# EMOTIONAL IMPACT OF MOVIES

by

Nihan Karslıoğlu İman

B.S., Computer Engineering, Bilkent University, 2015

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Computer Engineering

Boğaziçi University

2019

EMOTIONAL IMPACT OF MOVIES

APPROVED BY:

Prof. Lale Akarun .....
(Thesis Supervisor)

Assoc. Prof. Albert Ali Salah .....
(Thesis Co-supervisor)

Prof. Fikret Gürgen .....

Assist. Prof. Berk Gökberk .....

DATE OF APPROVAL: 25.07.2019

*to my angel sister Seran Cihan*

# ACKNOWLEDGEMENTS

First and foremost, I would like to deeply thank Prof. Lale Akarun and Assoc. Prof. Albert Ali Salah. This thesis wouldn't have come into existence without their invaluable guidance, motivation, support, and understanding. It is truly a great honor for me to work with them.

I am deeply thankful to Prof. Fikret Gürgen and Assist. Prof. Berk Gökberk for accepting to be on my thesis committee and their precious feedback.

I am deeply grateful to Assist. Prof. Heysem Kaya for his precious guidance, support, and ideas.

I would like to thank Yasemin Timar for her collaboration with me in this study. I want to thank İlhan Adıyaman, Mehran Bazargani, Burak Kurutmaz, Barış Demiröz, Alp Kındıroğlu, Furkan Gürpınar, Cihan Camgöz, and Batıkan Türkmen for their help throughout my master program and providing a nice studying environment. I also would like to express my thanks to my friends Ceren, Gözde, Ali Burak, Hüseyin, and many others in Bilkent University for their help in my studies during my undergraduate program and their friendship. I am thankful to my manager Fatih Yazıcı in Cybersoft for his support.

I owe thanks to many dogs and cats living on the campus of Boğaziçi University for their friendship and unconditional love.

I am eternally grateful to my precious family consisting of my mother Ayşe, father İrfan, sisters Cihan and Hilal, brother Atahan, nephews Mehmet and Ata for their unconditional love. I am especially thankful to my mother and my father for their huge guidance and endless support for all my education life.

Finally, I would like to express my heartfelt gratitude to my precious husband Mustafa for his encouragement, understanding, endless patience, guidance, and love. He has always stood by me, even the hardest moments of my life. It is undeniable that he has a huge share of whatever I have achieved in my graduate program.

# ABSTRACT

## EMOTIONAL IMPACT OF MOVIES

Video emotion estimation has been an active area of research in recent decades. This study area is crucial for personalized content delivery, video recommendation, video summarization, and inappropriate scene censorship. In this thesis, we study movie emotion estimation. The problem is explored with regards to feature extraction, feature summarization, feature selection, regression, and data synthesis. The audio and image features extracted from videos are Mel-frequency cepstral coefficients, hue saturation histogram, dense scale-invariant feature transform, facial action units, and the sixth fully connected layer's feature of VGG network. The features are summarized via descriptive statistics functionals and Fisher vector encoding. Feature selection technique based on canonical correlation analysis is applied to the features. Extreme learning and support vector machines are used as regression techniques. We construct train and validation set examining movie scene and movie mood distributions in the dataset. We synthesis data for the minority classes in the unbalanced dataset. Feature and score level fusion techniques are applied to the best features. We use smoothing techniques for sudden changes of consecutive labels. Our approach is evaluated on the Emotional Impact of Movies Task's dataset provided by Mediaeval 2017 organization; the movies of the dataset are selected from a challenging Liris-Accede dataset. Fusion of facial action units and hue saturation histogram features provide the best arousal results. Score fusion of sixth fully connected layer feature of VGG and hue saturation histogram achieve best valence results. We obtain good Pearson Correlation Coefficient results for the best valence result. In the best arousal model, some slopes in predictions are in line with slopes of actual label graph.

# ÖZET

## FİLMLERİN DUYGUSAL ETKİSİ

Videolarda duygu tahmini, son yıllarda aktif bir araştırma alanıdır. Bu çalışma alanı filmlerde kişiselleştirilmiş duygusal analizi, video önerisi ve uygunsuz sahne sansürü için önemlidir. Bu tezde, filmlerde duygu tahminini inceliyoruz. Problem öznitelik çıkarımı, öznitelik özetleme, öznitelik seçimi, regresyon ve veri sentezi açısından incelenmiştir. Videolardan çıkarılan ses ve görüntü özellikleri Mel-frekans kepstrum katsayıları, ton doygunluğu histogramı, yoğun ölçek değişmezlikli öznitelik dönüşümü, yüz hareket birimleri ve VGG ağının altıncı tam bağlantılı katmanından çıkarılmıştır. Öznitelikler, tanımlayıcı istatistikler fonksiyoneller ve Fisher vektör kodlaması ile özetlenmiştir. Kanonik korelasyon analizine dayalı öznitelik seçimi deneylenmiştir. Aşırı öğrenme ve destek vektör makineleri regresyon teknikleri olarak kullanılmıştır. Film sahnesi ve film ruh hali dağılımı incelenerek eğitim ve doğrulama seti kurulmuştur. Sınıf dengesizliği sorunu, veri setindeki azınlık sınıfları için sentetik veriler üretilerek çözülmüştür. Birleştirme teknikleri en iyi sonuç aldığımız özniteliklere skor ve öznitelik seviyesinde uygulanmıştır. Ardışık etiketlerin ani değişiklikleri için yumuşatma filtresi kullanıldı. Bizim yaklaşımımız, Mediaeval 2017 organizasyonu tarafından sağlanan Filmlerin Duygusal Etkisi çalışmasında test edildi. Bu veri seti, zorlu bir veriseti olan Liris-Accede veri setinden seçilerek oluşturulmuştur. Duygu olumluluğu modelinin en iyi sonucu, yüz hareket birimleriyle, renk doygunluğu histogramının skor füzyonuyla elde edildi. Duygu şiddetinin en iyi sonucu, VGG'nin altıncı tam bağlı katman özelliğiyle renk doygunluğu histogramının öznitelik füzyonuyla elde edildi. Duygu olumluğunda iyi bir sonuca ulaştık. Elde ettiğimiz en iyi duygu şiddeti modelinde, tahminlerin bazı eğimleri ile gerçek etiketleri arasında pozitif korelasyonlar görüldü.

# TABLE OF CONTENTS

ACKNOWLEDGEMENTS . . . . . iv
ABSTRACT . . . . . vi
ÖZET . . . . . vii
LIST OF FIGURES . . . . . x
LIST OF TABLES . . . . . xiii
LIST OF SYMBOLS . . . . . xv
LIST OF ACRONYMS/ABBREVIATIONS . . . . . xvi
1. INTRODUCTION . . . . . 1
1.1. Motivation . . . . . 1
1.2. Related Work . . . . . 2
1.3. Research Questions . . . . . 5
1.4. Contributions . . . . . 5
1.5. Outline of the Thesis . . . . . 5
2. METHODOLOGY . . . . . 6
2.1. Emotion and affective components of movies . . . . . 6
2.2. Feature Extraction . . . . . 10
2.2.1. Color Features . . . . . 10
2.2.2. Facial Features . . . . . 10
2.2.3. Scale Invariant Feature Transform . . . . . 12
2.2.4. Mel Frequency Cepstral Coefficients . . . . . 15
2.2.5. Deep Learning Features . . . . . 19
2.3. Unsupervised Clustering . . . . . 19
2.3.1. Gaussian Mixture Model . . . . . 19
2.4. Feature Summarization . . . . . 21
2.4.1. Fisher Vector Encoding . . . . . 21
2.5. Feature Selection . . . . . 23
2.5.1. Random Sample versus Labels Canonical Correlation Analysis Filter . . . . . 23
2.6. Oversampling of Data . . . . . 24

2.6.1. Adaptive Synthetic Sampling Approach for Imbalanced Dataset 24
2.7. Classification . . . . . 24
2.7.1. Support Vector Machine . . . . . 24
2.7.1.1. Kernel Trick . . . . . 27
2.7.2. Extreme Learning Machine . . . . . 29
3. PROPOSED METHODOLOGY . . . . . 31
4. EXPERIMENTS AND RESULTS . . . . . 35
4.1. Dataset . . . . . 35
4.1.1. Liris-Accede . . . . . 35
4.1.2. Development Dataset Examination . . . . . 36
4.2. Experiments . . . . . 39
4.2.1. Descriptive Statistics Functions Application on Unbalanced Dataset 39
4.2.2. Descriptive Statistics Functions Application and Classification on Balanced Dataset . . . . . 41
4.2.3. Descriptive Statistics Functions Application and Regression on Balanced Dataset . . . . . 42
4.2.4. Descriptive Statistics Definitions and Feature Selection Application 45
4.2.5. PCA and Fisher Vector Applied Regression Models . . . . . 46
4.2.6. Fisher Encoding and Feature Selection Applied Regression Models 48
4.2.7. Plot of Performance Results Experimented with Single Feature . 49
4.2.8. Score Fusion . . . . . 49
4.2.9. Feature Fusion . . . . . 53
4.3. Discussions . . . . . 53
5. CONCLUSIONS . . . . . 58
REFERENCES . . . . . 59
APPENDIX A: MOVIE DISTRIBUTION IN DEVELOPMENT SET . . . . . 65
APPENDIX B: PLOT OF ACTUAL AND PREDICTED LABELS . . . . . 68

# LIST OF FIGURES

Figure 2.1. The pipeline of our system as a summary. . . . . 6

Figure 2.2. Emotions in arousal-valence space. . . . . 8

Figure 2.3. Samples for each quarter of arousal-valence space. . . . . 9

Figure 2.4. Action units in FACS. . . . . 11

Figure 2.5. Upper face action units. . . . . 11

Figure 2.6. Lower face action units. . . . . 13

Figure 2.7. Scale Space and Difference-of-Gaussian. . . . . 15

Figure 2.8. Maxima and minima location after differences of Gaussians. . . . . 16

Figure 2.9. Feature descriptor step after orientation assignment. . . . . 16

Figure 2.10. The basic structure of an artificial neural network. . . . . 20

Figure 2.11. A classic example for a linear classifier and an example with complex structure data. . . . . 25

Figure 3.1. The pipeline of our system. . . . . 31

Figure 4.1. Development set information. . . . . 38

Figure 4.2. Total number of arousal annotations for train and validation sets. 38

Figure 4.3. Total number of valence annotations for train and validation sets. 39

Figure 4.4. Results of MFCC model for arousal space. . . . . 40

Figure 4.5. Results of VGG-FC6 model for valence space. . . . . 41

Figure 4.6. Arousal and valence classification results. . . . . 43

Figure 4.7. Arousal results with action unit models with non-smoothing and smoothing. . . . . 44

Figure 4.8. Arousal results. . . . . 50

Figure 4.9. Valence results. . . . . 51

Figure 4.10. Best results for arousal. . . . . 56

Figure 4.11. Best results for valence. . . . . 56

Figure A.1. Movie information in case of arousal space in the train set. . . . . 65

Figure A.2. Movie information in case of arousal space in the validation set. . 66

Figure A.3. Movie information in case of valence space in the train set. . . . . 66

Figure A.4. Movie information in case of valence space in the validation set. . 67

Figure B.1. Results with descriptive functions and ELM. . . . . 69

Figure B.2. Results with descriptive functions and SVM. . . . . 70

Figure B.3. Results with descriptive functions, feature selection and ELM. . . 71

Figure B.4. Results with descriptive functions, feature selection and SVM. . . 72

Figure B.5. Results with PCA, FV and ELM. . . . . 73

Figure B.6. Results with PCA, FV and SVM. . . . . 74

Figure B.7. Results with PCA, FV, CCA and ELM. . . . . 75

Figure B.8. Results with PCA, FV, CCA and SVM. . . . . 76

Figure B.9. Score fusion applied to ELM results. . . . . 77

Figure B.10. Score fusion applied to SVM results. . . . . 78

Figure B.11. Feature fusion results with ELM. . . . . 79

Figure B.12. PCA applied feature fusion results with ELM. . . . . 80

Figure B.13. Feature fusion results with SVM. . . . . 81

Figure B.14. PCA applied feature fusion results with SVM. . . . . 82

# LIST OF TABLES

Table 4.1. Arousal/valence prediction results of descriptive statistics functions and ELM. . . . . 39

Table 4.2. The performance results with descriptive statistics functions and ELM. . . . . 44

Table 4.3. The performance results with descriptive statistics functions and SVM. . . . . 45

Table 4.4. The performance results with descriptive statistics functions, SLCCA-Rand, and ELM. . . . . 45

Table 4.5. The performance results with descriptive statistics functions, SLCCA-Rand, and SVM. . . . . 46

Table 4.6. The performance results with PCA, FV, and ELM. . . . . 47

Table 4.7. The performance results with PCA, FV, and SVM. . . . . 47

Table 4.8. The performance results with PCA, FV, SLCCA-Rand, and ELM. 48

Table 4.9. The performance results with PCA, FV, SLCCA-Rand and SVM. . 49

Table 4.10. Test performance results with score fusion for valence. . . . . 52

Table 4.11. Test performance results with score fusion for arousal. . . . . 52

Table 4.12. Test performance results with feature fusion for valence. . . . . 53

Table 4.13. Test performance results with feature fusion for arousal. . . . . 54

# LIST OF SYMBOLS

| | |
|---|---|
| D(x, y, $\sigma$) | Difference of Gaussians |
| f | Frequency |
| G(x, y, $\sigma$) | Gaussian Blur Operator |
| h(n) | Hamming Window |
| $K(x_t, x)$ | Kernel Function |
| L(x, y, $\sigma$) | Blurred Image |
| M | Mel-Scale |
| $P_i(k)$ | Power Spectral of k |
| $p(X\|\lambda)$ | Probability Density Function |
| $s(n)$ | Signal of an n'th Frame |
| | |
| $\alpha^t$ | Lagrange Multiplier |
| $\xi$ | Slack Variable |
| $\pi$ | Phi Number |
| $\sigma$ | Amount of Blur |

# LIST OF ACRONYMS/ABBREVIATIONS

| | |
|---|---|
| ADASYN | Adaptive Synthetic Sampling Approach for Imbalanced Learning |
| ANN | Artificial Neural Network |
| AU | Action Units |
| BLSTM | Bidirectional Long-Short Term Memory |
| CCA | Canonical Correlation Analysis |
| CNN | Convolutional Neural Network |
| DCT | Discrete Cosine Transform |
| Dense-SIFT | Dense Scale Invariant Feature Transform |
| DFT | Discrete Fourier Transform |
| ELM | Extreme Learning Machine |
| EM | Expectation-Maximization |
| FACS | Facial Action Coding System |
| FV | Fisher Vector |
| GMM | Gaussian Mixture Model |
| HSH | Hue-Saturation Histogram |
| HSV | Hue-Saturation-Value |
| LSTM | Long-Short Term Memory |
| MFCC | Mel-Frequency Cepstral Coefficients |
| MSE | Mean Square Error |
| PCA | Principal Component Analysis |
| PCC | Pearson Correlation Coefficients |
| pdf | Probability Density Function |
| PHOW | Pyramid Histogram Of Visual Words |
| RBF | Radial-Basis Function |
| RGB | Red-Green-Blue |
| SIFT | Scale Invariant Feature Transform |
| SLCCA-Rand | Random Sample versus Labels Canonical Correlation Analysis Filter |

| | |
|---|---|
| SLFN | Single Hidden Layer Feed-Forward Network |
| SMOTE | Synthetic Minority Over-Sampling Technique |
| SVM | Support Vector Machine |
| SVR | Support Vector Regression |
| VGG-FC6 | Visual Geometry Group Neural Network - Sixth Fully Connected Layer |
| VGG-Net | Visual Geometry Group Neural Network |

# 1. INTRODUCTION

## 1.1. Motivation

The amount of available videos has rapidly increased as a result of the rising popularity of online video services. This improvement makes it easier to attain, create, and share a variety of videos. People use videos to communicate, share information, and pass time. Youtube, being just one of the online video service providers, is visited by over 1.9 billion logged-in users each month according to the data published in the year 2019. Every day, over a billion hours of videos are watched and billions of views are generated in a total of 80 different languages on Youtube [1].

With the increase in the number of videos, videos have become a big part of our daily lives and have led to some problems. Audiences and video providers may need to categorize videos as news, films, sports, animations, education and such. Movie genre classification and video indexing may take a long time. Inappropriate scene censorship is another issue because parents may not be able to keep track of what their children watch. Children may be exposed to inappropriate videos containing elements of fear, violence, porn and such. Moreover, video recommendation systems is a necessity for online video service providers to tailor the content according to customer's needs. Manually solving these problems with the enormous mass of videos is hard and time-consuming.

Video emotion estimation is an active research field for video analysis. Its purpose is automatically predicting emotions elicited by videos. There are some main problems on this subject. The subjectivity of emotions is a challenge in this research area. The emotion of video is a subjective concept. For instance, while watching a basketball match, the fan of the winning team will be happy; but the fan of the loser team will be sad. Similarly, watching a horror movie, one person may find some scenes scary whereas another person does not. Each viewer of a movie feels their unique feelings because everybody's ideas, experiences, culture, sex, and age are different. Therefore

the emotions they exhibit towards videos vary. This makes it harder to identify the emotion. Another problem in video affect recognition is that videos vary in many aspects. They change in terms of topic, purpose, shooting type, storytelling, and such. Moreover, not every film aims to transfer emotions to the viewers. These make it harder to get exact results in emotion prediction in videos.

Besides the difficulties to identify video emotion, there are frequently used video editing techniques in cinematography literature. Video makers or film directors use these techniques on their videos or films to stimulate certain feelings or manipulate the audience's emotion. Some of these techniques are actors' expressions and voice, sound and image effects, music, colors, lighting, camera and object/subject movements, scene decorations and such [2, 3].

## 1.2. Related Work

There are different approaches to estimating video emotion concerning feature representation and affective modeling. We first focus on the features investigated in the literature. According to Hanjalic *et al.*, the arousal and valence model of videos are represented by low-level features [4]. These are motion, color, sound energy and shot length. Soleymani *et al.* use audio and visual features. They use Mel-frequency cepstral coefficients (MFCC), linear predictive cepstral coefficient, zero-crossing rate, silence ratio, lighting key, color variance, shot change rate, hue-saturation-value histograms [5]. Probabilistic dependencies between lighting, color, and average short duration are explored in [6].

Soleymani *et al.* use textual features from the subtitles of films in [7]. Shot type, the closeness of the camera to the main recorded subject in a scene, is studied in [8]. SentiBank library [9], gives a sentiment score based on adjective-noun pair extracted from static images, is integrated in [10].

The features mentioned so far are static ones. There are also dynamic features such as dense trajectories and motion boundary descriptors [10, 11].

Physiological features which are galvanic skin response, blood flow, respiration, electromyography signals, eye blinking rate, skin temperature of the audience are acquired during watching each scene of the movie in [5]. In this work, audiences also tag their feelings for each scene. This is an interruptive process and users might misrepresent their feelings while tagging. Physiological features are dynamic responses of users. The correlation between physiological features, low-level features, and self-assessment labels are analyzed in this study.

There are different approaches for feature extraction in movie emotion estimations. [12] proposes a supervised dimensionality reduction algorithm called Arousal-Valence Discriminant Preserving Embedding to learn intrinsic factors in movies conveying emotion along with arousal-valence space.

Contextual information such as the genre of movie has an impact on the emotion of movie scenes. Soleymani *et al.* take account of context feature on [7] integrating probabilistic graphical model of the feature into movie emotion analysis.

The domain knowledge about emotion is used for the problem. Emotion values in consecutive frames do not change abruptly but low-level feature values vary significantly in consecutive frames in video segments. Kaiser window which smooths low-level features are used in [4,13] to solve abrupt changes in feature values of sequential frames. The lingering affect of emotion in movies modeling the gradually amplifying or decaying emotional flow of movies is worked on [14].

There are different regression models in video emotion estimation. Support vector regression is a method used to model affective content of movies [11, 12, 14–17]. Random forest is used in [11]. Deep learning features and models are also studied in the movie emotion estimation problem. Deep features extracted from VGG-Net [18] are modeled on Long-Short Term Memory (LSTM) [15]. LSTM and Bidirectional Long-Short Term Memory (BLSTM) are used in [17]. A neural network model is performed in [12]. Temporal and spatial Convolutional Neural Network (CNN) features are used in [16]. MFCC and color features are used as input for CNN to get their mid-

level representations [19]. Other works are extracting CNN features for video emotion estimation [11, 17, 20].

Liris-Accede dataset [21] is a movie emotion estimation dataset. One of the challenges of the dataset is that classes are unbalanced. SMOTE (Synthetic Minority Over-Sampling Technique) [22] is adapted to video affective estimation system generating new samples for the fear class which is the minority in the dataset [15]. Biased discriminant embedding is another approach to find more informative data for under-represented classes in the dataset [23].

Feature fusion techniques are used to improve movie affect estimation. Multi-kernel learning approach is investigated in [14]. Feature fusions such as early and late fusions are performed for multiple modalities in [11, 17, 20].

The MediaEval is a benchmark organization for multimedia evaluation. One of the tasks of the organization aims to automatically detect emotions evoked by videos. The task is to continuously predict the valence and arousal values during the movie. In year 2016, [24] extracts MFCC, dense SIFT, deep learning feature, hue saturation histogram and improved dense trajectories. The features are summarized by Principal Component Analysis (PCA) and Fisher vector encoding (FV). Support vector regression (SVR) is used as a regression technique. The other work [11] uses a similar approach in feature representation. Random Forest is applied as another regression method in this work. These two works obtain good results in Mediaeval 2016.

We participated in the MediaEval 2017 challenge. We constructed a baseline of the good studies in the MediaEval 2016 and experimented with different techniques in our study to improve the results.

## 1.3. Research Questions

The main research questions we ask in this thesis are as follows:

(i) Which image and video features are useful to estimate emotions evoked by a movie clip?
(ii) Which classifiers are adequate for estimating emotions elicited by a movie clip?
(iii) Can different fusion techniques be effectively used in movie emotion estimation?

## 1.4. Contributions

In this work, we propose a movie emotion estimation system working on Liris-Accede dataset. This unbalanced dataset is made balanced with ADASYN and various models are applied to this dataset. HSH, AU, MFCC, Dense-SIFT, and VGG-FC6 features are extracted from the videos. Descriptive statistics and Fisher vector encoding are used as feature summarization. Feature selection method SLCCA-Rand is also applied to feature vectors. ELM and SVR are performed as different regression techniques. Score and feature fusion are applied to the best models.

## 1.5. Outline of the Thesis

The rest of the thesis is organized as follows: Chapter 2 describes the methods used in our study. Chapter 3 describes a proposed system for movie emotion estimation. Chapter 4 introduces the experiments with the dataset and discusses our study. Finally, Chapter 5 concludes the report.

# 2. METHODOLOGY

Our approach to estimating emotional responses of movie segments follows the scheme depicted in Figure 2.1. We deal with each component of this approach individually, including feature extraction, feature summarization, feature selection, classification, and fusion. In this figure, the methods in bold (like Feature Summarization) are used in each experiment whereas the methods in light grey (like Feature Selection) are used in some of the experiments. They are explained in detail in Section 3.

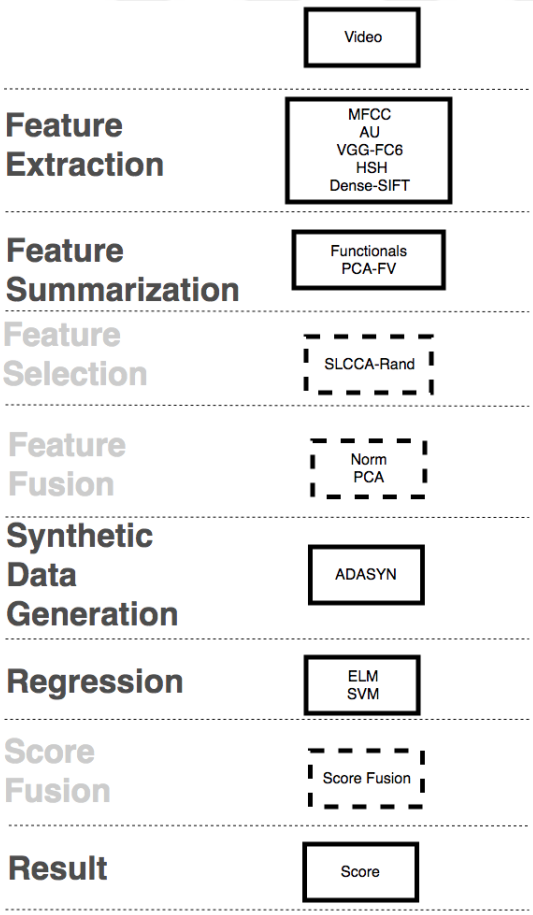


Figure 2.1. The pipeline of our system as a summary.

