by observers. If observers were adapting to the average of the patches to some degree, and not purely the background, their mixed adaptation state would be somewhere between tungsten and the halfway point. With the re-centering after each successive selection, the average of the patches would move closer to the tungsten point, but may not fully reach it. In contrast, when using real physical prints (as in experiment #2 from Gorzynski), the samples are typically constructed using surface reflectances that are relatively symmetric around a neutral (non-selective) point. This is necessary with real samples so that they can be used for different physical lighting conditions and maintain a reasonable sampling distribution. With this configuration, the patches will generally average to the color of the physical light and so partial adaptation to average patch color does not have the same potential to lessen the degree of adaptation as it does for the object display experiment sampling scheme. This question could be
tested in a future experiment using a starting point for the object display that is closer to the tungsten chromaticity. This was considered when testing possible sampling schemes for the current experiment, but presented practical issues as the tungsten point was too close to the edge of the display gamut to allow for symmetric sampling with a wide chromaticity range.

In both the current object display experiments and the original experiments performed by Gorzynski and Berns, the level of adaptation when viewing a display screen with simple flat patches on a tungsten-balanced background was evaluated under different ambient lighting conditions. In the case of the original study, there was no clear effect of the ambient conditions. In the object display experiment, there was an effect of ambient lighting, with the tungsten lighting increasing the adaptation to the tungsten white point and daylight-balanced light decreasing it. Further experiments are necessary to systematically evaluate the cause, but differences in the physical configurations for the two sets of experiments are a potential reason for the different set of results. In the Gorzynski and Berns experiments, the stimuli occupied a center portion of the screen with the remaining portions of the screen set to black. A flare shield was also placed above the CRT to block ambient light from falling on the screen. In the case of the object display experiments, the ambient lighting was more closely integrated with the stimuli displayed onscreen. There was a physical white border adjacent to the screen region displaying the stimuli and it was directly illuminated by the ambient lighting at a luminance level consistent with the stimuli shown onscreen. There are two potential reasons why this closer integration could be important. One potential reason is that a relatively seamless transition between a real physical mask and a virtual surface onscreen helps to give the appearance that the virtual surface is a reflective object. This may induce the mechanisms that lead to discounting of the illuminant with real objects. Another possible explanation is that with a physical mask at the edge of the
screen stimuli, observers begin adapting to the mask white point and this state is maintained when looking at samples onscreen if they appear to be similarly illuminated.

Finally, it is not possible to rule out that differences in the display hardware, or other factors, between the two experiments may have contributed to the different pattern of results. As the Gorzynski & Berns experiments were performed in early 1990’s, they were conducted using a CRT display, while the object display experiments were performed using an LCD panel. With these two different display technologies, there is the potential that differences in screen resolution, pixel fill factor, screen flicker, screen curvature, or other screen factors may have contributed to the difference in results.

10.3 Conclusions

One of the original motivations for developing an object display system capable of producing virtual reflections consistent with the real ambient lighting was to control for viewing condition differences that typically impact color appearance in cross-media situations. The absolute luminance of color stimuli and the illumination level of the surround (dark, dim, average) both factor into the color appearance. These often differ between screen viewing and object viewing and necessitate the use of color appearance models for cross-media reproductions. By using the same illuminated surround and reproducing stimuli on screen at the same absolute luminance levels as a physical object at that location, the intention was to match these viewing condition factors so that an appearance model adjustment would not be required. Additionally, virtual surfaces are rendered for the actual chromaticity of the illumination, which would provide a consistent adapting white point between the object and screen representations. Though the adapting white point is the same, there is still the potential for cross-media color
appearance differences if observers are not able to adapt the real illumination present when viewing the display.

The results of the chromatic adaptation experiment indicate that observers do adapt to a significant degree to the real illumination present when using the object display (with matched lighting), but not as fully as in the case of reflective objects. This would suggest that a chromatic adaptation transform may be necessary (though to a lesser degree than usual), even when using an object display approach. However, it is not clear to what extent the less complete adaptation (relative to a real object) found in the experiment is inherent to object display systems in general. There is the possibility that it resulted from some specific factor in the experiment (such as the color sampling procedure) or was a result of one of the limitations in the specific object display system implemented in the experiment. For future work, additional experiments would be necessary to draw a definite conclusion about whether any color appearance adjustments should be applied when using an object display system.
11 APPLICATIONS

There are several potential application domains where the ability to present computer-generated content in a way that simulates the appearance of real physical surfaces would be advantageous. In this chapter, a set of sample applications are described to demonstrate some potential uses of the object display system. The sample applications include a print soft-proofing application that simulates the material properties of different types of paper. The capabilities of the system to display virtual reproductions of artwork, as could be used for a virtual museum exhibit, are also demonstrated. Additionally, an object display system could provide a useful research platform for studying aspects of material and surface color perception.

11.1.1 Digital Print Proofing for Gloss and Texture

A digital print soft-proofing prototype was developed for the object display system to serve as a proof-of-concept of its potential capabilities in this domain. The application allows the user to preview the appearance of digital photographs when printed on simulated photo papers with a range of texture and gloss properties. A simple version, shown in Figure 103, was developed for use with the “tangiBook” system, the first generation tangible display [Darling & Ferwerda 2009]. The prototype shown in Figure 103 allowed the user to select between four virtual papers with different texture and gloss levels (matte, canvas, gloss, high-gloss). With the interactivity of the tangible display, the appearance of the virtual print updated as the user changed the orientation or viewed it from different directions to help convey the gloss and texture properties.
Figure 103. Images illustrating a simple interactive soft-proofing application on a tangible display system. Buttons in the interface allow a digital image to be proofed on simulated photo papers with different textures and gloss levels (canvas, left and high gloss, right). The user is able to interact with the virtual print by viewing it from different directions or changing the orientation of the screen to see the changing patterns of reflections.

A new version of this application was developed for the object display system that allows a virtual print to appear and behave more like a physical reflective print. In the application, the user may select from a set of photo paper textures (canvas, papyrus, textured, glossy) that were captured using photometric stereo [Woodham 1980]. The user can also set the gloss level of the virtual print to one of seven pre-measured gloss levels. Two different lighting options are available in the application. The virtual photos can be displayed for the overhead light booth illumination (described in Chapter 5). Additionally, to create highlights that accentuate glossy surfaces, there is the option to use the projector-based lighting workflow (described in Appendix D) that simulates the patterns of light from a source in front of the screen at specular angles with respect to the virtual surface. If a real light source of this type were used to light the environment, it would create a significant front surface reflection from the screen. With the projector source, the portion of light incident on the screen can be masked out to reduce the
unwanted reflection (this requires that the screen remain in a fixed location). In the system, it is possible to use either of the lighting types independently or use both at the same time. The user can select from a set of digital photographs (captured in RGB) that have been incorporated into the application with an approximate conversion to the six-channel primary representation. The prototype application is shown in Figure 104, displaying a photograph on the simulated canvas paper. In the figure, the outer black border and gray border are real while the textured white mat is rendered on the object display. Screen captures of the photograph displayed on different simulated papers are shown in Figure 105.

![Figure 104](image)

**Figure 104.** A simulated print on canvas-textured paper is presented on the object display. The black border and gray border are real. The textured white mat inside the gray border and the photograph are rendered on the display.
Figure 105. Cropped screen capture images illustrating different paper options in the printing example application. In the top image, the bird photo is simulated as a print on canvas paper. In the center image, the photo is simulated for papyrus paper and in the bottom image, for a glossy paper.
The example application is intended to demonstrate the potential for a digital proofing system based on the object display approach, but future work will be necessary for it to predict the appearance of the actual print that will be created for a specific printer. The color processing in the prototype is based on the six-channel multispectral workflow, so the real color of the print could be estimated if spectral information on the target print were available from a spectral printing model (e.g. Taplin & Berns [2001]). In the prototype, which uses typical 3-channel RGB photos as input (specified in sRGB), the six-channel color representation was calculated by a simple approximation where the sRGB photo colors were converted to CIE XYZ and transformed with a [3 x 6] matrix to the six-channel reflectance representation. In future work, a more accurate method for estimating the six-channel representation based on the spectral properties of the printer ink could be developed or, if a spectral printing model has been developed for the specific target printer, it could potentially be incorporated into the virtual proofing software. Additionally, in a real printing system, there is likely to be interaction between the different inks and types of paper that will impact the surface properties and so the modeling method described here, where they are treated independently, may not be sufficient. In the future, if BRDF models characterizing printer ink and paper interaction (as in research by Gatt et al. [2006]) become available for various printers, they could be combined with the display capabilities demonstrated in the prototype application to create a proofing system that previews the real material properties of the physical print in a real-world context.
11.1.2 Psychophysics of Material Appearance

Another potential application area for object display systems is as a platform for performing psychophysical experiments on material perception. In recent years, the study of surface appearance has been facilitated by computer graphics technologies that have allowed experiments to be conducted using high fidelity rendered images of surfaces with complex illumination, geometry, and material properties [Pellacini et al. 2000; Fleming et al. 2003; te Pas & Pont 2005; Ho et al. 2007a]. An object display system provides the means to bring computer-generated stimuli into the physical environment of the observer and simulate the experience of viewing physical samples.

An object display system could be used as a platform for performing studies on gloss perception. An example of a gloss experiment using an object display is illustrated in Figure 106 and Figure 107. A set of samples varying in gloss level, based on the NCS gloss sample set, was modeled in the object display framework (shown as screen capture images in Figure 106). One of the modeled samples is shown being presented on an object display system in Figure 107. To create a clear specular highlight to emphasize the gloss, an additional light source is used along with the overhead illumination from the light booth. A lamp with two linear bulbs was modeled in the object display framework. The projector-based lighting workflow (described in Appendix D) is used to project patterns of illumination for the lamp on the physical surfaces around the screen. With the projector-based lighting, the screen region can be masked out in the projector image to reduce the unwanted physical reflection from the screen surface.
Figure 106. Screen captures of a set of glossy samples modeled in the object display framework.
Figure 107. One of the modeled gloss samples being presented on the object display system. The highlight rendered to the screen updates with changes in the tracked viewing position.

In an experiment using an object display, it would be possible to systematically vary the viewing conditions or aspects of the gloss reproduction to evaluate how different factors influence the perception of gloss. Observers could evaluate the set of samples (using magnitude estimation or another method) to establish a perceptual scale for each condition and the derived scales could be compared across conditions. For example, with the tracking capabilities of the object display active, the virtual highlights rendered to the screen will update as the observer views the virtual object from different directions. However, this feature could be disabled and the samples could be judged in a static condition to evaluate the impact of dynamic viewing on
gloss perception. In future studies, the ability to vary factors under computer control while still presenting stimuli that behave in a manner similar to real objects could make the object display a useful platform for conducting research on material appearance.

