DOKUZ EYLÜL UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

MULTIMODAL EMOTION RECOGNITION IN VIDEO

by
Taner DANIŞMAN

June, 2008
İZMİR

MULTIMODAL EMOTION RECOGNITION IN VIDEO

A Thesis Submitted to the
Graduate School of Natural and Applied Sciences of Dokuz Eylül University
In Partial Fulfillment of the Requirements for the Degree of Doctor Philosophy in
Computer Engineering, Computer Engineering Program

by
Taner DANIŞMAN

June, 2008
İZMİR

PhD. THESIS EXAMINATION RESULT FORM

We have read the thesis entitled **"MULTIMODAL EMOTION RECOGNITION IN VIDEO"** completed by **TANER DANIŞMAN** under supervision of **ASSISTANT PROFESSOR DR. ADİL ALPKOÇAK** and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Doctor Philosphy.

.....

Assist.Prof.Dr. Adil ALPKOÇAK

Supervisor

.....	.....
Assist.Prof.Dr. Haldun SARNEL	Prof.Dr. Tatyana YAKHNO
Thesis Committee Member	Thesis Committee Member
.....	.....
Prof.Dr. Ahmet KAŞLI	Prof.Dr.Alp KUT
Examining Committee Member	Examining Committee Member

Prof.Dr. Cahit HELVACI
Director
Graduate School of Natural and Applied Sciences

ACKNOWLEDGEMENTS

At the very beginning, I would like to thank all those people for giving me support and help in everything I do for accomplishing this thesis. My dearest mother Asiye DANIŞMAN, my father Ahmet DANIŞMAN, my brother Ertan DANIŞMAN for all my life you will be my inspiration.

I would like to thank my colleagues at D.E.U. Computer Engineering Department for their efforts in creating EFN dataset. In addition, I would like to acknowledge the support from the Dokuz Eylul University and TUBITAK.

I would like to thank my thesis committee members Assist.Prof.Dr. Haldun SARNEL and Prof.Dr. Tatyana YAKHNO for their advices and suggestions in tracking meetings.

Finally, I am grateful to my thesis advisor, Assist.Prof.Dr. Adil ALPKOÇAK for his guidance, helpful suggestions, and encouragement during the course of my study. In addition, the readability of this thesis has greatly benefited from all his feedback.

Taner DANIŞMAN

MULTIMODAL EMOTION RECOGNITION IN VIDEO

ABSTRACT

This thesis proposes new methods to recognize emotions in video considering visual, aural, and textual modalities.

In visual modality, we proposed a new facial expression recognition algorithm based on curve fitting method for frontal upright faces in still images. Proposed algorithm considers the shape of mouth region to recognize happy, sad and surprise emotions. According to our experiments, our method achieves 89% average accuracy. In addition, we proposed a skip frame based approach for video segmentation.

In aural modality, we present an approach to emotion recognition of speech utterances that is based on ensembles of Support Vector Machine classifiers. In addition, we proposed a new approach for Voice Activity Detection in audio signal, and presented a new emotional dataset called Emotional Finding Nemo based on a popular animation film, Finding Nemo.

In textual modality, we proposed an emotion classification method based on Vector Space Model (VSM). Experiments showed that VSM based emotion classification on short sentences can be as good as other well-known methods including Naïve Bayes, SVM, and ConceptNet on predicting emotional class of a given sentence.

Finally, we use late fusion technique with a web-based interface for emotional browsing of TRECVID dataset, and we developed an emotion-aware video player to demonstrate the system performance.

Keywords: Multimodal Emotion Classification, Facial Expression Recognition, Voice Activity Detection, Emotion Classification of Speech, Emotion Classification of Text, Emotional Datasets, Late Fusion.

VİDEO İÇERİSİNDE ÇOK ALANLI DUYGU TANIMA

ÖZ

Bu tez, görsel, işitsel ve metinsel alanlar içeren video için yeni çok alanlı duygu tanıma yöntemlerini sunar.

Görsel alanda, resimlerdeki yüzlerin duygusal ifadesinin tanınması için eğri uydurma yöntemine dayalı yeni bir yüz ifadelerini tanıma algoritması önerilmiştir. Önerilen yöntem, ağız bölgesinin şeklini gözönüne alarak, mutluluk, üzgün olma ve şaşkınlık duygularını bulmaktadır. Yapılan deneyler sonucunda yöntemimiz 89% ortalama doğruluk oranına ulaşmaktadır. Buna ek olarak, yeni bir kare atlamalı video bölütleme yöntemi önerilmiştir.

İşitsel alanda, konuşma kesitlerinin duygusal sınıflarının, topluluk destek vektör makinaları ile sınıflandırılması amacıyla bir yaklaşım sunulmuştur. Buna ek olarak, ses sinyali içerisinden konuşma aktivitesi bulma alanında yeni bir yaklaşım ve Emotional Finding Nemo adında yeni bir duygu veriseti sunulmuştur.

Yazı alanında, Vektör Uzay Modeli tabanlı duygu sınıflandırması yöntemi sunulmuştur. Yapılan deney sonuçlarına göre önerilen yöntem, verilen bir cümlenin duygusal sınıfının tahminlenmesi konusunda kısa cümleler için en az, Bayes, Destek Vektör Makinları ve ConceptNet gibi bilinen diğer yöntemler kadar iyi sonuç üretmektedir.

Son olarak, web tabanlı bir arayüz ile geç birleştirme yöntemi kullanılarak, TRECVID verisetinin duygu içerikli olarak taranması sağlanmış, ve sistem performansını göstermek amacıyla duyguyu gösterebilen video oynatıcısı geliştirilmiştir.

Anahtar Kelimler: Çok Alanlı Duygu Sınıflandırma, Yüzsel İfade Bulma, Ses Aktivitesi Bulma, Ses İçerisinde Duygu Bulma, Metin İçerisinde Duygu Bulma, Duygu Tabanlı Veri Setleri, Geç Birleştirme

CONTENTS

CHAPTER ONE
INTRODUCTION

Over the last quarter century, there is increased body of research on recognition of emotional expressions on different environments. Emotions are complex psychophysical processes of human behavior that is a part of psychology, neuroscience, cognitive science, and artificial intelligence. On the other hand, emotional understanding is an important issue for intentional behaviors. Since, emotions convey our feeling to others, without emotions we behave like a robot.

Current state-of-art in computer human interaction largely ignores emotion whereas it has a biasing role in human-to-human communication in our everyday life. In the mean time, a successful computer human interaction system should be able to recognize, interpret, and process human emotions. The term "Affective Computing", first used by Picard (1997) at MIT Media Lab., deals with systems, which can process emotion signals. Affective computing could offer benefits in an almost limitless range of applications such as computer aided tutoring, customer relationship management, automatic product reviews and even card driver safety systems.

Human brain consider emotional stimulus during decision-making phase, therefore, it has advantages over rule-based systems used by computers. The difference also occurs in human computer interaction (HCI) where human tries to adapt to computer. In order to enhance the current state-of-art in HCI methods, first we first need to enhance Emotional Expression Recognition (EER) capabilities of computers.

1.1 Background and Problem Definition

Considering natural interaction mechanisms, human-to-human interaction not only occurs with facial expressions but also occurs with speech and content of conversation. Considering visual modality, emotions exists in facial muscle

movements and body gestures. In addition, aural modality has both the linguistic and paralinguistic information carrying the emotion signals. Text is another carrier for emotion. Recognition of emotional state of a user during HCI is desirable for more natural, intelligent, and human like interaction. Data gathered to recognize human emotion is often analogous to the cues that humans use to perceive emotions in other modalities. Human brain has a perfect fusion scheme for many different sources of signals, objects, relations, and events for emotions. Hence, human emotion recognition is multimodal in nature, which includes textual, visual, and acoustic features; Multimodal Emotion Recognition (MER) is required for more intelligent HCI.

Detecting emotion in video requires multimodal analysis but each modality has its own features, indexes, and interfaces, which make them difficult to combine with other modalities. Efficient access of desired information in terms of Human Emotion Recognition (HER) in huge amount of video resources requires a set of difficult and usually CPU intensive tasks such as segmentation, feature extraction, feature reduction, classification, high level indexing, and retrieval on each modality.

Perception of emotional expression occurs in visual, aural, and textual dimensions in human brain. Video is the most similar medium mimic to human-to-human communication channels. For this reason, EER studies on video, covering more than one modality getting more important. Video is the material that has the richest source of information for EER. It is a complex data having multiple data channels and it has widespread usage in many different areas. TV broadcasters, low cost digital cameras, and even cellular phones have the capability of capturing and storing moving pictures. Because of wide spread use of these devices, there is a corresponding need for efficient and effective access to those huge amount of video data produced. As the World Wide Web grows faster than advances in existing search engine technology, there is an urgent need to develop Next Generation Intelligent Multimedia Search Engines capable of content-based analysis and retrieval. Therefore, indexing and retrieval of video becomes more and more important.

1.2 Goal of Thesis

The goal of the thesis is to propose new methods for EER for a video-based emotion classification system by using multimodal features come from visual, textual, and aural modalities. To date, rich information carried on other modalities generally ignored including audio, texts, subtitles, and transcripts. For this reason, we aimed to develop methods to recognize emotion in visual, aural and text modalities of video.

1.3 Methodology

For each modality, we started with segmentation process where the source of data needs to be segmented into smaller units. First, video is segmented into shots and key frames, then audio is segmented into speech vs. non-speech segments, finally according to the boundaries of shots in visual modality, corresponding sentences segmented into group of sentences and/or snippets.

In visual domain, we employed rule-based and threshold-based methods for segmentation and emotion classification. In aural domain, we used ensembles of Support Vector Machines (SVM) for VAD and emotional speech recognition at utterance level. We developed emotion annotation tool and created a dataset for emotional speech experiments. Finally, in textual domain, we employed statistical information retrieval methods for textual emotion recognition using VSM.

1.4 Contributions of Thesis

The main contributions of this thesis can be presented in three groups such as visual, aural and textual that simply underlines the emotion recognition in different modalities of video.

In visual modality, we proposed a new facial expression recognition approach based on curve fitting technique, which is able to detect emotions in single still images rather than consecutive images. We also developed an effective video shot boundary detection method using skip frame approach for a better efficiency.