## 2.1. Emotion and affective components of movies

In this section, we define the key concepts and dichotomies related to emotion and its measurement for multimedia. Emotion or affect can be represented with some

basic categories such as happiness, sadness, and fear [25,26]. Appraisal theory, another emotion theory, extracts emotion from appraisals like evaluations and explanations of events [27]. Russell *et al.* propose a dimensional model to identify affect [28]. In this model, there are two dimensions called arousal and valence. The vertical axis corresponds to arousal value; the horizontal axis corresponds to valence value. While arousal measures calmness and excitement level of affect, valence is related to positivity and negativity of emotion. In Figure 2.2, 28 affect words are coordinated with direct circular scaling. For instance, annoyed emotion is in the space of high arousal and negative valence.

There are some movie scenes from the Liris-Accede dataset in Figure 2.3. Each scene is annotated with its corresponding quarter of arousal-valence space. Each scene has some components which are placed in them to manipulate the feelings of the audience. We will try to explain each of them.

The upper left scene is a scary one. The scene is about a woman shooting a zombie in the scene. The woman is scared of the zombie. The zombie shouts at her. She is breathing heavily. Finally, she shoots the zombie. Apart from the action, there are lighting features such as dim light to stimulate fear.

The lower left scene is a depressing one. The man in the scene is trying to solve a stressful problem on the computer. He is nervous; his face, voice, and gestures are used to convey this information.

The upper right scene is an exciting one. A boy defends himself in front of a jury. When he wins the argument, everyone in the room starts to clap. Victory march plays on the background. All these elements are used to stimulate excitement.

The lower right scene is a relaxed one. We see an island, the sea, and the sun while a relaxing song is played. The woman smiles. All these features are generally attributed to relaxing environments.

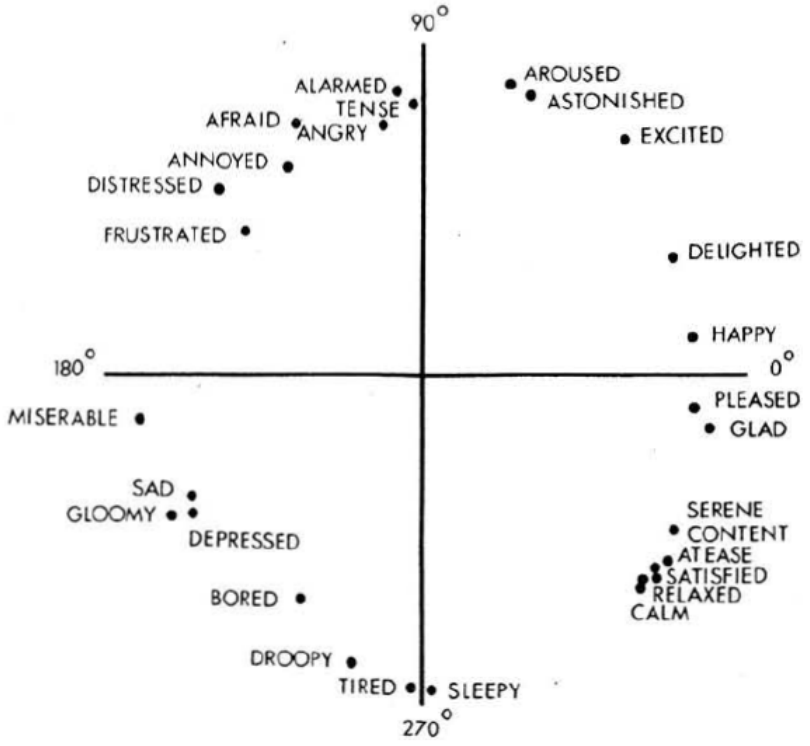


Figure 2.2. Emotions in arousal-valence space [28].

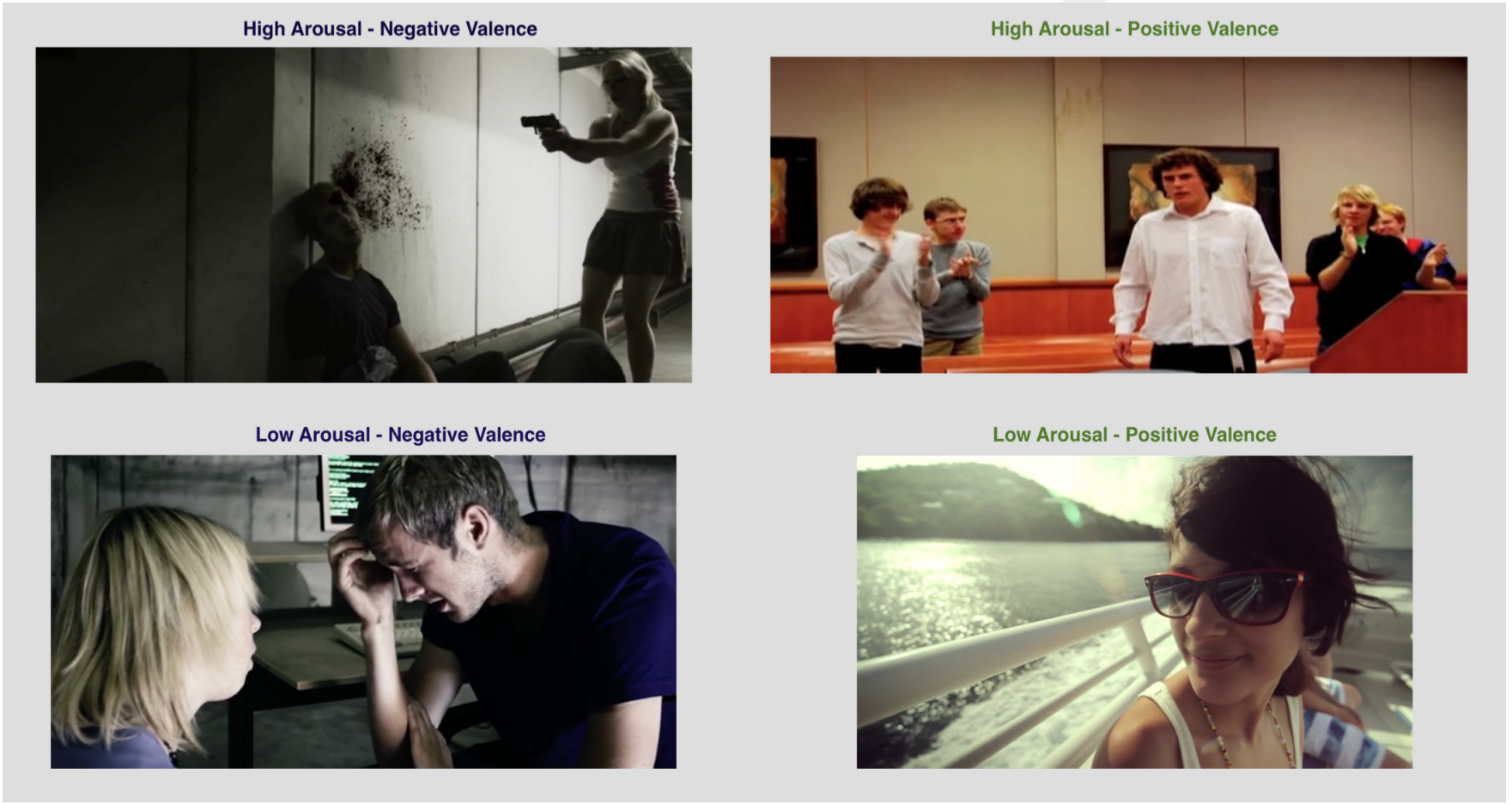


Figure 2.3. Samples for each quarter of arousal-valence space.

## 2.2. Feature Extraction

### 2.2.1. Color Features

Color is a significant property of objects. Multiple models define the color. One of them is the Red Green Blue (RGB) model. In the RGB model, color is defined based on the intensity of three main primary color components (red, green, blue). This model is also called the additive model.

The Hue Saturation Value (HSV) is another color model. It is a three-dimensional coordinate system. Dimensions are named as hue, saturation, and value. Hue defines color in different wavelengths. It is encoded in an angle ranging from 0 to $2\pi$. Saturation (chroma) is related to purity or intensity of color. It is in the range of [0,1]. Color is the clearest in the fullest saturation. There is no color in the lowest saturation. Value defines the brightness of the color. It is in the range of [0,1], too. Black has a brightness of zero whereas white has a brightness of one. 0.5 is pure color.

### 2.2.2. Facial Features

Facial expressions are an important part of non-verbal interaction. Facial expressions are formed by motions of muscles behind facial skin. Ekman and Friesen developed an anatomical system which is the facial action coding system (FACS). FACS breaks down facial expressions into actions of individual muscles. A set of muscle positions that forms a well-known expression is called an action unit. FACS defines 44 different action units. Alongside action units, FACS makes use of head and eye movements for expression recognition. Each action unit has a unique code matching one or more muscle movements [29]. The codes for action units, head movements, and eye movements are in Figure 2.4.

In Figure 2.5, there are some expression samples for the upper face action units. Some of the expressions are encoded with single action units whereas others are encoded with combined action units. Some lower face action units are shown in Figure 2.6.

| AU number | Descriptor | Muscular Basis |
|---|---|---|
| 1. | Inner Brow Raiser | Frontalis, Pars Medialis |
| 2. | Outer Brow Raiser | Frontalis, Pars Lateralis |
| 4. | Brow Lowerer | Depressor Glabellae, Depressor Supercilli; Corrugator |
| 5. | Upper Lid Raiser | Levator Palpebrae Superioris |
| 6. | Cheek Raiser | Orbicularis Oculi, Pars Orbitalis |
| 7. | Lid Tightener | Orbicularis Oculi, Pars Palebralis |
| 9. | Nose Wrinkler | Levator Labii Superioris, Alaeque Nasi |
| 10. | Upper Lip Raiser | Levator Labii Superioris, Caput Infraorbitalis |
| 11. | Nasolabial Fold Deepener | Zygomatic Minor |
| 12. | Lip Corner Puller | Zygomatic Major |
| 13. | Cheek Puffer | Caninus |
| 14. | Dimpler | Buccinator |
| 15. | Lip Corner Depressor | Triangularis |
| 16. | Lower Lip Depressor | Depressor Labii |
| 17. | Chin Raiser | Mentalis |
| 18. | Lip Puckerer | Incisivii Labii Superioris; Incisivii Labii Inferioris |
| 20. | Lip Stretcher | Risorius |
| 22. | Lip Funneler | Orbicularis Oris |
| 23. | Lip Tightener | Orbicularis Oris |
| 24. | Lip Pressor | Orbicularis Oris |
| 25. | Lips Part | Depressor Labii, or Relaxation of Mentalis or Orbicularis Oris |
| 26. | Jaw Drop | Masetter; Temporal and Internal Pterygoid Relaxed |
| 27. | Mouth Stretch | Pterygoids; Digastric |
| 28. | Lip Suck | Orbicularis Oris |

| AU number | FACS name |
|---|---|
| 19. | Tongue out |
| 21. | Neck Tightener |
| 29. | Jaw Thrust |
| 30. | Jaw Sideways |
| 31. | Jaw Clencher |
| 32. | Lip Bite |
| 33. | Cheek Blow |
| 34. | Cheek Puff |
| 35. | Cheek Suck |
| 36. | Tongue Bulge |
| 37. | Lip Wipe |
| 38. | Nostril Dilator |
| 39. | Nostril Compressor |
| 41. | Lid Droop |
| 42. | Slit |
| 43. | Eyes Closed |
| 44. | Squint |
| 45. | Blink |
| 46. | Wink |

Figure 2.4. Left: Single action units in FACS. Right: Other movements defined in FACS [29].

| *NEUTRAL* | AU 1 | AU 2 | AU 4 | AU 5 |
|---|---|---|---|---|
|  |  |  |  |  |
| Eyes, brow, and cheek are relaxed. | Inner portion of the brows is raised. | Outer portion of the brows is raised. | Brows lowered and drawn together | Upper eyelids are raised. |
| AU 6 | AU 7 | AU 1+2 | AU 1+4 | AU 4+5 |
|  |  |  |  |  |
| Cheeks are raised. | Lower eyelids are raised. | Inner and outer portions of the brows are raised. | Medial portion of the brows is raised and pulled together. | Brows lowered and drawn together and upper eyelids are raised. |
| AU 1+2+4 | AU 1+2+5 | AU 1+6 | AU 6+7 | AU 1+2+5+6+7 |
|  |  |  |  |  |
| Brows are pulled together and upward. | Brows and upper eyelids are raised. | Inner portion of brows and cheeks are raised. | Lower eyelids cheeks are raised. | Brows, eyelids, and cheeks are raised. |

Figure 2.5. Upper face action units [30].

### 2.2.3. Scale Invariant Feature Transform

Automatically detecting objects and describing an image is a challenging area in computer vision research. The reason is that scale space, rotation, illumination, and viewpoint of a single object vary from image to image. Lowe *et al.* [31] propose an algorithm called scale-invariant feature transform (SIFT). It is a feature extractor and a descriptor aiming for a robust solution to the above-mentioned problems. Well localized and distinctive features extracted by SIFT give high accuracy in image detection and description.

SIFT algorithm has four main steps. These are, in their respective order, scale-space extrema detection, keypoint localization, orientation assignment, and keypoint descriptor. While the first three steps are to identify keypoints in an image, the last step is to describe the chosen keypoints. The following is an explanation of the implementation of this algorithm.

Scale-space extrema step applies the cascading filtering approach. This approach firstly blurs out an original image with a Gaussian filter with different blurring parameters. This process is shown in Figure 2.7. The left part of the figure shows the scale-space extrema detection step. There are octaves and scales. In the first octave, there are five images. One is the original image. The other four images are generated by blurring out processes with various blurring parameters.

Gaussian blurring (convolution) formula is,

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \tag{2.1}$$

where $G(x, y, \sigma)$ is defined as:

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-\left(x^2 - y^2\right)/2\sigma^2} \tag{2.2}$$

| *NEUTRAL* | AU 9 | AU 10 | AU 12 | AU 20 |
|---|---|---|---|---|
|  |  |  |  |  |
| Lips relaxed and closed. | The infraorbital triangle and center of the upper lip are pulled upwards. Nasal root wrinkling is present. | The infraorbital triangle is pushed upwards. Upper lip is raised. Causes angular bend in shape of upper lip. Nasal root wrinkle is absent. | Lip corners are pulled obliquely. | The lips and the lower portion of the nasolabial furrow are pulled pulled back laterally. The mouth is elongated. |
| AU15 | AU 17 | AU 25 | AU 26 | AU 27 |
|  |  |  |  |  |
| The corners of the lips are pulled down. | The chin boss is pushed upwards. | Lips are relaxed and parted. | Lips are relaxed and parted; mandible is lowered. | Mouth stretched open and the mandible pulled downwards. |
| AU 23+24 | AU 9+17 | AU9+25 | AU9+17+23+24 | AU10+17 |
|  |  |  |  |  |
| Lips tightened, narrowed, and pressed together. | | | | |
| AU 10+25 | AU 10+15+17 | AU 12+25 | AU12+26 | AU 15+17 |
|  |  |  |  |  |
| AU 17+23+24 | AU 20+25 | | | |
|  |  | | | |

Figure 2.6. Lower face action units [30].

In this formula, I is an image. x and y are pixel coordinates. L is one of the blurred images. G is Gaussian blur operator, $\sigma$ is the amount of blur and * is convolution operator. After the filtering process, the original image is resized to quarter of its original size. This step is called second scale step. Same blurring process is applied recursively to resulting image until the last scale step.

Key point localization step is shown in Figure 2.7. The difference between each pair of consecutive images is called the difference of Gaussians.

$$\begin{aligned} D(x, y, \sigma) &= (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) \\ &= L(x, y, k\sigma) - L(x, y, \sigma) \end{aligned} \tag{2.3}$$

The next step is detecting keypoints. This is done by finding minima and maxima of differences of Gaussians computed in the previous step. This step is illustrated in Figure 2.8. Each pixel marked with an "X" is compared to its neighbors. Neighbors of a pixel are the pixels immediately next to it and the pixels located at the same coordinates in neighboring differences of Gaussians. A pixel has 26 neighbors according to this definition. If the value of pixel "X" is greater or smaller than all of its neighbors, it is chosen as a candidate keypoint. Further details for this step are in [31].

The next step is to assign orientation information to each keypoint. In this part, gradient directions and magnitudes of each key point are collected into a histogram so gradient orientations of the feature are identified. The number of bins in this histogram is 36. Each bin corresponds to 10 out of 360 degrees. The value of bin which is larger than 80% of the sum of orientation vectors is chosen as the orientation of this keypoint.

The last step of SIFT is describing the feature. This step is illustrated in Figure 2.9. In the left part of the figure, the image is divided into 8x8 windows. Gradient magnitudes and orientations are computed within each 8x8 window. These directions

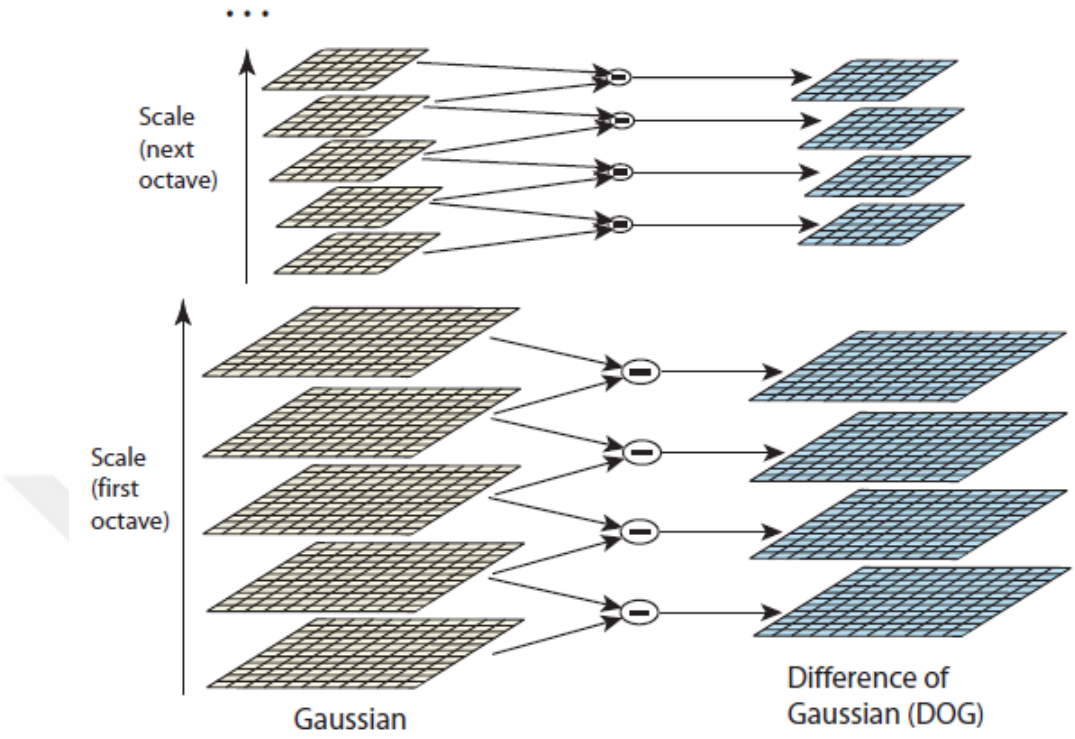


Figure 2.7. Scale Space and Difference-of-Gaussian [31].

are placed into an eight bin histogram as shown in the right part of the figure [31].

There are some extensions to the SIFT algorithm. One of them is Dense-SIFT. It collects more features. Every n'th pixel is a keypoint. Increasing the number of keypoints extracted from a dense region of an image helps to describe image better [32]. PHOW [33] algorithm is a method to improve Dense-SIFT. It extracts the features from multiple scales.

### 2.2.4. Mel Frequency Cepstral Coefficients

We extract the feature of Mel Frequency Cepstral Coefficients (MFCC) [34] for audio information. MFCC models spectral energy distribution. This algorithm is explained in various steps following the exposition [35].

*Pre-Emphasis:* Although speech signal may have rich information about its content, emotions, and such; there may be noise in this information. Before processing

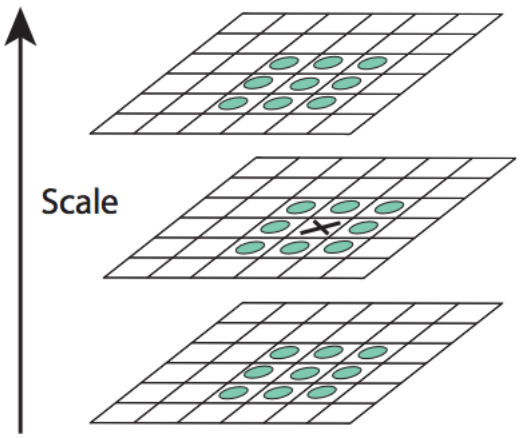


Figure 2.8. Maxima and minima location after differences of Gaussians [31].

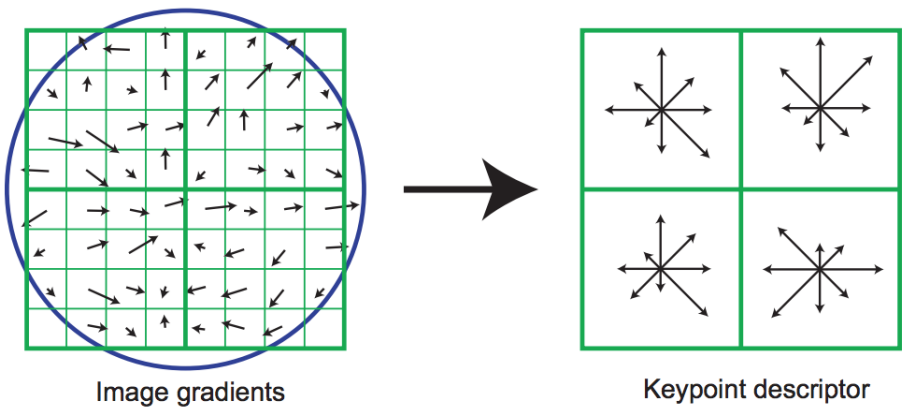


Figure 2.9. Feature descriptor step after orientation assignment [31].

the speech, we should minimize noise. Pre-emphasis step increases the magnitude of high-frequency bands and decreases the magnitude of lower frequency bands which are likely to be noise. The formula is,

$$s(n) = x(n) - a * x(n-1) \tag{2.4}$$

where x(n) is the value of a signal in an n'th time, $a$ is a high pass filter and s(n) is the output of a signal in an n'th time.

*Frame Blocking:* Audio signals are divided into short intervals before working with them. That is because long intervals of an audio signal may contain multiple distinct emotions which are not necessarily related. Shorter intervals are more likely to contain a single emotion. We segment the signal into N overlapping pieces.

*Hamming Window:* After frame blocking, each frame shares its first part with the previous frame and its last part with the next frame. However, continuity between adjacent frames may be disrupted; it may lead to data loss. To minimize these disruptions, the Hamming window algorithm is applied to each frame. It helps to make the signal sharper and more distinct.