11.1.3 Digital Reproductions of Cultural Heritage

Another potential application area for object display system is as a platform for displaying digitally-captured artwork and other cultural heritage objects. Advances in imaging and computer graphics technologies over the last decade have led to new methods for digitally capturing and modeling works of art and others items of cultural significance [Gardner et al. 2003; Tominaga 2005; Tominaga & Tanaka 2008; Chen et al. 2008; Ashbaugh et al. 2009, Berns & Chen 2012; Chen & Berns 2012]. Unlike photographs that capture a fixed view, these methods capture the underlying surface topography and gloss properties of paintings and other works of art. With these rich digital representations, captured models can be displayed interactively on an object display system to simulate the experience of viewing the original. An initial prototype application was developed for the original tangiBook system [Darling & Ferwerda 2009] to provide an interactive experience when viewing digital models of artwork. As shown in the screen capture image sequence of Figure 108, as the user interacted with the tangiBook system (displaying an illuminated manuscript model digitized by Gardner et al. [2003]), the rendered appearance updated allowing the user to see the glints off the gold leaf and the texture of the vellum.
Figure 108. An illuminated manuscript (digitized by Gardner et al. [2003]) is displayed on the “tangibleBook” system. The highlights in the gold leaf and vellum change, as shown in the left image to the right image, as the display is tilted by the user.

With the tangible display version, there were natural forms of interaction with the captured surface, but it did not provide the appearance of a real-world reflective object. The current object display system provides the means to more closely emulate the experience of viewing a real physical surface. This is illustrated in Figure 109, where a captured model displayed on the object display system is shown side by side with the original painting. The original painting was captured with a total appearance imaging system [Berns & Chen 2012] that provided multispectral reflectance data for the surface color along with a model of the surface texture properties. The multispectral reflectance data was converted to the six-channel rendering representation of the object display framework and used to render the surface color interactively for the real ambient illumination in the light booth environment.

Though often the goal is to reproduce the appearance of the original, it may be useful when studying a work of art to be able to emphasize one property (such as the texture) by removing other surface properties. In the sample application, surface properties such as the
diffuse color, surface texture, surface gloss, and shadowing can be toggled on or off interactively while viewing the surface. In Figure 110, painting is shown with its diffuse color removed to emphasize the shading and shadowing from it mesoscale texture.

**Figure 109.** A physical painting side-by-side with an object display system. The physical painting, shown on the left, was captured with a multispectral system for total appearance imaging of artwork [Berns & Chen 2012]. Right, the captured model is presented on the object display system.
Figure 110. A real painting alongside the surface model presented on an object display system. The virtual painting can be displayed without diffuse color information to emphasize the texture properties.
12 CONCLUSIONS AND FUTURE WORK

While traditionally display systems have been developed with a focus on recreating the light record of a scene captured in an image or video, there is a range of display applications where the primary goal is to convey the appearance of object surfaces. With recent advances in display technology as well as methods for capturing rich digital object content, there are opportunities to create new types of interactive display systems for these object appearance applications. The work in this dissertation was focused on developing a methodology for transitioning from display systems that present high-fidelity images to ones that act as digital surrogates for real world objects.

The principal goals of the dissertation were to create a general framework for reproducing objects with gloss, texture, and color properties that behave and appear like real physical objects and to develop a functional prototype based on this framework. In the dissertation, the process of creating the functional system was described to illustrate the set of technologies and components necessary and how they can be integrated to meet the goals of the conceptual framework. Using the prototype, it was possible to demonstrate the capabilities of a system based on the conceptual approach and evaluate the accuracy in physical experiments. In developing the general framework and a functional system, the goals were to:

1. Provide the means for natural modes of interaction with virtual surfaces onscreen

2. Facilitate the process of displaying captured objects by specifying a standard input form for color, texture, and gloss properties compatible with typical captured data
3. Provide an abridged spectral representation for surface reflectance and illumination compatible with a real-time rendering workflow to allow for accurate color for a range of real-world lighting types

4. Provide the capability to calculate photometrically-accurate gloss reflections, for real-world lighting, within a real-time rendering engine

5. Develop a display modeling component with the ability to produce the absolute luminance and colorimetry of the rendering engine output and the ability to maintain this output for a range of viewing angles (out to 30°)

6. Evaluate the accuracy of the object display system with measurement-based testing

7. Provide a demonstration of how an object display can be used to assess the accuracy of digital models by taking direct physical measurements

8. Assess to what extent the object display methodology allows observers to adapt to a self-luminous display screen

9. Evaluate which factors may contribute to the differences in adaptation between viewing display screens and physical objects

10. Demonstrate the potential utility of the object display methodology for real-world applications.

The work performed toward these goals was described in the dissertation document, beginning in Chapter 2 with a review of background material and past research that is related to the work in the dissertation. Chapter 3 outlined the high-level conceptual framework that was used in transitioning from a traditional pictorial display representation to an object-based one. In Chapter 4, the capabilities and technology for tangible display systems, interactive displays that
provide natural forms of interaction with virtual surfaces, were described. Tangible systems represented the early stages of the object display development process and their relation to the later systems was discussed. In Chapter 5, the overall design of a functional object display prototype was presented along with a set of example illustrating its capabilities. The surface and illumination models used by the system were identified and the forms of data input were described.

Chapters 6, 7, and 8 provided additional detail on three main components of the technology for an object display system. Chapter 6 provided a description of the multispectral color processing pipeline optimized for use in the object display framework, along with simulation data on its performance. In Chapter 7, the rendering techniques and mathematical modeling used in the gloss reproduction pipeline were described. The performance of the different rendering methods (median cut, filtered importance sampling, and a hybrid approach) was evaluated with a simulation. Chapter 8 described work on measuring and modeling the colorimetric and directional properties of the screen. An interactive model was implemented to correct for luminance changes as a function of viewing angle and was tested in a measurement experiment.

In Chapter 9, a measurement-based system evaluation was performed to evaluate color and gloss reproduction on the object display. To evaluate color accuracy, a virtual ColorChecker was displayed on the screen and measured for different real-world lighting conditions. To evaluate gloss reproduction, a set of real samples were modeled and goniometric measurements taken from the virtual objects onscreen were compared to measurements taken of the real samples in the same measurement configuration. This chapter illustrated the methodology that
could be used to assess the accuracy of captured digital models with physical measurements taken from the screen.

Chapter 10 described a visual experiment to assess the state of chromatic adaptation when using the object display system and evaluate its properties for use in cross-media color appearance applications. The experiment was designed to test whether observers adapt more fully when using the object display methodology than is typically the case with self-luminous displays screens. The use of a matched ambient lighting environment (as with object displays) was found to produce more complete adaptation than with no ambient or mismatched ambient lighting. The use of uniform flat patches vs. more realistic textured surface was also tested, but was not found to have a significant impact on adaptation. Though the degree of adaptation found in the experiment was not fully complete (as expected for a real object), the adaptation state was nearer to this full adaptation state than is typically the case for self-luminous display viewing.

In Chapter 11, a set of sample applications was presented to demonstrate the potential utility of object displays in three application areas. Examples were shown for uses of the object display for digital proofing, as a platform for conducting psychophysical experiments on material appearance, and as means to display digitally captured artwork and other items of cultural significance.

12.1 Limitations and Future Work

While the results of the current work with object display systems are promising, there are certain limitations and several potential avenues for future work.
Conceptually, the object display framework is based on an ideal screen that has the capability to produce the exact colorimetry (at the absolute luminance and chromaticity) of the virtual reflections calculated by the rendering engine. When using real screens, however, there are physical limitations that can prevent the calculated virtual reflections from being reproduced as needed. There are two types of departures from the ideal screen; the real screen may be unable to generate light at the absolute luminance level and chromaticity required, or conversely, the display may reflect unwanted light from the real environment. With more advanced display screens in the future, some of these limitations may be reduced and allow for object display systems that better implement the object display framework as conceptualized.

In current systems, the screen output limitations are partially overcome by selecting lighting configurations and lighting intensities that will produce reflections (on typical real-world objects) that generally fall within the range that the screen is able to produce. The screen used in the dissertation work was a high-luminance display (up to 900 cd/m²) with a backlight that could be globally adjusted to help scale the display output to the luminance levels needed for the lighting environment. Even with the range generally matched, however, dark shadows or sharp specular highlights are likely to fall outside the display capabilities. High-dynamic range displays are not yet readily available in a form conducive to use for an object display, but in the future could provide significant benefits for creating a system that can generate the full range of light output necessary to match a real-world object under a broad range of lighting conditions.

The color gamut limitations of standard display screens also present a challenge for implementing exact color matching for certain real-world light sources. In the object display ColorChecker measurement experiment (Section 9.1), the accuracy of the screen output was relatively high for the two fluorescent daylight simulator conditions. The rendering engine was
able to calculate reasonably high accuracy for the tungsten condition, but there were significant errors for some out-of-gamut patches (orange, red, yellow) when they were actually measured from the display screen. In the future, a specialized wide-gamut screen, potentially with more than three primaries, could help to provide accurate reproduction for a wider range of surface colors under multiple types of illumination.

An additional physical limitation of real display screen is the unwanted physical reflection from the front surface of the screen. The diffuse reflections from the screen are relatively minimal and can be corrected, but the specular reflections present a greater problem. The specular reflectance is typically high enough (a peak reflection of 1.1% the source luminance was found in Section 8.2.3) that it can result in a significant amount of undesired physical front surface reflection if a high luminance sources is located directly in front of the screen. When using the object display system in the dissertation, the screen was typically used in a configuration where the lights were not directly in front of the screen (as with the overhead light in Chapter 5). In this configuration, it is still possible to show specular highlights by simulating a surface with a curved top edge or adding enough virtual surface texture that upward facing facets are at a specular angle. The other solution to the screen surface reflectance problem was the use of a computational projector-based lighting workflow (described in Appendix D). This projector-based lighting was developed for situations where a high-intensity specular light source is needed to illuminate a flat virtual surface. The projector is used illuminate the real world scene, but masks out the portion of the projected light that would strike the screen. However, this technique has its own limitations and adds significant complexity to the use of an object display system. For future systems, the best solution would be to obtain a screen with no
significant surface reflectance, but there may be fundamental limits on how much the reflectance can actually be reduced.