In aural modality, we proposed an approach for emotional speech recognition problem using ensemble of Support Vector Machines. Our approach outperforms the state-of-the art results on same test sets. Additionally, we addressed the automatic creation of large-scale training and test set for Voice Activity Detection (VAD) task. We also introduced a new multimodal emotion dataset called Emotional Finding Nemo (EFN) having emotionally annotated speech and textual information in English language for emotion detection task. Furthermore, EFN can be easily transformed into other languages since different dubbed version of this movie is available.

Third, in textual modality, we proposed a new method, based on Vector Space Model, for text-based emotion recognition. The experimentations showed that the new approach classify short sentences better than ConceptNet, and it can be as good as other powerful text classifiers such as Naïve Bayes and SVM.

Finally, we also developed an emotion-aware video player, to demonstrate system performance.

1.5 Thesis Organization

The rest of the thesis is organized as follows. The next, Chapter 2 presents a literature survey on EER including visual, aural, and textual modalities.

The next three chapters, Chapter 3, 4 and 5 describe our proposal for visual, audio and text based EER, respectively. Chapter 3 presents details of our facial expression recognition algorithm for video. It includes segmentation using Shot Boundary Determination (SBD), face detection, and facial expression recognition tasks. In Chapter 4, we present our proposal on emotional speech recognition for video including VAD, a new method for automatic creation of large-scale speech training sets, and an approach for emotion recognition in speech using ensemble of SVM's. In addition, it presents a tool for speech based emotion annotation, a new emotional speech dataset called EFN, and experimental results on different emotional datasets. In Chapter 5, we introduce a new method for text based emotion classification using

VSM. Chapter 6 presents the experimental results on TRECVID2006 dataset for multimodal emotion recognition using proposed approaches in Chapter 3, 4 and 5.

Finally, Chapter 7 concludes the thesis, and presents possible future works on this topic.

CHAPTER TWO
DEFINITIONS and RELATED WORK

This chapter describes the problem definitions and related works for visual, audio and text modalities as well as multimodal approaches for emotion recognition in video.

Video is a complex data having multiple modalities including visual, audio, and textual information channels. Most of the research in this area is limited to use of single modality, which is usually visual modality and ignores the rich information carried on audio and textual modalities. Combined use of multiple modalities exists in video documents can produce highly efficient indexing & retrieval mechanisms. However, multimodal analysis of video requires high resource and computation time as well as it introduces multimodality integration problem.

2.1 Visual Modality

"A picture is worth a thousand words."

Napoleon Bonaparte (1769-1821)

What about thousands of pictures? Video indexing and retrieval is necessary for many fields, which uses large-scale of video collections such as TV stations, journalists, and even home users. There are gigabytes of video data generated every second and it is crucial to index those unformatted video data for efficient access. In recent years, advances in digital video technology make the digital video cameras and video recorders available to wide range of end user. Among others video has both spatial and temporal dimensions such as still images, motion and audio makes it the most complex multimedia object.

In the simplest case, available video browsers do not support content-based browsing. For example if the user wants to jump to a scene in a video where Bruce Willis appears and says "Help me!" user should seek the whole video randomly or

sequentially to find desired scene. Similarly, a typical movie has duration of 1-1.5 hours; consist of approximately 160,000 frames having rich content in addition to redundant information. Without any compression, such a standard file has approximately even 30 GB of data (320×240×24×30×60×90). To solve this problem MPEG, which stands for Moving Picture Experts Group, develops a family of standards used for coding audio-visual information (e.g., movies, video, music) in a digital compressed format. Compression based MPEG standards solves the storage problem but there is still an indexing & retrieval problem. Traditional textual indexing techniques do not satisfy user needs because they are limited on describing the rich multimodal content.

Without availability of video indexing applications, huge sized video data will need to be addressed by human intervention (usually librarians) that is not efficient in real world situations. Therefore, there is a corresponding need for tools that satisfies efficient indexing, browsing, and retrieval of video data. Dublin Core and especially MPEG-7 application metadata standards developed. Visual Information Retrieval systems deals with indexing and retrieval of visual data but they uses low level information, mostly depends on color histograms, edge detection, shape and texture properties and do not consider high level semantic information such as human centered events and objects. Instead, it is represented by structured components namely video, scene, shot and frames respectively.

A shot is simply the basic unit of video or the sequence of frames resulting from a continuous uninterrupted recording of video data Yongsheng & Ming (n.d.). Another word a shot is a continuous sequence of frames that presents continuous action captured from one camera. A scene is composed of one or more shots, which present different views of the same event, related in time or space. There are problems in automatically defining scene regions. For example, a person looking at a sport car would be one shot or scene also two camera shots showing different people looking at the sport car might also be one scene if the important object was the sport car and not the people. However, in reality, using those structures for video browsing does not fulfill expectations of the users during video browsing and retrieval. According to the research of Yeung, Yeo, & Liu (1996), there exist 300 shot in 15-minute video

segment of the movie “Terminator II – The Judgment Day” and total movie length is 139 minutes. In this case, it is difficult to browse entire movie using shot structure. Therefore, there exists a semantic gap between user requirements and the actual response of the systems. Semantic gap is the lack of coincidence between the information that one can extract from the data and the interpretation that the same data has for a user in a given situation, (Smeulders, Worring, Santini, Gupta, & Jain, 2000). Achievement in this domain can be used in next generation intelligent robotics and artificial intelligence, automatic product reviews and even in car driver safety systems.

2.1.1 Video Indexing and Retrieval

Studies in Video Indexing and Retrieval divided into two categories namely Compressed and Pixel domain. Analyzing video in compressed domain reduces the computational complexity by avoiding us from decompressing video into pixel domain. Fast shot and motion detection algorithms works in compressed domain.

Traditional Video Indexing segments each video into shots and then finds representative key frames for each shot. After that either Scene detection or Automatic/Semi-Automatic/Manual Feature extraction is applied on selected key frames. Finally, a high dimensional indexing technique is used to index and retrieve extracted information. Figure 2.1 shows a conceptual model for content based video indexing and retrieval (Zhong, Zhang & Chang, 1996).

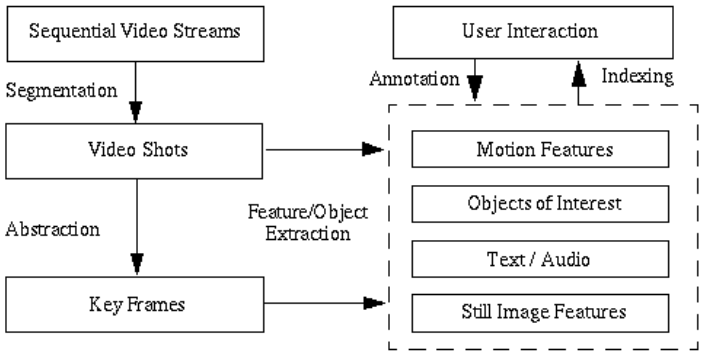

Figure 2.1 Conceptual model for content based video indexing and retrieval (Zhong et al., 1996)

Because of the complex structure of video domain, indexing problems usually reduced into frame domain for efficient use of existing image indexing techniques in literature. Image feature based indexing & retrieval is essential approach for video indexing and retrieval. State of art image indexing measurement utilities depends on a set of well-defined image features as seen in Table 2.1

Table 2.1 Common image features used in indexing

Color features
Color histograms, color correlogram
Texture features
Gabor wavelet features, Fractal features
Statistical features
Histograms, moments
Transform features in other domains
Fourier features, wavelet features, fractal features
Intensity profile features
Gaussian features

Wei-Ying & HongJiang (2000) and Gabbouj, Kiranyaz, Caglar, Cramariuc, & Cheikh et al. (2001) presented a multimedia browsing, indexing, and retrieval system that uses low-level features and supports hierarchical browsing system. Snoek & Worring (2005) presented a multimodal framework according to the perspective of the content author. Their framework considers different shot models such as camera shots, microphone shots and textual shots and answers the three questions “What to index? How to index? Which index?” According to their report, available semantic index hierarchy found in literature is as follows.

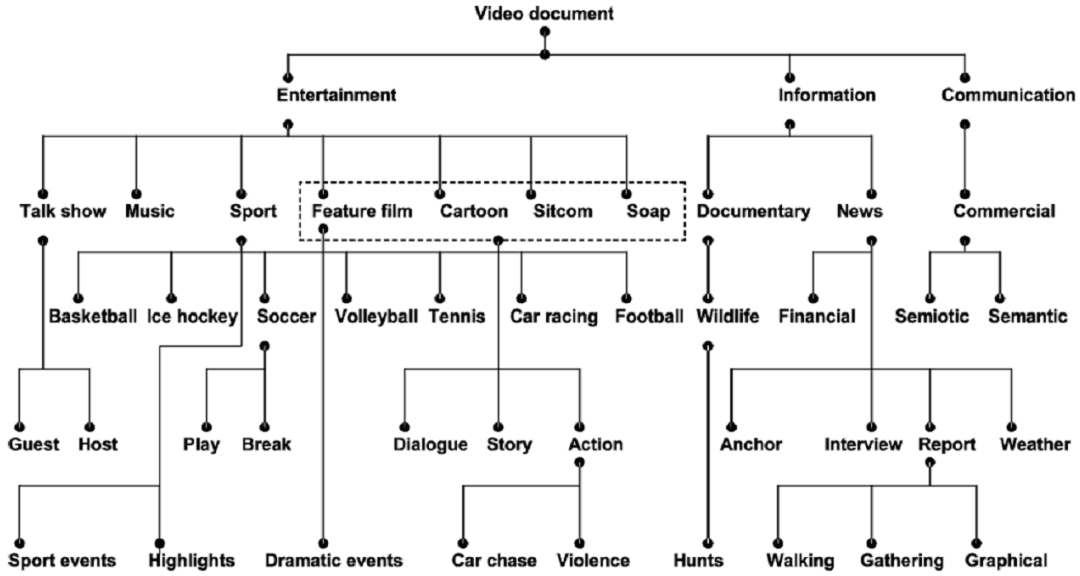

Figure 2.2 Semantic index hierarchy found in literature Snoek & Worring (2005)

When we look at the semantic index found in literature, it is easy to see that in most of the elements especially in Sport and News, the focus of subject is human as in Figure 2.2.