*Periodogram estimate of the power spectrum:* An audio signal includes different frequencies. Power spectrum of each frame of a signal is calculated to identify each frequency in the audio signal. This process is formulated as follows. Firstly, discrete Fourier transform (DFT) is applied to the signal. The formula is,

$$S_i(k) = \sum_{n=1}^{N} s_i(n)h(n)e^{-j2\pi kn/N} \qquad 1 \leq k \leq K \tag{2.5}$$

h(n) is Hamming window. i denotes frame number. $s_i(n)$ is frame of signal. $S_i(k)$ is DFT. K is length of DFT. h(n) is calculated as follows,

$$h(n) = (1 - a) - a * cos(2 * \pi * n/(N - 1)), \qquad 0 \leq n \leq N - 1 \tag{2.6}$$

$a$ is a coefficient. Then, power spectral is estimated with periodogram estimate. The formula is,

$$P_i(k) = 1/N * |S_i(k)|^2 \tag{2.7}$$

*Mel-scaled filterbank application:* Periodogram spectral process leads to closely spaced frequencies. It is important to distinguish between them. Periodogram bins are accumulated and summed to get the amount of energy in frequency regions. This is calculated with Mel-space filterbank. Mel is a kind of frequency measurement unit. At this stage, frequency units are converted from Hertz to Mel.

$$M(f) = 1125 \ln(1 + f/700) \tag{2.8}$$

Mel-space filterbank is a set of 20-40 triangular filters applying to periodogram power spectral estimate. Each vector is mostly zeros, only a certain section of the spectrum is non-zero. At the end of the mel-scaled filter bank application, we get how much energy there is in each filterbank.

### 2.2.5. Deep Learning Features

*Artificial Neural Networks:* Artificial Neural Networks (ANN) is an algorithm inspired by the human brain's learning process. The brain consists of neurons; neurons are connected in different ways. These connections constitute networks in the brain. ANN simply imitates this structure as shown in Figure 2.10. It is an interconnected group of nodes like neurons in the brain. Nodes are connected with links. Links have associated weights. These weights adjust to the learning process of ANN. Input layer communicates with the outside world to provide information. Hidden layer processes the inputs obtained by the previous layer, and performs transformation and computations from the input layer to the output layer. Output layer transfers the information to the outside world. This network is called the Single Hidden Layer Feed Forward Network (SLFN). There may be more hidden layers in an ANN. More details about this algorithm can be found in sources like [36].

*VGG Neural Network:* Deep learning neural network is an artificial neural network. It differs from ANN in the number of layers it has. One of the popular deep learning neural models is Convolutional Neural Networks (CNN). CNN has different kinds of layers such as convolutional, pooling, and fully connected layers. One of the well-known studies using CNN is [37]. VGG Neural Network is an extension to CNN [18]. It contains more layers than CNN. This architecture is trained using ImageNet [38], and it is used for the image classification task. We can use the same network for other tasks, too. This is called fine-tuning.

## 2.3. Unsupervised Clustering

### 2.3.1. Gaussian Mixture Model

Gaussian mixture model (GMM) is a probabilistic model to cluster features. GMM has some differences from the well-known clustering algorithm K-means [39]. K-means is a hard clustering algorithm which assigns each element to only one cluster. K-means clusters have strict boundaries and do not overlap. However, GMM is

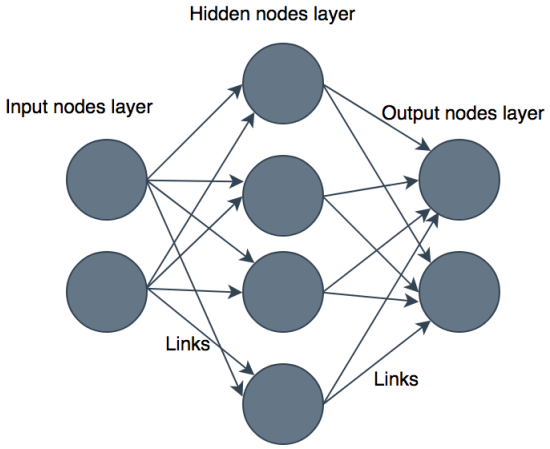


Figure 2.10. The basic structure of an artificial neural network.

a soft clustering algorithm which assigns elements to multiple clusters with varying confidence levels. Each GMM cluster is a probability distribution.

GMM model assumes that underlying data contain several Gaussian distributions. The density function of a mixture of K multivariate Gaussian distributions is defined as,

$$p(\mathbf{x}|\Theta) = \sum_{k=1}^{K} \pi_k p(\mathbf{x}_i|\mu_k, \Sigma_k) \tag{2.9}$$

$\pi_k$ is the prior probability of a Gaussian distribution. $\mu_k$ and $\Sigma_k$ are respectively the mean and covariance of each Gaussian distribution. $\Theta$ is the model parameter of $\{\pi_k, \mu_k, \Sigma_k; k = 1, \ldots, K\}$.

*Expectation-Maximization:* It is an algorithm to find the mean and covariance parameters of GMM. Expectation maximization (EM) is similar to K-means. Since the mean and covariance parameters of Gaussian components in GMM are unknown, it first assigns random values to these parameters. Then, the probability of data assignments to each cluster is estimated. This process is iteratively executed with expectation and maximization steps until a convergence criterion.

## 2.4. Feature Summarization

### 2.4.1. Fisher Vector Encoding

Fisher vector is an encoding technique to represent a sequence of features in a fixed vector [40]. It is a derivation of Fisher kernel framework [41].

*Fisher kernel:* There are generative and discriminative approaches in pattern recognition. While generative technique considers on modeling class-based probability density functions, discriminative techniques focus on classification. Fisher kernel combines the strength of the two approaches. It characterizes a signal with a gradient vector from a probability density function (pdf) [40]. This gradient vector characterizes sample $X = x_t, t = 1...T$ which is defined as,

$$\nabla_\lambda \log p(X|\lambda). \tag{2.10}$$

$\lambda$ is the parameter of a pdf. This is the gradient of log-likelihood that describes the orientation in which parameters should be changed to optimal fit the data [40]. It converts sample X into a fixed-length vector.

This gradient is normalized to be used with discriminative classifiers. Fisher information matrix is a normalization technique defined as,

$$F_\lambda = E_X[\nabla_\lambda \log p(X|\lambda) \nabla_\lambda \log p(X|\lambda)'] \tag{2.11}$$

The normalized gradient vector is defined as,

$$F_{\lambda}^{-1/2}\nabla_{\lambda}\log p(X|\lambda). \tag{2.12}$$

*Fisher vector:* Perronnin *et al.* apply Fisher kernels to visual vocabularies modeled by GMMs [40]. We follow the exposition in [32].

$\Theta = (\mu_k, \Sigma_k, \pi_k : k = 1, \ldots, K)$ is a parameter of GMM with K is the number of Gaussians in GMM. $I = (x_1, \ldots, x_N)$ is the set of D dimensional features extracted from an image. The GMM correlates each $x_i$ vector to the k'th Gaussian mixture with a strength provided by the posterior probability,

$$q_{ik} = \frac{\exp\left[-\frac{1}{2}(\mathbf{x}_i - \mu_k)^T\Sigma_k^{-1}(\mathbf{x}_i - \mu_k)\right]}{\sum_{t=1}^{K}\exp\left[-\frac{1}{2}(\mathbf{x}_i - \mu_t)^T\Sigma_k^{-1}(\mathbf{x}_i - \mu_t)\right]} \tag{2.13}$$

$$u_{jk} = \frac{1}{N\sqrt{\pi_k}}\sum_{i=1}^{N} q_{ik}\frac{x_{ji} - \mu_{jk}}{\sigma_{jk}}, \tag{2.14}$$

$$v_{jk} = \frac{1}{N\sqrt{2\pi_k}}\sum_{i=1}^{N} q_{ik}\left[\left(\frac{x_{ji} - \mu_{jk}}{\sigma_{jk}}\right)^2 - 1\right]. \tag{2.15}$$

where $j = 1, 2, \ldots, D$ of D dimensional feature. The image I is now represented with Fisher vector combining the vectors $u_k$ and $v_k$ for each of K Gaussian mixtures.

$$\Phi(I) = \begin{bmatrix} \vdots \\ \mathbf{u}_k \\ \vdots \\ \mathbf{v}_k \\ \vdots \end{bmatrix}. \tag{2.16}$$

## 2.5. Feature Selection

### 2.5.1. Random Sample versus Labels Canonical Correlation Analysis Filter

Feature selection is a technique to find the features which contribute the most to the prediction model. We use Random Sample versus Labels Canonical Correlation Analysis Filter (SLCCA-Rand) [42] as a feature selection method in our system. This feature selection method is first designed for paralinguistic speech analysis which deals with the basic meanings of the verbal messages. The state of the art methods in this field use brute-force feature extraction approach. Brute-forcing feature extraction approach leads to the problem of local minimum and the computational complexity. This work proposes an approach using the power of stochasticity to solve the problem of dimensionality by learning feature level and feature group level weights in many randomly selected contexts.

Canonical correlation analysis (CCA) explores the relationship between two different vectors. CCA finds the maximum correlation of linear combinations of these vectors. The idea of SLCCA-CCA is to perform CCA between features and annotations. After this step, the absolute values of the projection weights are sorted. Features that sum up to a selected percentage of the total weight for each modality are picked in this method.

## 2.6. Oversampling of Data

### 2.6.1. Adaptive Synthetic Sampling Approach for Imbalanced Dataset

Over-sampling is a technique used to adjust data distribution in imbalanced datasets. Imbalanced datasets involve a different number of samples for each class. Over-sampling techniques create samples for minority classes to compensate for the imbalance in the dataset.

Synthetic Minority Over-Sampling Technique (SMOTE) [22] is one of the over-sampling techniques. The minority class in a dataset is over-sampled by generating synthetic samples out of minority class samples. Synthetic samples along the line segments joining any/all of the k minority class nearest neighbors are introduced. Samples from the k nearest neighbors are randomly picked according to the amount of data needed. This process is explained in the following way. Difference between the sample and its nearest neighbor is taken. This difference is multiplied by a random number between zero and one. The result is added to the feature vector. This leads to choosing a random point along the line segment between two specific features [22].

Adaptive Synthetic Sampling Approach (ADASYN) [43] is an improved version of SMOTE. It oversamples using weighted distribution for different minority class examples according to their level of difficulty learning. More synthetic data are created for minority class samples that are hard to learn. This technique reduces bias in imbalanced datasets. It also improves the learning performance by shifting the classifier decision boundary to be more considered on those examples which are hard to learn [43].

## 2.7. Classification

### 2.7.1. Support Vector Machine

Support vector machine (SVM) is used for classification and regression tasks [44]. SVM builds a model for a set of training data. Each data belongs to a category. A

decision plane is created for this model by SVM. The decision plane separates the set of data with decision boundaries according to its category. We first would like to explain this basic idea of SVM with an example. In the left part of Figure 2.11, objects are belonging to either green or red class. These objects can be separated with a line. This line defines class membership of objects. If a new object falls to the left of the line, it is classified as red. If it falls to the right of the line, it is classified as green. This example is a basic explanation of the linearly classifiable model. However, there are more complex structured models like the one in the right part of Figure 2.11. The separation of this model is harder because the function is a curve.

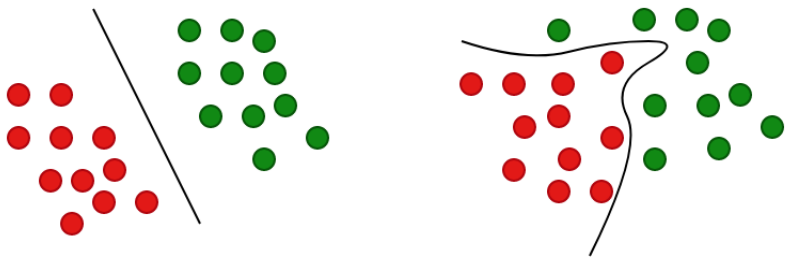

Figure 2.11. A classic example for a linear classifier and an example with complex structure data.

SVM solves this kind of non-linearly separable problems constructing a hyper-plane in a high dimensional space maximizing the margin between classes. We first focus on a classification problem with only two classes. We follow the exposition in [36]. The dataset is defined as,

$$X = \{x^t, r^t\}_t \quad where \quad r^t = \begin{cases} +1 & \text{if} \quad x_t \in C_1 \\ -1 & \text{if} \quad x_t \in C_2 \end{cases} \tag{2.17}$$

The formula defining optimal separation of hyperplane is,

$$\begin{aligned} w^T x^t + w_0 &\geq +1 \quad for \quad r^t = +1 \\ w^T x^t + w_0 &\leq +1 \quad for \quad r^t = -1 \end{aligned} \tag{2.18}$$

The aim is to find the coefficients of w and $w_o$. These two inequalities can be merged into one equation,

$$r^t(w^T x^t + w_0) \geq +1 \tag{2.19}$$

Margin is the maximum distance from discriminant to the closest data points for each side. Distance of x to the hyperplane is $\frac{|w_T x_t + w_0|}{||w||}$. This can be written as,

$$(r_t(w_T x_t + w_0)/||w||) \geq p, \forall t \tag{2.20}$$

with $r_t$ and the distance greater than some value p. For a unique solution, $p||w|| = 1$. It is needed to minimize $||w||$ to maximize margin according to this equation. The task becomes:

$$\min \frac{1}{2}||w||^2 \quad \text{subject to} \quad r^t(w^T x^t + w_0) \geq +1, \forall t \tag{2.21}$$

This is a constrained optimization problem and it is solved with Lagrange multipliers $\alpha^t$,

$$L_p = \frac{1}{2}||w||^2 - \sum_{t=1}^{N} \alpha^t [r^t (w^T x^t + w_0) - 1], \forall t \tag{2.22}$$

The solution to this equation is detailed in [44]. The final equation becomes:

$$L_d = -\frac{1}{2} \sum_t \sum_s \alpha^t \alpha^s r^t r^s (x^t)^T x^s + \sum_t \alpha^t \tag{2.23}$$

subjects to $\sum_t \alpha^t r^t = 0$, and $\alpha^t \geq 0$, $\forall t$.

*Soft margin:* When the data is not linearly separable, some mistakes are allowed. For this, a slack variable $\xi$ is defined in equation 2.19 and it becomes,

$$r^t (w^T x^t + w_0) \geq 1 - \xi^t \tag{2.24}$$

2.7.1.1. Kernel Trick. For the nonlinear classification problems, data are mapped to a new space. This process is called kernel trick. The model in the new space is the nonlinear model of the original space. The new dimension of data is in z space with k-dimensions. Original space is in x space with d-dimensions. The new dimension is calculated using a basis function $\phi$:

$$z = \phi(x) \quad \text{where} \quad z_j = \phi_j(x), j = 1, ..., k \tag{2.25}$$

Formula 2.19 becomes (ignoring $w_0$):

$$r^t * w^T \phi(x^t) \geq +1 \tag{2.26}$$

When this function is solved with Lagrange multipliers $\alpha_t$, as equation 2.21 solved, it becomes,

$$L_d = -\frac{1}{2} \sum_t \sum_s \alpha^t \alpha^s r^t r^s \phi(x^t)^T \phi(x^s) + \sum_t \alpha^t \tag{2.27}$$

The inner product part of $\phi(x^t)^T \phi(x^s)$ is replaced by a kernel function, $K(x_t, x_s)$. There are some kernel types,

*Linear Kernel:*

$$K(x_t, x_s) = x_t^T x_s \tag{2.28}$$

*Polynomials Kernel (Polynomials of degree q):*

$$K(x_t, x_s) = (x_t^T x_s + 1)^q \tag{2.29}$$

*Radial Basis Kernel:*

$$K(x_t, x_s) = \exp\left[-\left(\frac{||x_t - x_s||^2}{\sigma^2}\right)\right] \tag{2.30}$$

### 2.7.2. Extreme Learning Machine

Extreme Learning Machine (ELM) is a classification and a regression technique [45]. ELM is similar to SLFN. The weights of SLFN is unknown at first. It is found using backpropagation algorithms (For detail you can check some sources like [36]). The difference between ELM and SLFN is that input weights of ELM are assumed to be known.

We shortly want to explain ELM with first defining a standard SLFN. N training samples for a SLFN is defined as $(x_i\ ,\ t_i)$ where $x_i = [x_{i1}, x_{i2}, ...x_{in}]^T \in R^n$ and $t_i = [t_{i1}, t_{i2}, ...t_{im}]^T \in R^m$. The SLFN contains $\widetilde{N}$ hidden nodes. Its activation function is g(x). It is modeled as,

$$\sum_{1}^{\widetilde{N}} \beta_i g_i(x_j) = \sum_{1}^{\widetilde{N}} \beta_i g(w_i x_j + b_i) = o_j, j = 1, ..., N \tag{2.31}$$

$w_i = [w_{i1}, w_{i2}, ...w_{in}]^T$ is the weight vectors linking input nodes to hidden nodes. $\beta_i = [\beta_{i1}, \beta_{i2}, ...\beta_{im}]^T$ is the weight vectors linking hidden nodes to output nodes. $b_i$ is the bias of the $i$th hidden node. The aim of SLFN is to minimize the following cost,

$$\sum_{1}^{\widetilde{N}} ||o_j - t_j|| \tag{2.32}$$

This equation is solved by gradient-based learning technique such as back-propagation for SLFN. We now shortly explain this solution for ELM. The above equations are written as $H\beta = T$ where,

$$H(w_1,...,w_{\widetilde{N}},b_1,...,b_{\widetilde{N}},x_1,...,w_{\widetilde{N}}) = \begin{bmatrix} g(w_1x_1+b_1) & \dots & g(w_{\widetilde{N}}x_1+b_{\widetilde{N}}) \\ \vdots & \dots & \vdots \\ g(w_1x_N+b_1) & \dots & g(w_{\widetilde{N}}x_N+b_{\widetilde{N}}) \end{bmatrix} \tag{2.33}$$

$$B = \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_{\widetilde{N}}^T \end{bmatrix}_{\widetilde{N}\times m} \qquad T = \begin{bmatrix} t_1^T \\ \vdots \\ t_N^T \end{bmatrix}_{N\times m} \tag{2.34}$$

H is the hidden layer output matrix of the neural network. ELM algorithm assumes that H is known by fixing the weights $w_i$ and biases $b_i$. Then, the equation $H\beta = T$ becomes:

$$\beta = H^{\dagger}T \tag{2.35}$$

$H^{(\dagger)}$ is the Moore–Penrose generalized inverse of H. Lastly, kernels can be integrated into ELM. The details about the algorithm are in [45].

# 3. PROPOSED METHODOLOGY

Our feature extraction, feature summarization, feature selection, regression and fusion methods pipeline is illustrated in Figure 3.1. We build various models using different techniques. We apply score fusion and feature fusion to feature vectors of the best models.

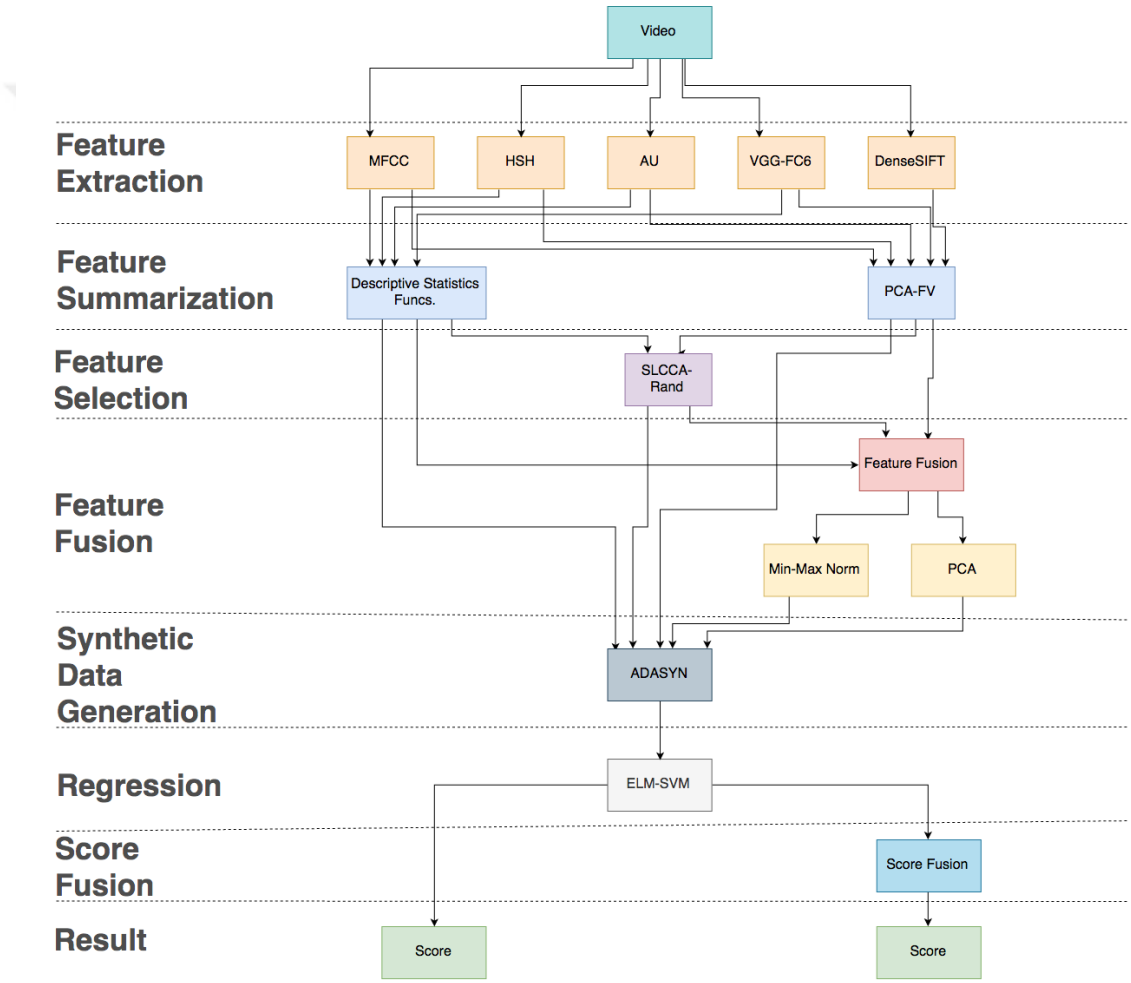


Figure 3.1. The pipeline of our system.

We test our models on a public dataset [46] derived from Liris-Accede [21]. The dataset consists of movies. The dataset is unbalanced both on arousal and valence axis. Initially, we built some models with this unbalanced dataset. These initial models were biased towards the majority class.

We apply data synthesis techniques to come up with an unbiased model. We quantize continuous labels into seven labels. Minority classes are oversampled via ADASYN [43]. Oversampling technique is performed before classification. We approach the problem as a classification problem first before applying regression.

We use various visual and audio features. We extract Hue Saturation Histogram (HSH), Dense-SIFT, VGG-FC6 feature, AU, and MFCC from movie scenes.

For HSH, we extract every 24th frame of each movie. Each image is scaled to 240x320 pixels. RGB images are converted to grayscale. Saturation histogram is in the range of 0 to 1 with steps of 1/32. Hue histogram is in the range of 0 to 1 with steps of 1/30.

Dense SIFT is extracted by using VLFeat library [32]. We extract Dense-SIFT for every 24th frame of each movie. Each image is scaled to 120x160 pixels. Then, RGB images are converted to grayscale. We use 'vl_phow' function of the library. Parameter 'step' is set to 7 and 'size' is set to [4,6,8] for this function. Parameter 'step' (in pixels) defines the step of the grid at which the Dense-SIFT features are extracted. Parameter 'size' defines scales at which the Dense-SIFT features are extracted. Each value is used as bin size for the 'vl_dsift' function. Other parameters are set to default values.

For MFCC, we extract the audio signals from movies. There are two signals for each movie; left and right channels. We merge these into one signal via the average function. Some of the functions we use can be found at [47]. MFCC is extracted via 'melfcc' function. We set 'hoptime' parameter to 0.016 and 'wintime' to 32. The rest of the parameters are left at defaults.

VGG-FC6 feature is provided by Mediaeval 2017 organization [46]. It is downloaded from [48]. The feature was extracted from VGG deep neural network trained for image recognition problem.

AU is extracted using Open Face library [49]. AU is extracted for every 30th frame of each movie. AU provides two kinds of information. One is which action units are present. The other is how intense the action unit is. We use the intensity. Resulting AU features is 17-dimensional.

Two different feature summarization methods are used and analyzed. Firstly, we apply some descriptive statistics functions such as minimum, maximum and median to features. We use maximum, minimum, median functions to represent AU, MFCC, and HSH. We use the median function to represent VGG-FC6. No descriptive function is used for Dense-SIFT.