In the future, new screen technologies for high-dynamic range, wide color gamuts, minimal viewing-angle dependencies, and lower surface reflectance could aid in creating a more capable and accurate object display system. Additionally, there is the possibility to expand the object display framework beyond the use of traditional 2D display screens. With the current technologies commonly used for 3D display, the user typically has to wear either LCD-based shutter glasses or polarized glasses so that each eye can receive a different set of stimuli from the screen. The decision was made not to utilize these types of 3D displays because wearing the glasses would inhibit simultaneous viewing of real-world objects, significantly reduce the absolute luminance, and impact the color accuracy of the light ultimately reaching each eye. When high fidelity autostereoscopic (glassless) 3D displays become available, this could provide the means to incorporate binocular disparity into the object display system. This would allow for more realistic gloss highlights that appear to be behind the screen in 3D space, as opposed to in the plane of the screen. This type of technology may also provide the opportunity to switch from an active observer-tracking paradigm to a passive one where the direction of output is modulated by the display. With current lenticular screens, there is a tradeoff between the spatial resolution of the elements of the image and the angular resolution of the light emitted for each picture element. With future technologies, there may be a high enough resolution both spatially and directionally to create passive high-fidelity surface reproductions. Though the screen would still need to be tracked with respect to real-world lighting, it would no longer be necessary to track the observer.
There are also some current limitations in the system that relate to the computational constraints of an interactive rendering pipeline. Currently, shadowing is the only height-based effect implemented in the object display system. The use of intermediate shadow maps allows this to be performed at interactive frame rates. Parallax and inter-reflections effects have not been implemented in the system. With new rendering methods and more powerful graphics hardware, it may be possible to incorporate these two effects in the future. Additionally, it may be possible in the future to directly calculate shadows from the height map at interactive rates, which would eliminate the need to create the intermediate horizon maps and simplify the data representation.

Currently, the only BRDF model used in object display systems is the isotropic form of the Ward-Dür model. In the future, more complex BRDF models or other types of models for simulating materials could be incorporated. Additionally, more complex lighting models could be incorporated to expand beyond the spatially-varying area sources currently used.

### 12.2 Impact

Though the current systems have limitations, this dissertation has presented a general conceptual framework for future systems and has taken a series of steps along the path of constructing display systems that can present digital objects in a way that recreates the experience of directly viewing real surfaces. With continued development and future advances in screen and graphics processing technologies, it may be possible to realize the final goal of presenting objects under computer control that are indistinguishable from viewing the real thing. An object-based approach for display systems has the potential for a broad impact, as it could provide benefits in a range of application areas where the goal is to preview, evaluate, or
faithfully reproduce the appearance of an object surface. It could allow for new types of high-fidelity computer-based design systems, where designers can interactively edit the properties of surfaces while directly viewing the results of their work in a real-world context. In the digital printing domain, an object display could be used to view digital proofs that simulate the surface properties of the real paper and behave like physical prints. It could provide the means to share digital archives of artwork and cultural heritage and present them in a way that helps to recreate the experience of viewing the original. As a computer-based platform for studying material perception, object displays could facilitate future research for gaining a greater understanding of object appearance phenomena.
13 REFERENCES

[Adelson & Bergen 1991]

[Adelson 2001]

[Ashbaugh et al. 2009]

[Bandyopadhyay et al. 2001]

[Bartleson & Breneman 1967]

[Bartleson 1968]

[Bartleson 1975]

[Becker 2002]

[Becker 2004]

[Becker 2006]

[Berns et al. 1993]

[Berns & Choh 1995]

[Berns 1997]

[Berns 2000]

[Berns 2001]
[Berns et al. 2003]

[Berns 2005]

[Berns et al. 2005]

[Berns & Chen 2012]

[Bimber et al. 2002]

[Bimber et al. 2005]

[Bimber & Raskar 2005]

[Blinn & Newell 1976]

[Blinn 1977]

[Blinn 1978]

[Borges 1991]

[Boynton & Kelley 1996]

[Braun et al. 1996]

[Buxton 2008]

[Chen et al. 2007]
[Chen & Berns 2012]

[Cheng 2007]

[Choh et al. 1996]

[Chung et al. 1989]

[CIE TC8-04 2004]

[Clark 1976]

[Colbert et al. 2006]

[Colbert & Křivánek 2007]

[Colbert et al. 2010]

[Cossairt et al. 2008]

[Cruz-Neira et al. 1993]

[Darling & Ferwerda 2009]

[Darling & Ferwerda 2010]
[Darling et al. 2011]

[Darling & Ferwerda 2012]

[Day et al. 2004]

[Debevec & Malik 1997]

[Debevec 1998]

[Debevec 2005]

[Devore 1995]

[Dorsey et al. 2008]

[Drascic et al. 1993]

[Drew & Finlayson 2003]

[Dür 2006]

[Edwards et al. 1993]

[Fairchild 1991]

[Fairchild 1992]

[Fairchild 1995]
[Fairchild 2005]

[Fairchild & Johnson 1999]

[Fairchild & Wyble 1998]

[Fairman et al. 1997]

[Ferwerda et al. 2001]

[Fisher et al. 1986]

[Fitzmaurice 1993]

[Fleming et al. 2003]

[Fleming et al. 2004]

[Forrest et al. 2000]

[Francois et al. 2003]

[Fuchs et al. 2008]

[Gardner et al. 2003]

[Gatt et al. 2006]
[Gortler et al. 1996]

[Gorzynski & Berns 1990]

[Gorzynski 1992]

[Gouraud 1971]

[Greenberg et al. 1997]

[Greene 1986]

[Hardeberg et al. 1999]

[Harker & O’Leary 2011]

[Heidrich & Seidel 1998]

[Heidrich & Seidel 1999]

[Henley & Fairchild 2000]

[Hill 2010]

[Ho et al. 2007a]
[Ho et al. 2007b]

[Hunt 1982]

[Hunt 1996]

[Hunt 2004]

[Hunter & Harold 1987]

[Imai et al. 2003]

[Ishii & Ulmer 1997]

[Jackson 1959]

[Joblove & Greenberg 1978]

[Johnson & Fairchild 1998]

[Johnson & Fairchild 1999]

[Kajiya 1986]

[Kamimigaki et al. 2009]

[Katoh et al. 1998]
[Katoh et al. 2001]

[Kelley et al. 1998]

[Kleffner & Ramachandran 1992]

[Klette et al. 1998]

[Koenderink & van Doorn 1996]

[Koike & Naemura 2008]

[Konieczny et al. 2005]

[Konieczny & Meyer 2006]

[Konieczny et al. 2008]

[Křivánek & Colbert 2008]

[Laihanen 1994]

[Lazzari et al. 2002]
[Levoy & Hanrahan 1996]

[Li et al. 2004]

[Lienhart & Maydt 2003]

[Masie et al. 1985]

[Max 1988]

[Meyer 1988]

[Meyer 2000]

[Meyer & Shimizu 2005]

[Milgram et al. 1991]

[Milgram et al. 1994]

[Miller & Hoffman 1984]

[Moroney et al. 2002]

[Nayar et al. 2004]

[Neider et al. 1993]
[Nicodemus et al. 1977]

[Nishida & Shinya 1998]

[O'Donnell & Billmeyer 1986]

[Okumura 2005]

[Oren & Nayar 1994]

[Palmer & Grant 2009]

[Patil et al. 2004]

[Pellacini et al. 2000]

[Peercy 1993]

[Peercy et al. 1996]

[Peercy et al. 1997]

[Pharr & Humphreys 2004]

[Pharr & Humphreys 2010]
[Phong 1975]

[Raskar et al. 1998a]

[Raskar et al. 1998b]

[Raskar et al. 2001]

[Raskar et al. 2003]

[Reinhard et al. 2010]

[Rolland et al. 2001]

[Rost 2006]

[Schenkman et al. 1999]

[Seetzen et al. 2004]

[Shimizu et al. 2003]

[Shimizu & Meyer 2005]

[Shirley 2005]

[Sloan & Cohen 2000]
[State et al. 1996]

[Stevens & Stevens 1963]

[Stockman et al. 1993]

[Strang 1993]

[Sueeprasan & Luo 2001]

[Süssstrunk et al. 2001]

[Sutherland 1968]

[te Pas & Pont 2005]

[Tamura et al. 2003]

[Taplin & Berns 2001]

[Tominaga 2005]

[Tominaga & Fukuda 2007]
[Tominaga & Tanaka 2008]

[Torrance & Sparrow 1967]

[Tsang et al. 2002]

[UmezU et al. 1998]

[Underkoffler et al. 1999]

[Unger et al. 2003]

[Veach & Guibas 1995]

[Viriyothai & Debevec 2009]

[Walter 2005]

[Ward 1992]

[Ward & Eydelberg-Vilesin 2002]

[Ward Larson & Shakespeare 1998]

[Ward 1994]

[Watt 1993]
[Westin et al. 1992]

[Whitted 1980]

[Wolfe 1998]

[Woodham 1980]

[Woolfe et al. 1997]

[Wyble & Rosen 2006]

[Yang et al. 2008]

[Yoshida et al. 2003]
Appendix A. Display Characterization Method for Channel Interaction

This section describes a display characterization method that incorporates techniques to compensate for channel interdependence and chromaticity changes over the primary ramps. The inverse model also includes methods to reduce the effects of quantization error and temporal drift. Due to the complexity of the model, the inverse interaction portion is implemented with a 3D LUT.

Display Model

An enhanced display characterization model was incorporated into the object display system to provide increased color accuracy for the chromatic adaptation experiment. The system typically uses the Day et al. [2004] approach, which provides a high enough level of color accuracy for general usage of the object display system (typically, CIEDE2000 mean = 0.5, max = 1.3). However, the display was found to exhibit some deviations from the assumptions of the Day et al. approach that led to small, but systematic error. In particular, the display was found to exhibit cross-channel interaction when multiple channels were driven at high levels, which violates the assumption of channel independence in the Day et al. method. In previous work, Woolfe et al. [1997] developed a method for characterizing CRT displays in the presence of cross-channel interaction using a 3D LUT to convert from channel independent scalars to channel scalars that include the effects of interaction. Tamura et al. [2003] developed a masking model for interaction that modeled the effects of channels in binary and tertiary combinations (RG, RG, GB, RGB) to account for channel interdependence. The approach used with the
object display models the conversion from the channel independent scalars to final channel scalars based on a polynomial model of the binary and tertiary channel combinations. This provides an analytical form for the forward model, which is then be used to generate a large set of estimated final channel scalars. An inverse 3D LUT is built from dense sampling with the forward model.

Additionally, there was a chromaticity shift in the blue channel at high digital count levels (above approximately 200 on a scale of 255), which results in a deviation from the scalability assumption in the Day et al. approach. These types of per-channel chromaticity changes were addressed in the model using LUTs, but ultimately to achieve better accuracy in the chromatic adaptation experiment, the luminance of the display backlight was raised to a high enough level that the values for RGB channels could be restricted to a maximum of 200 digital counts.