Furth & Saksobhavivat present a technique that can be used for fast similarity-based indexing and retrieval of both image and video databases in distributed environments. They assumed that image or video databases are stored in the compressed form such as JPEG or MPEG coded. Their technique uses selective distance metrics among weighted Euclidean distance, square distance and absolute distance of histograms of DC coefficients, so are computationally less expensive than other approaches. In case of video they partitioned of the video into clips, performed key frame extraction, indexing and retrieval. According to their experimental results, the proposed algorithm can be very efficient for similarity-based search of images and videos in distributed environments, such as Internet, Intranets, or local-area networks.

Pei & Chou (1999) used the patterns of macro block types for shot detection. Calic & Izquierdo (2002) present a technique to the multi-resolution analysis and scalability in video indexing and retrieval. Their technique is based on real-time

analysis of MPEG motion variables and scalable metrics simplification by discrete contour evolution. They use scalable color histogram for hierarchical key-frame retrieval. Table 2.2 shows their results.

Table 2.2 Shot change detection results

	Detect	Missed	False	Recall	Prec.
News	87	2	6	98%	94%
Soap	92	2	9	98%	91%
Comm	127	9	16	94%	88%

2.1.1.1 Dublin Core Metadata Initiative

The Dublin Core Metadata Initiative (DCMI) is a META data standard whose development began in 1995 at an OCLC meeting in Dublin, OH. Its objective is to develop a META data standard to enhance core set of META data elements or attributes to structure the description of networked resources. The DCMI assists the simple description of a networked resource, but is not accepted by all search engines.

DCMI Element Set Version 1.1 consists of 15 descriptive data elements relating to content, intellectual property and instantiation. These elements include title, creator, publisher, subject, description, source, language, relation, coverage, date, type, format, identifier, contributor, and rights. Each Dublin Core element defines a set of ten attributes from the ISO/IEC 11179 [ISO11179] standard for the description of data elements, (Dublin Core Metadata Element Set, 1999). Details of data element information are as follows;

Name: The label assigned to the data element

Identifier: The unique identifier assigned to the data element

Version: The version of the data element

Registration Authority: The entity authorized to register the data element

Language: The language in which the data element is specified

Definition: A statement that clearly represents the concept and essential nature of the data element

Obligation: Indicates if the data element is required to always or sometimes be present (contain a value).

Datatype: Indicates the type of data that can be represented in the value of the data element.

Maximum Occurrence: Indicates any limit to the repeatability of the data element.

Comment: A remark concerning the application of the data element.

These 15 DC META data elements grouped in three main classes. These are,

Content: Title, Subject, Description, Source, Language, Relation, Coverage

Intellectual Property: Creator, Publisher, Contributor, Rights

Instantiation: Date, Type, Format, Identifier

2.1.1.2 MPEG-7 Multimedia Content Description Interface

MPEG-7 formally named "Multimedia Content Description Interface" is an ISO/IEC standard being developed by Moving Picture Experts Group (MPEG) for describing the multimedia content data that supports some degree of interpretation of the information's meaning, which can be passed onto, or accessed by, a device or a computer code (Martínez, 2002). MPEG will not regulate or evaluate applications or is not aimed at any one application in particular; somewhat, the elements that MPEG-7 standardizes shall support as wide range of applications as possible.

MPEG-7 aims at offering a comprehensive set of audiovisual description tools and richest set of features to create descriptions, which will form the basis for

applications enabling the needed quality access to content, good storage solutions, accurate filtering, searching and retrieval.

2.1.1.3 Semantic Video Indexing and Retrieval

Semantic video indexing researches try to close the gap between the high-level semantic information and low-level features extracted by algorithms. Current trend is to use textual information such as subtitles and ASR generated text for high-level concept detection task as in TRECVID (TREC Video Retrieval Evaluation) which tries to promote progress in content-based retrieval from digital video via open, metrics-based evaluation.

Semantic gap can be decreased by selecting effective semantic concepts. Detecting particular concepts in video is an important step toward semantic understanding of visual imagery. Concepts itself are the semantic entities and has visually distinguishable parts. These concepts should be sufficient to be recognizable by humans such as events, camera effects, people, building, cars etc. Therefore, most of the semantic video indexing research concentrates on concept detection where the concept can be human, vehicle, animal, hands etc. For this reason each year TRECVID conferences tries to find important semantic concepts which gives important clues on current trend in semantic video retrieval.

Simplest way of building semantic video indexing system is to build hierarchical representations of multimedia data or semantic index. Some researchers use the term “concept detection” for determining the theme of the documents having both audiovisual and textual information. In spite of its easiness of implementation, it has the disadvantage that not all real life objects are hierarchical.

Semantic networks like WordNet, Miller (1995) or Semantic Relation Graphs (SRG) is a solution to the problem. A semantic network consists of a set of clusters that each cluster has its own set of words from a language. These networks usually have many to many relationships. It means that a word can be member of more than one cluster. Each cluster has one or more centroids or representative terms. These

networks provide synonym or alternative word representations to the original query. Another words, we can use these systems to find the semantic meaning behind the original query. This kind of extensions (query expansion) on initial query provides more powerful query representations on multimedia objects. In the simplest case, if the original query includes the term “football” then it is a big probability that the term “soccer” is in the result set.

When we look at state of art researches in semantic video indexing and retrieval, there is a reduction in the number of general scenes in datasets and increase in more detailed objects and events. Indexing and retrieval of video using semantic labels are one of the most challenging research areas in field of information retrieval. Low-level operations are not enough to fulfill high-level semantic query in users mind. To solve the problem, many researchers suggest studying video indexing and retrieval to more complicated form called Semantic Video Indexing and Retrieval. Many research projects try to close the gap between the high-level semantic space and low-level features space. To close the semantic gap usually a semantic index is created and classic hierarchical approach (video, scene, shot, and key frame) known as table of contents of a video is combined with the semantic index.

According to Long, Feng, Peng & Siu (2001) it is a difficult task to detect Semantic Video Objects (SVO) because there is no unique definition of a SVO exists and SVO detection infects a segmentation process, which is one of the most difficult problems in the computer vision, and image processing. Traditional homogeneity criteria do not lead us to semantically meaningful objects in real world.

Rasheed, Sheikh, & Shah (2003), presented a framework that uses unsupervised learning technique on film previews for the classification of films by using cinematic features into four broad categories namely comedies, action films, dramas, and horror films. According to their research, like the natural languages, films also have a “film grammar” generated by the director. If we find a way to find the *computable video features,* which are, infect any statistics on video data, then we reduce the semantic film classification problem to computing video feature problem. They used computable video features such as;

Average shot length: For measuring the tempo of scenes

Color feature: To find the genre of film. Bright colors mean comedies whereas darker color means horror videos.

Motion content: To find the genre of the film. High motion represents action films and low motion represents dramatic or romantic movies.

Light: If the direction of the light is known then it is used to find the key feature of the shot. Figure 2.3 shows high key and low-key shots used to differentiate genres.

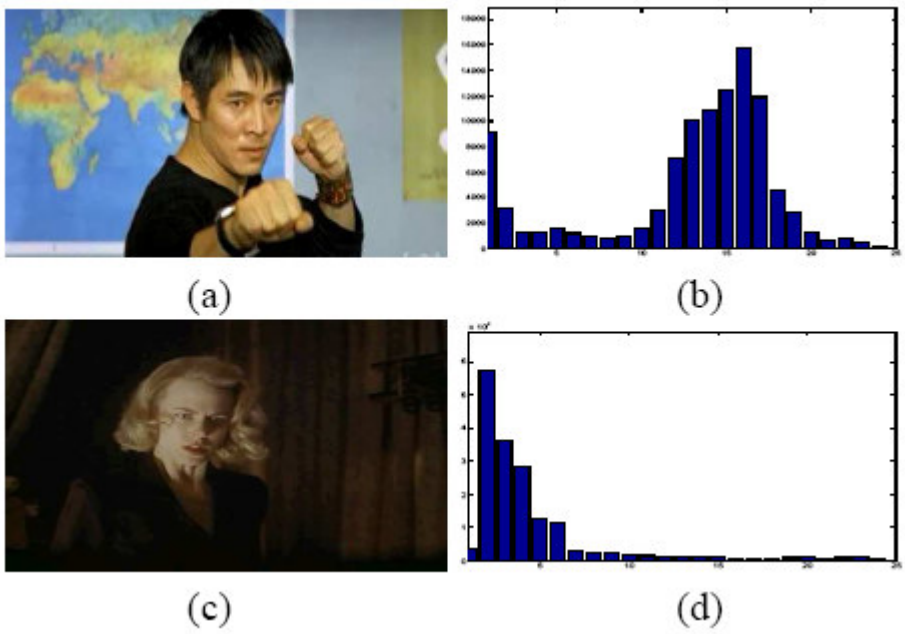

Figure 2.3 High-key, low-key shots on left and corresponding histograms on the right

Adams, Amir, Dorai, Ghosal, & Iyengar et al. (2002) developed a video retrieval system that explores fully automatic content analysis, shot boundary detection, multi-modal feature extraction, statistical modeling for semantic concept detection, speech recognition, and indexing. They have used SVM to map the generated feature vectors into high dimensional space through nonlinear function and HMM for concept detection task. Their lexicon design has three types of concepts. These are (person,

building, bridge, car, animal, flower), scenes (beach, mountain, desert, forest), and events (explosion, picnic, wedding). They also implemented Spoken Document Retrieval (SDR) system that allows the user to retrieve video shots based on the speech transcript associated with the shots.

Naphade, Krisljansson, Frey, & Huang (1998) and Naphade, Kozintsev, & Huang (2002) proposed a domain independent novel approach for bridging the semantic gap using probabilistic framework. They generate multijects from low-level features by using multiple modalities. A multiject is a probabilistic object that has a semantic label and summarizes a temporal duration of low-level features of multiple modalities in the form of probability. Their fundamental concepts are *sites*, *objects,* and *events*. Set of multijects builds the multinet (a unidirected multiject network having + and - signs) and can handle queries at semantic level. Figure 2.4 show a sample view of a multinet.