The second summarization technique we use is Principal Component Analysis (PCA) with the Fisher Vector (FV). For PCA, we set the explained variance to 0.95. Our experiments are memory limited. The maximum size of features that we supply to the PCA step is 0.5GB. FV uses GMM for encoding. While applying GMM to features, different parameters for the features are set. 16 GMM is extracted for AU, Dense-SIFT, and MFCC. One GMM is extracted for HSH and VGG-FC6. We separately apply the GMMs to Fisher vector and normalize the Fisher vectors with signed square root and L2. The PCA coefficients and GMM's parameters for the Fisher vector are separately obtained during training for each feature. VLFeat library [32] is used for GMM and FV.

We observe that the models predict fluctuating values on consecutive labels but the actual consecutive labels are changing smoothly. For this reason, we apply a smoothing technique [50]. We use 'movingmean' function with 'window' set to 20.

Random Sample versus Labels Canonical Correlation Analysis Filter (SLCCA-Rand) [42] is used as a feature selection method. This method selects features that sum up to 90% of the total weight for each modality. We use 10 iterations for SLCCA projections.

Extreme Learning Machine (ELM) and Support Vector Regression (SVM) are applied as regression techniques for both arousal and valence models. Grid search is applied for both techniques to find the optimal parameters of models. Regularization coefficient is searched in [0.01,100] interval with exponential steps. Radial basis function (RBF) and linear kernels are tested. The RBF kernel scale parameter is optimized in [0.01,100] interval, also with exponential steps. For ELM and SVM we use an open-source libraries [51, 52]. Our performance measures are the Pearson Correlation Coefficient (PCC) and Mean Square Error (MSE). PCC results are prioritized.

We apply score fusion and feature fusion to feature vectors of the best models. We perform score fusion to the outputs of the best models. We also implement some feature fusion techniques. We concatenate the feature vectors of the best models. Then, we apply the regression techniques to the feature vectors. We apply PCA to the feature vectors before applying regression techniques.

The experiments performed in this thesis are implemented in Matlab. We perform the experiments in a MacBook Pro (late 2013) with 2.4 GHz Intel Core i5 processor, and 4GB 1500 MHz DDR3 memory.

# 4. EXPERIMENTS AND RESULTS

## 4.1. Dataset

### 4.1.1. Liris-Accede

The Liris-Accede dataset contains 160 professional and amateur movies. There are examples of several movie genres like horror, comedy, drama, and action. The movies are mostly English but there are a small number of movies in Italian, Spanish, and French [21].

The development set used in MediaEval 2017 challenge consists of 30 movies. The videos are between approximately two minutes and one hour 15 minutes. The total length of the videos is approximately seven hours. The test set has 14 movies which are different than the original 160 movies of Liris-Accede. They are between approximately three minutes and two hours. The total length of the 14 movies is approximately eight hours. The test set was sequestered during the Mediaeval 2017 challenge.

The Mediaveal 2017 organization shared not only the movies but also the features extracted from them [46]. We use only VGG-FC6 feature out of the provided features.

Movies are divided into 10 second-segments sliding over the entire movie with a shift of 5 seconds. Each segment is annotated according to its position on the valence-arousal plane. Range of valence is from -1 to +1, -1 being the most negative and +1 being the most positive. Arousal ranges from -1 to +1, -1 being the calmest and +1 being the most active. Evaluation metrics are Mean Square Error (MSE) and Pearson Correlation Coefficient (PCC). PCC measures the correlation directions between actual labels and predictions. MSE measures the mean square error between actual labels and predictions.

### 4.1.2. Development Dataset Examination

We need to properly split development set into train and validation sets. Videos are annotated with continuous values. We quantize the labels to better visualize the distribution of arousal and valence data.

We need to be careful about the distribution of labels into training and validation sets. These sets must contain roughly equal amounts of scenes for each class. Models created from imbalanced datasets become biased; they favor the majority class in predictions. Minority classes are treated as noise in imbalanced datasets [53]. For these reasons, we benefit from the quantization technique to visualize data distribution of continuous values. We categorize arousal values into low, neutral and high; valence values into negative, neutral and positive labels.

We create a histogram for quantizing arousal and valence labels. The histogram consists of five categories for [-1,1] interval. For arousal, we name them as "low", "low close to neutral", "neutral", "high close to neutral", and "high". For valence, we name them as "negative", "negative close to neutral", "neutral", "positive close to neutral", and "positive".

After quantizing the scenes with this method, we get 412 low, 3096 neutral and 1766 positive values for arousal. "low" refers to the total number of "low" and "low close to neutral" labels. "high" refers to the total number of both "high" and "high close to neutral" labels. "neutral" refers to the number of "neutral" labels. It shows that the neutral category is the majority in arousal.

There are 1245 negative, 2680 neutral and 1349 positive values for valence. "negative" refers to the total number of "negative" and "negative close to neutral" values. "positive" refers to the total number of "positive" and "positive close to neutral" labels. "neutral" refers to the total number of "neutral" labels. It shows that the neutral category is the majority in valence.

Video scenes are cut out of 30 movies. As we mentioned earlier, each video is a 10-second segment of a movie and consecutive videos share five seconds of footage. So it only makes sense to put overlapping videos into the same dataset; either training or validation. Also, each movie consists of many similar scenes, music effects, actors and such itself. For these reasons, all the scenes from one movie must be either in the training set or in the validation set, never in both.

All the segments in each movie are already labeled. However, the movies themselves are not labeled as a whole with any genre. We need to distribute movies into training and validation sets properly taking their genre into account. For example, if we create a training set containing only thriller movies, we cannot use it to predict comedy movies. For this reason, we need to quantize movies into categories as we do to scenes. We quantize the movies using the labels of the scenes they contain.

Each movie scene is quantized to the low, neutral and high label. The total number of scenes with each label is calculated for a movie. If the total number of high labels is 20% more than the total number of low labels, this movie is categorized as high arousal movie. Low arousal movies are labeled similarly. If a movie is neither high arousal nor low arousal, it is categorized as neutral. The same method is applied to valence values. Resulting labels are in Figure 4.1.

We attempt to keep 60% of movies for each category in the training set. While doing that we also want to have 70% of scenes for each category in the training set. This might not be possible for all categories. 10% deviation is allowed. An open-source algorithm is used for this process [54].

The train and validation set for arousal and valence models can be found in the appendix (Figure A.1, A.2, A.3 and A.4). The total number of video distribution is in Figure 4.2 and Figure 4.3.

| Movie Names | Valence Mood | Arousal Mood | Number of annotations |
|---|---|---|---|
| **After The Rain** | Negative | Neutral | 117 |
| **Attidude Matters** | Positive | High | 273 |
| **Barely Legal Stories** | Neutral | Neutral | 197 |
| **Between Viewings** | Positive | High | 176 |
| **Big Buck Bunny** | Positive | High | 118 |
| **Chatter** | Negative | High | 101 |
| **Cloudland** | Neutral | Neutral | 139 |
| **Damaged Kung Fu** | Positive | High | 202 |
| **Decay** | Negative | High | 912 |
| **Elephant's Dream** | Neutral | High | 130 |
| **First Bite** | Neutral | High | 127 |
| **Full Service** | Positive | High | 223 |
| **Islands** | Positive | Neutral | 33 |
| **Lesson Learned** | Negative | Neutral | 155 |
| **Norm** | Positive | High | 77 |
| **Nuclear Family** | Neutral | Neutral | 339 |
| **On Time** | Positive | Low | 61 |
| **Origami** | Positive | Neutral | 99 |
| **Parafundit** | Negative | High | 157 |
| **Payload** | Neutral | High | 214 |
| **Riding The Rails** | Positive | Neutral | 179 |
| **Sintel** | Neutral | High | 176 |
| **Spaceman** | Positive | High | 185 |
| **Superhero** | Negative | Neutral | 227 |
| **Tears of Steel** | Neutral | High | 146 |
| **The Room of Franz Kafka** | Negative | Low | 49 |
| **The Secret Number** | Negative | High | 185 |
| **To Claire From Sonny** | Negative | Low | 82 |
| **Wanted** | Positive | High | 22 |
| **You Again** | Positive | Neutral | 173 |

Figure 4.1. Development set information.

| | Number of Total Low Arousal Scene | Number of Total Neutral Arousal Scene | Number of Total High Arousal Scene |
|---|---|---|---|
| **Train set** | 201 | 1634 | 943 |
| **Validation set** | 211 | 1462 | 823 |

Figure 4.2. Total number of arousal annotations for train and validation sets.

| | Number of Total Negative Valence Scene | Number of Total Neutral Valence Scene | Number of Total Positive Valence Scene |
|---|---|---|---|
| **Train set** | 504 | 1457 | 829 |
| **Validation set** | 741 | 1223 | 520 |

Figure 4.3. Total number of valence annotations for train and validation sets.

## 4.2. Experiments

### 4.2.1. Descriptive Statistics Functions Application on Unbalanced Dataset

We extract MFCC, HSH, VGG-FC6 and AU features. Since we study time series data, we need to summarize the time-series features. When we apply a feature extraction method on a movie scene, we get multiple results per scene, generally one per second. For instance, if we want to represent a video using MFCC, we get approximately 300 MFCC features. We use descriptive functional statistics to summarize these. After summarization, we normalize each feature to [-1,1] range, and we create regression models with ELM. The results are in Table 4.1.

Table 4.1. Arousal/valence prediction results of descriptive statistics functions and ELM.

| **Feature** | **Dataset** | **Valence-PCC** | **Valence-MSE** | **Arousal-PCC** | **Arousal-MSE** |
|---|---|---|---|---|---|
| **HSH** | dev | 0.0424 | 0.1474 | 0.0841 | 0.1340 |
| **HSH** | test | 0.1145 | 0.0856 | 0.0968 | 0.1399 |
| **MFCC** | dev | 0.4081 | 0.1323 | 0.0314 | 0.1212 |
| **MFCC** | test | -0.1894 | 0.1074 | -0.0263 | 0.1728 |
| **AU** | dev | 0.2656 | 0.1334 | 0.0358 | 0.1118 |
| **AU** | test | 0.1207 | 0.0853 | 0.0394 | 0.1060 |
| **FC6** | dev | 0.1919 | 0.1394 | 0.1146 | 0.0984 |
| **FC6** | test | -0.1721 | 0.0911 | -0.2064 | 0.1290 |

The model with AU exhibit the best PCC and MSE results on the valence test set. MFCC and VGG-FC6 show negative PCC results. The model with HSH exhibit the best PCC result on the arousal test set. VGG-FC6 and MFCC show negative PCC

correlation.

Some plots of actual and predictions of the test set is shown in Figure 4.4 and 4.5. We observe that there are sudden changes in consecutive predictions whereas actual emotion labels of consecutive scenes change smoothly. We explain smoothing techniques that help mitigate this issue in the next section. MFCC model achieves to predict labels only in [0, 0.4] range. It cannot predict low arousal labels. Since the training set is skewed towards neutral and high arousals, this is expected. The results of the valence model with VGG-FC6 are similar. This model achieves to predict only neutral values.

Imbalanced dataset decreases the performance due to bias towards majority classes. Minority class samples may be treated as noise [53]. We apply an oversampling technique to make the dataset balanced in the following section.

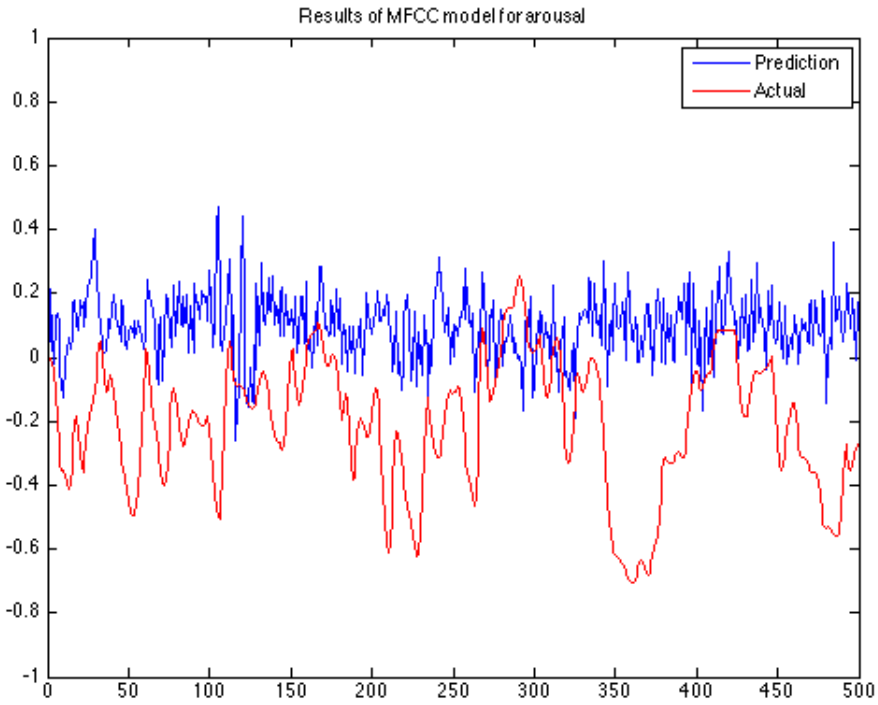


Figure 4.4. Results of MFCC model for arousal space.

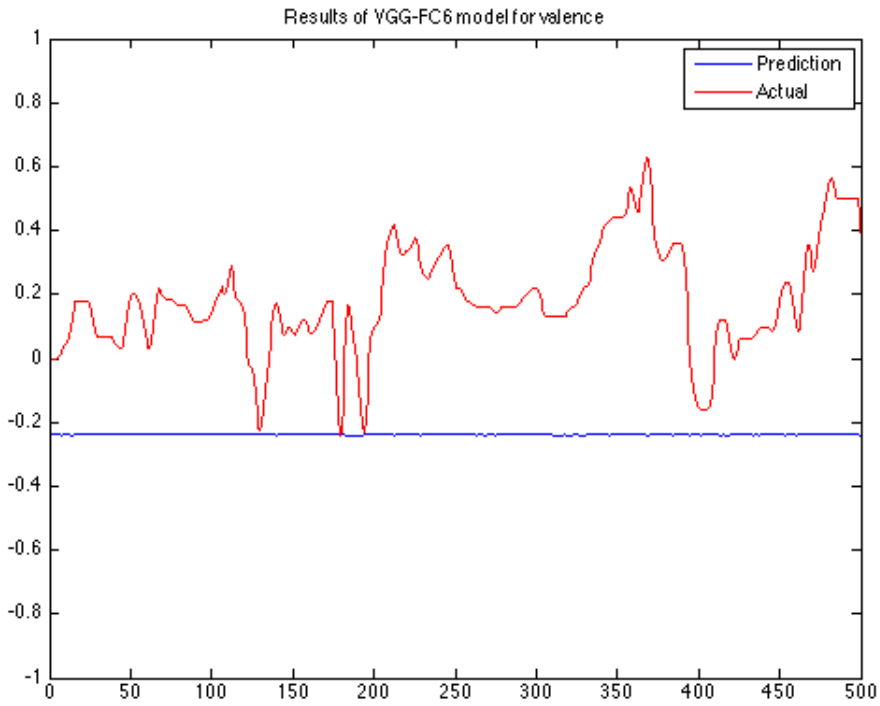


Figure 4.5. Results of VGG-FC6 model for valence space.

### 4.2.2. Descriptive Statistics Functions Application and Classification on Balanced Dataset

We synthesis more samples of minority classes via ADASYN [43]. ADASYN requires quantization of the continuous labels first. We need to determine how many categories we will divide [-1, 1] interval into. We divide the label range into seven equal parts to obtain a reasonably balanced yet varied data. As the number of categories increases, it requires more storage and processing power to run the models. However, the results do not improve enough to justify the additional complexity.

The problem becomes a classification problem after applying ADASYN. We obtain the results in Figure 4.6. High recall, low precision valence results in VGG-FC6 feature shows that many labels are misclassified as negative valence. Valence specificity results with MFCC and AU shows that neutral scenes are predicted more reliably compared to the others. Valence specificity results with VGG-FC6 shows that negative

scenes are predicted more reliably compared to the others. F1 scores of all features show that neutral results are predicted more reliably to the others.

In arousal experiments, precision values of all features except VGG-FC6 indicate that our system predicts "low neutral" and neutral values more successfully but many samples are misclassified as "low arousal". Looking at specificity, neutral samples are predicted more accurately. The model with AU predicts low arousal samples. F1 scores indicate that neutral samples are predicted better than the others. Based on the experiments, we anticipate that neutral samples are easier to predict.

### 4.2.3. Descriptive Statistics Functions Application and Regression on Balanced Dataset

We repeat the experiment in the previous section with the actual continuous annotations. We use AUs to validate the usefulness of smoothing techniques. Figure 4.7 shows that smoothing leads to better predictions by reducing fluctuations. Therefore, we apply smoothing to all experiments.

We apply ELM to the features. The performance results are shown in Table 4.2. The plot of actual labels and predictions are in Figure B.1. In valence models, the PCC results show that the model with AU is the best but the model is biased towards neutral labels. The model with HSH and MFCC have a similar problem. The model with VGG-FC6 has good predictions. The slopes in predictions are in line with slopes of actual label graph. However, the predictions do not match with the actual labels. Especially, positive valences are not predicted well.

In arousal models, the PCC results show that the model with HSH is the best. The slopes in predictions are in line with the slopes of actual label graph. However, the predictions do not match with the actual labels. The model is biased towards neutral and "high close to neutral" labels. The model with MFCC is biased towards "high close to neutral" label. The slopes in predictions of the model with AU are in line with slopes of actual label graph. The model is biased towards neutral labels. The model

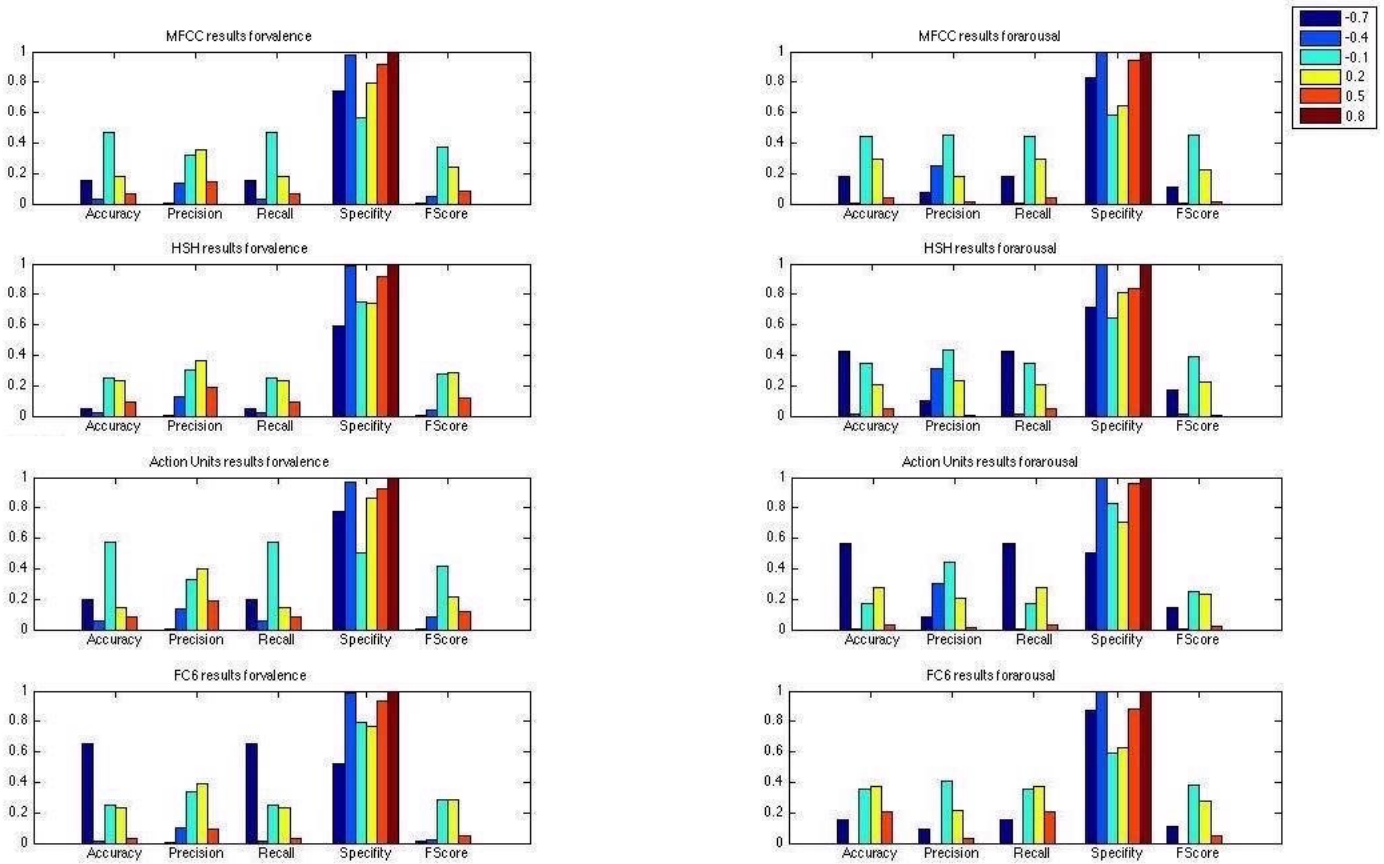


Figure 4.6. Arousal and valence classification results.

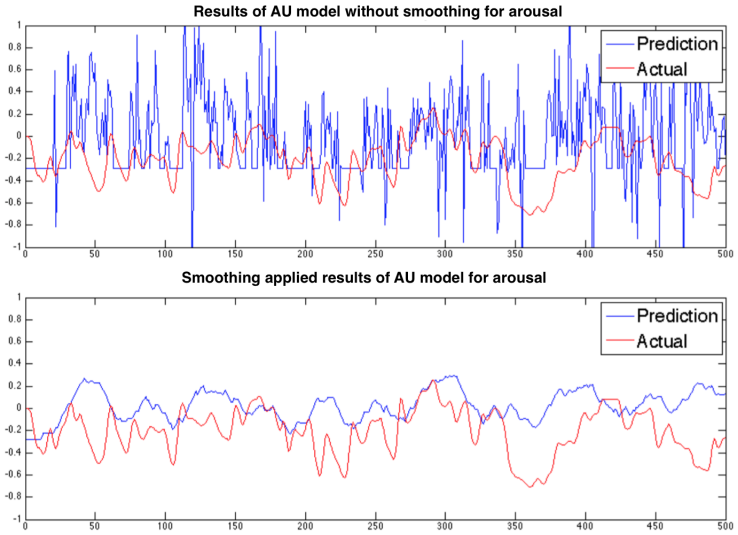


Figure 4.7. Arousal results with action unit models with non-smoothing and smoothing.

with VGG-FC6 has a similar problem.

Table 4.2. The performance results with descriptive statistics functions and ELM.

| Feature | Valence-PCC | Valence-MSE | Arousal-PCC | Arousal-MSE |
|---|---|---|---|---|
| **MFCC** | -0.0318 | 0.0858 | 0.0183 | 0.1406 |
| **AU** | 0.2195 | 0.0930 | -0.0060 | 0.2251 |
| **HSH** | 0.1284 | 0.0822 | 0.1103 | 0.1432 |
| **VGG-FC6** | 0.1454 | 0.0982 | -0.0195 | 0.0984 |

We apply SVM to the features. The performance results are shown in Table 4.3. The plot of actual labels and predictions are in Figure B.2. In valence models, the PCC results show that the model with VGG-FC6 has the best result. The slopes in predictions have some correlations with the slopes in actual labels. The model can not predict positive valence labels. The models with other features are biased towards to neutral values. Almost all the predictions are close to each other.

In arousal models, the PCC results show that the model with MFCC has the best result. The slopes in predictions have some correlations with the slopes in actual labels. However, the model can not predict low arousal labels. The second best PCC

Table 4.3. The performance results with descriptive statistics functions and SVM.

| Feature | Valence-PCC | Valence-MSE | Arousal-PCC | Arousal-MSE |
|---|---|---|---|---|
| **MFCC** | 0.0598 | 0.0925 | 0.1205 | 0.1814 |
| **AU** | -0.1156 | 0.0836 | -0.0399 | 0.1175 |
| **HSH** | 0.0743 | 0.0863 | 0.0123 | 0.1833 |
| **VGG-FC6** | 0.1685 | 0.0957 | 0.0592 | 0.1739 |

result is the model with VGG-FC6. The predictions of the model with other features are biased towards neutral labels. Almost all the predictions are close to each other.