*Forward Model:*

An optimized black-level correction, $XYZ_X$ is first calculated to minimize the standard deviation in the channel ramps, in $u'v'$ space. The initial value for the optimization is the $XYZ$ measured with the display at its lowest output level $DC_{rgb} = (0, 0, 0)$. A set of LUTs are used to convert from the digital counts driving the display to a set of radiometric scalars on the channels:

\[
R = \text{LUT}(DC_r) \\
G = \text{LUT}(DC_g) \\
B = \text{LUT}(DC_b)
\]
The radiometric scalars used to construct the LUTs are calculated using the sum of the XYZ values at each digital count level, relative to the sum of X, Y, and Z at the channel maximum, after black level corrections:

\[
R = \frac{(X_{R,DCr} + Y_{R,DCr} + Z_{R,DCr}) - (X_K + Y_K + Z_K)}{(X_{R,DCr_{Max}} + Y_{R,DCr_{Max}} + Z_{R,DCr_{Max}}) - (X_K + Y_K + Z_K)}
\] (152)

These initial radiometric scalars are directly related to the digital counts controlling an individual channel when that channel alone is active (i.e. in the absence of any interaction with other channels). An interaction model is created to account for the additional effects of cross-channel interaction. Using the interaction model, the R, G, and B radiometric scalars derived from the independent channel ramps are converted to a set of final radiometric scalars on the primaries \( (P_r, P_g, P_b) \), which include the added interaction effect when more than one channel is active at a time. This approach of converting between independent and final scalars was used in the interaction method developed by Woolfe et al. [1997].

The interaction model is based on a set of intermediate variables used to represent the possible combinations of the R, G, and B channels. These intermediate variables \( (N, C, M, Y) \) are the secondary and tertiary combinations of RGB described by Tamura et al. [2003] in their masking model for LCD characterization. In the object display work, the intermediate variables are calculated on RGB after conversion to a radiometrically linear space, while Tamura et al. performed the calculations directly on input digital counts. Additionally, in the object display work, the secondary combinations are not calculated from the residuals of the tertiary combination but are used instead of it if the RGB tertiary combination does not reach a threshold.
value. The neutral variable (N) represents the amount of the RGB combination present, the yellow variable (Y) represents the amount of the RG combination present, the cyan variable (C) represents the GB present, and the magenta variable (M) represents the RB present, calculated by:

\[
N = \begin{cases} \min(R,G,B) & \text{if } \frac{\min(R,G,B)}{\max(R,G,B)} > k_N, \\ 0 & \text{otherwise} \end{cases}
\]

\[
C = \begin{cases} \min(G,B) & \text{if } R = \min(R,G,B) \text{ and } N = 0, \\ 0 & \text{otherwise} \end{cases}
\]

\[
M = \begin{cases} \min(R,B) & \text{if } G = \min(R,G,B) \text{ and } N = 0, \\ 0 & \text{otherwise} \end{cases}
\]

\[
Y = \begin{cases} \min(R,G) & \text{if } B = \min(R,G,B) \text{ and } N = 0, \\ 0 & \text{otherwise} \end{cases}
\]

where \(k_N\) is the ratio threshold for using the neutral variable instead of the pairwise variables (0.9 is the default). The excess R, G, and B, above the amount accounted for by the selected pairwise combination, is calculated by:

\[
R_{ex} = \max(0, R - M - Y - N)
\]

\[
G_{ex} = \max(0, G - C - Y - N)
\]

\[
B_{ex} = \max(0, B - C - M - N)
\]

The interaction model incorporates first and second order terms for the pairwise and tertiary combinations C, M, Y, N, as well as terms for the interaction between the secondary combinations C, M, Y and the excess values \(R_{ex}, G_{ex}, B_{ex}\). Note that the model does not
incorporate interaction terms between \(N\) and the \(R, G, B\) excess terms (as the resulting increase in the number model terms may lead to over-fitting), so the \(k_N\) threshold should be close enough to 1 that the excess from any individual channel is minimal. The square root of the variables \(C, M, Y\) and the excess \(RGB_{ex}\) variables are used to model their interaction:

\[
\begin{bmatrix}
P_r \\
P_g \\
P_b
\end{bmatrix}
= 
\begin{bmatrix}
1 & 0 & 0 & m_{1,4} & \ldots & \ldots & m_{1,39} \\
0 & 1 & 0 & m_{2,4} & \ldots & \ldots & m_{2,39} \\
0 & 0 & 1 & m_{3,4} & \ldots & \ldots & m_{3,39}
\end{bmatrix}
\text{PredVec}
\tag{155}
\]

where:

\[
\begin{bmatrix}
\sqrt{R_{ex}C} & \sqrt{R_{ex}M} & \sqrt{R_{ex}Y} & \sqrt{G_{ex}C} & \sqrt{G_{ex}M} & \sqrt{G_{ex}Y} & \sqrt{B_{ex}C} & \sqrt{B_{ex}M} & \sqrt{B_{ex}Y} & \ldots \\
C\sqrt{R_{ex}} & M\sqrt{R_{ex}} & Y\sqrt{R_{ex}} & C\sqrt{G_{ex}} & M\sqrt{G_{ex}} & Y\sqrt{G_{ex}} & C\sqrt{B_{ex}} & M\sqrt{B_{ex}} & Y\sqrt{B_{ex}} & \ldots \\
R_{ex}C & R_{ex}M & R_{ex}Y & G_{ex}C & G_{ex}M & G_{ex}Y & B_{ex}C & B_{ex}M & B_{ex}Y & 1
\end{bmatrix}^T
\]

These final radiometric scalars are multiplied by the XYZ values of the primaries to estimate the output of the display. Because the chromaticities of the primaries were found to vary over the ramps, they are stored in a set of LUTs that are indexed by the ramp radiometric scalar values (R, G, or B, pre-interaction). Note that \(Y\) is held constant, so even though XYZ values are stored in the LUTs for computational reasons, only the chromaticity coordinates of the primary are actually being varied. The chromaticity ramps are calculated using the chromaticity of the measured XYZ values at each level of the ramp, but are heavily smoothed in the \(u'v'\) space before conversion to XYZ values, using robust LOESS smoothing and a moving average filter, to avoid high frequency changes due to measurement noise. The \(Y\) value used in each ramp is
calculated in a linear regression (with intercept 0) from the Y-values at each ramp level to radiometric scalar value at each level.

\[
XYZ_{R,prim} = \text{LUT}(R) \\
XYZ_{G,prim} = \text{LUT}(G) \quad (156) \\
XYZ_{B,prim} = \text{LUT}(B)
\]

Using the XYZ values for the primaries from the LUTs, the optimized black-level, and the calculated final radiometric scalars, the display output is estimated as:

\[
\begin{bmatrix}
X \\
Y \\
Z
\end{bmatrix} =
\begin{bmatrix}
X_{R,prim} & X_{g,prim} & X_{b,prim} & X_K \\
Y_{R,prim} & Y_{g,prim} & Y_{b,prim} & Y_K \\
Y_{R,prim} & Z_{g,prim} & Z_{b,prim} & Z_K
\end{bmatrix}
\begin{bmatrix}
P_r \\
P_g \\
P_b \\
1
\end{bmatrix}
\quad (157)
\]

**Inverse Model:**

The inverse model converts from target XYZ values to the \(DC_r, DC_g, DC_b\), digital count values used to control the screen. Given a target XYZ, the radiometric scalars that include interaction \((P_r, P_g, P_b)\) are estimated with a matrix inversion:

\[
\begin{bmatrix}
P_r \\
P_g \\
P_b
\end{bmatrix} =
\begin{bmatrix}
X_{R,prim} & X_{g,prim} & X_{b,prim} \\
Y_{R,prim} & Y_{g,prim} & Y_{b,prim} \\
Z_{R,prim} & Z_{g,prim} & Z_{b,prim}
\end{bmatrix}^{-1}
\begin{bmatrix}
X \\
Y \\
Z
\end{bmatrix} - 
\begin{bmatrix}
X_K \\
Y_K \\
Z_K
\end{bmatrix}
\quad (158)
\]
However, because the XYZ values of the primaries \((XYZ_{RGB,\text{prim}})\) are dependent on the ramp level, it is necessary to first obtain an initial estimate of the radiometric scalars. This is found using the standard Day et al. [2004] approach, which has a single optimized primary matrix. A separate 3x3 matrix is maintained for this purpose:

\[
\begin{bmatrix}
R_{\text{EST}} \\
G_{\text{EST}} \\
B_{\text{EST}}
\end{bmatrix} =
\begin{bmatrix}
X_{R,\text{max,opt}} & X_{G,\text{max,opt}} & X_{B,\text{max,opt}} \\
Y_{R,\text{max,opt}} & Y_{G,\text{max,opt}} & Y_{B,\text{max,opt}} \\
Z_{R,\text{max,opt}} & Z_{G,\text{max,opt}} & Z_{B,\text{max,opt}}
\end{bmatrix}^{-1}
\begin{bmatrix}
X \\
Y \\
Z
\end{bmatrix}
\]

(159)

Using these estimates, the XYZ values for the three primaries are determined from the XYZ LUTs:

\[
XYZ_{R,\text{prim}} = \text{LUT}_{XYZ}(R_{\text{EST}})
\]

\[
XYZ_{G,\text{prim}} = \text{LUT}_{XYZ}(G_{\text{EST}})
\]

\[
XYZ_{B,\text{prim}} = \text{LUT}_{XYZ}(B_{\text{EST}})
\]

Note that because the chromaticity ramps were heavily smoothed, it is only necessary that the estimates \(R_{\text{EST}} G_{\text{EST}} B_{\text{EST}}\) are in the same approximate region of the curve as the true radiometric scalars \(R, G, B\) to provide an acceptable level of accuracy on the primary chromaticities.

In testing, it was found that in subsequent uses of the screen after calibration, there were small variations in the absolute magnitude of the channel primaries relative to one another. While the chromaticity for each primary at a given drive level was relatively stable, the white point exhibited chromaticity shifting. Though the luminance changes were generally small in magnitude (<1% of the target), in some cases channels varied in opposite directions (e.g. the blue channel increased, the red channel decreased) leading to the chromaticity shift for a neutral
A drift-correction procedure was adopted to address this effect without requiring a full recalibration of the display before every usage:

\[
\begin{align*}
XYZ_{R,\text{prim}} &= C_R \cdot \text{LUT}_{XYZ}(R_{\text{EST}}) \\
XYZ_{G,\text{prim}} &= C_G \cdot \text{LUT}_{XYZ}(G_{\text{EST}}) \\
XYZ_{B,\text{prim}} &= C_B \cdot \text{LUT}_{XYZ}(B_{\text{EST}})
\end{align*}
\]

where \(C_R\), \(C_G\), and \(C_B\) are scaling constants for the red, green, and blue primaries that are multiplied by the XYZ values stored in the LUTs.

