The Emotional Impact of Audio - Visual Stimuli

Titus Pallithottathu Thomas
tpt7797@rit.edu

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

Recommended Citation

This Thesis 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 ritscholarworks@rit.edu.
The Emotional Impact of Audio - Visual Stimuli

By,

Titus Pallithottathu Thomas

A Thesis Submitted in Partial Fulfillment of the Requirements for the Degree of Master of Science in Computer Engineering

Supervised by

Dr. Raymond Ptucha
Department of Computer Engineering
Kate Gleason College of Engineering
Rochester Institute of Technology
Rochester, NY
July 2017

Approved By:

Dr. Raymond Ptucha
Primary Advisor – R.I.T. Dept. of Computer Engineering

Dr. Emily Prud'hommeaux
Secondary Advisor – R.I.T. Dept. of English

Dr. Andres Kwasinski
Secondary Advisor – R.I.T. Dept. of Computer Engineering
I would like to dedicate this research to God and my family.
Acknowledgements

“The fear of the Lord is the beginning of wisdom, and I can do all things through Christ my Lord who strengthens me.”

Dr. Ptucha, I am thankful for your unwavering guidance and relentless support from the day I started my master's program all the way to its completion. I would like to thank my thesis committee members, Dr. Prud'hommeaux and Dr. Kwasinski for guiding me during my time at Rochester Institute of Technology (RIT).

I would also like to thank my family and my comrades in arms at the Machine Intelligence Lab.
Abstract

Induced affect is the emotional effect of an object on an individual. It can be quantified through two metrics: valence and arousal. Valence quantifies how positive or negative something is, while arousal quantifies the intensity from calm to exciting. These metrics enable researchers to study how people opine on various topics. Affective content analysis of visual media is a challenging problem due to differences in perceived reactions. Industry standard machine learning classifiers such as Support Vector Machines can be used to help determine user affect. The best affect-annotated video datasets are often analyzed by feeding large amounts of visual and audio features through machine-learning algorithms. The goal is to maximize accuracy, with the hope that each feature will bring useful information to the table.

We depart from this approach to quantify how different modalities such as visual, audio, and text description information can aid in the understanding affect. To that end, we train independent models for visual, audio and text description. Each are convolutional neural networks paired with support vector machines to classify valence and arousal. We also train various ensemble models that combine multi-modal information with the hope that the information from independent modalities benefits each other.

We find that our visual network alone achieves state-of-the-art valence classification accuracy and that our audio network, when paired with our visual, achieves competitive results on arousal classification. Each network is much stronger on one metric than the other. This may lead to more sophisticated multimodal approaches to accurately identifying affect in video data.
This work also contributes to induced emotion classification by augmenting existing sizable media datasets and providing a robust framework for classifying the same.
## Table of Contents

DEDICATION ........................................................................................................................................ II

ACKNOWLEDGEMENTS .................................................................................................................... III

ABSTRACT ........................................................................................................................................ IV

TABLE OF CONTENTS ....................................................................................................................... VI

LIST OF FIGURES ............................................................................................................................. IX

LIST OF TABLES ............................................................................................................................... XI

GLOSSARY .......................................................................................................................................... XIII

CHAPTER 1  INTRODUCTION ........................................................................................................... 1

1.1. MOTIVATION ............................................................................................................................. 1

1.2 OUTLINE ................................................................................................................................... 6

CHAPTER 2  LITERATURE REVIEW ................................................................................................. 8

2.1. LEGACY AFFECT CLASSIFICATION ...................................................................................... 8

2.2. CONTEMPORARY AFFECT CLASSIFICATION .................................................................... 11

CHAPTER 3  BACKGROUND ............................................................................................................ 14

3.1. HANDCRAFTED VISUAL FEATURE CLASSIFICATION ...................................................... 14

3.2. AUDIO FEATURE CLASSIFICATION .................................................................................... 15

3.3. K–NEAREST NEIGHBOR ....................................................................................................... 17

3.4. SUPPORT VECTOR MACHINES .............................................................................................. 18

3.5. CONVOLUTIONAL NEURAL NETWORKS ............................................................................ 19

3.6. DEEP IMAGE CLASSIFICATION AND CAPTIONING ....................................................... 21

3.7. TRANSFER LEARNING .......................................................................................................... 24
6.6. AROUSAL NORMALIZATION EXPERIMENTS .......................................................... 57

CHAPTER 7 CONCLUSION .......................................................................................... 58

BIBLIOGRAPHY .......................................................................................................... 59
List of Figures

Figure 1: Image that elicits a positive valence(left) or confusing valence (right) [4].

Figure 2: Graph that shows the emotions induced by varying valence and arousal.

Figure 3: Images that induce a high (positive) valence and low (left) or high (right) arousal.

Figure 4: Images to demonstrate the difference in of valence and arousal levels depending on the object of attention. The gun in the original image [7] to the left has been obscured to change the primary object of attention from the weapon to the man holding it.

Figure 5: Wheel of Emotion as defined by Dr. Plutchik [21].

Figure 6: Lam et al.’s video affect analysis network architecture [8].

Figure 7: Yi et al.’s affect analysis network architecture [9].

Figure 8: Traditional Image classification model.

Figure 9: (Left) Original image and (Middle) image with SIFT key points overlaid and its Global color histogram (Right).

Figure 10: Traditional Audio classification model.

Figure 11: Spectrogram of an audio file from the LIRIS-ACCEDE dataset [30] generated using SoX [39] with default parameters.

Figure 12: Toy K-NN classification.

Figure 13: Toy linear SVM classification.
Figure 14: LeCun et al.'s [39] LeNet-5 CNN used for handwritten character recognition on the Mnist Dataset.

Figure 15: AlexNet CNN architecture [41].

Figure 16: A simple transfer learning implementation.

Figure 17: An LSTM memory cell [51].

Figure 18: Venugopalan et al.'s S2VT Model [58].

Figure 19: Normalized film distribution of IMDB, LIRIS-ACCEDE, and ScreenRush [30].

Figure 20: LIRIS-ACCEDE Dataset class spread in percent.

Figure 21: LIRIS-ACCEDE Dataset class spread in percent.

Figure 22: Mediaeval Dataset class spread in percent.

Figure 23: Mediaeval Test Set class spread in percent.

Figure 24: Emotion data set class spread for dominant emotion.

Figure 25: Visual affect classification model.

Figure 26: Audio Affect classification network architecture.

Figure 27: The Pyaudioanalysis feature vector [68, 73].

Figure 28: Spectrogram from Figure 11 processed for use in the Deep Audio Models.

Figure 29: Video Description Classification model.

Figure 30: Late Fusion Affect Ensemble Model.

Figure 31: Early Fusion Affect Ensemble Model.
List of Tables

Table 1: Global accuracies in percent for best affect classification models with the best accuracies in bold. 47

Table 2: Global accuracies in percent for different audio only affect classification models with highest in bold. 49

Table 3: Mean Square Errors for different Deep audio only affect classification models with lowest in bold. 50

Table 4: Video Descriptions generated by S2VT trained using different datasets. 50

Table 5: Global accuracies in percent for different oracle text affect classification models with highest in bold. 52

Table 6: Mean Square Errors for different oracle text affect classification models with lowest in bold. 52

Table 7: Global accuracies in percent for Late Fusion Ensemble models explored in this study with highest in bold. 53

Table 8: Mean Square Error for Late Fusion Ensemble models explored in this study with lowest in bold. 54

Table 9: Global accuracies in percent for early fusion ensemble models explored in this study in highest in bold. 54

Table 10: Mean Squared Error for early fusion models explored in this study with lowest in bold. 55
Table 11: Feature vectors used for SVM classification with their Accuracy and MSE.