### 4.2.4. Descriptive Statistics Definitions and Feature Selection Application

We repeat the previous experiment adding SLCCA-Rand feature selection technique to the system. We apply ELM to the features. The performance results are shown in Table 4.4. The plot of actual labels and predictions are in Figure B.3. The results show that we obtain similar outcomes with the ELM experiments of the previous section. In valence models, the best PCC result is obtained with VGG-FC6 model. However, the feature selection technique decreases PCC. In arousal models, the best PCC result is obtained via HSH model. Feature selection increases PCC.

Table 4.4. The performance results with descriptive statistics functions, SLCCA-Rand, and ELM.

| Feature | Valence-PCC | Valence-MSE | Arousal-PCC | Arousal-MSE |
|---|---|---|---|---|
| **MFCC** | -0.0371 | 0.0893 | -0.0150 | 0.1416 |
| **AU** | 0.2202 | 0.0934 | -0.0519 | 0.1285 |
| **HSH** | 0.1252 | 0.0823 | 0.1264 | 0.1408 |
| **VGG-FC6** | 0.1428 | 0.0979 | -0.0207 | 0.0984 |

We apply SVM to the features. The performance results are shown in Table 4.5. The plot of actual labels and predictions are in Figure B.4. The results show that we obtain similar outcomes with the SVM experiments of the previous section.

Table 4.5. The performance results with descriptive statistics functions, SLCCA-Rand, and SVM.

| Feature | Valence-PCC | Valence-MSE | Arousal-PCC | Arousal-MSE |
|---|---|---|---|---|
| **MFCC** | 0.0418 | 0.0928 | 0.1166 | 0.1802 |
| **AU** | -0.0143 | 0.0835 | -0.0399 | 0.1175 |
| **HSH** | 0.0601 | 0.0864 | -0.0059 | 0.1837 |
| **VGG-FC6** | 0.1697 | 0.0952 | 0.0646 | 0.1743 |

In valence models, the best PCC result is obtained via VGG-FC6 model. In arousal models, the best PCC result is obtained via the MFCC model. Feature selection does not significantly change PCC.

### 4.2.5. PCA and Fisher Vector Applied Regression Models

We change the feature summarization technique with FV. We first reduce the feature dimension via PCA. Then, we encode the feature vectors with FV. The feature vectors are normalized with signed square root and L2.

We apply ELM to the features. The performance results are shown in Table 4.6. The plot of actual labels and predictions are in Figure B.5. In valence models, the PCC results show that the model with AU has the best result. The model is biased towards to neutral values. Almost all the predictions are close to each other. The models with MFCC, Dense-SIFT, and HSH have the same problem. Summarizing VGG-FC6 feature with FV instead of descriptive function increases the PCC. The slopes in predictions have some correlations with the slopes in actual labels. However, the model can not predict positive and negative valence labels.

In arousal models, there are similar outcomes with those of experiments in previous sections. The model with Dense-SIFT is biased towards to neutral labels. Almost all the predictions are close to each other. Applying FV to the features instead of descriptive functions does not improve PCC.

Table 4.6. The performance results with PCA, FV, and ELM.

| Feature | Valence-PCC | Valence-MSE | Arousal-PCC | Arousal-MSE |
|---|---|---|---|---|
| **MFCC** | -0.0444 | 0.0894 | 0.0237 | 0.1396 |
| **AU** | 0.2239 | 0.0933 | -0.0581 | 0.1322 |
| **Dense-SIFT** | -0.2613 | 0.0943 | -0.1080 | 0.0940 |
| **HSH** | 0.0849 | 0.0832 | 0.1193 | 0.1428 |
| **VGG-FC6** | 0.1896 | 0.0825 | -0.0153 | 0.0980 |

Table 4.7. The performance results with PCA, FV, and SVM.

| Feature | Valence-PCC | Valence-MSE | Arousal-PCC | Arousal-MSE |
|---|---|---|---|---|
| **MFCC** | 0.0625 | 0.0925 | 0.1195 | 0.1827 |
| **AU** | -0.1771 | 0.0855 | -0.0397 | 0.1175 |
| **Dense-SIFT** | -0.2567 | 0.0988 | -0.0696 | 0.1626 |
| **HSH** | 0.0417 | 0.0864 | 0.0851 | 0.1438 |
| **VGG-FC6** | 0.1313 | 0.0988 | 0.0690 | 0.1729 |

We apply SVM to the features. The performance results are shown in Table 4.7. The plot of actual labels and predictions are in Figure B.6. In valence models, the PCC results show that the model with VGG-FC6 has the best result. The slopes in predictions have some correlations with the slopes in actual labels. However, the model can not predict positive labels. Applying FV to VGG-FC6 feature instead of descriptive statistics decreases the PCC result. The second-best PCC result is obtained via the MFCC model. The predictions of the models with other features are biased towards neutral labels. Almost all the predictions are close to each other.

In arousal models, the PCC results show that the model with MFCC has the best result. The slopes in predictions are in line with the slopes of actual label graph. However, the model is biased towards low labels. Applying FV to MFCC instead of descriptive functions does not significantly change the PCC. The second best PCC result is obtained with the HSH model. The model can not predict low labels. Applying FV to HSH instead of descriptive functions improves the PCC, decreases the MSE.

Applying FV to the other features instead of descriptive functions do not significantly change the performance results. The slopes in the predictions of VGG-FC6, HSH, and Dense-SIFT have some correlations with the slopes in actual labels.

### 4.2.6. Fisher Encoding and Feature Selection Applied Regression Models

We repeat the previous experiment applying SLCCA-Rand feature selection technique. We apply ELM to the features. The performance results are shown in Table 4.8. The plot of actual labels and predictions are in Figure B.7. In valence models, the PCC results show that the model with VGG-FC6 has the best results. Applying feature selection to the feature considerably decreases PCC. In arousal models, the best PCC result is obtained with the HSH model. Applying feature selection to HSH and other features do not significantly change PCC. However, applying feature selection to Dense-SIFT leads to more correlation between predictions and actual labels.

Table 4.8. The performance results with PCA, FV, SLCCA-Rand, and ELM.

| **Feature** | **Valence-PCC** | **Valence-MSE** | **Arousal-PCC** | **Arousal-MSE** |
|---|---|---|---|---|
| **MFCC** | -0.0379 | 0.0893 | 0.0016 | 0.1352 |
| **AU** | 0.2152 | 0.0935 | -0.0452 | 0.1279 |
| **Dense-SIFT** | -0.2569 | 0.0941 | -0.0495 | 0.1508 |
| **HSH** | 0.1130 | 0.0828 | 0.1211 | 0.1418 |
| **VGG-FC** | 0.1386 | 0.0977 | -0.0225 | 0.0984 |

We apply SVM to the features. The performance results are shown in Table 4.9. The plot of actual labels and predictions are in Figure B.8. In valence models, VGG-FC6 and MFCC models show the best PCC results. However, the MFCC model is biased towards neutral labels. In arousal models, the best PCC result is obtained with the MFCC model. Applying feature selection to HSH model considerably decreases PCC.

Table 4.9. The performance results with PCA, FV, SLCCA-Rand and SVM.

| Feature | Valence-PCC | Valence-MSE | Arousal-PCC | Arousal-MSE |
|---|---|---|---|---|
| **MFCC** | 0.0999 | 0.0925 | 0.1201 | 0.1784 |
| **AU** | -0.1854 | 0.0853 | -0.0395 | 0.1175 |
| **Dense-SIFT** | -0.2552 | 0.0987 | -0.0939 | 0.1516 |
| **HSH** | 0.0420 | 0.0863 | -0.0218 | 0.1831 |
| **FC6** | 0.0986 | 0.1025 | 0.0586 | 0.1744 |

### 4.2.7. Plot of Performance Results Experimented with Single Feature

Arousal and valence results of all experiments with single features are illustrated in Figure 4.8 and Figure 4.9.

In valence ELM models, the best PCC result is obtained with VGG-FC6 model (PCA and FV applied). The other best PCC results are obtained with AU (PCA and FV applied) and HSH (descriptive functions applied). In arousal ELM models, the best PCC is obtained with HSH model (descriptive functions applied). The other good models are MFCC (PCA and FV applied), AU (descriptive functions applied), and Dense-SIFT (PCA, FV, and CCA applied).

In valence SVM models, the best PCC result is obtained with VGG-FC6 model (descriptive functions and SLCCA-Rand applied). The second best PCC results are obtained with MFCC (FV and SLCCA-Rand applied). In arousal SVM models, the best PCC result is obtained with the MFCC model (descriptive functions applied). The other good PCC results are VGG-FC6 (FV applied) and HSH (FV applied).

### 4.2.8. Score Fusion

We apply score fusion to the best models with linear regression technique. We first apply score fusion to the ELM models. The performance results are shown in Table 4.10 and 4.11. The plot of actual labels and predictions are in Figure B.9. We

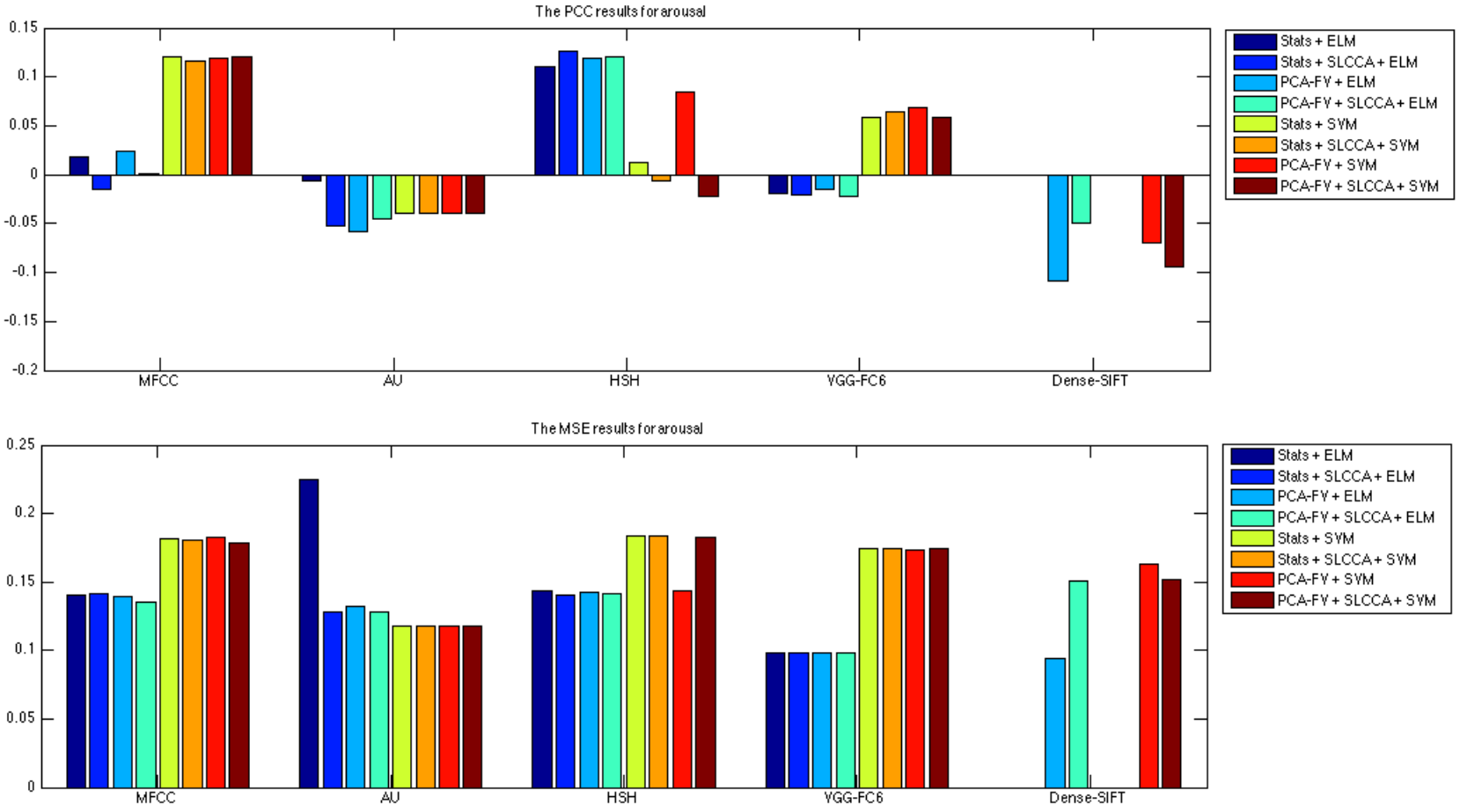


Figure 4.8. Arousal results.

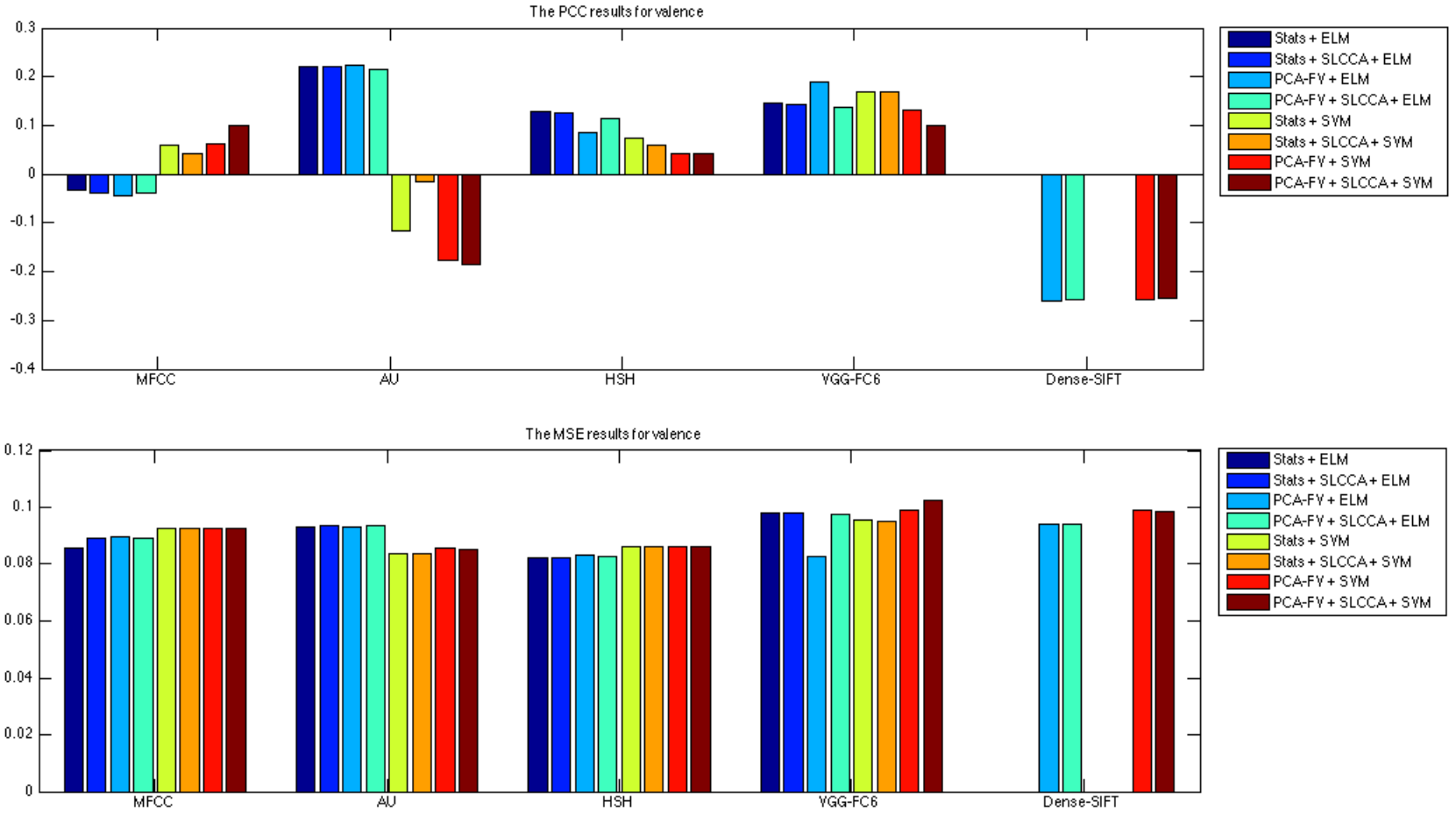


Figure 4.9. Valence results.

fuse the scores of VGG-FC6 model with the scores of AU and HSH models separately. The score fusions do not provide significant improvement to PCC but we obtain the highest PCC result with the fusion of VGG-FC6 and HSH among all the experiments of valence. The predictions are biased towards neutral values. In arousal models, we fuse the scores of HSH model with the scores of MFCC, AU, Dense-SIFT separately. Score fusions do not significantly affect PCC.

We apply score fusion to the SVM models. The performance results are shown in Table 4.10 and 4.11. The plot of actual labels and predictions are in Figure B.10. We fuse the scores of VGG-FC6 model with the scores of MFCC. The score fusion results increase PCC. Some slopes in predictions are in line with the slopes of actual label graph. However, the model can not predict positive labels. In arousal models, we fuse the scores of MFCC model with the scores of VGG-FC6 and HSH. The score fusion decreases PCC.

Table 4.10. Test performance results with score fusion for valence.

| **Classifier** | **Score Fused Features** | **Valence-PCC** | **Valence-MSE** |
|---|---|---|---|
| **ELM** | VGG-FC6 + AU | 0.2044 | 0.0818 |
| | VGG-FC6 + HSH | 0.2091 | 0.0808 |
| **SVM** | VGG-FC6 + MFCC | 0.1821 | 0.0946 |

Table 4.11. Test performance results with score fusion for arousal.

| **Classifier** | **Score Fused Features** | **Arousal-PCC** | **Arousal-MSE** |
|---|---|---|---|
| **ELM** | HSH + MFCC | 0.1065 | 0.1371 |
| | HSH + Dense-SIFT | -0.0397 | 0.1441 |
| | HSH + AU | 0.1236 | 0.1446 |
| **SVM** | MFCC + VGG-FC6 | 0.0491 | 0.1880 |
| | MFCC + HSH | 0.0758 | 0.1416 |

### 4.2.9. Feature Fusion

We use two feature fusion techniques to the best models; applying min-max normalization or PCA. We apply ELM to the features. The performance results are shown in Table 4.12 and 4.13. The plot of actual labels and predictions are in Figure B.11 and B.12. In valence models, neither of the feature fusion techniques show improvements to PCC. In arousal models, the PCC results do not significantly change. However, we obtain the highest PCC by the fusions of HSH and AU features with PCA among all the experiments of arousal.

We apply SVM to the features. The performance results are shown shown in Table 4.12 and 4.13. The plot of actual labels and predictions are in Figure B.13 and B.14. In arousal and valence models, both feature fusion technique decrease PCC.

Table 4.12. Test performance results with feature fusion for valence.

| **Classifier** | **Fusion Type** | **Features fused** | **Valence-PCC** | **Valence-MSE** |
|---|---|---|---|---|
| **ELM** | Feat. Fus. | VGG-FC6 + AU | -0.0333 | 0.0892 |
| | | VGG-FC6 + HSH | 0.1340 | 0.0824 |
| **ELM** | Feat. Fus. + PCA | VGG-FC6 + AU | -0.0355 | 0.0879 |
| | | VGG-FC6 + HSH | 0.1758 | 0.0807 |
| **SVM** | Feat. Fus. | VGG-FC6 + MFCC | 0.0075 | 0.1006 |
| **SVM** | Feat. Fus. + PCA | VGG-FC6 + MFCC | 0.0030 | 0.0975 |

## 4.3. Discussions

In this section, we discuss the results of each step in the pipeline we build.

We construct training and validation sets out of our unbalanced dataset. We take mood distribution into account while doing so. In the training and validation sets, the number of neutral arousal videos is higher than the number of low and high arousal videos; the number of neutral valence videos is higher than the number of negative and positive valence videos. Models created with the training and validation set may

Table 4.13. Test performance results with feature fusion for arousal.

| Classifier | Fusion Type | Features fused | Arousal-PCC | Arousal-MSE |
|---|---|---|---|---|
| **ELM** | Feat. Fus. | HSH + MFCC | -0,0966 | 0,1396 |
| | | HSH + Dense-SIFT | -0.2272 | 0.1138 |
| | | HSH + AU | 0.0511 | 0.1885 |
| **ELM** | Feat. Fus. + PCA | HSH + MFCC | -0.0964 | 0,1445 |
| | | HSH + Dense-SIFT | -0.1945 | 0.1477 |
| | | HSH + AU | 0.1322 | 0.1676 |
| **SVM** | Feat. Fus. | MFCC + VGG-FC6 | 0.1156 | 0.1379 |
| | | MFCC + HSH | 0.0288 | 0.1507 |
| **SVM** | Feat. Fus. + PCA | MFCC + VGG-FC6 | 0.1025 | 0.1376 |
| | | MFCC + HSH | 0.0256 | 0.1568 |

lead to an outcome biased towards the neutral majority class. We make the dataset balanced using ADASYN to solve this problem. Balanced dataset helps mitigate this problem, but some models are still biased towards the neutral class.

Consecutive scenes generally change emotions smoothly whereas our predictions exhibit sudden fluctuations. For this reason, we apply the smoothing filter on the prediction labels. It helps to reduce fluctuations.

We would like to emphasize some results of the models with ELM classifier. For valence, we obtain the best PCC result with AU. However, when we compare the predictions and actual labels, most of the predictions are around zero. Hence, we analyze the second best PCC result for valence. We achieve the best performance result on the outcomes with VGG-FC6. We see a significant improvement when Fisher vector encoding is applied to VGG-FC6 instead of descriptive statistics function. However, there is a significant decrease in performance when we use the feature selection algorithm with Fisher encoding. For arousal, we obtain the best result with HSH. Fisher vector and feature selection do not cause any improvement.

There are important takeaways from the experiments with the models with ELM. Firstly, we expect that applying Fisher vector encoding instead of descriptive statistics functions for feature summarization improves the results. However, this expectation does not hold for all features. Similarly, we also expect to get improved results when we apply feature selection but this expectation does not hold either. We observe that VGG-FC6 features are the best for valence.

SVM models and ELM models exhibit similar behavior. Applying FV and feature selection algorithms to feature vectors do not always improve the results on SVM models. The best feature for valence is VGG-FC6 with SVM as it is with ELM. However, the best feature for arousal with SVM model is obtained with MFCC.

The best performance results for both arousal and valence are obtained with ELM. We see that ELM is a strong alternative to commonly used SVM for movie emotion estimation. Training speed is not among our criteria. However, we think it is important to note that ELM is extremely faster than SVM in training [45].

The best arousal model is obtained with the fusion of HSH and AU features via PCA (PCC: 0.1322, MSE: 0.1676). The plot of actual labels and predictions are in Figure 4.10. The model is biased towards neutral and "neutral close to high" labels. The model can not reliably predict low arousal labels. The slopes in predictions are in line with slopes of actual label graph

The best valence model is obtained with score fusions of VGG-FC6 and HSH models (PCC: 0.2091, MSE:0.0808). The plot of actual labels and predictions are in Figure 4.11. The model is biased towards neutral labels. The best PCC result of valence is higher than the best PCC result of arousal.

To evaluate the performance of our system, we compare it to a baseline model. Our baseline is a model that predicts only a single label regardless of input. This single label is the average of all the labels in the training set. For arousal, the baseline model's PCC is zero and MSE is 0.1057. For valence, the baseline model's PCC is

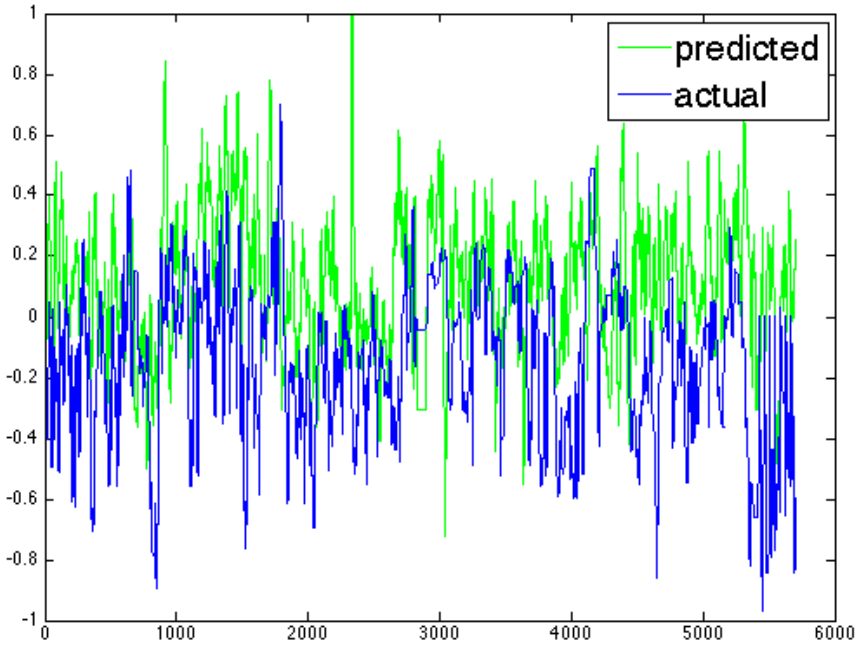


Figure 4.10. Best results for arousal.