The matrix of primaries, selected from the LUTs, is inverted and multiplied by the target XYZ (after black level correction) to calculate the radiometric scalars that include interaction:

\[
\begin{bmatrix}
P_r \\
P_g \\
P_b
\end{bmatrix} =
\begin{bmatrix}
X_{R,\text{prim}} & X_{g,\text{prim}} & X_{b,\text{prim}}
\end{bmatrix}^{-1}
\begin{bmatrix}
X \\
Y \\
Z
\end{bmatrix} -
\begin{bmatrix}
X \\
Y \\
Z
\end{bmatrix}
\]

Due to the complexity of inverting the forward model for interaction, the amount of interaction present is estimated using a 3D lookup table:

\[
\begin{align*}
\Delta R_{\text{int}} &= \text{LUT3D}(P_r, P_g, P_b) \\
\Delta G_{\text{int}} &= \text{LUT3D}(P_r, P_g, P_b) \\
\Delta B_{\text{int}} &= \text{LUT3D}(P_r, P_g, P_b)
\end{align*}
\]

To construct the 3D LUT, a 3D grid with 128 x 128 x 128 levels of independent \(R\), \(G\), \(B\) scalars is processed with the forward model. This produces a \(P_r\), \(P_g\), \(P_b\) estimate for every value in the \(R\), \(G\), \(B\) grid. At each grid location, the \(R\), \(G\), \(B\) input value is subtracted from the \(P_r\), \(P_g\), \(P_b\)
results to provide estimates of $\Delta R_{\text{int}}, \Delta G_{\text{int}}, \Delta B_{\text{int}}$ at each grid location. The $P_r, P_g, P_b$ output values are rescaled to [0 1] by dividing each channel by its maximum allowable value (determined in the forward model optimization) and clipping any negative values to 0. These $P_r, P_g, P_b$ output values are in a regular grid for R, G, B, but not for $P_r, P_g, P_b$. Therefore, it is necessary to perform interpolation on this scattered $P_r, P_g, P_b$ data to build the final 3D LUT relating $P_r, P_g, P_b$ values to $\Delta R_{\text{int}}, \Delta G_{\text{int}}, \Delta B_{\text{int}}$ values. The linear interpolation was performed in Matlab(‘TriScatteredInterp’) using an interpolant based on Delaunay triangulation to create a regular 128 x 128 x 128 grid in terms of $(P_r, P_g, P_b)$. All not-a-number elements in the LUT were replaced with the $\Delta R_{\text{int}}, \Delta G_{\text{int}}, \Delta B_{\text{int}}$ values from the nearest point $(P_r, P_g, P_b)$ that contained valid numbers.

Using the $\Delta R_{\text{int}}, \Delta G_{\text{int}}, \Delta B_{\text{int}}$ estimates from the 3D LUT, the radiometric scalars on the independent channels are calculated by subtracting the estimated amount of interaction per channel:

$$R = P_r - \Delta R_{\text{int}}$$
$$G = P_g - \Delta G_{\text{int}}$$
$$B = P_b - \Delta B_{\text{int}}$$

(164)

The digital counts for controlling the display are determined from these independent channel R, G, B radiometric scalars using the inverse lookup tables for the channel ramps:

$$DC_r = \text{LUT}^{-1}(R)$$
$$DC_g = \text{LUT}^{-1}(G)$$
$$DC_b = \text{LUT}^{-1}(B)$$

(165)
The video card only supports 8-bit per channel output to the display screen, so the digital count values \((DC_r, DC_g, DC_b)\), though calculated as floating point values, are rounded to 256 levels when sent to the display screen. This rounding leads to quantization error when producing a target XYZ value, in particular if the digital count value necessary to produce the target is approximately halfway between the possible levels (shown in Figure A1). The quantization effect is reduced by adding a uniform random variable with a mean of 0 and range \([-0.5 \text{ DC level}, +0.5 \text{ DC level}]\) to the floating point DC values calculated. The random variable varies both spatially, with a different value for each pixel, and temporally so that the pixel noise values vary on a frame by frame basis.

**Figure A1.** Quantization in the display output for floating point control values between 127.5 and 129.5 digital counts. Raw measured data is shown in the left panel and the data averaged over three replications is shown in the right panel. The results for standard display output are shown in red and the results with random noise added to the control values are shown in blue.
Model Results

The CIEDE2000 color error for one of the characterizations performed on the EIZO Radiforce RX220 for use in the chromatic adaptation experiment is shown in Figure A2 (for the first characterization) and A3 (for the second characterization). The CIEDE2000 values were calculated by comparing the predictions of the forward model for sets of input \((DC_r, DC_g, DC_b)\) values to the actual measured XYZ values with the PhotoResearch PR-655 (averaged over three replications in the first dataset and four measurement replications in the second dataset). The color error for a characterization with the Day et al. approach is represented by red dotted lines and for the enhanced model by the blue dotted lines. The results are shown for the primary ramps (red, green, blue), the gray ramp, the three pairwise combination ramps (RG, RG, GB), and a 5-step factorial. Additionally, because many of the samples in the chromatic adaptation experiment were expected to be near the tungsten background \((u'v'=[.5316, .2681])\), additional ramps were measured in that region of the color space. A set of six ramps each at a \(u'v'\) chromaticity in the range from \((0.2405, 0.5090)\) to \((0.2716, 0.534)\) were measured (shown as “Ramps Near Tungsten” in the plots). A ramp at the \(u'v'=(.5316, 2681)\) chromaticity of the background was also measured (“Tungsten Ramp”). Finally, two sets consisting of 100 randomly generated RGB combinations were measured. The first 100 samples were included as training data in the characterization. The second 100 samples were not included in the characterization and were considered as testing data.
Figure A2. Comparison of color error between the standard Day et al. approach and the interaction model on the ramp, factorial, and random data used in characterization and for 100 random test samples not used during the model fitting. (Experiment characterization data, week 1)
Figure A3. Comparison of color error between the standard Day et al. approach and the interaction model on the ramp, factorial, and random data used in characterization and for 100 random test samples not used during the model fitting. (Experiment characterization data, week 2)
In addition to evaluating the color difference error for the enhanced model, the ability of the enhanced model to account for the channel interaction was evaluated. The excess XYZ resulting from channel interaction (after black-level correction) was estimated by subtracting the sum of the independent channel measurements from the measured XYZ from pairwise or tertiary combinations of the channels, for channel values (R=Rc, G=Gc, B=Bc):

\[
\Delta XYZ = \left( XYZ_{(R=Rc, G=Gc, B=Bc)} - XYZ_K \right)
- \left( XYZ_{(R=Rc, G=0, B=0)} - XYZ_K \right)
- \left( XYZ_{(R=0, G=Gc, B=0)} - XYZ_K \right)
- \left( XYZ_{(R=0, G=0, B=Bc)} - XYZ_K \right).
\]

(166)

The excess XYZ results are shown in Figures A4 and A5 for the measured combinations (dots) and for the model predictions of the combination XYZ values (solid lines). Results are shown for the gray ramp, the pairwise combination ramps (RG, RB, GB), and the ramp for the tungsten background chromaticity.
Figure A4. Modeling results for the interaction between channels on the gray ramp, RG (yellow) ramp, RB (magenta) ramp, GB (cyan) ramp, and the tungsten chromaticity ramp. The ΔXYZ values represent the excess measured XYZ when multiple channels are active, above the sum of the individual channels for the same control levels. The dotted lines represent the measured values for the excess. The solid lines represent the model predictions. (Experiment characterization data, week 1)
**Figure A5.** Modeling results for the interaction between channels on the gray ramp, RG (yellow) ramp, RB (magenta) ramp, GB (cyan) ramp, and the tungsten chromaticity ramp. The ΔXYZ values represent the excess measured XYZ when multiple channels are active, above the sum of the individual channels for the same control levels. The dotted lines represent the measured values for the excess. The solid lines represent the model predictions. (Experiment characterization data, week 2)
Appendix B. GPU-based Method for Constructing Horizon Maps

The shadowing in the rendering engine is calculated using a set of horizon maps [Max 1988, Sloan & Cohen 2000] that store information on the minimum elevation angle above the surface, at each surface position, required so that a light source will be not occluded by an intervening surface point. The construction of the horizon maps from the height map is a computationally intensive process. For each pixel of the height map, it is necessary to determine the angle from the starting pixel’s surface height to the surface height of every other pixel on the surface. To support the object display system, software for generating horizon maps was developed that could be run on the GPU so that its parallel processing capabilities could be used to generate the maps in a reduced amount of time. The algorithm implemented to construct the shadow maps for the object display uses the following steps:

0. Set the initial elevation angle value ($\theta_{thresh}$) to 0° for every pixel in the set of horizon maps. (For computational reasons, the tangent of the elevation angle is used in the algorithm, so tan($\theta_{thresh}$) = 0 is stored in the maps.)

1. Select a starting pixel, represent its surface position as $(u_0,v_0)$ and height as $w_0$.

2. Iterate over every other pixel in the height map, with surface positions specified by $(u_i,v_i)$ and heights specified by $w_i$:

   a. Compare $w_0$ to $w_i$. If the height $w_i$ is shorter than $w_0$, then the test pixel will not block the light on the starting pixel and no further testing is needed in this iteration of step 2.
b. Determine the azimuthal direction $\phi$ from the starting pixel to the current test pixel by calculating the angle from surface position $(u_0, v_0)$ to $(u_i, v_i)$:

$$\phi = \tan^{-1}\left(\frac{v_i - v_0}{u_i - u_0}\right)$$  \hspace{1cm} (167)

c. Select the map whose 20° range of azimuth angles contains the calculated $\phi$ angle, and look up the stored elevation angle threshold value $\tan(\theta_{\text{thresh}})$ for this selected azimuth angle range.

d. Calculate the elevation $\tan(\theta)$ value from the starting pixel to test pixel:

$$\tan(\theta) = \frac{w_i - w_0}{\sqrt{(u_i - u_0)^2 + (v_i - v_0)^2}}$$  \hspace{1cm} (168)

e. If the calculated elevation $\tan(\theta)$ value is larger than the $\tan(\theta_{\text{thresh}})$ that is currently stored in the map, update the map with the value of $\tan(\theta)$.

f. Repeat step 2 until every other pixel is tested against the given starting pixel.