Table 12: Feature vectors used for Arousal normalization experiments with their Accuracy and MSE.
Glossary

CNN

Convolutional Neural Network

FC6

Fully Connected Layer - 6

FC7

Fully Connected Layer – 7

FC8

Fully Connected Layer – 8

GMM

Gaussian Mixture Model

KNN

K-Nearnest Neighbour

LSTM

Long Short-Term Memory

MFCC

Mel Frequency Cepstrum Coefficients

MSVD

Microsoft Research Video Description Corpus

MSE

Mean Squared Error
NLP

Natural Language Processing

S2VT

Sequence to sequence - Video to Text

SVM

Support Vector Machine

SVR

Support Vector Regression

SoX

Sound eXchange
Chapter 1  Introduction

1.1.  Motivation

Internet communication has evolved from primary text-based bulletin boards and forums to complex multimedia social networks. Social media enables users to express their feelings on a topic - not just through natural language, but through images, video, and emoji. This potentially lucrative problem space is of keen interest to data scientists and machine learning engineers. The surge of media-only social networks such as Instagram and YouTube see content creators and casual users expressing themselves in a combination of media such as text, images, and video. Analyzing the emotions of social media posts is an active area of research, but often visual media attached to these posts are removed during the preprocessing step [1, 2]. Focusing only on text constrains the analysis to study statistical dependencies between words. Additionally, most studies concentrate on English-language data. As of 2017, only 25 percent of Internet users are native English speakers [3]. Useful affect analysis algorithms need to take a more global approach. This can be done by analyzing the media’s visual and audio content, which can include language-agnostic information that can help an emotion analysis system generalize away from English-based attributes.
Figure 1: Image that elicits a positive valence (left) or confusing valence (right) [4].

The emotional impact of the surrounding environment on humans is known as induced affect. Induced affect is categorized by a combination of valence (sentiment) and arousal. Valence is the affinity towards an object; positive, neutral or negative. Arousal is the intensity of the reaction towards that object: calm, neutral or active. Valence is often referred to as sentiment in studies. However, unlike sentiment which often takes on discrete values, arousal and valence are often represented as either discrete or continuous values. Combinations of valence and arousal encode emotion (Figure 2). For example, high valence and low arousal can indicate serenity, while high valence and high arousal can indicate excited joy (Figure 3). Figure 1 (left) shows an example of a picture that likely suggests a high positive valence. The subject in the picture has a confident, energetic smile and is posing with a cute cat. It is unlikely that a viewer would ascribe a negative valence to that image.
However, the emotional response can often be ambiguous and differ from person to person. For example, Figure 1 (right) shows an image that may be construed as negative or positive. In general, images of babies elicit a positive response, and many viewers may come away with that impression. However, in this case, the creator intended to post a negative picture illustrating a lesion on the baby’s face [4]. This ambiguity makes affect analysis a challenging task. Chen et al. [5] defined the above separation as Publisher Affect Concepts (PAC) and Viewer Affect Concepts (VAC). Due to this distinction between PAC and VAC, affective analysis of visual media is an interesting field that is ripe for examination because most studies are based on PAC [4, 6]. While affective analysis is a challenging task, it can still be considered a classification problem of sorts as each affective value is associated with an appropriate label.
Affect is highly dependent on the object of attention. Figure 4 shows an image of a man holding a gun. A person looking at that image will usually experience low valence (negative) and high arousal (agitated) because of the presence of a weapon. However, if the weapon is obscured, the focus of attention shifts to the man. A person viewing Figure 4-left sees a smiling man and may experience high valence (positive) and low arousal (calm). Thus, the presence of the gun in the Figure 4-right image, which becomes the primary object of attention, changes the entire tone of the visual stimulus. Since a video is a moving set of image frames, the sudden appearance of certain objects (such as weapons) establishes the tone for the entire emotional classification.
Figure 4: Images to demonstrate the difference in valence and arousal levels depending on the object of attention. The gun in the original image [7] to the left has been obscured to change the primary object of attention from the weapon to the man holding it.
1.2. Outline

Estimating affect from video offers several advantages over other media. Images are limited to describing color, shape, and texture. Videos contain motion and often sound information that can help disambiguate its affective meaning. Recent works have made use of features from several domains to attempt to classify valence and arousal [8-11]. In these works, large collections of features are built from video content, packed into intermediate representations such as PCA or Fisher vectors, and passed through machine-learning-based classifiers such as Support Vector Machines (SVMs). While legacy research has shown that additional information modalities such as text subtitles and user profile information can improve accuracy [12-14]; contemporary research seems to prefer working with information directly extracted from the audiovisual content of the videos.

Rather than pushing forward in this vein, we step back and seek to understand better how visual and audio information each uniquely contribute to the understanding of affect. To this end, we train independent networks on a common video dataset: The MediaEval 2015 dataset [15]. Some networks are trained only on the visual video data, and some are only trained on audio data. Additionally, in order to explore alternate modalities for affect classification, text descriptions extracted from the video are studied.
This thesis research makes the following contributions:

- We show that a valence classifier based on Convolutional Neural Network (CNN) features from raw video can alone achieve state-of-the-art results.
- We show that an arousal classifier based on CNN features from audio spectrograms when paired with visual features can achieve results comparable to state-of-the-art.
- We show that video features alone are less capable of classifying arousal, and audio features alone are less capable of classifying valence.
- We show that deep audio feature representations outperform traditional handcrafted audio features for affect classification.
- We show that video descriptions using state-of-the-art video captioners, and sentence encoding are capable of producing good feature representations for affect classification.
- We partially augment the LIRIS-ACCEDE dataset with manually collected ground truth, such as video captions and the universal emotions: (happy, neutral, sad, anger, disgust, and fear).

These contributions will hopefully lead to future affect classification architectures that combine data in ways that best make use of each modality’s individual strengths.
Chapter 2  Literature Review

2.1. Legacy Affect Classification

Early attempts to extract affect (text sentiment) began in the late 1950’s on typewritten and handwritten documents- a task that was often time and labor intensive [16]. However, in recent times this process has largely been automated and has been applied to digital text from vastly diverse sources. One approach to affect analysis (sentiment) is to use lexicon (rule) based classifiers that automatically build lexicons from text using certain criteria (rules) such as the semantic orientation of adjectives [17]. Another popular approach is to use sizable text datasets annotated for affect (sentiment) along with traditional classifiers such as Support Vector Machines (SVM) [18]. SVMs are a popular choice as classifiers since they boast relatively easy implementation and impressive performance and hence have been a preferred method for many machine learning problems [19, 20].

The most popular induced affect dataset is the International Affective Picture System (IAPS) [21]. This dataset has become a standard dataset used in literature because it contains detailed information about induced affect and image descriptions. Although popular, this dataset contains relatively few images which are a vital concern since affect analysis is a challenging task and all models benefit from additional training data. Glauser and Scherer [22] argue that repeated exposure to the same images lower their affective impact. Thus, affective analysis of videos is a potentially lucrative problem space as
relatively less work (compared to images) has been conducted whereby the chance of a repeat participant being exposed to similar data is reduced.

Inspired by film theorists and psychologist’s studies analyzing the affective content of videos [23], researchers have been studying the correlation of valence and arousal with changes in motion, audio, and scene content. Wang and Cheong [24] studied the effects of visual features such as lighting, motion, color energy and audio features extracted from speech and music content. They were one of the first papers to propose and incorporate the use of more comprehensive audio features such as Mel Frequency Cepstrum Coefficients (MFCCs) and audio spectral characteristics for affective video analysis. Subsequent studies studying video affect analysis incorporate other modalities alongside the above features such as studying text captions [14] or user profile information [12, 14] and subsequently using Bayesian or Support Vector Regression (SVR) to build affective video models.