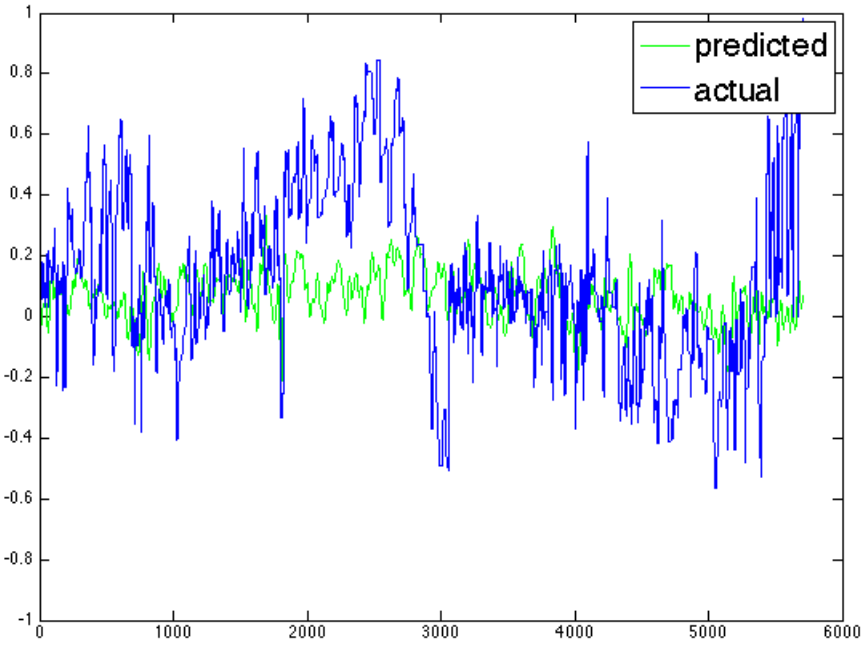


Figure 4.11. Best results for valence.

-0.6624 and MSE is 0.0885. Comparing these results to our pipeline's results, there is no improvement in MSE but our pipeline shows better PCC. This holds for both arousal and valence.

Finally, we compare our results with the best results from Mediaeval 2017. [15] achieves the highest results for both arousal (PCC: 0.3213, MSE: 0.1166) and valence (PCC: 0.3707, MSE: 0.1833). Both of their models use VGG-FC6 features. The features are modeled by SVR. We cannot reproduce their results because our training and validation sets are constructed differently than theirs. Another work achieving a good result in MediaEval 2017 is [14]. It uses visual and audio features given by the MediaEval task organizers. The lingering feature is fused with these features. It proved useful. Their arousal prediction success rate is higher than ours (PCC: 0.21485, MSE: 0.11335).

# 5. CONCLUSIONS

In this thesis, we study movie emotion estimation tasks in arousal and valence space extracting various features, applying different feature summarization and selection methods and synthesizing data.

In valence models, VGG-FC6 and HSH features are useful for predictions. They show good PCC results. HSH, AU, and MFCC are good features to predict arousal labels. They achieve some positive correlation between the actual labels and prediction. Some works show that deep learning features are useful for video emotions estimations models [10, 15]. The mid-level representations of HSV and MFCC are discriminative features for this problem.

SVM models are highly used in movie emotion estimation [11, 12, 14–17]. Our results show that ELM models are strong alternatives to SVM models because they are extremely faster than SVM [45] while being competitive in terms of performance.

Fusion techniques are used for multiple modalities in [11, 17, 20]. In our experiments, fusion techniques do not show a significant effect on the performance results.

Even though we synthesize data to balance the dataset, many models still fail. They are biased towards neutral values. It may indicate a problem arising from limited data. Although we use more sophisticated methods, we could not achieve the expected results. This can be indicative of the need for training with a larger set of data.

# REFERENCES


1. YouTube, *YouTube for Press*, 2019, `https://www.youtube.com/intl/en-GB/yt/about/press/`, accessed in April 2019.

2. Smith, G. M., *Film Structure and The Emotion System*, Cambridge University Press, 2003.

3. Plantinga, C. and G. M. Smith, *Passionate Views: Film, Cognition, and Emotion*, BRILL, 1999.

4. Hanjalic, A. and L.-Q. Xu, "Affective Video Content Representation and Modeling", *IEEE Transactions on Multimedia*, Vol. 7, No. 1, pp. 143–154, 2005.

5. Soleymani, M., G. Chanel, J. J. Kierkels and T. Pun, "Affective Characterization of Movie Scenes Based on Multimedia Content Analysis and User's Physiological Emotional Responses", *Multimedia, 2008. ISM 2008. Tenth IEEE International Symposium on*, pp. 228–235, IEEE, 2008.

6. Chen, T., Y. Wang, S. Wang and S. Chen, "Exploring Domain Knowledge for Affective Video Content Analyses", *Proceedings of the 2017 ACM on Multimedia Conference*, pp. 769–776, ACM, 2017.

7. Soleymani, M., J. J. Kierkels, G. Chanel and T. Pun, "A Bayesian Framework for Video Affective Representation", *Affective Computing and Intelligent Interaction and Workshops, 2009. ACII 2009. 3rd International Conference on*, pp. 1–7, IEEE, 2009.

8. Canini, L., S. Benini and R. Leonardi, "Affective Analysis on Patterns of Shot Types in Movies", *Image and Signal Processing and Analysis (ISPA), 2011 7th International Symposium on*, pp. 253–258, IEEE, 2011.

9. Borth, D., T. Chen, R. Ji and S.-F. Chang, "SentiBank: Large-Scale Ontology and Classifiers for Detecting Sentiment and Emotions in Visual Content", *Proceedings of the 21st ACM International Conference on Multimedia*, pp. 459–460, ACM, 2013.

10. Acar, E., F. Hopfgartner and S. Albayrak, "A Comprehensive Study on Mid-Level Representation and Ensemble Learning for Emotional Analysis of Video Material", *Multimedia Tools and Applications*, Vol. 76, No. 9, pp. 11809–11837, 2017.

11. Chen, S. and Q. Jin, "RUC at MediaEval 2016 Emotional Impact of Movies Task: Fusion of Multimodal Features.", *MediaEval*, 2016.

12. Liu, Y., Z. Gu, Y. Zhang and Y. Liu, "Mining Emotional Features of Movies.", *MediaEval*, 2016.

13. Chan, C. H. and G. J. Jones, "Affect-Based Indexing and Retrieval of Films", *Proceedings of the 13th Annual ACM International Conference on Multimedia*, pp. 427–430, ACM, 2005.

14. Yoon, S., "TCNJ-CS at MediaEval 2017 Emotional Impact of Movie Task.", *MediaEval*, 2017.

15. Jin, Z., Y. Yao, Y. Ma and M. Xu, "THUHCSI in MediaEval 2017 Emotional Impact of Movies Task.", *MediaEval*, 2017.

16. Yil, Y., H. Wang and J. Wei, "MIC-TJU in MediaEval 2017 Emotional Impact of Movies Task.", *MediaEval*, 2017.

17. Ma, Y., Z. Ye and M. Xu, "THU-HCSI at MediaEval 2016: Emotional Impact of Movies Task.", *MediaEval*, 2016.

18. Simonyan, K. and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition", *arXiv preprint arXiv:1409.1556*, 2014.

19. Acar, E., F. Hopfgartner and S. Albayrak, "Understanding Affective Content of Music Videos through Learned Representations", *International Conference on Multimedia Modeling*, pp. 303–314, Springer, 2014.

20. Yi, Y., H. Wang and Q. Li, "CNN Features for Emotional Impact of Movies Task", *MediaEval*, 2018.

21. Baveye, Y., E. Dellandrea, C. Chamaret and L. Chen, "Liris-accede: A Video Database for Affective Content Analysis", *IEEE Transactions on Affective Computing*, Vol. 6, No. 1, pp. 43–55, 2015.

22. Chawla, N. V., K. W. Bowyer, L. O. Hall and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over-Sampling Technique", *Journal of Artificial Intelligence Research*, Vol. 16, pp. 321–357, 2002.

23. Liu, Y., G. Zhonglei and T. H. Ko, "HKBU at MediaEval 2017 Emotional Impact of Movies Task.", *MediaEval*, 2017.

24. Anastasia, T. and H. Leontios, "Auth-Sgp in Mediaeval 2016 Emotional Impact of Movies Task", *MediaEval 2016 Workshop*, 2016.

25. Ekman, P., "Basic Emotions", *Handbook of Cognition and Emotion*, pp. 45–60, 1999.

26. Ekman, P., "An Argument for Basic Emotions", *Cognition & emotion*, Vol. 6, No. 3-4, pp. 169–200, 1992.

27. Moors, A., P. C. Ellsworth, K. R. Scherer and N. H. Frijda, "Appraisal Theories of Emotion: State of the Art and Future Development", *Emotion Review*, Vol. 5, No. 2, pp. 119–124, 2013.

28. Russell, J. A., "A Circumplex Model of Affect.", *Journal of Personality and Social Psychology*, Vol. 39, No. 6, p. 1161, 1980.

29. Ekman, P. and E. L. Rosenberg, *What the Face Reveals: Basic and Applied Studies of Spontaneous Expression Using the Facial Action Coding System (FACS)*, Oxford University Press, USA, 1997.

30. Tian, Y.-I., T. Kanade and J. F. Cohn, "Recognizing Action Units for Facial Expression Analysis", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 23, No. 2, pp. 97–115, 2001.

31. Lowe, D. G., "Distinctive Image Features from Scale-Invariant Keypoints", *International Journal of Computer Vision*, Vol. 60, No. 2, pp. 91–110, 2004.

32. Vedaldi, A. and B. Fulkerson, "VLFeat: An Open and Portable Library of Computer Vision Algorithms", *Proceedings of the 18th ACM International Conference on Multimedia*, pp. 1469–1472, ACM, 2010.

33. Bosch, A., A. Zisserman and X. Munoz, "Image Classification Using Random Forests and Ferns", *2007 IEEE 11th International Conference on Computer Vision*, pp. 1–8, Ieee, 2007.

34. Davis, S. and P. Mermelstein, "Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences", *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. 28, No. 4, pp. 357–366, 1980.

35. Lyons, J., *Mel Frequency Cepstral Coefficient (MFCC) Tutorial*, `http://practicalcryptography.com/miscellaneous/machine-learning/guide-mel-frequency-cepstral-coefficients-mfccs/`, accessed in April 2019.

36. Alpaydin, E., *Introduction to Machine Learning*, MIT Press, 2009.

37. Krizhevsky, A., I. Sutskever and G. E. Hinton, "Imagenet Classification with Deep Convolutional Neural Networks", *Advances in Neural Information Processing Sys-*

*tems*, pp. 1097–1105, 2012.

38. Deng, J., W. Dong, R. Socher, L.-J. Li, K. Li and L. Fei-Fei, "Imagenet: A Large-Scale Hierarchical Image Database", *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, IEEE, 2009.

39. MacQueen, J. *et al.*, "Some Methods for Classification and Analysis of Multivariate Observations", *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, Vol. 1, pp. 281–297, Oakland, CA, USA, 1967.

40. Perronnin, F. and C. Dance, "Fisher Kernels on Visual Vocabularies for Image Categorization", *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, IEEE, 2007.

41. Jaakkola, T. and D. Haussler, "Exploiting Generative Models in Discriminative Classifiers", *Advances in Neural Information Processing Systems*, pp. 487–493, 1999.

42. Kaya, H., T. Özkaptan, A. A. Salah and F. Gürgen, "Random Discriminative Projection Based Feature Selection with Application to Conflict Recognition", *IEEE Signal Processing Letters*, Vol. 22, No. 6, pp. 671–675, 2015.

43. He, H., Y. Bai, E. A. Garcia and S. Li, "ADASYN: Adaptive Synthetic Sampling Approach for Imbalanced Learning", *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, pp. 1322–1328, IEEE, 2008.

44. Cortes, C. and V. Vapnik, "Support-Vector Networks", *Machine learning*, Vol. 20, No. 3, pp. 273–297, 1995.

45. Huang, G.-B., Q.-Y. Zhu and C.-K. Siew, "Extreme Learning Machine: Theory and Applications", *Neurocomputing*, Vol. 70, No. 1-3, pp. 489–501, 2006.

46. Dellandrea, E., M. Huigsloot, L. Chen, Y. Baveye and M. Sjoberg, "The Mediaeval 2017 Emotional Impact of Movies Task", *The MediaEval 2017 Emotional Impact of Movies Task. In Proceedings of MediaEval 2017 Workshop. Dublin, Ireland*, 2017.

47. Ellis, D. P. W., *PLP and RASTA (and MFCC, and Inversion) in Matlab*, 2005, `http://www.ee.columbia.edu/~dpwe/resources/matlab/rastamat/`, accessed in April 2019.

48. Baveye, Y., E. Dellandrea, C. Chamaret, L. Chen, J.-N. Bettinelli and T. Li, *Liris-Accede*, `http://liris-accede.ec-lyon.fr/database.php`, accessed in April 2019.

49. Baltrušaitis, T., P. Robinson and L.-P. Morency, "OpenFace: an Open Source Facial Behavior Analysis Toolkit", *Applications of Computer Vision (WACV), 2016 IEEE Winter Conference on*, pp. 1–10, IEEE, 2016.

50. Glen, A., *Moving Average Function*, `https://www.mathworks.com/matlabcentral/fileexchange/41859-moving-average-function`, accessed in May 2018.

51. Zhou, H.-M. and G.-B. Huang, *Matlab Codes of ELM Algorithm*, 2012, `https://www.ntu.edu.sg/home/egbhuang/elm_kernel.html`, accessed in June 2019.

52. Chang, C.-C. and C.-J. Lin, *LIBSVM - A Library for Support Vector Machines*, 2018, `https://www.csie.ntu.edu.tw/~cjlin/libsvm/`, accessed in June 2019.

53. Elrahman, S. M. A. and A. Abraham, "A Review of Class Imbalance Problem", *Journal of Network and Innovative Computing*, Vol. 1, No. 2013, pp. 332–340, 2013.

54. van der Geest, J., *nchoosecrit(S, FUN)*, 2018, `https://www.mathworks.com/matlabcentral/fileexchange/40301-nchoosecrit-s-fun`, accessed in April 2019.

# APPENDIX A: MOVIE DISTRIBUTION IN DEVELOPMENT SET

| Movie Names | Number of Low Arousal Scene | Number of Neutral Arousal Scene | Number of High Arousal Scene |
|---|---|---|---|
| On Time | 41 | 20 | 0 |
| The Room of Franz Kafka | 24 | 25 | 0 |
| After The Rain | 3 | 95 | 19 |
| Barely Legal Stories | 30 | 124 | 43 |
| Cloudland | 22 | 117 | 0 |
| Islands | 0 | 33 | 0 |
| Lesson Learned | 34 | 106 | 15 |
| Origami | 18 | 74 | 7 |
| Attitude Matters | 0 | 98 | 175 |
| Between Viewings | 0 | 122 | 54 |
| Big Buck Bunny | 4 | 24 | 90 |
| Damaged Kung Fu | 13 | 133 | 56 |
| First Bite | 0 | 91 | 36 |
| Full Service | 0 | 79 | 144 |
| Norm | 7 | 43 | 27 |
| Parafundit | 2 | 118 | 37 |
| Payload | 0 | 132 | 82 |
| Sintel | 0 | 65 | 111 |
| Spaceman | 3 | 135 | 47 |

Figure A.1. Movie information in case of arousal space in the train set.

| Movie Names | Number of Low Arousal Scene | Number of Neutral Arousal Scene | Number of High Arousal Scene |
|---|---|---|---|
| To Claire From Sonny | 44 | 18 | 20 |
| Nuclear Family | 27 | 290 | 22 |
| Riding The Rails | 19 | 160 | 0 |
| Superhero | 14 | 161 | 52 |
| You Again | 17 | 156 | 0 |
| Chatter | 0 | 58 | 43 |
| Decay | 86 | 346 | 480 |
| Elephant's Dream | 0 | 47 | 83 |
| Tears of Steel | 4 | 103 | 39 |
| The Secret Number | 0 | 116 | 69 |
| Wanted | 0 | 7 | 15 |

Figure A.2. Movie information in case of arousal space in the validation set.

| Movie Names | Number of Negative Valence Scene | Number of Neutral Valence Scene | Number of Positive Valence Scene |
|---|---|---|---|
| After The Rain | 73 | 44 | 0 |
| Chatter | 43 | 56 | 2 |
| Lesson Learned | 58 | 96 | 1 |
| Parafundit | 59 | 98 | 0 |
| Superhero | 90 | 132 | 5 |
| The Room of Franz Kafka | 26 | 23 | 0 |
| Barely Legal Stories | 50 | 127 | 20 |
| Cloudland | 23 | 87 | 29 |
| Elephant's Dream | 26 | 94 | 10 |
| First Bite | 10 | 117 | 0 |
| Payload | 24 | 190 | 0 |
| Big Buck Bunny | 3 | 12 | 103 |
| Damaged Kung Fu | 0 | 65 | 137 |
| Full Service | 0 | 48 | 175 |
| Islands | 0 | 3 | 30 |
| Norm | 0 | 34 | 43 |
| On Time | 0 | 26 | 35 |
| Origami | 0 | 20 | 79 |
| Riding The Rails | 0 | 76 | 103 |
| Spaceman | 19 | 109 | 57 |

Figure A.3. Movie information in case of valence space in the train set.

| Movie Names | Number of Negative Valence Scene | Number of Neutral Valence Scene | Number of Positive Valence Scene |
|---|---|---|---|
| **Decay** | 527 | 385 | 0 |
| **The Secret Number** | 87 | 98 | 0 |
| **To Claire From Sonny** | 26 | 47 | 9 |
| **Nuclear Family** | 46 | 206 | 87 |
| **Sintel** | 50 | 88 | 38 |
| **Tears of Steel** | 0 | 118 | 28 |
| **Attitude Matters** | 0 | 83 | 190 |
| **Between Viewings** | 5 | 92 | 79 |
| **Wanted** | 0 | 7 | 15 |
| **You Again** | 0 | 99 | 74 |

Figure A.4. Movie information in case of valence space in the validation set.

# APPENDIX B: PLOT OF ACTUAL AND PREDICTED LABELS

The plots of actual labels and predictions of the experiments are shown in this part.

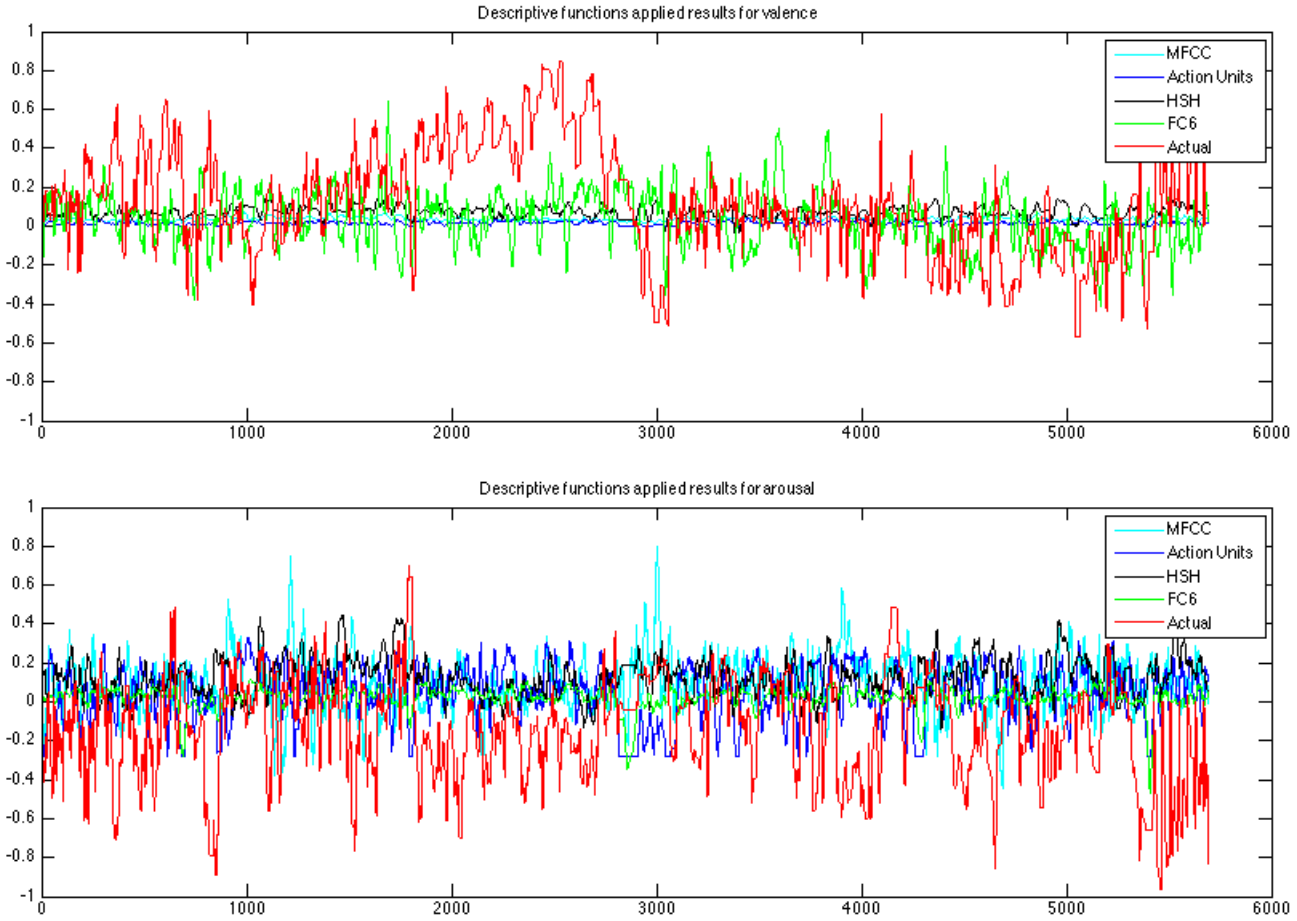


Figure B.1. Results with descriptive functions and ELM.