3. Return to step 1 and repeat the process for a new starting pixel until a value is determined for every map pixel.

To perform the calculation for every starting pixel and check every possible test pixel, there will be $n^2$ pixel height comparisons (where $n$ is the number of pixels in the height map). For example, a height map at an HD display resolution of 1920 x 1080, with approximately 2 million pixels, would involve $4 \times 10^{12}$ comparisons. Using the parallel processing capabilities of a high-end video card GPU (NVIDIA GTX 670, 1344 shader cores), it was possible to perform this processing on typical height maps in under 1 hour.
Appendix C. Method for Estimating 3D Viewpoint on Tangible Displays using Camera-based Head-tracking

In the tangible display systems [Darling & Ferwerda 2009, 2010], the first step in estimating the 3D world coordinates of the eye-point from the camera data is to determine the camera’s position in $xyz$. The physical distance from the origin point to the camera, $d_{\text{cam}}$, is measured, and the camera is placed on the $v$-axis (Figure C1), allowing the camera position ($p_{\text{cam}}$) to be found by:

$$p_{\text{cam}} = d_{\text{cam}} v.$$  \hfill (169)

Starting from the camera position, the eye-point position in 3D space is determined from the camera tracking data using an ideal pinhole camera model [Klette et al. 1998]. The distance along the camera’s principal ray to the perpendicular plane containing the eye-point is estimated from the size of the viewer’s head radius in the camera image, $d_{\text{rad,pix}}$. Using the effective focal length of the camera in image pixels ($f_{\text{pix}}$), calculated from calibration data, and an estimate of the physical vertical radius of the viewer’s head ($d_{\text{rad,cm}}$), the distance along the principal ray to the plane ($d_{\text{plane,cm}}$) is determined from the head size in the image using the pinhole camera equation:

$$d_{\text{plane,cm}} = d_{\text{rad,cm}} \left( f_{\text{pix}} / d_{\text{rad,pix}} \right).$$  \hfill (170)

The position in world coordinates where the principal ray intersects the plane containing the eye-point is found by moving along the principal ray direction (which corresponds to the $w$-axis) a distance $d_{\text{plane,cm}}$ from the camera:

$$p_{\text{pr,plane}} = p_{\text{cam}} + d_{\text{plane,cm}} w.$$  \hfill (171)
In the final step, the position of the eye-point in the camera image is used to determine the eye-point position in the world. The up-vector of the camera coincides with the direction of \( \mathbf{v} \), and is the physical direction corresponding to a vertical offset from center in the camera image. The \( \mathbf{u} \) vector is the physical direction corresponding to a horizontal offset from center in the camera image. The eye-point offsets in image pixels are scaled to physical units using the pinhole camera equation and are applied as physical offsets from the principal ray-plane intersection point, giving a final viewing position, \( \mathbf{p}_{\text{view}} \), of:

\[
\mathbf{p}_{\text{view}} = \mathbf{p}_{\text{pr,plane}} + d_{\text{plane,cm}} \left( \left( \frac{x_{\text{img,eye}} - x_{\text{img,ctr}}}{f \text{pix}} \right) \mathbf{u} + d_{\text{plane,cm}} \left( \frac{y_{\text{img,eye}} - y_{\text{img,ctr}}}{f \text{pix}} \right) \mathbf{v} \right).
\] (172)
Appendix D. Computational Illumination for an Object Display

Overview and Motivation

A principal objective of the object display framework is to render virtual objects to the display screen to match the real physical lighting present in the environment. To expand on this concept, a complementary projector-based lighting framework has been developed for generating lighting patterns onto the physical surfaces surrounding the display. In this way, it is possible to have computer-based control over both the real-world lighting and the virtual surface on the display screen. Additionally, it provides a computational solution for a potential physical limitation of object display systems that arises when small, high luminance light sources are needed to emphasize virtual surface gloss. Though it is possible to find display screens with very low diffuse surface reflectance, the specular reflectance of screens is typically higher and can result in a significant amount of undesired physical front surface reflection for these types of high luminance sources. A projector-based source allows the light pattern to be modulated to avoid impacting the screen surface, while still lighting the real-world surfaces around the screen.

The implementation of the projector-lighting system is similar to the Shader Lamps system [Raskar et al. 2001], but is used for the opposite purpose. With shader lamps, light is projected onto a blank real-world object to give it the appearance of material properties, and generally, this is only the object being lit in an otherwise dark environment. In the object display case, the projector is used specifically to avoid projecting light onto the element producing the virtual surface. It is instead used to light all the other surfaces around it, to give the environment the appearance that it has ambient illumination.
Though the computational lighting implementation in this work is based on two-dimensional projector lighting, ideally, a four-dimensional programmable light source, capable of both spatial and directional light modulation, would be used to provide the real-world illumination. The 2D projector-lighting capability increases the flexibility of the system, but it does lead to a conceptual departure, to some degree, from the pure objectives of the object display framework. Instead of strictly maintaining photometric consistency with the real-world light sources present, the object display output instead is maintaining consistency with the lighting implied by other physical surfaces that are visible around the screen. With the projector-based lighting, the virtual surface rendered to the display screen has lighting that is consistent with any diffuse physical surfaces surrounding it, but not strictly consistent with the actual 6D light field created by the real light source (the projector) present.

The principle benefit of the projector-based lighting framework is that it is possible to create the desired patterns of light on the physical surfaces surrounding the screen, but minimize the physical light directly shining on the object display screen surface (as shown in Figure D1). This allows for the use of small high intensity light sources in the virtual surface rendering without creating unwanted physical reflections on the real screen surface that have too high of an intensity to mitigate. These types of specular sources are important for emphasizing the gloss properties of the rendered virtual surface on the object display, but correspondingly, also have the unwanted effect of emphasizing any real gloss on the front surface of the display screen.

The diffuse reflection from the front surface of the display screen used for the prototype is minimal (approximately 0.2%) and can be modeled and corrected, but specular reflection from the front surface of the screen is more difficult to mitigate. The diffuse portion of reflectance acts like a low-pass filter on the incident light sources, so that the resulting light
pattern has only gradual changes in intensity across the surface of the screen. The diffuse reflections are relatively uniform across the screen and without high intensity peaks. In contrast, the specular portion retains high frequency changes in the intensity of light sources. Modeling and subtracting these specular reflections is not practical for two reasons. First, because they are specular, small changes in viewing direction will cause the patterns to move across the screen, requiring perfect tracking of the viewer to avoid any ghosting artifacts or misalignment. Additionally, the luminance of these reflections, when dealing with high luminance sources (or portions of sources, such as a bulb filament) will often be too high in intensity to subtract from the calculated screen output for the virtual surface.

Figure D1. In the left panel luminance image, the booth is illuminated with a real light bulb placed in front of the screen, which produces a large front surface reflection. In the right panel, a projector is used to simulate the light from the bulb so that the screen region can be masked out to reduce the physical reflection.

A current limitation of the projector-based workflow is that it uses off-line rendering (ray-tracing) to calculate the illumination patterns and projects them on the real-world surfaces based on an off-line geometric registration. When using the current workflow, the screen must stay in a fixed position for the masked projector region (used to reduce the unwanted screen
reflection) to remain aligned with the screen location. In future work, an interactive workflow could potentially be developed and combined with tracking information on the screen location to overcome this limitation.

**System Design**

The computational lighting framework uses projector-based illumination to simulate real-world light sources and recreate their diffuse patterns of reflection on the real world surfaces surround the display. Though the output modality differs, the underlying representations are relatively similar to the object display framework. In particular, it uses the same input path for captured light sources as the object display framework, so that the light source data used to light the virtual surfaces on the display screen can be used directly for projector-based illumination of the real physical surfaces surrounding the screen.

The computational lighting framework uses a 3D geometric model of the surround surfaces, spatial luminance and geometric information about the light source, and a radiometric model of the projector light transport to recreate the set of patterns for a light source using the projector. The 3D model for the surround surfaces is rendered using a physically-based ray-tracer to estimate the resulting illumination patterns on each surface. A simple geometric calibration is performed to map each real-world surface plane to the corresponding pixels on the image plane of the projector. An inverse radiometric model of the projector light transport is used to calculate the luminant intensity of each pixel required to produce the desired illuminance at the surface positions.
**Geometric Surface Modeling**

Each physical surface in the real environment is represented with a parameterized plane. The origin of the plane is specified by the XYZ position (in cm) of the top left corner. The orientation of the plane is specified by a set of three unit vectors: the surface normal, a vector \( \mathbf{u} \) representing the left-to-right direction in the plane, and a \( \mathbf{v} \) vector representing the top-to-bottom direction of the plane. The size of the surface is specified by two scalars, \( l_u \) and \( l_v \), specified in cm. The \( \mathbf{u} \) and \( \mathbf{v} \) directions serve as the axes for an image-based parameterization of the planar surface. Using this parameterization, image pixel coordinates \((s, t)\) can be easily mapped to a physical position in space, \( \mathbf{p} \), based on the formula:

\[
\mathbf{p} = \mathbf{o}_{TL} + l_u \left( \frac{s}{s_{\text{max}}} \right) \mathbf{u} + l_v \left( \frac{t}{t_{\text{max}}} \right) \mathbf{v}.
\]  

(173)

**Modeling a Target Light Source**

The projection lighting process starts with the same captured spatial luminance maps of light sources that are used in the object display framework. These luminance images are mapped to a physical planar surface in 3D space using the parameterization described above, and so they occupy a physical location with respect to the object display screen and the surrounding surface model. Using the variance-preserving median-cut algorithm [Viriyothai & Debevec 2009] spatial light maps are converted to a set of light-cut points that store summed luminance values at physical XYZ positions in space.
The illumination on each surround surface that results from the simulated light source is estimated in a multistage process using a physically-based ray tracer. The ray-tracing is performed using PBRT v2.0 [Pharr & Humphreys 2010]. The overall processing is controlled by a Matlab script that automatically generates the PBRT code necessary to render the different stages of the simulation and ultimately form the geometrically warped, characterized image for the projector. Each rectangular surface of the environment is represented in PBRT as a triangular mesh with the material properties of a white matte material (Kd = [1 1 1]). The scene is rendered multiple times, once for each surface. For each rendering, the PBRT camera is placed perpendicular to one of the surfaces and a corresponding field of view is set that will encompass only that surface. The result is a set of (unprojected) rectangular textures that can later be mapped to the planes in the scene.