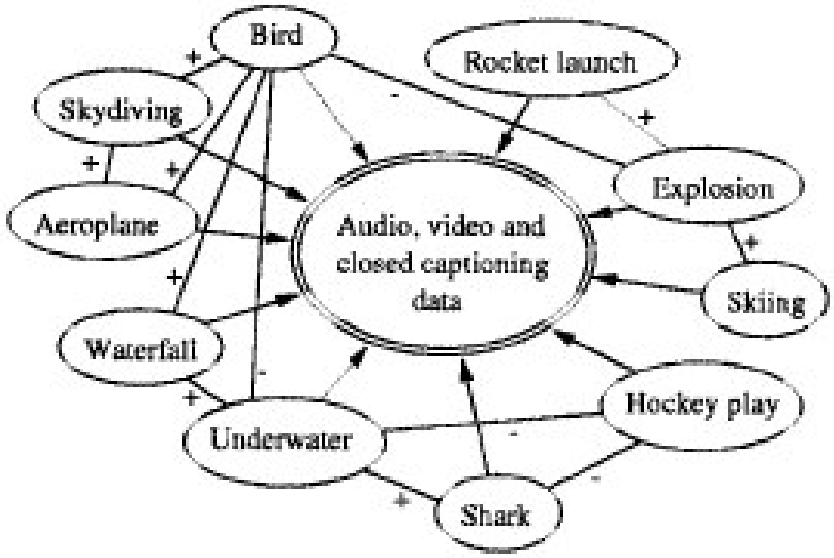

Figure 2.4 Conceptual figure of a multinet Naphade et al. (2002)

According to Naphade & Smith (2004) research, the following Table 2.3 shows the available Concept detection algorithms in literature.

Table 2.3 Concept detection algorithms, Naphade & Smith (2004)

Active Learning
Appearance Templates
Boosting: Adaboost
Context Models
Decision Trees (C4.5)
Face ID
Fisher LDA
Gaussian Mixtures
Hidden Markov Models
K Nearest Neighbor
Keyframe Based Modeling
Latent Semantic Analysis
Maximum Entropy Model
Media Synchronization
Metadata-based Models
Multi-Frame Based Modeling
Motion Templates
Neural Networks
Rule-Based Detection and Filtering
Shape Templates
Support Vector Machines
Unsupervised Clustering
Video OCR
Weighted Averaging

Snoek & Worring (2005), developed a generic approach for semantic concept classification using the semantic value chain, which extracts lexicon of 32 semantic concepts from video documents based on content, style, and context links. Figure 2.5 show semantic value chain used by (Snoek & Worring, 2005). According to their research TRECVID 2004 dataset would take about 250 days on the fastest sequential machine available, therefore they used a Beowulf cluster having 200 dual 1Ghz CPU's and reduced the time to 48 hours.

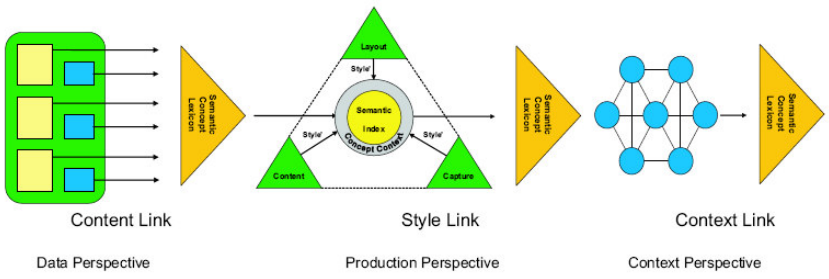

Figure 2.5 Semantic value chain (Snoek & Worring, 2005)

Colombo, Bimbo & Pala (1999) divides the video into four semiotic categories namely, practical, playful, utopic and critical and proposes a set of rules defining the semiotic class of video by looking at the low level features such as , color, shape, video effects etc.

According to Jaimes & Smith (2003), semantic ontology construction process uses either data driven or concept driven approach. They have build a system that allows the videos in multiple ways including textual search on metadata, ASR text, syntactic features (color, texture, etc.) and semantic concepts such as, face, indoor, sky, music etc.

Salway & Graham (2003) presented a method for character's emotions in films. They suggested that it could help to describe higher level of semantics. They have extracted audio information from the video sequence and then find the semantics. They have created a list of emotion tokens by using the WordNet.

Table 2.4 Emotion tokens, Naphade et al. (1998)

Emotion Type	Total tokens	Example emotion tokens
JOY	47	euphoria, elation, happy, jolly, pleased
DISTRESS	50	distraught, anguished, miserable, depressed
LIKE	31	love, passionately, adoration, fondness
DISLIKE	33	hatred, loathing, disgust, aversion, distaste
HOPE	31	anticipation, excited, expectant, optimistic,
FEAR	115	terrified, panicked, worried, concerned

Then they tested the emotion tokens on the film Captain Corelli's Mandolin and draw the plot of emotion tokens found in this film as seen in Figure 2.6

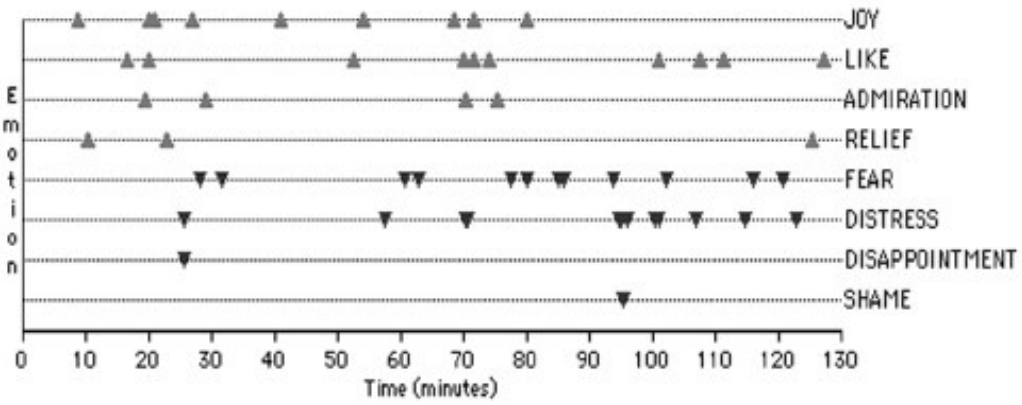

Figure 2.6 Plot of emotion tokens from Captain Corelli's Mandolin

A common way for retrieving subject of interest is to use query by example paradigm Naphade, Yeung, & Yeo (2000). Small video clips can be submitted for retrieval of similar results but in this case the person should have a similar video clip and it is not possible for a person to have a video clip that has similar high-level semantic properties that what he has in his mind in real world. Therefore, we need to find a better way to understand the high-level semantic concepts in human's brain.

Rautiainen, Seppänen, Penttilä, & Peltola (2003) used temporal gradient correlograms to capture temporal correlations from sampled shot frames. They have tested algorithms on TRECVID 2002 video test set and detected shots containing people, cityscape, and landscape.

Visser, Sebe, & Lew (2002) used Kalman filter to track the detected objects and sequential probability ratio test to classify the moving objects in streaming video.

Garg, Sharma, Chaudhury, & Chowdhury (2002) suggested a new model for organizing video objects in appearance based hierarchy. They have used SVD based Eigen-space merging algorithm.

Guo, JongWon, & Kuo (2000) developed SIVOG system that adaptively selects processing regions based on the object shape. They used temporal skipping and interpolation procedures to slow motion objects and the system is able to extract

simple semantic objects with pixel wise accuracy. Figure 2.7 show the result of extraction the human object with background removed.

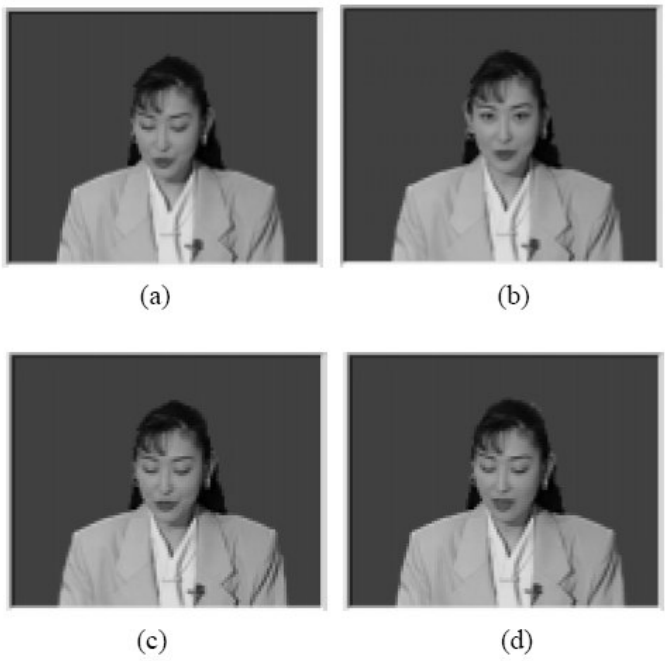

Figure 2.7 Example sequence (frames: 50,150,200,300) from Guo et al. (2000)

Izquierdo, Casas, Leonardi, Migliorati, & O'Connor et al. (2003) summarized the common features of semantic objects such that;

- Objects of interests tend to be homogenous.
- Objects composed of different parts should be spatially linked.
- Shape complexity (squared contour length divided by the object area) of objects are usually low.
- Objects usually satisfy the symmetry property.

Figure 2.8 shows the structure analysis using these features.

Figure 2.8 Structure analysis using the common features of objects, Izquierdo et al. (2003)

Wang, Ma, Zhang, & Yang (2000) considered the human as a whole and developed a new multimodal approach to people-based video indexing. They defined people similarity according to both clothing similarity and speaking voice similarity by using Support Vector Machines. Figure 2.9 shows the tree-based structure proposed by Wang et al. (2000).