To address the lack of academically relevant datasets for visual affective analysis, Borth et al. [25] created a dataset of affect annotated images. Images from Flickr and YouTube were labeled with adjectives that describe objects (nouns) in accordance with the wheel of emotion defined by Plutchik (Figure 5) [26]. For example, the adjective angry when paired with nouns such as I or bull consist of images with a negative valence, whereas the adjective happy when paired with the nouns baby or smile consist of images with a positive valence. The wheel of emotion is a well-established representation of emotions within the psychological community. Over time, similar affect (sentiment) annotated datasets such as the Photo Tweet Sentiment Benchmark dataset (PTSB) [4] have been
created. This dataset is often augmented with similar images collected from social media [27].

![Figure 5: Wheel of Emotion as defined by Dr. Plutchik [21].](image)

Attempts have been made to extract *publisher affective concepts* from videos. Morency and Mihalcea [28] proposed a multimodal affect (sentiment) analysis model that extracted text, audio, and video features to help attempt this classification. They used a YouTube video dataset with people voicing their opinions while directly facing the camera. Rosas et al. [29] proposed a similar classifier to fuse audio and visual cues along with transcribed text. They compiled a similar YouTube video dataset containing content in both English
and Spanish languages. Both models were trained to work with videos containing people talking while facing the camera. The focus of these studies are limited to the affect (sentiment) of the people in the videos.

### 2.2. Contemporary Affect Classification

LIRIS-ACCEDE [30] and its extension MediaEval 2015 [15] are popular benchmark video datasets annotated for induced valence and arousal. Baveye et al. [31], the LIRIS-ACCEDE dataset curators, analyzed the performance of Support Vector Regression (SVR) and CNNs for affect classification. The SVR was fed normalized visual and audio features such as image hue and zero-crossing rate. It was found that fine-tuned CNNs performed the best but the SVR results highlighted the correlation between arousal samples. This finding motivated subsequent affective studies to incorporate audio information alongside deep visual features. The 2015 Medieval Affective Impact of Movies task is a video classification challenge using the Mediaeval 2015 dataset [15]. Inspired by previous work, several architectures use CNNs [9, 10]. CNNs such as Alexnet [9, 10] and VGG-19 [8] were also used for feature extraction. Audio features were mostly limited to MFCCs. These audio features, as well as both handcrafted and deep visual features, have been merged using Gaussian Mixture Model (GMM) vector encoding [4, 5]. Studies that incorporated motion features alongside RGB features have yielded promising results [8, 10].
Lam et al. [8] achieved the best valence accuracy (42.95%) by using handcrafted features for visual information, motion alongside deep CNN features (FC6, FC7, and FC8) and MFCCs (Figure 6). GMM vector encoding was employed for all the features, and models were trained with and without PCA for motion and audio features respectively. The average pooled motion and audio representations are used to train an SVM which resulted in impressive valence and arousal accuracies.

![Figure 6: Lam et al.’s video affect analysis network architecture [8].](image)

Yi et al. [9] proposed a linear late fusion model incorporating handcrafted visual features, deep CNN visual features, and MFCCs (Figure 7). With the exception of CNN visual features, all other feature vectors were GMM vector encoded followed by SVM classification. Linear late fusion was employed to merge the results of each SVM classifier, and the CNN classification produced a final valence accuracy score of 55.93%.
Figure 7: Yi et al. ’s affect analysis network architecture [9].
Chapter 3 Background

3.1. Handcrafted Visual Feature Classification

Figure 8: Traditional Image classification model.

Figure 8 shows the workflow of a generic image classification model. Salient features are extracted from each image and are used in conjunction with a class label to train models using supervised machine learning algorithms. Once trained, this model is used to classify new images. Image classification has always seen handcrafted features such as Scale-Invariant Feature Transforms (SIFT - Figure 9 middle), Global Color Histograms (GCH - Figure 9 right), or Histogram over Gradients (HoG) paired with robust classifiers such as Support Vector Machines (SVM) [20, 32, 33] or K-Nearest Neighbor [34-36]. These handcrafted features extract salient vector representations from images and
offer several advantages over working with raw image data. These feature representations work extremely well for classifying images from small homogeneous datasets, but its performance exponentially wanes on heterogeneous datasets of increasing size [32, 37, 38].

![Figure 9](image)

**Figure 9:** (Left) Original image and (Middle) image with SIFT key points overlaid and its Global color histogram (Right).

### 3.2. Audio Feature Classification.

![Figure 10](image)

**Figure 10:** Traditional Audio classification model.
Analogous to Image classification in the previous section, Audio classification sees feature representations extracted from audio files and this alongside its associated labels are used to train a classification model. This workflow is detailed in Image 10. The feature representations can be simple vectors obtained by analyzing the pitch or temporal characteristics such as zero crossing rate, or signal energy.

Spectral characteristics of the audio signal provide more detailed representations by using Fourier transform to convert the signal to work in the frequency domain to extract features such as spectral flux and spectral entropy. Figure 11 is an example of a linear spectrogram extracted from a video file from the LIRIS-ACCEDE dataset [30]. More complex feature representations such as MFCCs and chroma vectors are popular for many speech recognition and music recognition tasks.
3.3. **K–Nearest Neighbor**

K–Nearest Neighbor is a machine learning algorithm that is popular within the machine learning community because of its simple but intuitive approach for classification. As the name implies, for each unforeseen test sample a set of $K$ samples that are closest to the test sample is returned by this algorithm. The distance is calculated using metrics such as Euclidean, Manhattan or Minkowski. A majority vote of the returned $K$ samples determines the final class of the test sample. Figure 12, shows a toy example of a simple KNN classification. Here, three samples are returned and based on the majority vote, semi-circle is the predicted class of the test sample.
3.4. Support Vector Machines

Support Vector Machines are one of the more prominent machine learning classifiers in the machine learning community. Researchers have gravitated towards them because of their great trade-off between classification accuracy and speed of training. Software companies use it for its low computational overhead and simple deployment.

To understand how the SVM algorithm works, consider the toy example in Figure 13 which consists of a two-class problem. The SVM algorithm is a binary classifier that linearly separates points from two classes using a line called a 'hyperplane.' The points from either class that are closest to the hyperplane are known as support vectors. With respect to Figure 13, an unforeseen test sample that falls to the left of the hyperplane will
get predicted as a star class and conversely if it falls to the right of the hyperplane, it will be predicted as a semi-circle class. The distance from the hyperplane represents the confidence of the prediction.

Sometimes it is likely to encounter cases where it is not possible to generate a hyperplane that linearly separates the two classes, for e.g. Figure 12. This can be mitigated by using the ‘kernel’ trick, whereby the 2D plane is rotated to a higher dimensional space where the data points can be easily separated using a higher order hyperplane.

3.5. **Convolutional Neural Networks**

Advancements in computing and internet speeds have made it easier to mine large heterogeneous image datasets across a broad range of problem spaces. This sudden availability of data (along with significant advancements in parallel computing on GPUs
and incremental software and hardware advances) led to the proliferation of Convolutional Neural Networks (CNNs).

CNN’s as the name implies is a popular variant of neural networks. Neural networks consist of an input layer, various hidden layers and an output layer all made up of individual neurons. Each neuron takes an input and fires an output when a certain energy threshold is met. An activation function determines the energy threshold. Sigmoid, Rectified Linear Units (ReLU) and tanh are popular activation functions with ReLU being the most used as it offers many advantages such as helping mitigate the vanishing gradient problem.

Figure 14 is an example of one of the first popular CNN architectures: - LeNet-5 [39] which was used for character recognition (digits). The input to this CNN is a 32×32 image consisting of a single handwritten character, and the output of the network is a 1×10 vector containing the predicted label. CNNs can learn feature representations unique to a problem [40]. Unlike handcrafted features (Section 3.1), CNN’s work exceptionally well with large and unforeseen image datasets but require a considerable amount of data and compute power. While this was an issue for early CNNs, faster broadband speeds and
improved compute capabilities of current hardware allow easy collection and processing of sizable image datasets.