During rendering, the median cut points of the illumination source are represented as a set of modified PBRT spotlights. In PBRT, spotlights have an associated 3D position in space, a principal lighting direction specified by a “LookAt” parameter, a “coneangle” parameter specifying the angular extent (from the principal axis) in which there is uniform illumination, and a “conedeltaangle” parameter that specifies the starting angle (from the principal axis) at which a smooth falloff in intensity begins [Pharr & Humphreys 2010]. To simulate median cut lighting, the 3D position of the spotlight is set the 3D coordinates of the median cut point, and the principal axis direction is set to the normal direction of the area light plane used to generate the median cut points. To simulate the light distribution of a median cut light that represents a portion of a diffuse area source (as used in the object display framework), a modification was
made to the default implementation of the PBRT spotlight. It was changed to incorporate a

cosine falloff, as a function of the angle from the principal axis, within the central “coneangle”

region that is typically uniform in intensity. With this modification, the cone angle is then set to

approximately 90° (89.999°) so that the falloff occurs over the hemisphere surrounding the

principal axis. The cone delta start angle is set to nearly 90°, so that the default intensity falloff

behavior is not used. The cosine falloff term is used to simulate the reduction in illuminance

received at a surface point that is not perpendicular to the approximated area light at the position

of the median cut point. This cosine term corresponds to the \( \mathbf{N}_{\text{light}} \cdot -\mathbf{I}_n \) factor (the dot product

of the area light normal and light-to-surface vector) in the illuminance calculation performed in

the diffuse lighting portion of the object display framework:

\[
E = \sum_{n=1}^{32} \frac{L_n \left( \mathbf{N}_{\text{surf}} \cdot \mathbf{I}_n \right) \left( \mathbf{N}_{\text{light}} \cdot -\mathbf{I}_n \right) \left( A \right) \left( d_n^2 \right) \left( P \right)}{d_n^2}.
\]  

(174)

**Calculating the Luminous Intensity Image for the Projector**

The output produced by the rendering software is a spatial luminance pattern for each of

the surround surfaces, due to the simulated light source. Working backwards from the rendering

simulation output, the objective is to calculate the luminous intensity image for the projector

needed to recreate these lighting patterns. The luminance from the simulation is first used to
calculate the incident illuminance on the surfaces, which is then used to estimate the intensity
image for the projector. Because each surface is at a different orientation and distance with
respect to the projector, an inverse model of the radiometric transport from the projector to
surface is used in the step converting from illuminance patterns to the projector luminous intensity.

In the rendering simulation, the surround surfaces were set as perfect reflecting diffusers so that the simulation output could be used to estimate the incident illumination on the surfaces. For a Lambertian surface, the luminous exitance can be calculated from the outgoing luminance by applying a factor of $\pi$ (the $\pi$ factor used to relate radiant exitance and radiance for Lambertian surfaces, as described in the radiometry text by Wolfe [1998]):

$$M_v = \pi L_v$$  \hspace{1cm} (175)

where $M_v$ is the luminous exitance and $L_v$ is the luminance. For a perfect reflecting diffuser, the luminous exitance $M_v$ from the surface will be equal in magnitude to the illuminance $E_v$ incident upon it. The illuminance $E_v$ on a surface varies with the square of the distance between the source and surface. Additionally, the illuminance at the surface is dependent on the angle ($\theta$) between the source-to-surface vector and the surface normal. Accounting for these factors in the illuminance calculation (in the manner used with the radiometric quantities of irradiance and radiant intensity, as given in the text by Palmer & Grant [2009]), the formula used to approximate the surface illuminance from the luminous intensity of the projector is:

$$E_v = \frac{I_v \cos(\theta)}{r^2}$$  \hspace{1cm} (176)

where $r$ is the distance in meters and $I_v$ is the luminous intensity. Solving this equation for the luminous intensity needed to produce a target illuminance at the surface gives:

$$I_v = \frac{E_v r^2}{\cos(\theta)}$$  \hspace{1cm} (177)
In point-vector notation, the complete formula for calculating the projector luminous intensity, $I_v$, from the luminance, $L_v$, of the rendered image is:

$$I_v = \frac{\pi L_v \| \mathbf{p}_{proj} - \mathbf{p}_{surf} \|^2}{\| \mathbf{p}_{proj} - \mathbf{p}_{surf} \| \cdot \mathbf{n}_{surf}}$$  \hspace{1cm} (178)$$

where $\mathbf{p}_{proj}$ is the 3D physical position of the projector (in meters), $\mathbf{p}_{surf}$ is the 3D position on the surface, and $\mathbf{n}_{surf}$ is the surface normal.

**Projector Characterization in Luminous Intensity**

The projector was colorimetrically characterized to convert between the digital counts controlling the projector and colorimetric output on a luminous intensity scale (Y in units of candelas). The Day et al. approach [2004] was used for characterizing the projector (Optoma HD66). The HD66 is a four-channel DLP projector with a white channel in addition to the RGB channels, but was set to a cinema mode (where only the RGB channels are used) to allow for use of the three-channel Day et al. approach. To characterize the projector output on a luminous intensity scale, the set of RGB ramp, gray ramp, and factorial measurements were taken using a spectroradiometer (PhotoResearch PR655) of the light reflected from a white fluorilon patch. The patch position and projector position were measured to provide the necessary information on the measurement geometry and this was used to convert from the reflected luminance measured by the spectroradiometer to luminous intensity using Eq. (178).
Appendix E. Six-Channel Primary Spectral Curves (10 nm)

<table>
<thead>
<tr>
<th>Wavelength (nm)</th>
<th>Ch1</th>
<th>Ch2</th>
<th>Ch3</th>
<th>Ch4</th>
<th>Ch5</th>
<th>Ch6</th>
</tr>
</thead>
<tbody>
<tr>
<td>380</td>
<td>0.0001</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>390</td>
<td>0.0008</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>400</td>
<td>0.0050</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>410</td>
<td>0.0216</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>420</td>
<td>0.0661</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>430</td>
<td>0.1425</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>440</td>
<td>0.2166</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>450</td>
<td>0.2321</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>460</td>
<td>0.1754</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>470</td>
<td>0.0934</td>
<td>0.0229</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>480</td>
<td>0.0351</td>
<td>0.8522</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>490</td>
<td>0.0093</td>
<td>0.1248</td>
<td>0.0004</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>500</td>
<td>0.0017</td>
<td>0.0001</td>
<td>0.0215</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>510</td>
<td>0.0002</td>
<td>0.0000</td>
<td>0.2399</td>
<td>0.0008</td>
<td>0.0001</td>
<td>0.0000</td>
</tr>
<tr>
<td>520</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.5121</td>
<td>0.0201</td>
<td>0.0008</td>
<td>0.0000</td>
</tr>
<tr>
<td>530</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.2095</td>
<td>0.1597</td>
<td>0.0056</td>
<td>0.0000</td>
</tr>
<tr>
<td>540</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0164</td>
<td>0.4037</td>
<td>0.0271</td>
<td>0.0000</td>
</tr>
<tr>
<td>550</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0002</td>
<td>0.3252</td>
<td>0.0864</td>
<td>0.0001</td>
</tr>
<tr>
<td>560</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0835</td>
<td>0.1819</td>
<td>0.0005</td>
</tr>
<tr>
<td>570</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0068</td>
<td>0.2527</td>
<td>0.0032</td>
</tr>
<tr>
<td>580</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0002</td>
<td>0.2317</td>
<td>0.0138</td>
</tr>
<tr>
<td>590</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.1402</td>
<td>0.0438</td>
</tr>
<tr>
<td>600</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0560</td>
<td>0.1022</td>
</tr>
<tr>
<td>610</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0147</td>
<td>0.1751</td>
</tr>
<tr>
<td>620</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0026</td>
<td>0.2201</td>
</tr>
<tr>
<td>630</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0003</td>
<td>0.2029</td>
</tr>
<tr>
<td>640</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.1373</td>
</tr>
<tr>
<td>650</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0681</td>
</tr>
<tr>
<td>660</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0248</td>
</tr>
<tr>
<td>670</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0066</td>
</tr>
<tr>
<td>680</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0013</td>
</tr>
<tr>
<td>690</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0002</td>
</tr>
<tr>
<td>700</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>710</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>720</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>730</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
</tbody>
</table>
Appendix F. Approximating the Distance Squared Metric by the Area

In the error metric for filtered importance sampling, it is necessary to estimate the sum of the squared distances, \( d_j \), from each pixel in an image region to the region center:

\[
\text{Distance Metric} = \sum_{j=1}^{M} d_j^2
\]  

(179)

For the real-time rendering process, it is not feasible to directly calculate the distance metric by iterating over all pixels in each region. Instead, an approximation based on the area of the region is necessary. If certain shapes are assumed, such as a square, the distance metric can be estimated.

For a square region with area \( A \) and side length \( s (s = A^{1/2}) \), the distance metric can be calculated with a double integral over all the squared distances from the center of the square. This is simplified by placing the center of the square at the origin, with edges of length \( s \) extending from positions \(-s/2\) to \( s/2\), so that no subtraction of the center is necessary in the distance formula:

\[
\text{Distance Metric} \approx \int_{-s/2}^{s/2} \int_{-s/2}^{s/2} \left( \sqrt{x^2 + y^2} \right)^2 \, dx \, dy
\]  

(180)
Solving the integral:

\[
\begin{align*}
\int_{-s/2}^{s/2} \int_{-s/2}^{s/2} \left( \sqrt{x^2 + y^2} \right)^2 \, dx \, dy
&= \int_{-s/2}^{s/2} \int_{-s/2}^{s/2} \left( x^2 + y^2 \right) \, dx \, dy \\
&= \int_{-s/2}^{s/2} \left[ \int_{-s/2}^{s/2} \left( x^2 + y^2 \right) \, dy \right] \, dx \\
&= \int_{-s/2}^{s/2} \left[ x^2y + \frac{y^3}{3} \right]_{-s/2}^{s/2} \, dx \\
&= \int_{-s/2}^{s/2} \left( x^2 \frac{s}{2} + \frac{s}{2} \right)^3 - \left( x^2 \frac{-s}{2} + \frac{-s}{2} \right)^3 \, dx \\
&= \int_{-s/2}^{s/2} \left( x^2 s + \frac{s^3}{12} \right) \, dx \\
&= \left[ \frac{x^3 s}{3} + \frac{s^3 x}{12} \right]_{-s/2}^{s/2} \\
&= \left[ \frac{x^3 s}{3} + \frac{s^3 x}{12} \right]_{-s/2}^{s/2} \\
&= \left( \frac{(s/2)^3 s}{3} + \frac{(s/2)^3}{12} \right) - \left( \frac{(-s/2)^3 s}{3} + \frac{(-s/2)^3}{12} \right) \\
&= \left( \frac{s^4}{24} + \frac{s^4}{24} \right) - \left( \frac{-s^4}{24} + \frac{s^4}{24} \right) \\
&= \frac{s^4}{6}
\end{align*}
\]