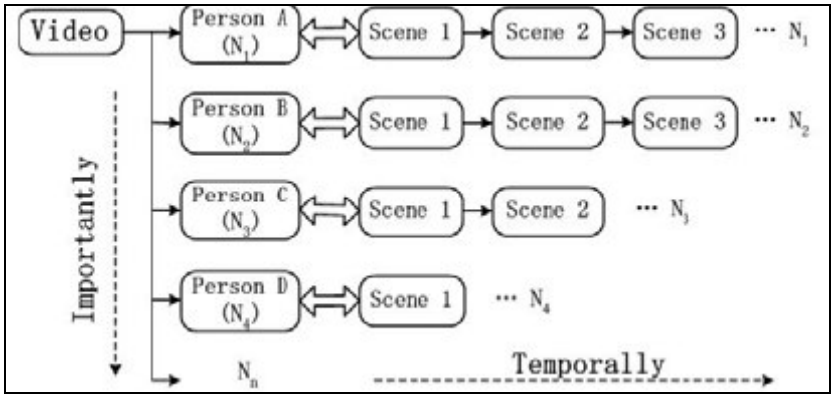

Figure 2.9 Human based video indexing

Tran, Hua & Vu (2000) studied a video data model called SemVideo, which tries to achieve the problem of limiting the semantic meaning with the temporal dimension of video. According to their researches classical segmentation based approaches has incapability of representing semantics because of the overlapped segments.

Babaguchi & Nitta (2003) proposed a strategy for semantic content analysis by using multimodalities (audio, visual, textual content) for detection of semantic events in sports videos.

Arslan, Donderler, Saykol, Ulusoy, & Gudukbay (2002) developed a semi-automatic semantic video annotation tool that considers activities, actions, and objects of interest for semantic indexing. They have designed a semantic video model for storing semantic data as seen in Figure 2.10.

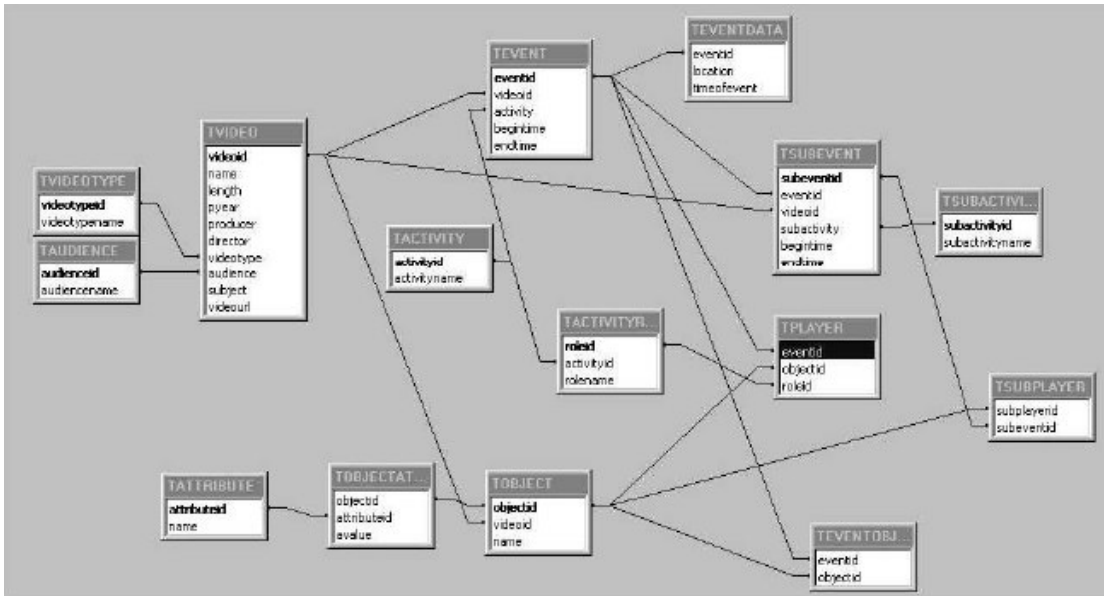

Figure 2.10 Database design of the semantic video model, (Arslan et al., 2002)

2.1.2 Shot Boundary Detection (SBD)

Video segmentation is dividing the video into sequential frames that is either has a spatial or temporal relation. One or more sequential frames build up a shot. There are a number of different segmentation techniques in literature for shot detection and most of them use a threshold value for detecting shot regions.

There are wide range of shot types exist such as, fades, dissolves, wipes, and editing effects. The most common shot type is cuts or breaks. A cut occurred whenever a transition from one shot to another occurs between the two sequential frames. The importance of shots in video indexing and retrieval similar like importance of words in textual indexing and retrieval methods, thus shot locations and types gives important clues about the video itself. There are many methods in literature to find shot regions (Albanese, Chianese, Moscato, & Sansone, 2004),

(Gargi, Kasturi, & Strayer, 2000), (Lienhart, 1999, 2001a), (Truong, Dorai, & Venkatesh, 2000), (Zhang, Kankanhalli, & Smoliar, 1993).

Shot Boundary Determination (SBD) is a process to identify the boundaries of shots from a sequence of video frames, where a shot is the smallest meaningful unit of video. In video processing, SBD appears at the very early phase of the video processing. In order to detect shot boundaries within a video it needs to find for some changes across the boundary. Most of the previous works focused on cut detection. The more recent works have focused on detecting gradual transitions. According to Boreczky & Rowe (1996) there are a number of different types of transitions or boundaries between shots.

Cut: A cut is an abrupt shot change that occurs in a single frame.

Fade: A fade is a slow change in brightness usually resulting in or starting with a solid black frame.

Dissolve: A dissolve occurs when the images of the first shot get dimmer and the images of the second shot get brighter, with frames within the transition showing one image superimposed on the other.

Wipe: A wipe occurs when pixels from the second shot replace those of the first shot in a regular pattern such as in a line from the left edge of the frames.

Cut detection process also tries to find camera operations such as, dollying (back/forward), zooming (in/out), tracking (left/right), panning (left/right), tilting (up/down), and booming (up/down).

2.1.2.1 Pair wise Pixel Differences

Pair wise pixel difference is the obvious metric to consider first. Considering two sequential frames; Let $P_i(k,l)$ represents (k,l) pixels of i^{th} frame and DP_i .indicates that in i^{th} frame, whether pixels are changed or not.

$$DP_i = \begin{cases} 1 & \text{if } \left|P_i(k,l) - P_{i+1}(k,l)\right| > t \\ 0 & \text{otherwise} \end{cases}$$

where *t* is the threshold value.

For M × N size frames, a shot boundary exists if:

$\sum_{k,l=1}^{M,N} \frac{DP_i(k,l)}{M * N} > T_b$ However, this is very sensitive to both camera and object motion.

2.1.2.2 Histogram Comparison

Histograms are one of the most commonly used methods to detect shot boundaries within video data. In this method, histograms of colored or gray scale pixels in each frame are used to detect shot boundaries. It assumes that the background information does not change either so frequently or strongly among the boundaries of a shot region. Another say, it assumes that the number of pixel belongs to background is dominating. If the bin wise difference between the two histograms exceeds the threshold value then this state results finding of shot boundaries.

If there are *n* frames each of size M×N and let $H_i(j)$ be the histogram value of j^{th} bin of the i^{th} frame. Then the difference between the i^{th} and $(i+1)^{th}$ frame can be defined as;

$SD_i = \sum_{j=1}^{m} \left|H_i(j) - H_{i+1}(j)\right|$ If the SD_i value is greater than the threshold T_b then a shot boundary is detected.

2.1.2.3 Three Frames Approach

Sethi, & Patel (1995) considered three consecutive frames. These are formally called *r*, *s,* and *t*. D_{rs} and D_{st} are the measure of the frame dissimilarities. According to these values Observer Motion Coherence OMC, defined by;

$$OMC(r,s,t) = \left| \frac{D_{rs} - D_{st}}{D_{rs} + D_{st}} \right|$$

If the $OMC(r,s,t)$ value is, a number that close to 1, it means that there is no change between the consecutive frames *r*, *s*, and *t*. A shot is detected when the value of the $OMC(r,s,t)$ close to number zero.

2.1.2.4 Twin-Comparison Method

This approach is widely used for detecting edit effects and gradual transitions within video sequence. The basic idea is to mark frames before and after the gradual transitions.

The important problem of this method is that basic camera operations including pan and zoom can be misinterpreted as special effects. Threshold based solutions cannot be used because pan and zoom operations produce the same change effects as special effects. In addition, if a threshold based method is used then the value of the threshold must be lower value compared with standard threshold value of shot detection.

Motion feature is a solution that can be used to detect this kind of camera operations. During the pan and zoom operation of the camera, motion vector fields have the same direction with almost a fixed angle value. The algorithm works as follows;

Let SD_i represents Standard Deviation of frame *i* and T_b is the threshold value.

Compute SD_i for all frames in the video.

Find camera breaks where $SD_i > T_b$

Mark potential gradual transition subsequences defined by *GT* of the video wherever $SD_i > T_b$ for $F_s \le i \le F_e$ where F_s is start frame and F_e is end frame of gradual transition.

For each gradual transition, frame-to-frame difference (1) is as follows:

$$AC = \sum_{i=F_s}^{F} SD_i \qquad \textbf{(1)}$$

If $AC > T_b$, then declare $[F_s, F_e]$ as a gradual transition effect

2.1.2.5 Compression Differences

Little, Ahanger, Folz, Gibbon, & Reeve et al. (1993) used differences in the size of JPEG compressed frames with same compression rate to detect shot boundaries as a supplement to a manual indexing system.

Arman, Hsu & Chiu (1994) used differences in the discrete cosine transform (DCT) coefficients of JPEG compressed frames as a measure of frame similarity, as a result of this challenging situation, they have decreased computation time by not making compression on frames. They have also considered the differences between color histogram values in order to detect potential shot boundaries.

2.1.2.6 Average Pixel Method

The first step for this method is shot boundary detection. If we assume that, each frame has size of M×N and shot region *k* starts with frame F_b, ends at frame F_e and has S_k number of frames in shot *k* then average pixel values of shut *k* can be defined as average value of same pixels of each frame. As a result, the averages frame F_{avg} usually a blur image and the object motion can affect it. Figure 2.11 explain the algorithm.