![Figure 14](image)

*Figure 14: LeCun et al.’s[39] LeNet-5 CNN used for handwritten character recognition on the Mnist Dataset.*

### 3.6. Deep Image classification and captioning.

The resurgence of deep learning within the computer vision community has resulted in significant advancements in image captioning tasks. This rise was made possible due to large datasets such as the ImageNet dataset which has over 14 million images from over 20,000 categories. Annotation datasets such as Microsoft’s MSCOCO data set provides segmented and captioned images [42].

AlexNet is a deep CNN that was trained on Google’s ImageNet dataset [41] (Figure 15). This network architecture performed exceptionally well on the ImageNet dataset.
which was a notoriously hard classification dataset. Prior networks made liberal use of handcrafted features. AlexNet popularized the rapid adoption of CNNs in both academia and industry for image classification and captioning tasks. The input to the AlexNet CNN is a 224x224 image. The network consists of five convolution, max pooling and dropout layer blocks followed by two fully connected layers that each consist of 4,096 neurons. The network is then connected to a final fully connected layer that consists of 1,000 neurons, which makes the final class prediction. This CNN implementation also popularized the use of the ReLU activation functions and dropout layers to shave off training time and reduce overfitting in deep image classification architectures respectively.

Figure 15: AlexNet CNN architecture [41].
Subsequent networks trained on the ImageNet dataset saw improvements in classification accuracy by increasing the number of layers - VGGnet [43], ResNet [44] or by making layers ‘smarter’ - GoogleLeNet [45].

The primary motivation of the VGGnet implementation was expanding the Alexnet architecture in simple, nonobtrusive ways while improving classification accuracy. They did this by replacing large filter convolution layers (in AlexNet) with multiple smaller filter convolution layers. This saw a notable reduction in the number of parameters while maintaining and improving the classification accuracy. Multiple convolution layers also have the added benefit of the architecture having exponentially more ReLU activation functions. The input, output, fully connected and max pooling layers remain the same as its AlexNet counterparts.

ResNet took this concept further by expanding AlexNet by adding an exponentially large number of convolution layers with its final implementation being 152 layers deep. This CNN implementation has one of the lowest error rates of 3.6% on the ImageNet dataset, but it results in a significant number of parameters and subsequently longer training times.
3.7. **Transfer Learning**

Transfer learning is the process of extracting meaningful information from an existing model and refactoring it to achieve meaningful performance gains in a new model in the same or related domain [46]. The weights from a CNN trained on one dataset are used to initialize the weights of a new CNN of similar architecture, then fine-tuned on a new dataset. This has the added benefit of cutting down on the training time of deep CNN architectures. Using pre-trained CNN models such as AlexNet [41], GoogleLeNet [45], Microsoft ResNet [44] and VGGnet [43] are popular choices for most transfer learning implementations. Figure 16 details the workflow showing a simple implementation of transfer learning using a CNN pretrained on the ImageNet Dataset.

![Figure 16: A simple transfer learning implementation.](image-url)
3.8. **Long-Short Term Memory Neural Networks**

Long Short-Term Memory (LSTM) neural networks are modern variants of Recurrent Neural Networks (RNNs) that improve the ability to reason over temporal sequences. LSTM neural networks, while initially conceived by Hochreiter and Schmidhuber [47], have seen cumulative improvements to its functionality over the years, eventually leading to its more widespread adoption in various problem spaces in computer vision and Natural Language Processing (NLP). Figure 17 shows an LSTM cell.

CNN’s have been successfully used in conjunction with LSTMs for object recognition as well as image and video captioning. These architectures quickly outperformed traditional methods which used handcrafted features paired with independent classifiers [48-50]. In image captioning tasks, CNNs are used to extract features from images, and these features are fed to LSTMs. LSTM layers take in CNN features as well as one-hot encoded vector representations of words. The length of these one-hot encoded vectors, or the dictionary length, can be dictated by word counts in the dataset.

Recent studies have shown that LSTMS perform on par with traditional methods such as SVMs for NLP tasks such as text generation [51-53] and machine translation [54-56]. Due to their impressive performance, technology giants, such as Apple and Google have deployed LSTMS for state-of-the-art speech and text processing.
3.9. **Video Captioning**

Video captioning is a challenging yet rewarding problem space that can potentially have major real-world ramifications such as providing real-time captioning for the visually impaired. Perhaps the biggest challenge arises from having to identify and observe a large number of objects and their interactions (activities). Thomason et al. [57] tried to solve this problem by using a language model alongside SVM’s to identify relevant events and objects in the videos to create a novel sentence using a predefined set sentence templates. Since sentences are constrained to follow a fixed set of templates, they are not always as intuitive to read when compared to natural language.

LSTM based deep learning models are seeing increased use in video processing tasks such as video captioning [58-60] and video summarization [61-63]. Venugopalan et al. [64] proposed a novel framework for video captioning, where image frames from videos...
are sampled at constant timesteps and in turn sent to CNNs pre-trained on ImageNet. The outputs of these CNNs (FC7 layer) are mean pooled and then forwarded to a two-layer LSTM network of 100 hidden neurons each, to generate a sentence which describes the input video.

Venugopalan et al. [58] further refined this model by exploiting temporal information in the video as well. This new architecture dubbed the Sequence to Sequence Video to Text model (S2VT model) (Figure 18) adapt captioning for videos by calculating the visual information in the frames similarly to the previous approach. In addition, optical flow between two successive frames can be fed to a separate CNN, and the FC7 feature vector is used for LSTM architectures. Whether using image or optical flow features, the LSTM model has two stages, an ‘encoding’ stage where the LSTM learns the feature vectors by creating a linear feature representation and a ‘decoding’ stage where the LSTM generates the sentence from the newly learned feature vectors. These models are trained on video description datasets such as the Microsoft Video Description Corpus (MSVD) [65]. The MSVD dataset consists of roughly 2,000 videos collected from YouTube with one-line video descriptions in multiple languages (around 120,000 sentences).

The S2VT model utilizes zero padding added to the end of the last frame to manage variable frame length. Upon encountering the first zero-padded input, the ‘encoding’ LSTM layer generates and sends a begging of sentence ‘bos’ token to the ‘decoding’ LSTM layer to start sentence generation using the linear feature representation. Once the ‘decoding’ layer outputs a complete sentence, it outputs an end of sentence ‘eos’ token.
During training time, the log likelihood of the predicted words is used to calculate the loss for backpropagation.

Figure 18: Venugopalan et al.’s S2VT Model [58].
Chapter 4  Datasets

4.1. LIRIS-ACCEDE Affect Dataset

LIRIS-ACCEDE is a sizable video dataset made up of 9,800 short video clips extracted from 160 independent movies. This dataset is used as a baseline in many studies. The movies in this dataset are from one of the nine most common film genres. The spread of the movies in the dataset mirrors the normalized distribution on IMDB as shown in Figure 19. These movies come in a variety of different languages such as English, German, Italian, French, Hindi with the majority being English. The non-English movies have English subtitles, and a small subset of English movies are provided with non-English subtitles. A tiny percent of the video clips are extracted from silent films.

Figure 19: Normalized film distribution of IMDB, LIRIS-ACCEDE, and ScreenRush [30].
Each video segment is roughly 8 to 12 seconds in length, and a considerable amount of videos have more than one scene in each segment. These segments contain a wide variety of different scenes ranging from serene mountain landscapes to scenes of brutal fights. The dataset comes with both continuous and discrete versions. The discrete dataset is used in subsequent experiments conducted in this thesis. The dataset curators recommend a train and test split of 6,144 and 3,656 videos respectively. The class distribution for valence and arousal labeled videos are shown in Figure 20. The recommended training and testing splits set by the dataset curators are 6,144 (Figure 21) and 3,656 respectively.