Substituting the square root of the area \(A\) for the side length \(s\) yields:

\[
\text{Distance Metric} = \frac{(A^{1/2})^4}{6} = \frac{A^2}{6}.
\]
Appendix G. Color Measurement Experiment CIELAB Data Tables
Table G1. D65 Light Booth Condition
Number
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24

Patch,Name
dark,skin
light,skin
blue,sky
foliage
blue,flower
bluish,green
orange
purplish,blue
moderate,red
purple
yellow,green
orange,yellow
blue
green
red
yellow
magenta
cyan
white
neutral,8
neutral,6.5
neutral,5
neutral,3.5
black

Full,Spectral,Calculation
L*
a*
b*
36.9
12.1
13.3
65.4
13.1
17.3
50.3
0.3
D22.1
43.9
D16.4
21.9
55.6
11.2
D24.9
70.8
D30.6
3.2
60.3
28.3
55.6
40.6
17.2
D43.6
50.2
42.4
12.6
29.5
22.4
D23.7
71.9
D26.5
58.8
71.3
12.2
66.2
29.5
25.6
D51.6
55.8
D39.4
36.6
39.3
49.5
23.0
81.5
D3.9
78.6
50.1
48.3
D18.0
50.5
D20.3
D24.6
95.4
D0.4
1.0
80.4
D0.7
0.6
65.4
0.2
D0.3
50.9
D0.7
D0.2
35.0
D0.5
D0.2
19.8
0.1
D0.2

L*
36.9
65.4
50.2
43.9
55.7
70.7
60.5
40.6
50.5
29.5
71.9
71.5
29.7
55.6
40.6
81.6
50.7
50.5
95.4
80.4
65.4
50.9
35.0
19.8

Multispectral,Calculation
a*
b*
DE00
11.8
13.6
0.4
12.8
17.4
0.3
0.0
D22.1
0.3
D16.2
21.5
0.2
11.2
D24.5
0.3
D30.0
2.7
0.4
27.2
56.0
0.8
16.8
D43.2
0.2
42.6
13.2
0.4
22.2
D22.8
0.5
D26.2
57.9
0.2
12.5
66.4
0.2
25.3
D51.2
0.2
D39.5
35.8
0.4
51.9
25.6
1.7
D3.4
78.1
0.3
49.1
D16.3
1.1
D20.2
D24.9
0.2
D0.1
1.3
0.6
D0.3
0.8
0.5
0.5
D0.2
0.4
D0.4
D0.1
0.3
D0.3
D0.1
0.3
0.3
D0.2
0.2

L*
37.0
65.4
50.3
43.7
55.9
70.8
60.3
40.8
50.4
29.6
71.8
71.3
29.7
55.2
40.4
81.1
50.9
50.8
95.4
80.8
65.8
51.2
35.0
19.9

Object,Display,Screen
a*
b*
12.1
13.5
13.0
17.6
0.1
D21.1
D15.4
21.3
11.1
D23.6
D29.6
3.5
27.3
55.7
16.4
D42.0
42.3
13.1
21.9
D22.1
D24.9
57.6
13.2
66.4
24.6
D50.0
D38.5
35.6
51.4
25.3
D2.3
76.5
49.0
D16.1
D19.6
D23.3
0.4
1.5
0.3
1.3
1.0
0.4
D0.1
0.5
0.0
0.2
0.3
0.3

DE00
0.2
0.2
0.5
0.6
0.8
0.5
0.6
0.5
0.4
0.8
0.7
0.6
0.5
0.7
1.4
1.1
1.3
0.7
1.4
1.6
1.3
1.0
0.9
0.6

L*
37.2
65.7
50.1
43.7
55.7
70.2
61.1
40.6
51.2
29.8
71.7
72.0
29.7
55.2
41.6
81.8
51.5
50.0
95.4
80.4
65.4
50.9
35.0
19.9

Multispectral,Calculation
a*
b*
DE00
12.3
13.9
0.3
13.9
17.7
0.3
D0.1
D22.4
0.4
D16.7
20.9
0.2
11.5
D24.5
0.2
D31.1
1.7
0.4
27.9
56.5
0.6
16.5
D43.2
0.2
44.5
14.4
0.2
22.3
D22.2
0.5
D27.2
56.7
0.2
12.7
66.3
0.1
24.9
D51.1
0.2
D41.1
34.5
0.3
54.1
27.1
1.5
D3.4
77.2
0.3
50.7
D14.9
1.0
D20.3
D25.8
0.3
0.0
1.1
0.4
D0.4
0.6
0.4
0.4
D0.3
0.3
D0.5
D0.2
0.3
D0.4
D0.2
0.2
0.2
D0.2
0.2

L*
37.3
65.8
50.3
43.7
56.1
70.4
60.9
40.7
51.3
29.9
71.6
71.5
29.7
55.0
41.4
81.6
51.6
50.0
95.3
80.6
65.5
51.0
35.1
19.8

Object,Display,Screen
a*
b*
12.5
13.9
14.0
17.7
D0.1
D21.6
D15.8
20.9
11.3
D23.4
D30.9
2.3
28.3
54.7
16.1
D42.2
44.3
14.3
22.1
D21.5
D25.6
56.5
13.4
63.9
24.4
D50.1
D40.0
34.6
53.8
27.5
D1.8
71.5
50.5
D14.8
D20.6
D24.7
0.3
1.4
D0.2
0.9
0.8
0.1
D0.2
0.2
D0.1
0.3
0.4
0.4

DE00
0.2
0.1
0.4
0.6
0.8
0.4
0.5
0.4
0.2
0.9
0.8
1.0
0.4
0.5
1.6
1.8
1.0
0.4
0.9
0.9
0.9
0.9
0.8
0.8

Table G2. D50 Light Booth Condition
Number
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24

Patch,Name
dark,skin
light,skin
blue,sky
foliage
blue,flower
bluish,green
orange
purplish,blue
moderate,red
purple
yellow,green
orange,yellow
blue
green
red
yellow
magenta
cyan
white
neutral,8
neutral,6.5
neutral,5
neutral,3.5
black

Full,Spectral,Calculation
L*
a*
b*
37.2
12.7
13.8
65.7
14.2
17.7
50.1
0.2
D22.3
43.7
D16.9
21.3
55.7
11.5
D24.7
70.2
D31.7
2.2
61.1
29.0
56.4
40.6
16.9
D43.4
51.2
44.2
14.1
29.7
22.8
D23.2
71.7
D27.5
57.6
71.9
12.5
66.3
29.5
25.2
D51.2
55.3
D41.2
35.3
40.4
52.1
24.7
81.7
D3.8
77.9
51.0
50.1
D16.5
49.9
D20.5
D25.3
95.4
D0.3
0.9
80.4
D0.7
0.5
65.4
0.2
D0.3
50.9
D0.7
D0.3
35.0
D0.5
D0.2
19.8
0.1
D0.2

390


Table G3. Tungsten (III) Light Booth Condition

<table>
<thead>
<tr>
<th>Number</th>
<th>Patch Name</th>
<th>Full Spectral Calculation</th>
<th>Multispectral Calculation</th>
<th>Object Display Screen</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>L*</td>
<td>a*</td>
<td>b*</td>
</tr>
<tr>
<td>1</td>
<td>dark skin</td>
<td>38.3</td>
<td>15.3</td>
<td>14.4</td>
</tr>
<tr>
<td>2</td>
<td>light skin</td>
<td>67.0</td>
<td>19.1</td>
<td>17.5</td>
</tr>
<tr>
<td>3</td>
<td>blue sky</td>
<td>49.6</td>
<td>-1.3</td>
<td>-21.6</td>
</tr>
<tr>
<td>4</td>
<td>foliage</td>
<td>42.8</td>
<td>-16.6</td>
<td>17.3</td>
</tr>
<tr>
<td>5</td>
<td>blue flower</td>
<td>55.9</td>
<td>11.9</td>
<td>-22.0</td>
</tr>
<tr>
<td>6</td>
<td>bluish green</td>
<td>68.4</td>
<td>-34.3</td>
<td>-4.3</td>
</tr>
<tr>
<td>7</td>
<td>orange</td>
<td>63.8</td>
<td>32.4</td>
<td>53.7</td>
</tr>
<tr>
<td>8</td>
<td>purplish blue</td>
<td>40.5</td>
<td>12.2</td>
<td>-39.6</td>
</tr>
<tr>
<td>9</td>
<td>moderate red</td>
<td>54.5</td>
<td>51.2</td>
<td>19.0</td>
</tr>
<tr>
<td>10</td>
<td>purple</td>
<td>30.8</td>
<td>23.3</td>
<td>-15.8</td>
</tr>
<tr>
<td>11</td>
<td>yellow green</td>
<td>70.6</td>
<td>-28.4</td>
<td>45.9</td>
</tr>
<tr>
<td>12</td>
<td>orange yellow</td>
<td>73.4</td>
<td>15.7</td>
<td>60.5</td>
</tr>
<tr>
<td>13</td>
<td>blue</td>
<td>29.7</td>
<td>19.8</td>
<td>-47.9</td>
</tr>
<tr>
<td>14</td>
<td>green</td>
<td>53.3</td>
<td>-45.2</td>
<td>24.4</td>
</tr>
<tr>
<td>15</td>
<td>red</td>
<td>44.7</td>
<td>63.8</td>
<td>29.3</td>
</tr>
<tr>
<td>16</td>
<td>yellow</td>
<td>82.3</td>
<td>-0.7</td>
<td>66.1</td>
</tr>
<tr>
<td>17</td>
<td>magenta</td>
<td>54.4</td>
<td>56.7</td>
<td>-6.5</td>
</tr>
<tr>
<td>18</td>
<td>cyan</td>
<td>48.2</td>
<td>-20.5</td>
<td>-29.9</td>
</tr>
<tr>
<td>19</td>
<td>white</td>
<td>95.3</td>
<td>0.5</td>
<td>0.4</td>
</tr>
<tr>
<td>20</td>
<td>neutral 8</td>
<td>80.3</td>
<td>-0.7</td>
<td>0.3</td>
</tr>
<tr>
<td>21</td>
<td>neutral 6.5</td>
<td>65.4</td>
<td>0.0</td>
<td>-0.3</td>
</tr>
<tr>
<td>22</td>
<td>neutral 5</td>
<td>50.8</td>
<td>-1.0</td>
<td>-0.4</td>
</tr>
<tr>
<td>23</td>
<td>neutral 3.5</td>
<td>35.0</td>
<td>-0.7</td>
<td>-0.3</td>
</tr>
<tr>
<td>24</td>
<td>black</td>
<td>19.8</td>
<td>0.2</td>
<td>-0.2</td>
</tr>
</tbody>
</table>

*The CIEDE2000 values of six out-of-gamut patches are highlighted in the last column of Table G3.*