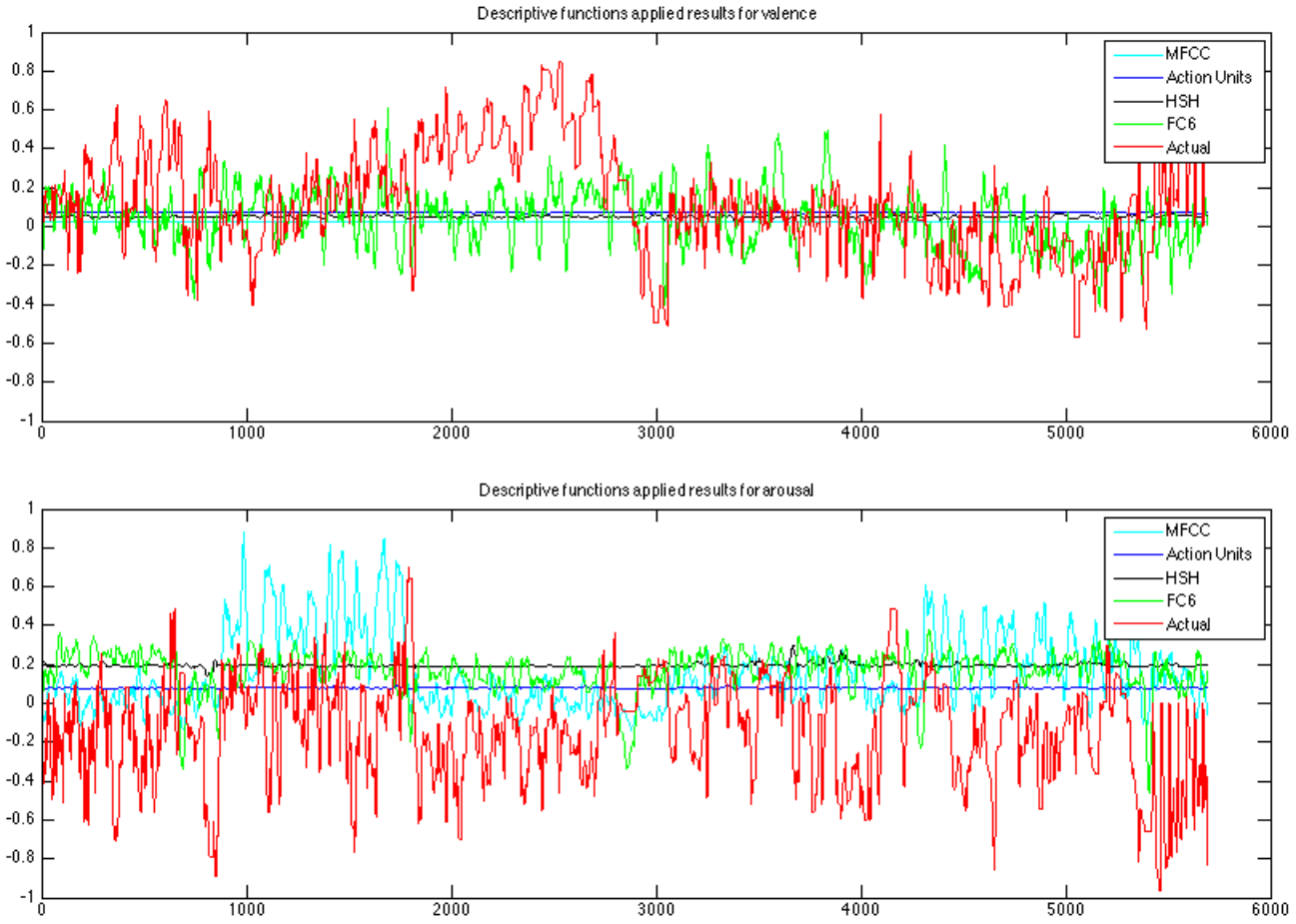


Figure B.2. Results with descriptive functions and SVM.

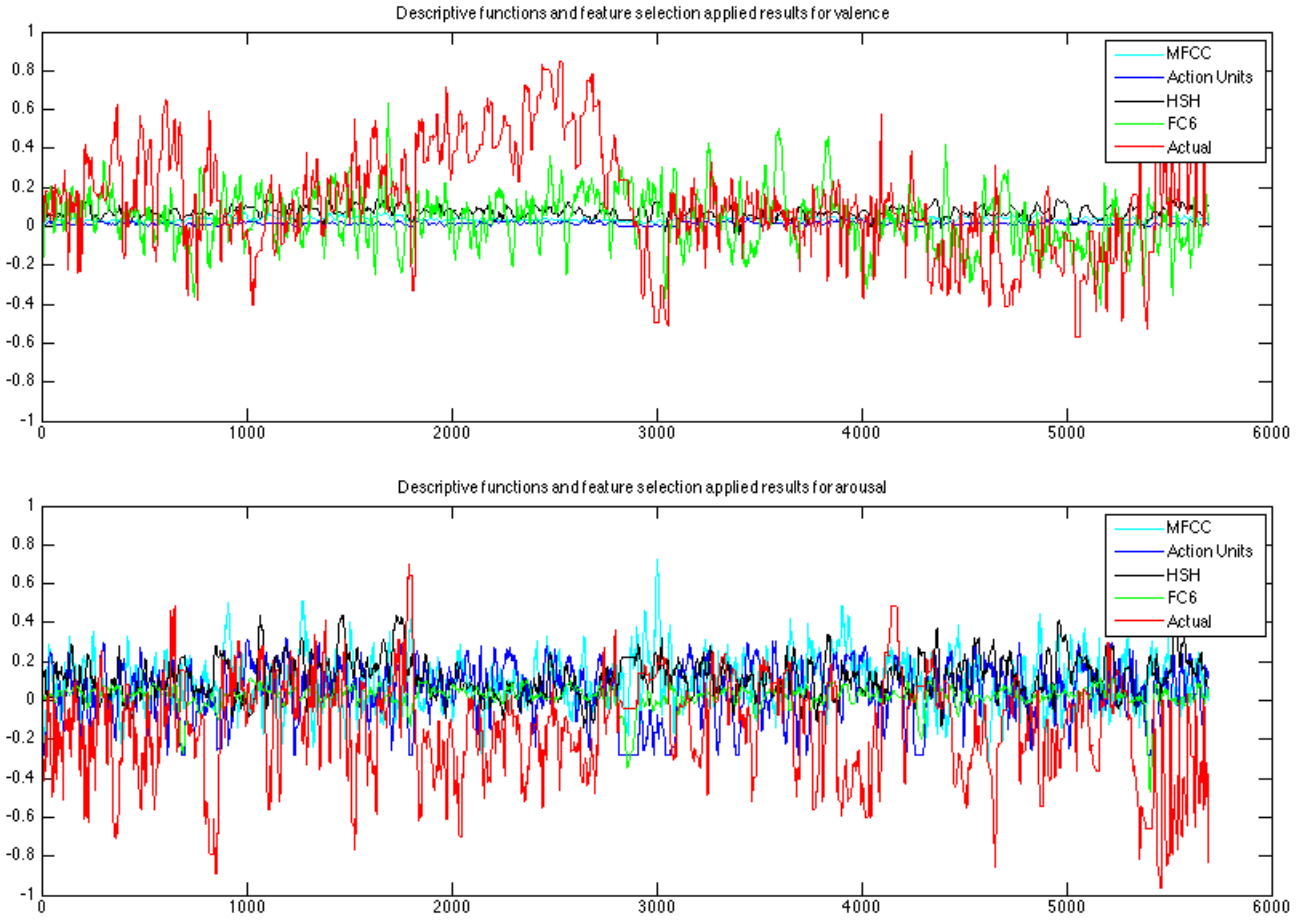


Figure B.3. Results with descriptive functions, feature selection and ELM.

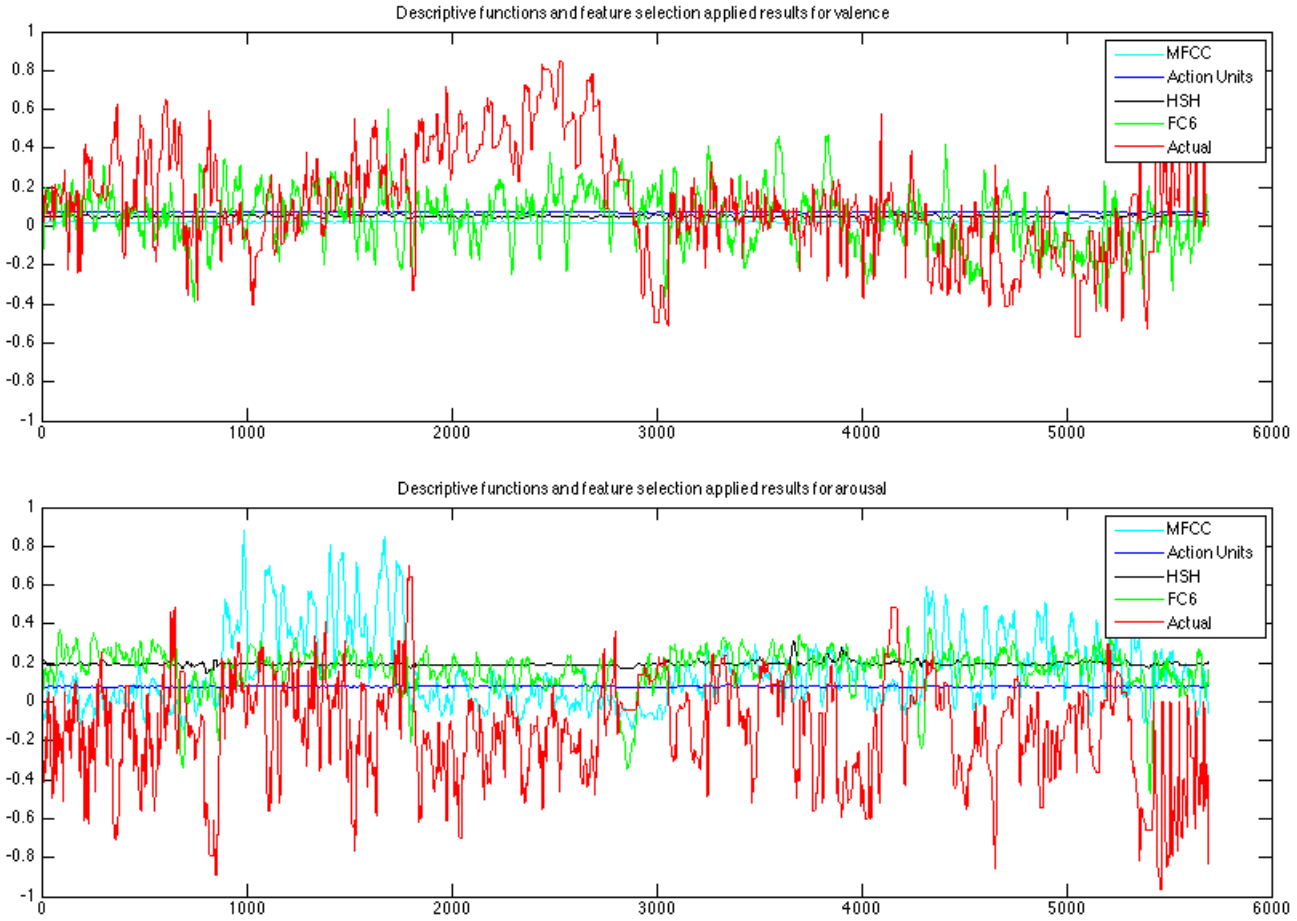


Figure B.4. Results with descriptive functions, feature selection and SVM.

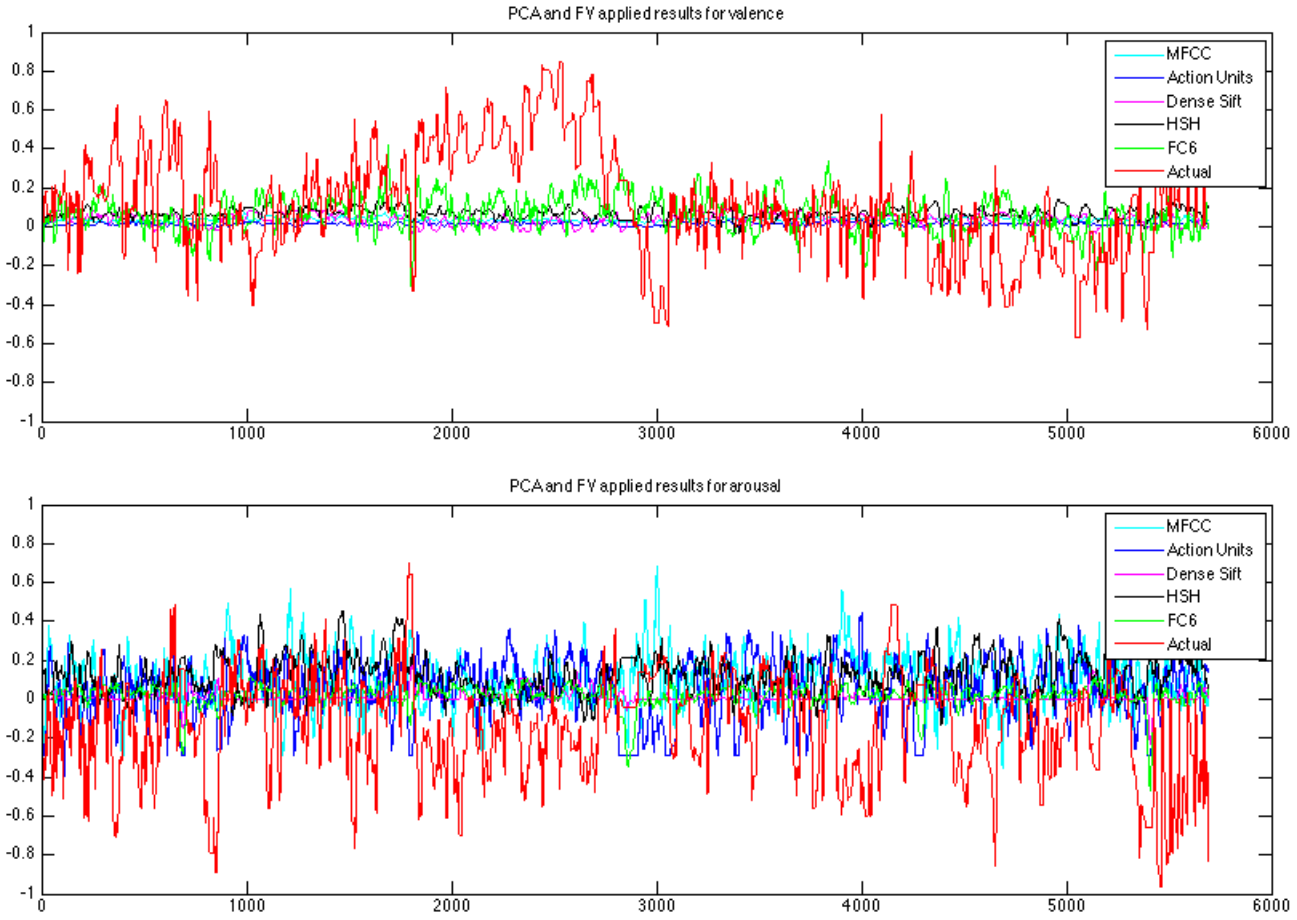


Figure B.5. Results with PCA, FV and ELM.

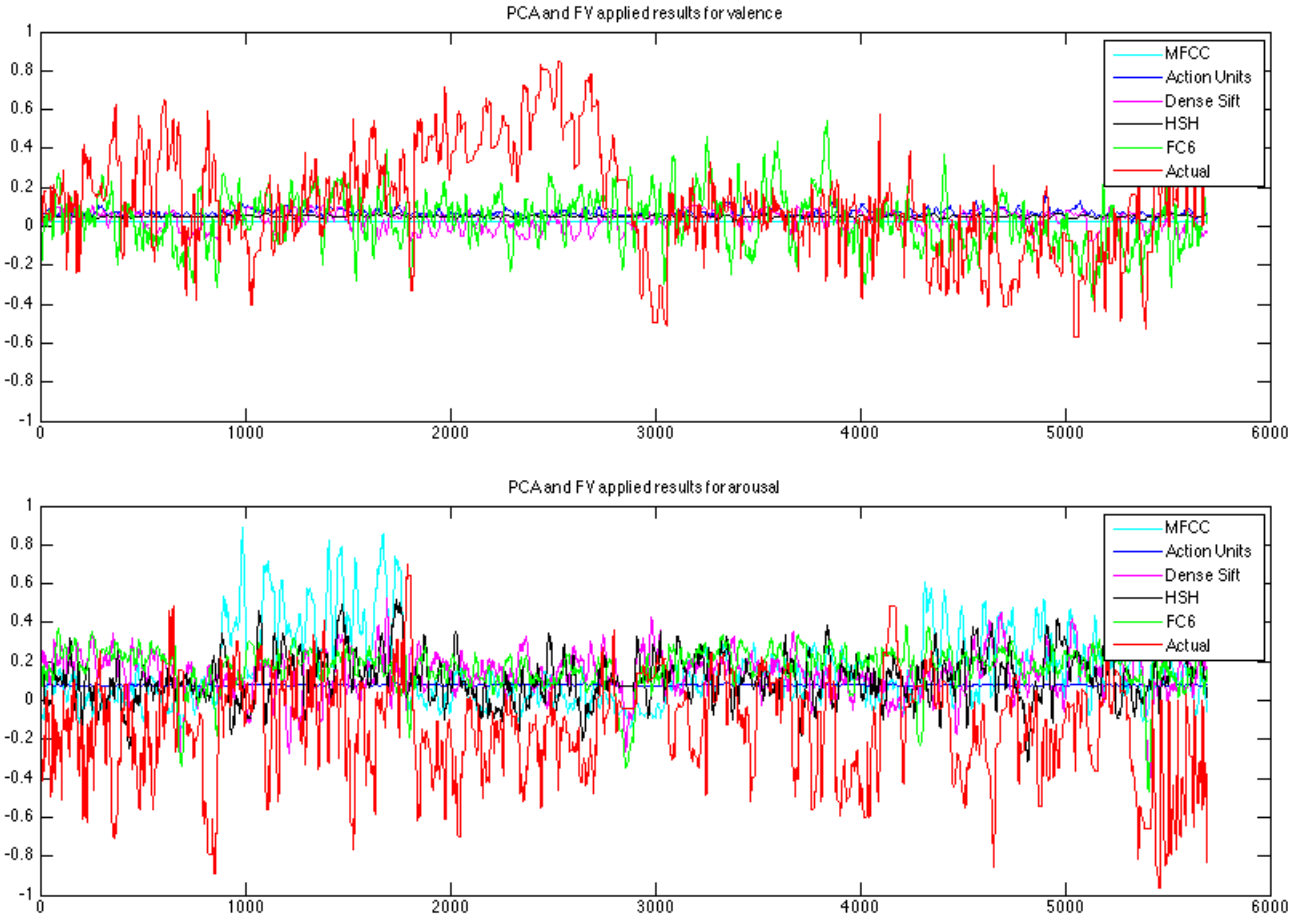


Figure B.6. Results with PCA, FV and SVM.

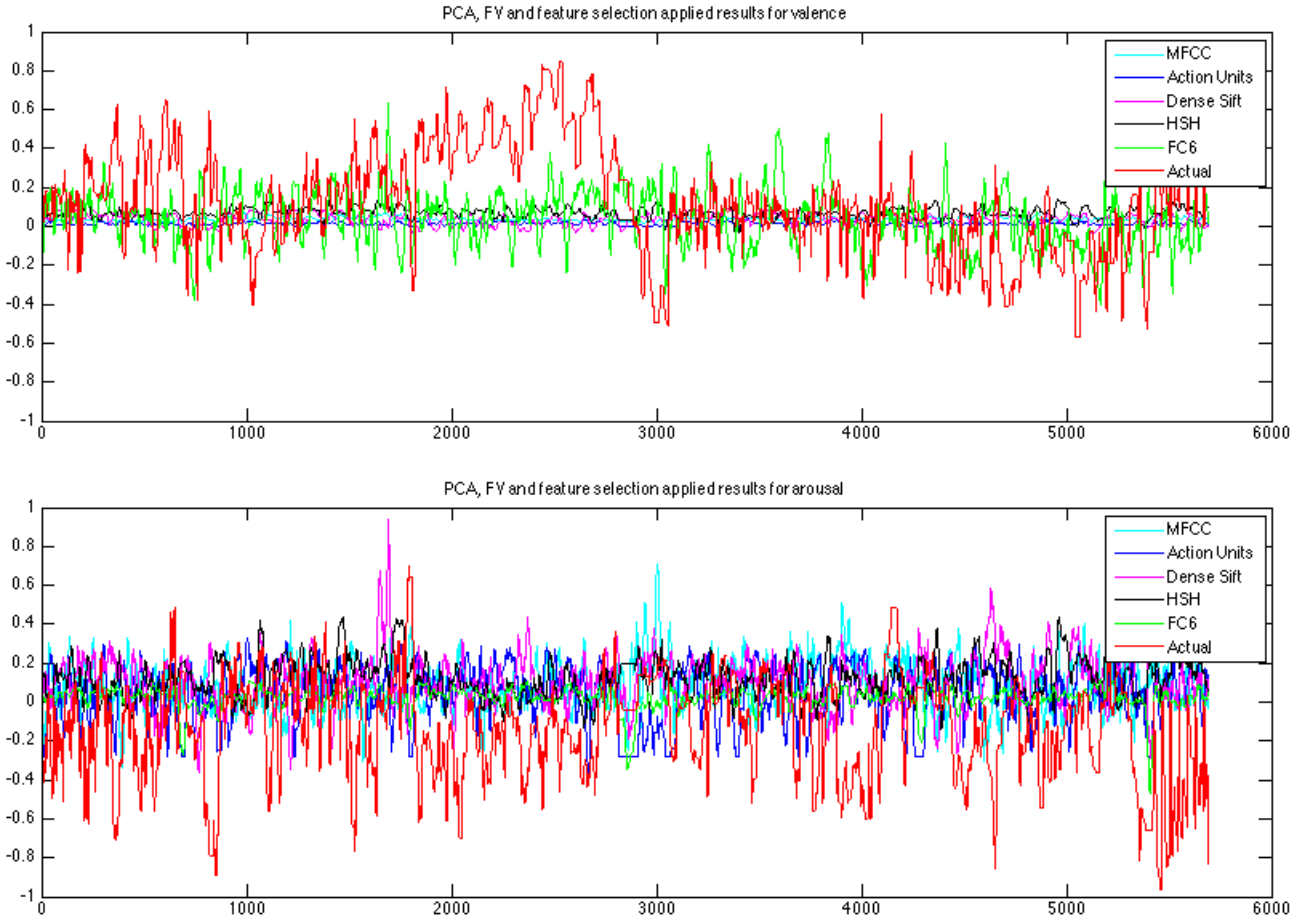


Figure B.7. Results with PCA, FV, CCA and ELM.

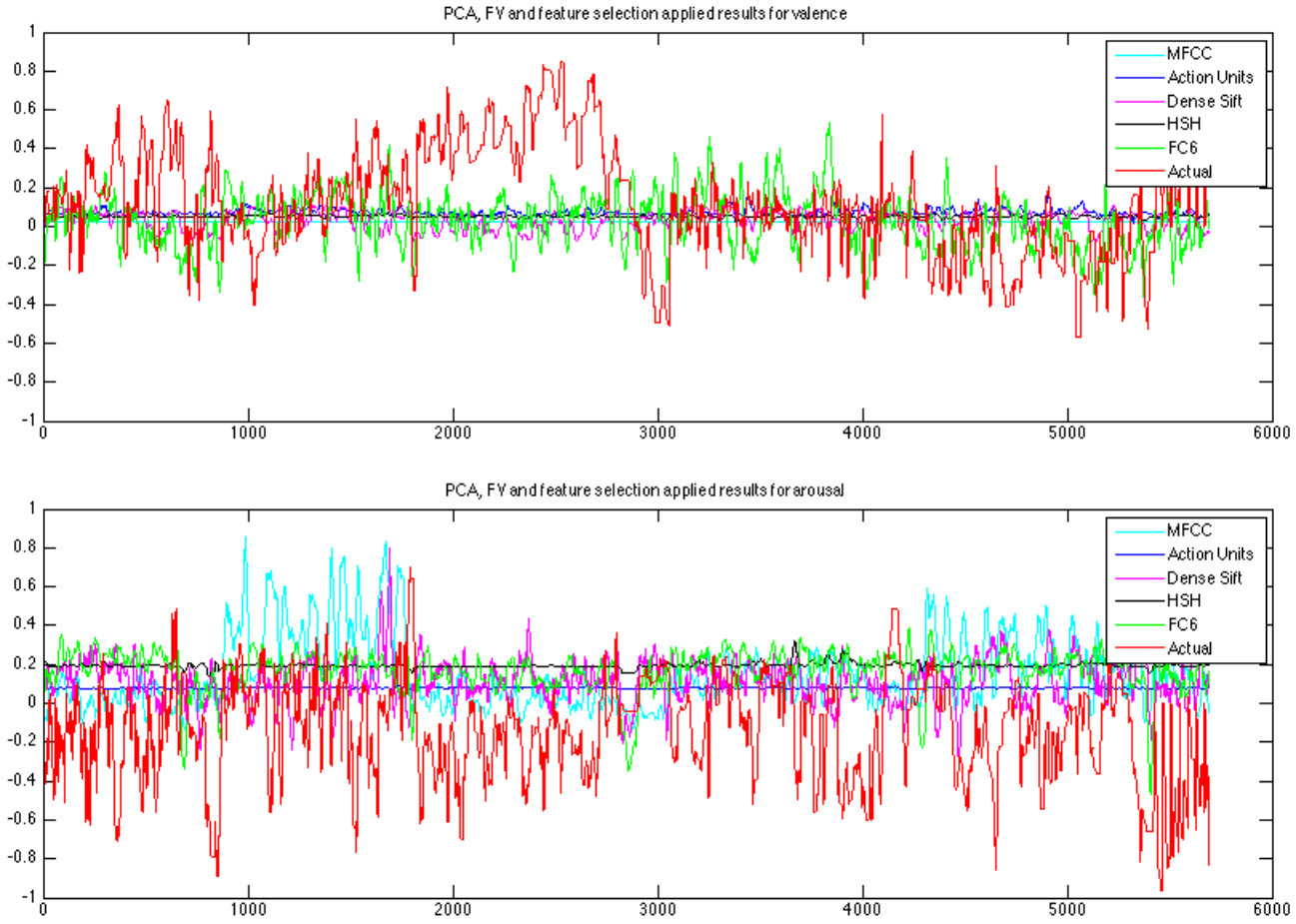


Figure B.8. Results with PCA, FV, CCA and SVM.