![Figure 20: LIRIS-ACCEDE Dataset class spread in percent.](image)
4.2. **MediaEval 2015 Affect Dataset**

The MediaEval Video dataset consists of 10,900 video clips taken from 199 movies. This dataset is an extension of the LIRIS-ACCEDE dataset from the previous section augmented with an additional 1,100 clips. Each clip is between 8 to 12 seconds in length. The videos in this dataset cover the entire affect range. The clips range from civil conversation to intense action sequences. Each video in this dataset has a discrete label for valence (Positive, Neutral, or Negative) and arousal (Calm, Neutral and Active). The class distribution of videos in this dataset is shown in Figure 22. The dataset curators recommend training and testing splits of 6,144 and 4,175 videos, respectively (Figure 21 and Figure 23). Ten percent of the training videos were held out as validation for hyperparameter fine-tuning. Valence and arousal accuracy on the test set is the official metric (as recommended by the dataset curators) for evaluating the performance of models on this dataset.
Figure 22: Mediaeval Dataset class spread in percent.

Figure 23: Mediaeval Test Set class spread in percent.
4.3. Data Collection

In order to explore alternate modalities for affect classification, the Mediaeval 2015 dataset was partially augmented with text descriptions for 7,483 out of the original 10,900 videos. Each captioned video was provided a succinct description detailing the major objects in each scene and their interactions. There were utmost two sentences in American English per video sample, and these sentences were collected using only the visual elements present in the image. For fairness sake, visual information subtitles in the videos are not considered.

Induced Emotion for each video was based on the following classes - happiness, fear, surprise, sadness, anger, disgust and neutral. These emotion categories are based on the six universal human facial emotions as popularized by Dr. Ekman [66] with an additional neutral class to account for videos that elicit a reaction that does not fall into the former categories. Both dominant emotion and combination of emotions felt are collected for 6,402 videos out of the original 10,900 videos. Figure 24 shows the class distribution for the train and test sets for the dominant emotion.
Figure 24: Emotion data set class spread for dominant emotion.
Chapter 5  Methodology

5.1. Visual Classification

A series of temporal frames are extracted from each video and fed to separate VGG-19 CNNs pre-trained on the ImageNet Dataset. The final 4,096 fully connected output of the CNN’s is used to train separate SVMs for valence and arousal.

\[ N = 80 \] evenly-spaced frames are extracted from each video and fed into a VGG-19 network [43] pre-trained on the ImageNet image classification dataset [41]. We extract \( N \) FC7 feature vectors \( x_n \). The image representations are average pooled, as shown in (1). The VGG Network is not fine-tuned on image or video affect datasets. A custom python script using Caffe and Numpy was used for the vector extraction and averaging.
\[ x_\mu = \frac{1}{N} \sum_{n=1}^{N} x_n \]  

Figure 25 details the model architecture employed for visual feature classification. The FC7 features extracted in the previous section are used to train separate kernel SVMs using 5-fold cross-validation on the training set. A grid search is performed to find the optimal hyper parameters for the SVM classifier, maximizing both precision and recall. Both the valence and arousal SVM use the radial basis function kernel and a \( \gamma = 0.001 \). For the valence SVM, \( C = 1 \), and for the arousal SVM, \( C = 100 \).

5.2. Audio Classification

The first step of audio preprocessing for audio affect analysis involves extracting high-quality lossless audio from each video in both datasets. The Pazera Free Audio Extractor [67], a free audio extraction library, is used to extract high-quality audio in the *.wav format from the video files without a loss in audio quality.

Figure 26: Audio Affect classification network architecture.

The following independent audio models (Figure 26) are explored in this study:
5.2.1 Handcrafted Audio Classification

A feature vector composed of classical audio features including MFCCs, Zero-Crossing Rate, and Spectral Flux is extracted from the audio of each video file. The Python library PyAudioanalysis [68] extracts a feature vector $x \in \mathbb{R}^{68}$ that represents the mean and standard deviations of 34 audio features sampled in 50ms windows (with 25ms overlap). Pyaudioanalysis is a popular tool that is used in many audio analysis studies for feature extraction [69, 70] and optionally audio feature classification [71, 72].

Figure 27 details the features used in this portion of the study. This feature representation is used to train SVM and K-Nearest Neighbor (KNN) classifiers. Features 1-3 represent statistics mined directly from the audio file. Zero crossing rate is the number of times the audio signal changes sign (positive to negative); the energy is the sum of squares of the actual signal values. The entropy of the normalized energy, which is calculated by measuring sudden audio signal changes is also calculated.

Features 4-8 extract statistics from the signal spectrum. A signal spectrum details the distribution of the audio signal’s amplitude with respect to frequency. The spectral centroid is the center of gravity (center of mass) of the audio spectrum. The spectral spread is the second central moment, spectral flux is the squared differences between the normalized spectral values, spectral roll off is the frequency value below which most of the spectral values are concentrated.
Features 9-21 extract MFCC feature representations from the video. MFCC which stands for Mel Frequency Cepstral coefficients is a feature representation of the audio signal converted to the Mel scale using mel filter banks. The signal is split into frames, and for each frame, a power spectrum is calculated for each mel filter bank. The log of all the filter energies is taken followed by DCT. The DCT coefficients between 2-13 is kept and the rest discarded. Features 22-33 house the chroma vector which is the entire audio spectrogram stored is stored in 12 bins. Feature 34 is the standard deviation of the 12 chroma vector bins.

<table>
<thead>
<tr>
<th>Feature ID</th>
<th>Feature Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Zero Crossing Rate</td>
<td>The rate of sign-changes of the signal during the duration of a particular frame.</td>
</tr>
<tr>
<td>2</td>
<td>Energy</td>
<td>The sum of squares of the signal values, normalized by the respective frame length.</td>
</tr>
<tr>
<td>3</td>
<td>Entropy of Energy</td>
<td>The entropy of sub-frames' normalized energies. It can be interpreted as a measure of abrupt changes.</td>
</tr>
<tr>
<td>4</td>
<td>Spectral Centroid</td>
<td>The center of gravity of the spectrum.</td>
</tr>
<tr>
<td>5</td>
<td>Spectral Spread</td>
<td>The second central moment of the spectrum.</td>
</tr>
<tr>
<td>6</td>
<td>Spectral Entropy</td>
<td>Entropy of the normalized spectral energies for a set of sub-frames.</td>
</tr>
<tr>
<td>7</td>
<td>Spectral Flux</td>
<td>The squared difference between the normalized magnitudes of the spectra of the two successive frames.</td>
</tr>
<tr>
<td>8</td>
<td>Spectral Rolloff</td>
<td>The frequency below which 90% of the magnitude distribution of the spectrum is concentrated.</td>
</tr>
<tr>
<td>9-21</td>
<td>MFCCs</td>
<td>Mel Frequency Cepstral Coefficients form a cepstral representation where the frequency bands are not linear but distributed according to the mel-scale.</td>
</tr>
<tr>
<td>22-33</td>
<td>Chroma Vector</td>
<td>A 12-element representation of the spectral energy where the bins represent the 12 equal-tempered pitch classes of western-type music (semitone spacing).</td>
</tr>
<tr>
<td>34</td>
<td>Chroma Deviation</td>
<td>The standard deviation of the 12 chroma coefficients.</td>
</tr>
</tbody>
</table>

Figure 27: The Pyaudioanalysis feature vector [68, 73].
5.2.2 Deep Audio Feature Classification

A spectrogram is computed from the lossless audio extracted from each video file. This is a single spectrogram that represents the entire video. Various linear and logarithmic spectrograms were extracted from the audio files of each video file using the audio processing libraries Sox [74] and Librosa [75]. Color and monochrome spectrograms as well as channel separated and channel averaged spectrograms were tested. Figure 11 (Section 2) shows an example of the spectrogram extracted from Sox.