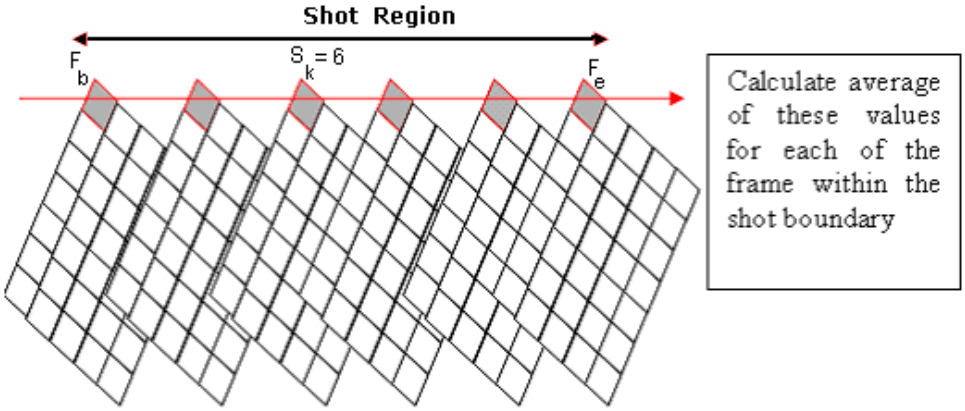

Figure 2.11 Average pixel method

In another words;

$$\mu(i,j) = \frac{1}{S_k}\sum_{1}^{Sk} pixel(i,j) \qquad \textbf{(2)}$$

By using the formula (2), an average frame of shot *k* is calculated for each M×N pixel. After calculating the average frame F_{avg}, compute the distance of every frame within the shot *k* to the average frame F_{avg_k}. Let FK_k (3) will be the key frame and F_i is a frame within the shot region shot *k* then,

$$FK_k = \left\{F_i \quad \exists i \left|(F_i - F_{avg_k})\right| \le \forall j \left|(F_j - F_{avg_k})\right| \text{where} \quad 0 \le i, j \le S_k \right. \qquad \textbf{(3)}$$

2.1.2.7 Cross Fade Detection

Detection of gradual transition effects are one of the most important problems in video indexing because a shot is the elementary unit of a video and it is the first step to find boundaries of shots in video segmentation researches. On the other hand, cross fade detection is not easy to detect effect because it contains both spatially and temporally different frames. Therefore, we need to deal with both of the problems instead of one.

Automatic detection of the cuts and transition effects may increase the probability of extracting semantic information from the video. For example, according to the research on videos by Fischer, Lienhart & Effelsberg (1995), feature films and documentary films include dissolve effect more often than sports or comedy shows. We can categorize the types of dissolves in two distinct groups. Two frames that produce the dissolve effect can have;

Different color layout information. In this case, it is easy to detect the dissolve region because there is almost no correlation between the successive frames. Fade-in by appearing from a solid color frame and fade-out by disappear to make solid color frame effect is examples for this type of gradual transition.

Having similar color layout but different spatial layout. Histogram based methods fails but edge detection is the solution to the problem.

There are some other types of such as morphing from one object to another is the special case of dissolve effect but they are so rare that most of the researchers concentrate on the two categories.

Table 2.5 shows the classification of the transitions according to the spatial and temporal properties of the transition frames.

Table 2.5 Transition classifications (Lienhart, 2001a)

Type of transition	The two involved sequences are	
	Spatially separated	Temporally separated
Hard cut	Yes	Yes
Fade	Yes	Yes
Wipe	Yes	No
Dissolve	No	No

A formal definition of a cross-fade is the combination of a fade out and fades in, superimposed on the same filmstrip, Arijon (1976). This simultaneous fade-in of one video frame source or lighting effect while another fades out and may overlap

temporarily. It is known as dissolve or cross-dissolve effect that provides smooth transition. Figure 2.12 shows two dissolves occurring simultaneously.

Figure 2.12 Cross-fade effect

As seen in Figure 2.12, there is a small change in the background area of the shot boundaries. Usually start and end frames of the dissolve region do not changes and freezes during the dissolve effect. Duration of a dissolve effect is between 10-60 frames.

Use of the similarity between the consecutive frames is the dominant technique for dissolve detection. Techniques that use threshold value to measure the boundary of shot region is not suitable for gradual transitions (dissolve) because of the small spatial and temporal change in frames (Lienhart, 2001b).

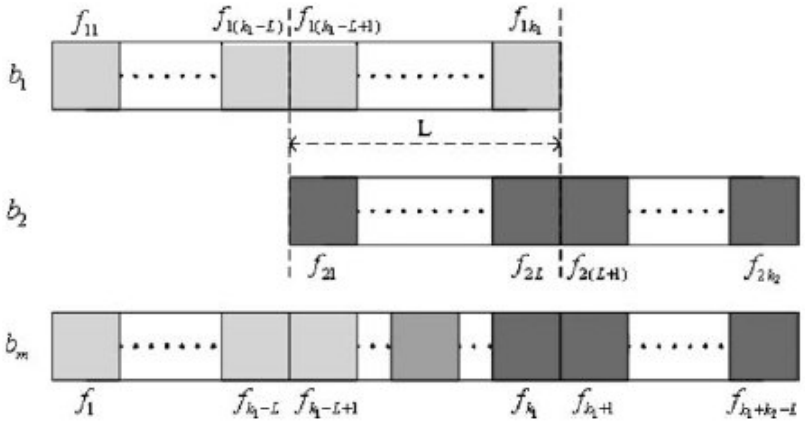

Figure 2.13 Cross Fade in temporal dimension (Lienhart, 2001b)

According to Figure 2.13, formal definition of cross-fade is shown in (4);

$$f_i(p)=\begin{cases} f_{1i}(p) & i\in[1,k_1-L] \\ \dfrac{(k_1-i)}{L-1}f_{1i}(p)+\left(1-\dfrac{(k_1-i)}{L-1}\right)f_{2(i-k_1+L)}(p) & i\in(k_1-L,k_1] \\ f_{2(i-k_1+L)}(p) & i\in(k_1,k_1+k_2-L] \end{cases} \quad \textbf{(4)}$$

Where $f_i(p)$ represents the i^{th} frame in cross-fade region, L is the length of region (# of frames), and k_1 is the start frame of the cross-fade effect. For frames between $(k_1-L,k_1]$, the effects of the first frame decrease while the latter increases. Therefore this effect combines both fade-in and fade-out transitions.

2.1.2.8 Key Frame Selection Strategy

Key frames have an important role in video indexing and retrieval. Shots are basic units of video but there is a necessity of having a handle that represents content of video. It is a simple method. If it is used with boundary detection methods then key frame selection can operate in better way. In that case, first frame of the current shot can be selected as a key frame. But in some circumstances the first frame of the shot can be meaningless and cannot cover the content of the shot region therefore some other techniques can be used such as in a simple way, selecting the middle frame

within a shot region as a key frame. These methods can be improved to select the best key frame for shot region.

Number of key frames can be adaptive within a shot. In this case, mean and standard deviation of frame sizes will be computed and the frames which size is greater than the mean frame size plus standard deviation then it will be selected as key frame for the shot.

A good key frame selection strategy provides reduction in temporal dimension thus increase performance. Traditional key frame selection methodologies select single or multiple key frames per shot such that;

- First, last or middle frame of the shot sequence.
- Average frames in the shot sequence.
- I-Frames in shot region.
- Frames having a desired object

2.1.3 Facial Expression Recognition

Duchenne du Boulogne first expresses facial expressions in 1862. He was a pioneering neurophysiologist and photographer. Most researchers acknowledge their debt to Duchenne and his book "The Mechanisms of Human Facial Expression".

Ekman & Friesen (1978) presented the most important comprehensive study in the content of facial expression recognition, called Facial Action Coding System (FACS). They have defined a method for describing and measuring facial behaviors and facial movements based on anatomical analysis of facial action.

Measurement unit of the FACS system is Action Units (AUs). They have defined a set of 44 Action Units (AUs) in original work that having a unique numeric code, which represents all possible distinguishable facial movements because of change in

muscular actions. 30 of them are related to a specific contraction of muscles and 14 of them are unspecified.

Most of the researchers use six basic “universal facial expressions” corresponding to happiness, surprise, sadness, fear, anger, and last disgust. Figure 2.14 shows sample set from the (Cohen, 2000, pp. 8-30).

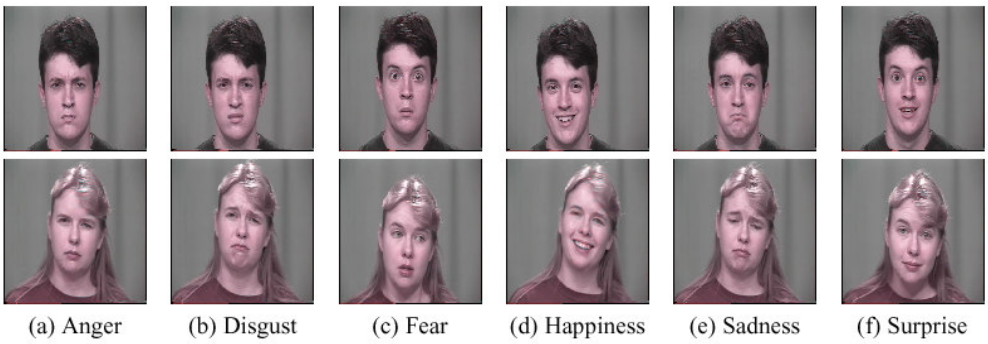

Figure 2.14 Sample six basic facial expression data set from (Cohen, 2000, pp. 8-30)

Ekman studied on video tapes in order to find changes in human face when there is an emotion exists. According to the work, a smile exists if the corners of the mouth lift up through movement of a muscle called zygomaticus major, and the eyes crinkle, causing "crow's feet," through contraction of the orbicularis oculi muscle.

Changes in location and shape of the facial features are observed. Score of a facial expression consists of a set of Action Units. Duration and intensity of the facial expression are also used. Observed raw FACS scores should be analyzed in order to produce behavior that is more meaningful. FACS has four main steps;

Observe movements and then match the AUs with the observed movements.

An intensity score is given for each one of the actions

Determine the action unit’s type as asymmetry or joint

Determining the face and facial feature positions during the movement of the face in the sequence.

Interpreting AU is a difficult task. For example, there are six main emotional states exists but each of them has many variations. Figure 2.15 shows two different type of smile of the same person. Therefore, usually each emotional state is represented by a set of action units. Thus most of the action units are additive.