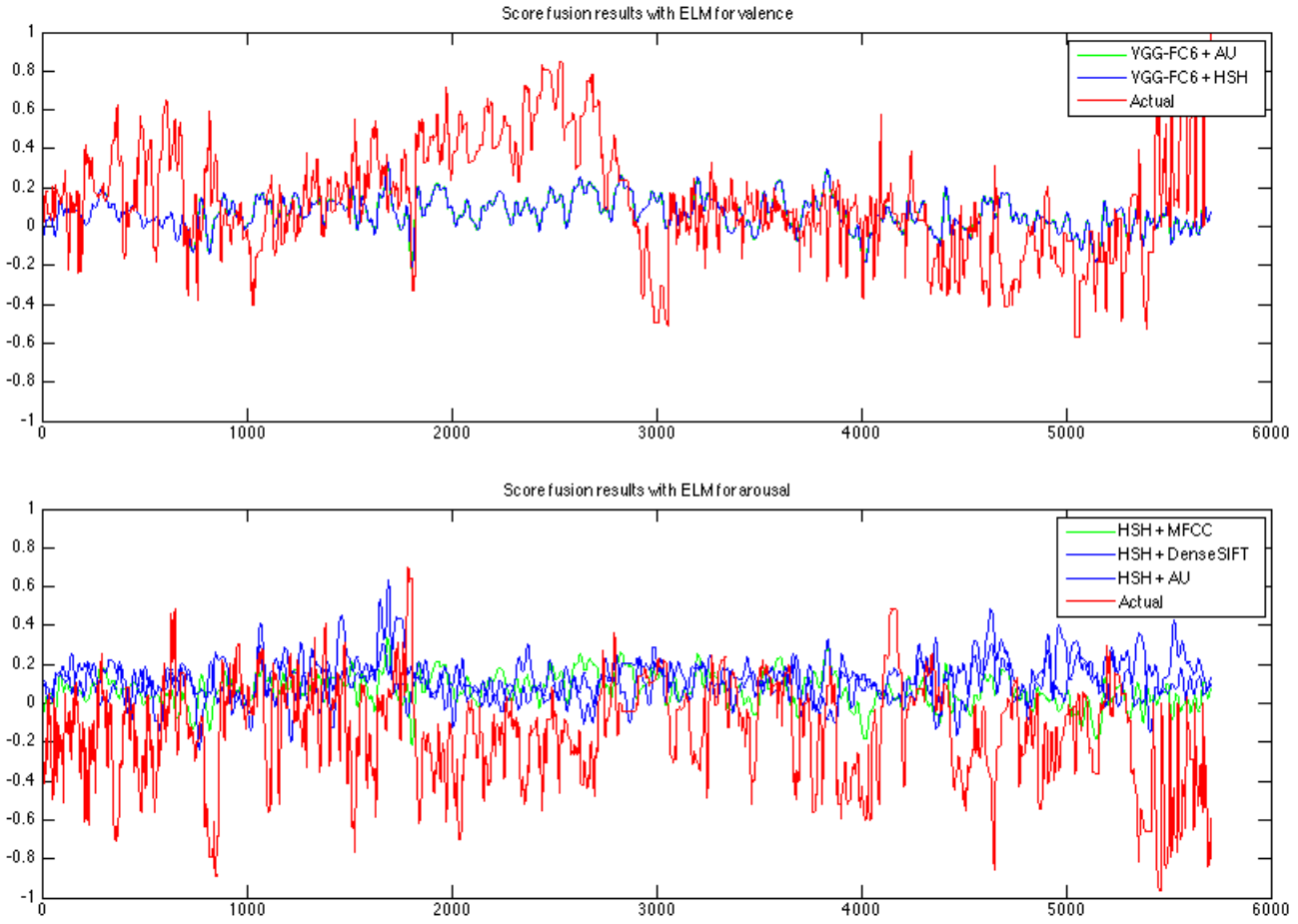


Figure B.9. Score fusion applied to ELM results.

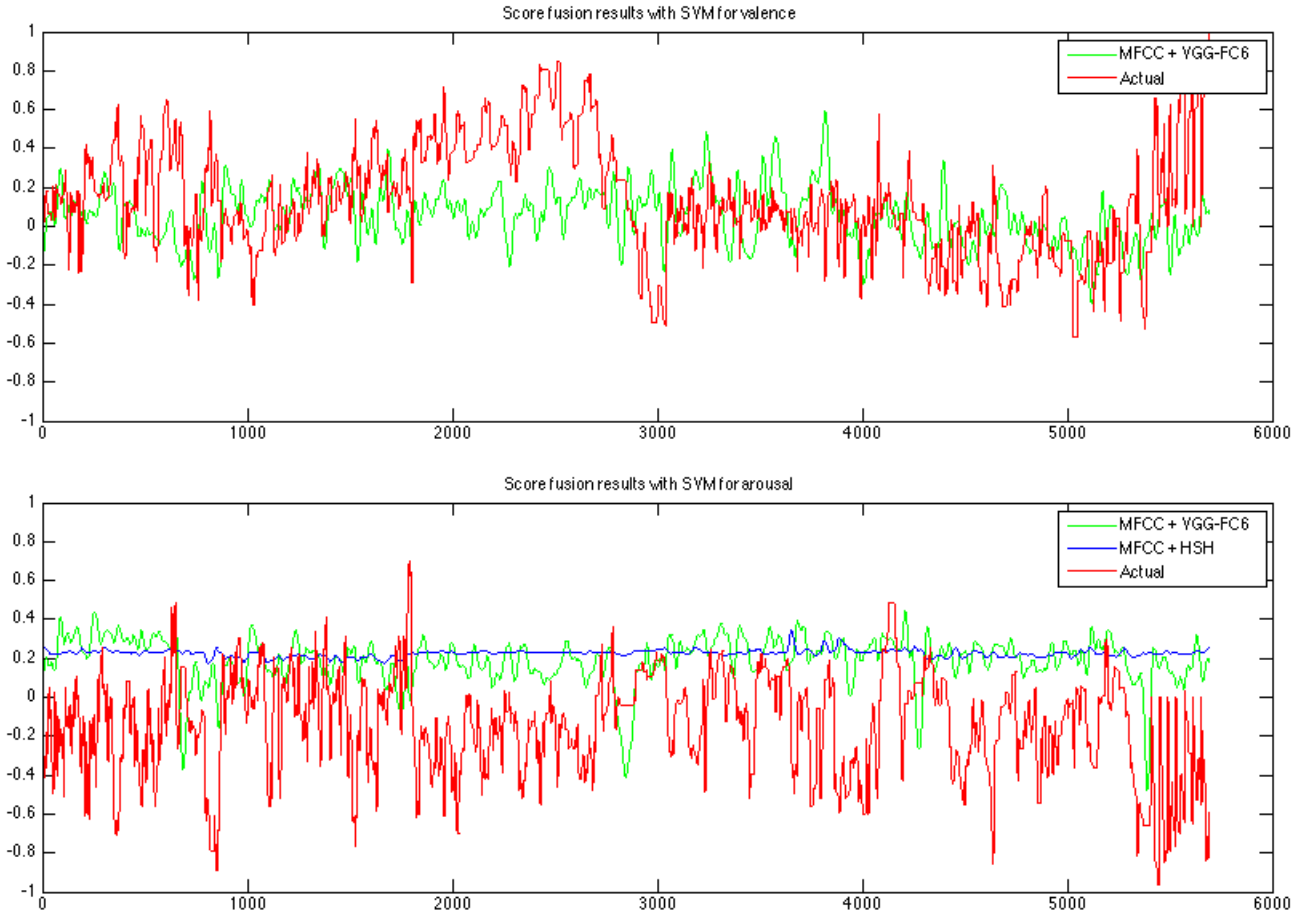


Figure B.10. Score fusion applied to SVM results.

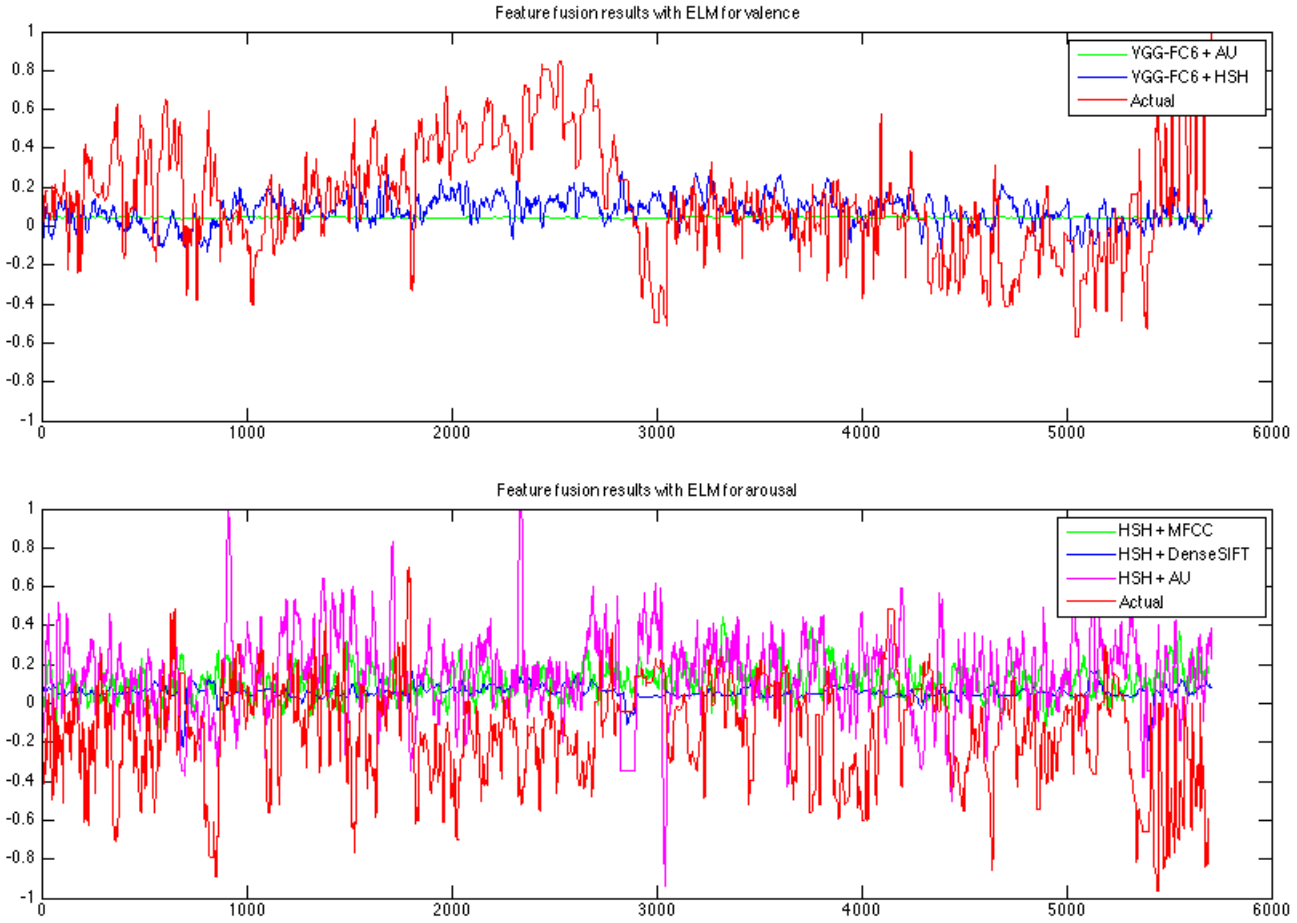


Figure B.11. Feature fusion results with ELM.

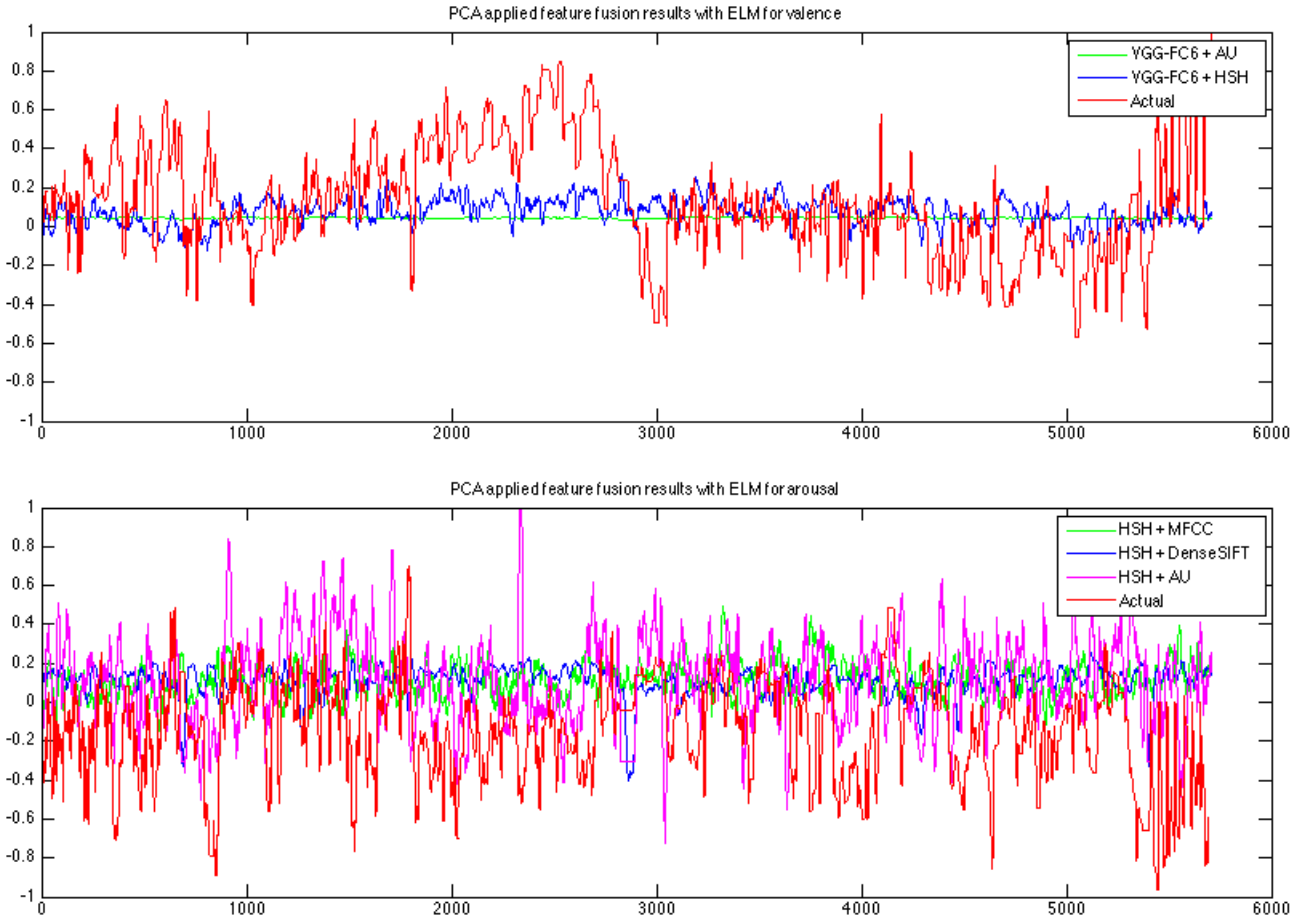


Figure B.12. PCA applied feature fusion results with ELM.

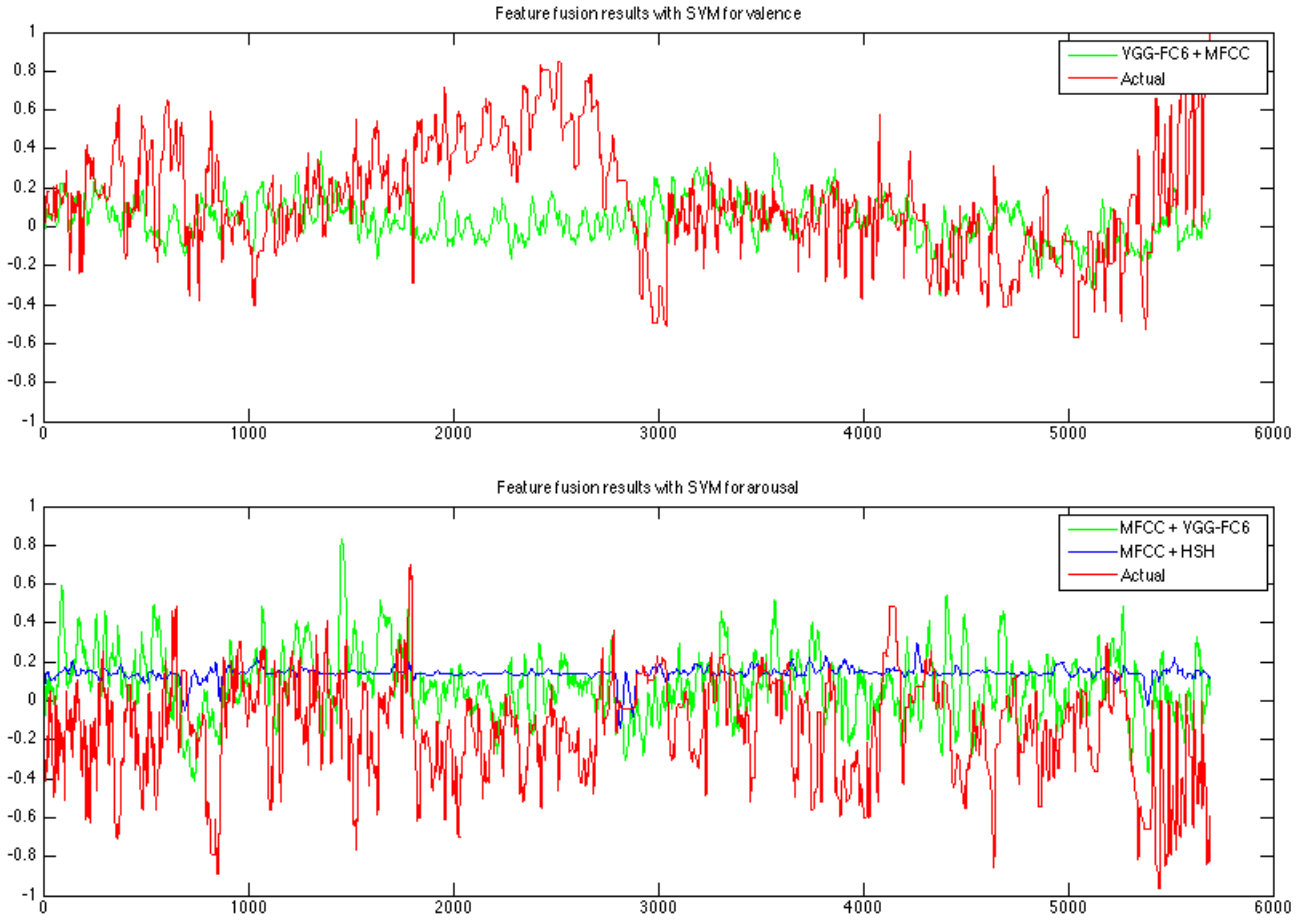


Figure B.13. Feature fusion results with SVM.

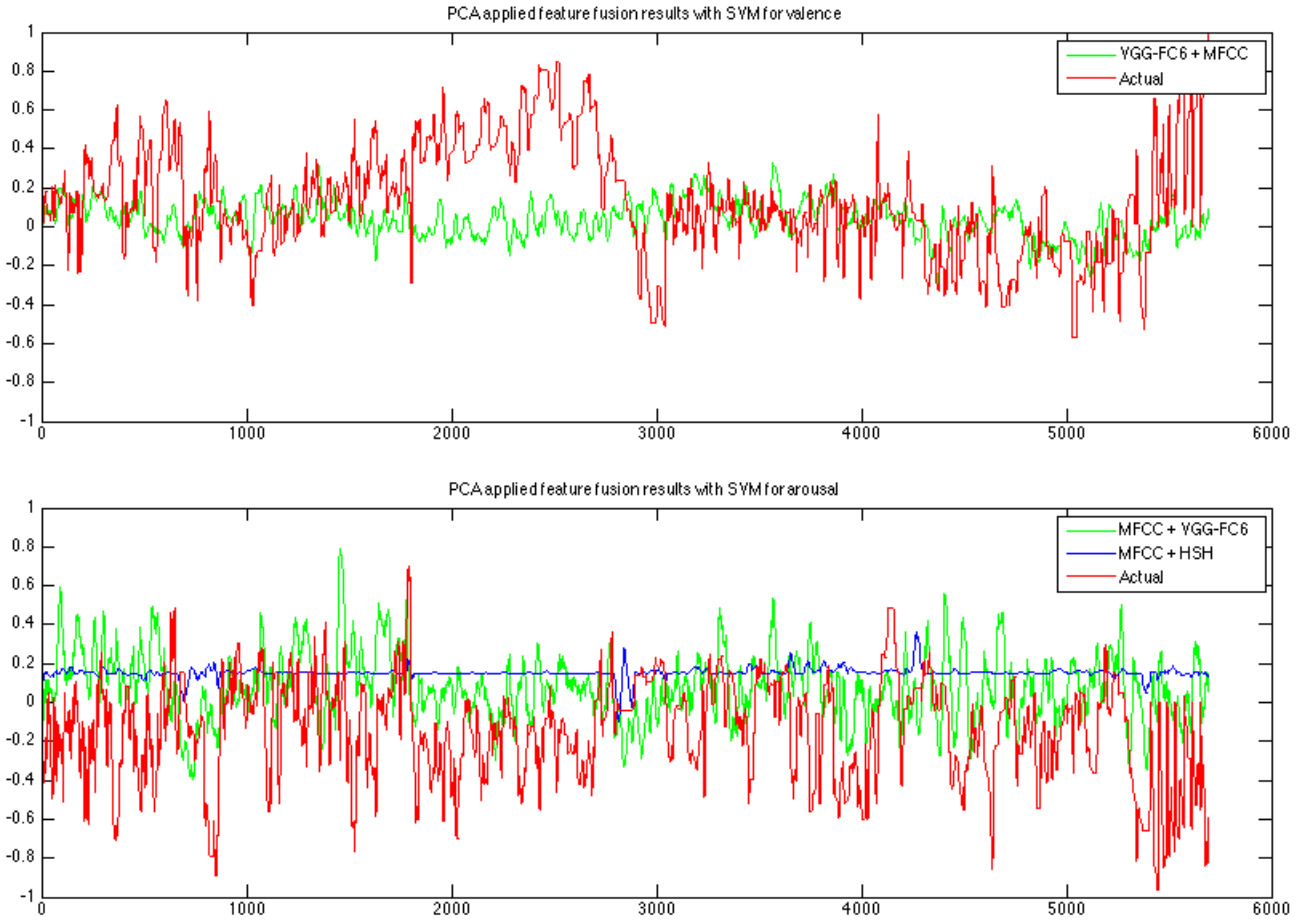


Figure B.14. PCA applied feature fusion results with SVM.