It was found that monochrome channel separated linear spectrograms were the most effective spectral representation for the Deep Audio Models. Figure 28 shows an example of the extracted spectrogram used for subsequent analysis. This spectrogram has time on the x-axis, frequency on the y-axis (between 0-24 kHz) and the intensity of the color on the plane (z-axis) represents the dynamic range (amplitude) which is set to the default value of 120 dB.
The spectrograms are used to train separate CNNs using AlexNet [41], GoogleLeNet [45], and VGG-19 [43] architectures. A small set of training samples with an even number of classes is used as the train and test to maximize the performance of each architecture. Subsequently, the CNN is scaled up to the full development set with 10 percent of the training set held out as a test set to evaluate the CNN architectures. Hyperparameters such as learning rate, step size, and $\gamma$ were modified, and various weight initiations and batch normalization are explored. Caffe was used to train and deploy these models.
In order to combat the disproportionate number of classes in the arousal dataset and to improve the classification accuracy, each training sample is augmented with variable length subsamples extracted from the same. Lower density classes are given more subsamples.

The final valence model is an AlexNet CNN trained with a learning rate of 0.001, momentum parameter of 0.9, step-size of 100, and \( \gamma = 0.1 \). The final arousal model was a VGG-19 CNN trained using Adam with a fixed learning policy, base learning rate of 0.0001 momentum parameters of 0.9 and 0.999, weight decay of 0.0005 and \( \gamma = 0.001 \).

The best performing valence and arousal CNN models are used to extract 4,096 vector representations of the second to last fully connected layer (FC7) from the image spectrograms. These vectors are used to train SVM classifiers analogously to the visual classification section. Five-fold classification alongside grid search is used to find the optimal hyperparameters using grid search maximizing both precision and recall. The valence SVM is trained using a linear kernel with \( C = 1 \), and for the arousal SVM, the radial basis function kernel is used with \( \gamma = 0.0001 \), and \( C = 100 \). A custom Python script using the libraries scikit-learn and Numpy is used to implement the SVM. Additionally, a late fusion ensemble model that merges all the paths in Figure 26 consisting of handcrafted and deep audio classification probabilities is implemented.
5.3. **Video Description Classification**

![Diagram](image)

*Figure 29: Video Description Classification model.*

To evaluate the feasibility of using existing video captioning methods, S2VT [58] (Figure 18) was run on Mediaeval [15] and MSVD [65] datasets. It was found that S2VT trained on MSVD performs well with MSVD and but gives below average performance for other datasets such as Mediaeval. To improve the performance of the video captioner, ground truth for the video datasets was manually collected (were not provided).

The video captioner model was trained on the new dataset (pre-trained and from scratch) and the best performing model was S2VT trained on the Mediaeval captions. The collected sentences are converted into a vector representation using Kairos et al.’s skip-thoughts [76] sentence to vector encoding scheme. As per the author’s recommendation, both the unigram and bigram feature vectors are generated and combined to get the final feature representation. These feature recommendations are used to train SVM classifiers. A grid search was used to optimize hyperparameters to maximize precision and recall using 5-fold cross-validation on the training set. The final text models are run only once on the complete testing set. The network architecture for this model is shown in Figure 29.
To study whether perfect captions improve affect classification, an ‘oracle’ classifier is trained using the above model. The collected captions (7,486) were trained using 5,303 as the training set and 2,184 as the testing set. As with the previous section, grid search was used to optimize hyperparameters to maximize precision and recall using 5-fold cross-validation on the training set. The final SVM models are trained using the radial basis function with $\gamma = 0.001$. The valence SVM has $C = 100$, and arousal $C = 1000$. To evaluate the performance of perfect captions, MSVD and MeV generated captions using the same train-test splits are used to train new SVM classifiers using the same setup. The MSVD captions are trained using SVMs using a linear kernel with $C = 10$. The MeV captions are trained using SVM classifiers with the radial basis function kernel and $C = 1000$. The valence SVM uses $\gamma = 0.0010$, and arousal $\gamma = 0.001$.

5.4. **Ensemble Classification models**

Ensemble models are tested to explore whether the combined visual, audio, and text knowledge improves overall affect classification. In the late fusion ensemble models, the results from the best independent models are fused to obtain the final classification scores. The early fusion ensemble classification model takes the best vector representations of the visual, audio, and text data and is subsequently used to train a single classifier from scratch.
5.4.1 Late Fusion Ensemble Model

To evaluate the combined classification ability of the visual, audio, and text models, the outputs from the previous sections are combined into an ensemble. The class probabilities $P_m(c)$ for each class $c$ from the different model’s $m \in M$ are merged according to the energy as shown in (2). The predictions output by one network will potentially be reinforced by the other. Figure 30 shows the Late Fusion Ensemble classification model explored in this thesis.

\[
C_{\text{best}} = \arg \max_c \sum_{m=1}^{M} P_m(c)
\]  

(2)

Figure 30: Late Fusion Affect Ensemble Model.
5.4.2 Early Fusion Ensemble Model

In order to find out whether these independent networks benefit from the knowledge of each other, the best feature vectors extracted in the previous section are used to train a single SVM using 5-fold cross-validation on the training set. A grid search is performed to find the optimal hyper-parameters for the SVM classifier, maximizing both precision and recall. Figure 31 details the model architecture employed for this classification model. Subsequently, a dual stream CNN is trained using visual and audio information. Each stream in the CNN uses a VGGnet Architecture and the FC7 layer of both streams are fused (concatenated) before the final FC8 layer of size 3. All the frames from each video were averaged to get a single image which is used as the input for the visual stream. The audio stream uses the spectrogram images extracted in the previous section. The final valence CNN uses a base learning rate of 0.001 with a momentum parameter of 0.9, step-size of 10, and $\gamma = 0.01$. The final arousal uses a base learning rate of 0.1 with a momentum parameter of 0.9, step-size of 10, and $\gamma = 0.01$.

The Audio - Visual Late fusion SVM and the Audio- Visual – Text Late fusion model was trained with both the valence and arousal SVMs using the radial basis function kernel. For the valence model, $\gamma = 0.0001$ and $C = 1$; whereas the arousal SVM uses $\gamma = 0.001$ and $C = 10$. 
5.5. Arousal Normalization Experiments

In the Mediaeval 2015 Train set, a sizable percent (63%) of videos annotated for arousal is from the “calm” class. This class imbalance makes it difficult to train well-performing machine learning models. In order to investigate further, the “calm” class was brought in line with the remaining classes by randomly sampling and removing 75 percent of the videos. The test set was retained in its entirety. The best SVM models from the previous section are used with this new training set.
Chapter 6  Tests and Results

We now compare the results of our methods to see how our networks compare overall in the MediaEval 2015 dataset. Table 1 presents our results as well as the results of other works in this field. The works that were a product of the MediaEval 2015 competition itself [8-10] were allowed five attempts to predict valence and arousal on the test set. We only report the top-performing scores for valence and arousal for each architecture. In our case, we only evaluated the test set once, with the hyperparameters chosen by evaluating the validation set or by using 5-fold cross validation on the entire training set. We also list the results of a “Trivial” predictor, which is the accuracy if one always predicted “neutral” valence and “calm” arousal. The high arousal score shows that MediaEval 2015 has a significant bias towards calm arousal. The bias towards neutral valence is much less pronounced.