Figure 2.15 Two different types of smile. (Lien, 1998)

One of the limitations of the FACS system is nonexistence of a time element for the action units. Electro-Myo-Graphy (EMG) studies, which are based on the measurement of electrical activity of muscles, showed that facial expressions occur in a time-aligned sequence beginning with application, continuing with release and finally relaxation.

Face tracking is needed to compute the movements of each facial feature. Terzopoulos, & Water (1993) used making up facial features and Cohen, (2000) used different face tracking algorithms including 3D-based models.

Chen modified the Piecewise Bezier Volume Deformation (PBVD) tracker of Tao & Huang to extract facial feature information (Chen, 2000), (Tao, & Huang, 1998). In this work, the first frame of the image sequence is selected and processed to find facial features like eye and mouth corners. Then generic face model is warped to fit the selected features. Their face model consists of 16 surface patches embedded in

Bezier volumes. In this way, the surface is guaranteed to be continuous and smooth. The shape of the mesh changes by changing control points in the Bezier volume.

Terzopoulos & Water developed a model that tracked facial features in order to observe required parameters for a three-dimensional wire-frame face model (Terzopoulos, & Water, 1993, pp. 569-579). However, their work has a limitation in which the humans facial features should be marked up to robustly track these facial features.

The most difficult task in facial expression recognition is tracking and extracting facial features from a set of image sequence. A huge number of parameters and features should be considered (Cohen, 2000, pp. 8-30). Therefore, it is necessary to decrease the number of points that are required to track facial features thus decreasing computational time. Principal Feature Analysis (PFA) makes this task and finds the most important feature points that need tracking. (Cohen, 2000, pp. 8-30) used PFA method to find the best facial feature points. Cohen initially marked up the face to be tracked to get robust results and then tracked a video of 60 seconds at 30fps. Figure 2.16 shows example images from the video sequences.

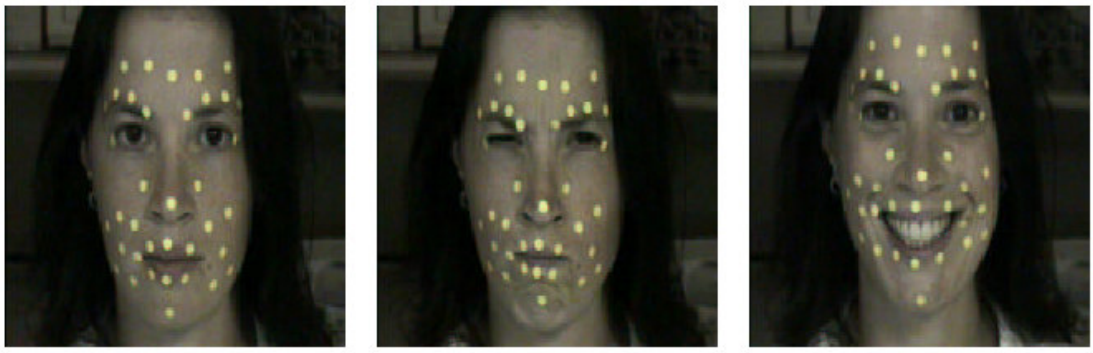

Figure 2.16 Example images from the video sequences. (Cohen, 2000, pp. 8-30)

Cohen has used 40 facial points each having two directions, horizontal and vertical to be tracked. For the PFA, these points divided into two groups, namely

upper face (eyes and above) and lower face. Then the correlation matrix computed. After applying the principle feature analysis, the resulting image showed in Figure 2.17.

In Figure 2.17, selected feature points are marked by arrows. According to Cohen's work, PFA is able to model complex face motions and reduces the complexity of existing algorithms.

In model based recognition systems, a feature vector should be defined for each expression and a similarity metric should be used to compute the difference between these expressions.

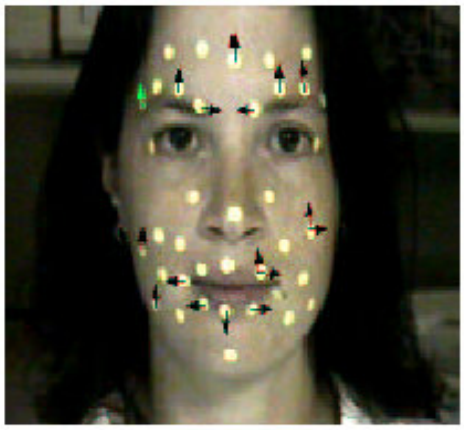

Figure 2.17 Result of PFA method. Arrows shows the principal features chosen

Pantic, & Rothkrantz (2000) developed an Integrated System for Facial Expression Recognition (ISFER) which is an expert system for emotional classification of human facial expressions from still full-face images. The system has two main parts. The first part is ISFER Workbench, used for feature detection and the latter is an inference engine called HERCULES.

First part of the system, ISFER Workbench presents a system for hybrid facial feature detection. In this part, multiple feature detection techniques are applied in

parallel. Therefore, it gives a chance to use redundant parts with eliminating uncertain or missing data. It has several modules, each doing different types of pre-processing, detection, and extraction. They have used both frontal view and side view of human faces. Figure 2.18 shows the frontal-view template from their work. Figure 2.19 shows algorithmic representation of ISFER Workbench. ISFER is complete automated system that is able to extract facial features from digitized still images. It does not deal with image sequences. Automatic encoding of facial Action Units (Ekman & Friesen, 1978) and automatically classifies six basic universal emotional expressions, happiness, anger, surprise, fear, sadness, and disgust.

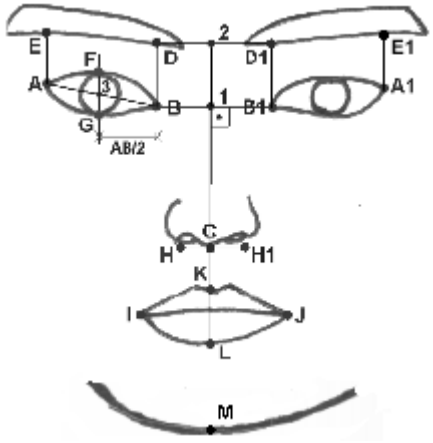

Figure 2.18 Facial points of the frontal-view (Pantic, & Rothkrantz, 2000, pp.881-905)

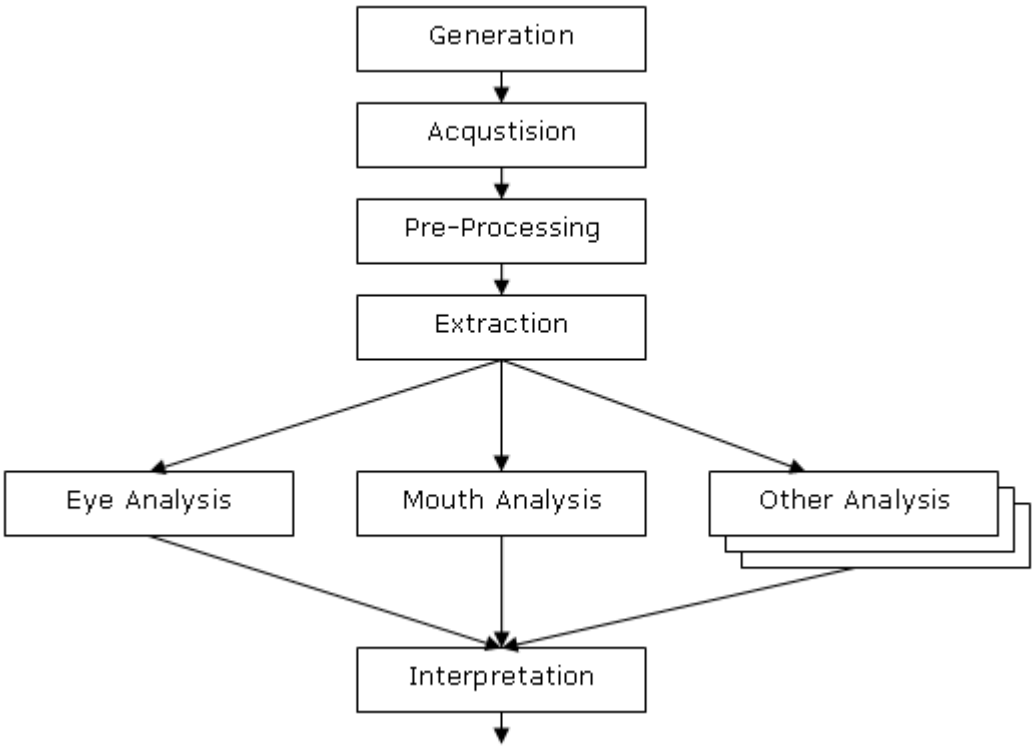

Figure 2.19 Algorithmic representation of ISFER Workbench

The second part of the system, HERCULES, converts low-level face geometry in high-level facial actions. Details of these points are described in Table 2.6.

Table 2.6 Details of facial points in Figure 2.18

Point	Description	Point	Description
B	Left eye inner corner, stable point	F	Top of the left eye, non-stable
B1	Right eye inner corner, stable point	F1	Top of the right eye, non-stable
A	Left eye outer corner ,stable point	G	Bottom of the left eye, non-stable
A1	Right eye outer corner, stable point	G1	Bottom of the right eye, non-stable
H	Left nostril centre, non-stable	K	Top of the upper lip, non-stable
H1	Right nostril centre, non-stable	L	Bottom of the lower lip, non-stable
D	Left eyebrow inner corner, non-stable	I	Left corner of the mouth, non-stable
D1	Right eyebrow inner corner, non-stable	J	Right corner of the mouth, non-
E	Left eyebrow outer corner, non-stable	M	Tip of the chin, non-stable
E1	Right eyebrow outer corner, non-stable		

Dailey, Cottrell, Padgett, & Adolphs (2002) showed that a simple biologically neural network model, trained to classify facial expressions matches a variety of

psychological data into six universal basic emotions. They have considered categorization, similarity, reaction times, discrimination, and recognition difficulty, in both qualitatively and quantitatively. Figure 2.20 shows morphing from happiness to disgust. They have used Morphs software version 2.5.