*Table 1: Global accuracies in percent for best affect classification models with the best accuracies in bold.*

<table>
<thead>
<tr>
<th>Model</th>
<th>Valence</th>
<th>Arousal</th>
</tr>
</thead>
<tbody>
<tr>
<td>NII-UIT [8]</td>
<td>42.96</td>
<td>55.91</td>
</tr>
<tr>
<td>MIC-TJU [9]</td>
<td>41.95</td>
<td>55.93</td>
</tr>
<tr>
<td>Fudan-Huawei [10]</td>
<td>41.8</td>
<td>48.8</td>
</tr>
<tr>
<td>Trivial</td>
<td>37.868</td>
<td>55.55</td>
</tr>
<tr>
<td>Best Visual Classification (ours)</td>
<td><strong>44.64</strong></td>
<td>48.2</td>
</tr>
<tr>
<td>Best Audio</td>
<td>38.58</td>
<td>48.34</td>
</tr>
<tr>
<td>Classification (ours)</td>
<td>Best Text Classification (ours)</td>
<td>36.97</td>
</tr>
<tr>
<td>-----------------------</td>
<td>---------------------------------</td>
<td>-------</td>
</tr>
<tr>
<td>Best Ensemble</td>
<td>Classification (ours)</td>
<td>44.15</td>
</tr>
</tbody>
</table>

### 6.1. Visual Feature Classification

The visual-only network achieves state-of-the-art accuracy on valence classification (44.63%) but scored lower on arousal (48.2%). This is curious because this is not the first architecture to use VGG FC7 features and an SVM. However, this network only used the FC7 features and did not combine them with HOG, SIFT, and HOF features or PCA or Fisher vector encodings. It may be that simply combining many features is less helpful than focusing on a few high-quality features. If the VGG-19 network was fine tuned on the MediaEval 2015 dataset and used as a classifier, it may be possible to get even better results. These same video features that were useful in classifying valence were less helpful for classifying arousal. The inherent bias in the dataset for arousal may be the reason for the reduced accuracies. The Mean Squared Error for the valence classification is 0.880 and the arousal classification 1.186.

This may be because the presence of certain objects tends to bias valence more than arousal as explained by the case presented in the Introduction section. In both cases, the viewer will experience low arousal, but the valence will be positive for Figure 1 (left)
and negative for Figure 1 (right). CNNs are good at learning and predicting such patterns and may be the reason why valence accuracy is higher than arousal.

### 6.2. Audio Feature Classification

Table 2: Global accuracies in percent for different audio only affect classification models with highest in bold.

<table>
<thead>
<tr>
<th>Model</th>
<th>Valence</th>
<th>Arousal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Handcrafted-SVM Network</td>
<td>27.63</td>
<td>21.36</td>
</tr>
<tr>
<td>Handcrafted-KNN Network</td>
<td>30.76</td>
<td>20.54</td>
</tr>
<tr>
<td>Deep Audio Network</td>
<td>37.62</td>
<td>44.66</td>
</tr>
<tr>
<td>Deep Augmented Audio Network</td>
<td><strong>38.58</strong></td>
<td>47.02</td>
</tr>
<tr>
<td>Deep Audio SVM Network</td>
<td>33.77</td>
<td><strong>48.34</strong></td>
</tr>
<tr>
<td>Handcrafted SVM – Deep CNN (aug) – Deep SVM Fusion</td>
<td>36.56</td>
<td>23.38</td>
</tr>
</tbody>
</table>

Table 2 displays the global accuracies for valence and arousal for audio analysis. Handcrafted audio features as per Section 5.2.1 were used to train an SVM and a KNN. Spectrograms were used to train deep CNNs from scratch. Table 3 displays the Mean Squared Errors of the valence and arousal of the deep audio models. The independent audio models seem to be better at predicting arousal classes. This may be because audio tends to have thematic constancy. Consider two fight sequences in movies, one being a flashy fight scene and the other a one-sided fight with a person being injured. In both cases, the arousal
may be high due to loud and pronounced music, but the valence will be positive in the first sequence and negative in the latter.

Table 3: Mean Square Errors \( t \) for different Deep audio only affect classification models with lowest in bold.

<table>
<thead>
<tr>
<th>Model</th>
<th>Valence</th>
<th>Arousal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deep Audio Network</td>
<td>1.369</td>
<td>1.5181</td>
</tr>
<tr>
<td>Deep Augmented Audio Network</td>
<td>1.472</td>
<td>1.149</td>
</tr>
<tr>
<td>Deep Audio SVM Network</td>
<td>1.448</td>
<td>1.226</td>
</tr>
</tbody>
</table>

6.3. Text Feature Classification

Table 4 shows video descriptions produced for the same sample videos using S2VT trained on a different dataset. When directly compared, the S2VT Mediaeval trained captioner produces more vivid and detailed descriptions when running on S2VT trained on MSVD. The former often fails at captioning videos that consist of empty scenery as shown by entry 5 (Table 4).

Table 4: Video Descriptions generated by S2VT trained using different datasets.

<table>
<thead>
<tr>
<th>ID</th>
<th>S2VT MSVD</th>
<th>S2VT Mediaeval</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>a woman is talking</td>
<td>a man opens his eyes and looks down at something</td>
</tr>
<tr>
<td></td>
<td>a woman is eating a baby</td>
<td>a baby is laughing a woman smiles and kisses the baby</td>
</tr>
<tr>
<td>---</td>
<td>-------------------------</td>
<td>------------------------------------------------------</td>
</tr>
<tr>
<td>2</td>
<td>a man is talking</td>
<td>a sad man is talking to someone</td>
</tr>
<tr>
<td>3</td>
<td>a woman is talking</td>
<td>a man and a woman are seriously talking</td>
</tr>
<tr>
<td></td>
<td></td>
<td>a man is looking at something</td>
</tr>
<tr>
<td>4</td>
<td>a woman is slicing a piece of a knife</td>
<td>it is a scene of a field</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The Text classification model with the text descriptions from the MSVD trained S2VT captioner achieves an accuracy of 33.96% for valence and 53.77% for arousal. The Mean Squared Error for the valence SVM is 1.586 and arousal us 1.182.

The Text classification model with text descriptions extracted using the S2VT captioner trained on the manually collected captions (S2VT Mediaeval) sees an increased accuracy of 36.97% for valence and a similar accuracy of 53.09% for arousal. The Mean squared error for both the valence and arousal models have come down to 1.463 and 1.168 respectively. This improved valence accuracy was expected, but it is surprising that the improvement was rather minor. Additionally, arousal models did not see any improvement.

Tables 5 and 6 show the accuracy and the MSE of the oracle predictors. The perfect captions show the highest global accuracy and lowest mean squared errors which was expected. As with the previous set of experiments the captioner generated classification is
closer to each other. This may be due to the skip-thoughts sentence to vector representations and the video captioner itself. Skip-thoughts model clusters sentence based on their syntax and semantic attributes. These vectors are label independent and deeply generic in nature. The video captioner generated sentences are lower complexity. Thus, swapping out skip-thoughts for different sentence to vector representations and using more complex video captioner models may yield better classification results.

*Table 5: Global accuracies in percent for different oracle text affect classification models with highest in bold.*

<table>
<thead>
<tr>
<th>Model</th>
<th>Valence</th>
<th>Arousal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Perfect oracle classification</td>
<td>52.34</td>
<td>49.77</td>
</tr>
<tr>
<td>MSVD oracle classification</td>
<td>36.86</td>
<td>46.52</td>
</tr>
<tr>
<td>MEV oracle classification</td>
<td>35.53</td>
<td>45.32</td>
</tr>
</tbody>
</table>

*Table 6: Mean Square Errors for different oracle text affect classification models with lowest in bold.*

<table>
<thead>
<tr>
<th>Model</th>
<th>Valence</th>
<th>Arousal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Perfect oracle classification</td>
<td>1.210</td>
<td>1.41</td>
</tr>
<tr>
<td>MSVD oracle classification</td>
<td>1.531</td>
<td>1.595</td>
</tr>
<tr>
<td>MEV oracle classification</td>
<td>1.543</td>
<td>1.536</td>
</tr>
</tbody>
</table>
6.4. Ensemble Methods Results

The best visual, audio and text classification models are used to create ensemble models. The final class probabilities and concise feature vectors obtained from the above models are used to create late fusion and early fusion ensemble models respectively. The next two sections discuss the results of these ensemble models in detail.

6.4.1 Late Fusion Ensemble Model Results

*Table 7: Global accuracies in percent for Late Fusion Ensemble models explored in this study with highest in bold.*

<table>
<thead>
<tr>
<th>Model</th>
<th>Valence</th>
<th>Arousal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audio (Deep SVM) - Visual</td>
<td>35.13</td>
<td>54.04</td>
</tr>
<tr>
<td>Audio (Deep SVM) – Visual – Text</td>
<td>38.45</td>
<td>54.03</td>
</tr>
</tbody>
</table>

The class probabilities of the visual streams and the two audio streams are fused to get the final class probabilities for valence and arousal. Combinations of visual model use the audio model trained on the original spectrograms and the audio model trained on the augmented spectrograms. The ensemble of the results of the above two models with text features is also considered. Table 7 and 8 show the global accuracies and Mean Squared Errors of the late fusion models respectively. The results for the ensemble models seem to drag down the accuracies from the best independent model for valence but pulls up the
same for arousal. For valence, the results creep close to its best independent model, and further training may help reach it.

Table 8: Mean Square Error for Late Fusion Ensemble models explored in this study with lowest in bold.

<table>
<thead>
<tr>
<th>Model</th>
<th>Valence</th>
<th>Arousal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audio (Deep SVM) -</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Visual</td>
<td>1.793</td>
<td>1.198</td>
</tr>
<tr>
<td>Late Fusion</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Audio (Deep SVM) –</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Visual – Text</td>
<td>1.636</td>
<td>1.198</td>
</tr>
<tr>
<td>Late Fusion</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

6.4.2 Early Fusion Ensemble Model Results

Table 9: Global accuracies in percent for early fusion ensemble models explored in this study in highest in bold.

<table>
<thead>
<tr>
<th>Model</th>
<th>Valence</th>
<th>Arousal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audio - Visual Early Fusion</td>
<td>44.15</td>
<td>54.5</td>
</tr>
<tr>
<td>Audio Visual Dual Stream CNN</td>
<td>33.77</td>
<td>54.40</td>
</tr>
<tr>
<td>Audio – Visual – Text Early Fusion</td>
<td>44.13</td>
<td>54.5</td>
</tr>
</tbody>
</table>
The best audio, visual and text feature representations from the previous sections are combined to train SVMs independently for valence and arousal. Table 9 and 10 show the global accuracies and Mean Squared Errors of the late fusion models respectively. The inclusion of text feature vectors does not seem to make a noticeable difference and even slightly brings down the accuracy of valence. These results imply a sequential correlation between the audio and visual feature representations. Future affective content studies should continue to incorporate both audio-visual modalities. It should be noted that state-of-the-art accuracy was achieved for two of the valence models that have original visual stream results.

Table 10: Mean Squared Error for early fusion models explored in this study with lowest in bold.

<table>
<thead>
<tr>
<th>Model</th>
<th>Valence</th>
<th>Arousal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audio - Visual Late Fusion</td>
<td>1.163</td>
<td>1.143</td>
</tr>
<tr>
<td>Audio – Visual – Text Late Fusion</td>
<td>1.369</td>
<td>1.143</td>
</tr>
</tbody>
</table>

6.5. Emotion Classification

The best visual, audio, text and ensemble classification network models are used to classify the videos with the new dominant emotion class labels (Table 11). It was found that the deep arousal audio FC7 features gave the best class spread but the lowest overall
accuracy. However, when paired with audio, visual and text features extracted from the MSVD captioner lead to the best global classification accuracy and with the MeV captioner, the lowest Mean Squared error.

Table 11: Feature vectors used for SVM classification with their Accuracy and MSE.

<table>
<thead>
<tr>
<th>Features</th>
<th>Accuracy</th>
<th>MSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Visual FC7 features</td>
<td>45.09%</td>
<td>7.02</td>
</tr>
<tr>
<td>Audio Valence FC7</td>
<td>37.73%</td>
<td>7.471</td>
</tr>
<tr>
<td>Audio Arousal FC7</td>
<td>33.10%</td>
<td>7.483</td>
</tr>
<tr>
<td>Concatenated Audio FC7</td>
<td>32.82%</td>
<td>7.451</td>
</tr>
<tr>
<td>Averaged Audio FC7</td>
<td>37.73%</td>
<td>7.471</td>
</tr>
<tr>
<td>MSVD S2VT Skip-thoughts</td>
<td>42.00%</td>
<td>7.001</td>
</tr>
<tr>
<td>MeV S2VT Skip-thoughts</td>
<td>39.50%</td>
<td>7.0482</td>
</tr>
<tr>
<td>Early Fusion of Visual, Arousal Audio, and MSVD S2VT Skip-thoughts</td>
<td>45.45%</td>
<td>7.080</td>
</tr>
<tr>
<td>Early Fusion of Visual, Arousal Audio, and MeV S2VT Skip-thoughts</td>
<td>41.91%</td>
<td>6.54</td>
</tr>
</tbody>
</table>
6.6. Arousal Normalization Experiments

Table 12 details the results of the Arousal Normalization set of experiments. The early fusion ensemble models performed the best, followed closely by the visual classification model. These accuracies are lower than the best accuracy models that were trained using the full development set (Table 1), but the MSE of each model is significantly lower than their original counterparts.

The Mediaeval dataset expands the LIRIS-ACCEDE dataset by augmenting its test set with an additional 1,100 videos. 58% and 27% of these videos are classified as “neutral” and “calm” respectively. However, the full LIRIS-ACCEDE dataset has 64% and 13% of the videos classified as “calm” and “neutral” respectively. Since these datasets were independently annotated and curated, there may have been a disparity in how the ground truth was collected. This may be the cause of the lower arousal accuracies despite the considerably lower MSE.

Table 12: Feature vectors used for Arousal normalization experiments with their Accuracy and MSE.

<table>
<thead>
<tr>
<th>Features</th>
<th>Accuracy</th>
<th>MSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Visual FC7 features</td>
<td>47.48%</td>
<td>0.9676</td>
</tr>
<tr>
<td>Audio FC7 features</td>
<td>35.04%</td>
<td>1.327</td>
</tr>
<tr>
<td>Text Features</td>
<td>38.56%</td>
<td>0.9513</td>
</tr>
<tr>
<td>Audio – Visual Features</td>
<td>48.22%</td>
<td>0.9525</td>
</tr>
<tr>
<td>Audio – Visual -Text Features</td>
<td>48.40%</td>
<td>0.9527</td>
</tr>
</tbody>
</table>
Chapter 7  Conclusion

Affective analysis of videos is a challenging but interesting task. In this study, valence and arousal of videos are analyzed using independent visual and audio streams. Compact visual features are extracted from video frames using the FC7 layer from CNNs. When these features are used to train an SVM, we achieve state of the art valence accuracy. Independent audio models explored extracting handcrafted audio features vs. CNNs. Training Deep CNNs using spectrograms distilled from the audio files gave better results. Class probabilities for the independent models are used to create ensemble models, and they gave results comparable to literature and better valence accuracy than state of the art. Future work in this domain can further fine-tune CNN models to push arousal accuracies over state of the art.

These affect classification models can be used to expand existing social media studies such as Schrading’s [77] work which deals with socially relevant issues. Often, social media posts contain a combination of text and some form of visual media. Studying how users view these multi-modal posts will be of interest to sociologists and psychologists. Market analysts can also use this model to predict product trends, and advertisers can tailor their ads to target and reach their demographics. These models may allow manufacturers better product reviews to get relevant feedback about their products.
Bibliography