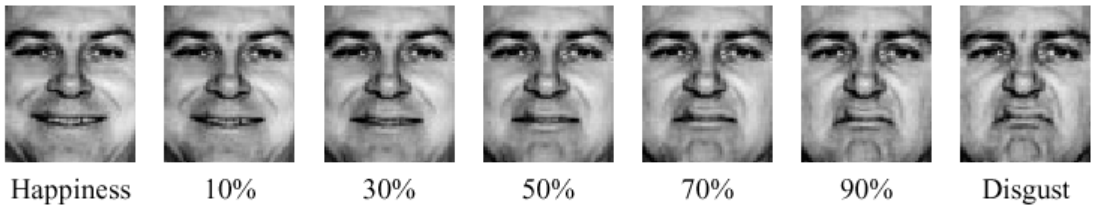

Figure 2.20 Morphs from happiness to disgust (Dailey et al., 2002).

Franco & Treves (1997) inserted a local unsupervised processing stage within a neural network to recognize facial expressions (Franco, & Treves, 1997). They worked with Yale Faces database and their neural net architecture has four layers of neurons. They have success at rate of 84.5% on unseen faces and 83.2% when principal component analysis processing applied at the initial stage.

Another method is use of Hidden Markov Models that solves classification problems especially for speech recognition systems because of its ability to model or classify non-static events. However, compared with other models, time required to solve the problem is significantly higher. Figure 2.21 shows a maximum likelihood classifier for emotion specific HMM.

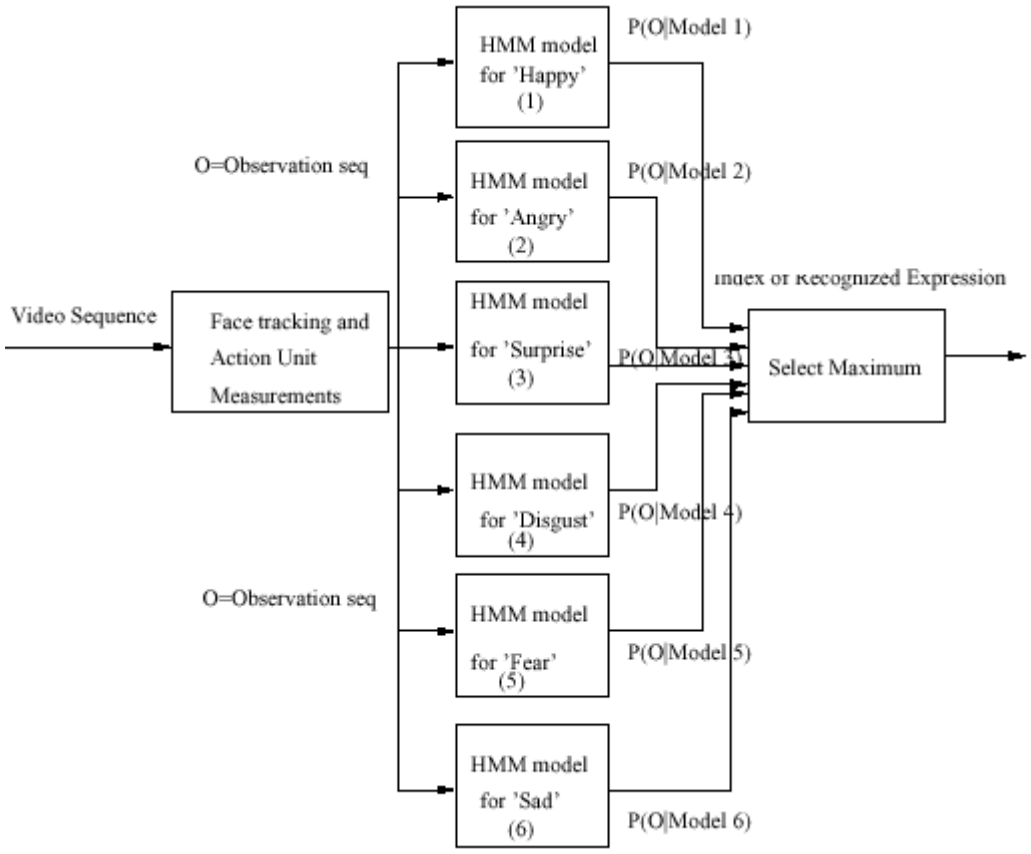

Figure 2.21 Maximum likelihood classifier for emotion specific HMM case (Cohen, 2000, pp. 8-30)

According to Figure 2.21, after making face tracking and Action Unit measurements, each one of the six Hidden Markov Model representing six universal emotions produces a result showing the probability of belonging to a specific type of emotion. At the end of the system, the emotion having the maximum probability is chose as observed emotion.

2.2 Speech Modality

2.2.1 Voice Activity Detection

Speech vs. nonspeech segmentation of audio signals widely used in automatic speech recognition, discrete speech recognition and speaker recognition areas to improve robustness of these systems. Aim of the Voice Activity Detection task is to find the presence or absence of human speech in a given audio signal. Elimination of

nonspeech segments within spoken content reduces the computational complexity while improving classification performance. A good speech-vs.-nonspeech segmentation method should be successful on unseen data and real world sounds where background noise exists. These methods intended to solve speech classification problem that requires high dimensional feature vectors which is in fact a fusion of a number of different feature sets.

Like other modalities, auditory modality needs the segmentation process. Audio shots or microphone shots are uninterrupted sound recording blocks, which provide boundaries of the speech signal. First audio signal must be cleaned to reduce the noise effect and then it must be segmented into speech, environmental and musical sounds. Continuity on these signals gives more semantic clues about the emotional content. For example, statistical analysis of loudness, brightness, harmonicity, timbre, and rhythm values can give clues about laughter, crowds, water sound, explosions, thunder etc.

According to the research of Murray & Arnott (1993), Table 1 shows the acoustic characteristics of the emotions.

Table 2.7 Acoustic characteristics of emotions

	Anger	Happiness	Sadness	Fear	Disgust
Speech rate	Slightly faster	Faster or slower	Slightly slower	Much faster	Very much slower
Pitch average	Very much higher	Much higher	Slightly slower	Very much higher	Very much lower
Pitch range	Much wider	Much wider	Slightly narrower	Much wider	Slightly wider
Intensity	Higher	Higher	Lower	Normal	Lower
Voice quality	Breathy, chest tone	Breathy, blaring	Resonant	Irregular voicing	Grumbled chest tone
Pitch changes	Abrupt, on stressed syllables	Smooth, upward inflections	Downward inflections	Normal	Wide downward terminal inflections
Articulation	Tense	Normal	Slurring	Precise	Normal

Previous works on VAD uses Mel Frequency Cepstral Coefficients (MFCC), pitch frequencies as formants, speech rate, and Teager Energy Operator (TEO) for features extraction purposes. Classification techniques used in emotion classification

task usually includes Multi-Class Support Vector Machines (MC-SVM), Artificial Neural Networks (ANN), Hidden Markov Models (HMM), Linear Discriminant Analysis (LDA) and K-Nearest Neighbor (K-NN) classifiers.

Vandecatseye & Martens (2003), used GMM and HMM on Hub4 News dataset and their speech detection accuracy is 99.5% while non-speech detection is 76.44%. Shafran & Rose (2003), used Bagging MLP method on SPINE corpus and get 95,8% accuracy. Casagrande, Eck, & Kigl (2005) used AdaBoost method with Haar-like features and smoothing technique on Scheirer-Slaney dataset and get 93% accuracy. Meinedo & Neto (2005) used ANN-MLP on Cost278-BN (Vandecatseye, Martens, & Neto, 2004) dataset and get 97.5% accuracy for speech, 70.6% for non-speech classification. An interesting point in VAD studies is that, classification methods requires large-scale datasets for training purposes (Byrne, Beyerlein, Huerta, Khudanpur, & Marthi et al., 2000), (Foo & Yap, 1997). Difficulties in finding large-scale datasets caused researchers to use limited datasets.

2.2.2 Emotional Speech Classification

Previous works on this area use Mel Frequency Cepstral Coefficients (MFCC) (Shami &Verhelst, 2007), (Altun & Polat, 2007), (Le, Quenot, & Castelli, 2004), pitch frequencies as formants (Zervas, Mporas, Fakotakis, & Kokkinakis, 2006), (Ververidis, Kotropoulos, & Pitas, 2004), (Hammal, Bozkurt, Couvreur, Unay, Caplier, 2005), (Datcu & Rothkrantz, 2005), (Shami & Verhelst, 2007), (Teodorescu & Feraru, 2007), (Lugger & Yang, 2006,2007), (Sedaaghi, Kotropoulos, & Ververidis, 2007), (Altun and Polat, 2007), (Zhongzhe, Dellandrea, Dou, & Chen, 2006), (Sedaaghi et al., 2007), (Pasechke & Sendlmeier, 2000) speech rate (Hammal et al., 2005), zero crossing rate (Lugger & Yang, 2007), Fujisaki parameters (Fujisaki & Hirose, 1984), (Zervas et al., 2006), energy (Zhongzhe, Dellandrea, Dou, & Chen,, 2006), (Ververidis, Kotropoulos, & Pitas, 2004), (Hammal et al., 2005), (Altun & Polat, 2007), (Sedaaghi et al., 2007), (Lugger & Yang, 2007), linear predictive coding (LPC) (Altun & Polat, 2007), (Le et al., 2004) for feature extraction purposes.

(Zhongzhe, Dellandrea, Dou, & Chen,, 2006), (Ververidis, Kotropoulos, & Pitas, 2004), (Sedaaghi et al., 2007) and (Lugger & Yang, 2007) used sequential floating forward selection (SFFS) method to discover the best feature set for the classification.

Classification techniques used in emotion classification task includes Support Vector Machines (SVM) (Hammal et al., 2005), (Shami &Verhelst, 2007), (Altun & Polat, 2007), Neural Networks (NN) (Zhongzhe et al., 2006), Hidden Markov Models (HMM) (Le et al., 2004), Linear Discriminant Analysis (LDA) (Hammal et al., 2005), (Lugger & Yang, 2006), Instance Based Learning (Zervas et al., 2006), Vector Quantification (VQ) (Le et al., 2004), C4.5 (Zervas et al., 2006), (Shami & Verhelst, 2007), GentleBoost (Datcu & Rothkrantz, 2005), Bayes Classifiers (Ververidis, Kotropoulos, & Pitas, 2004), (Hammal et al., 2005), (Lugger & Yang, 2007) and K-Nearest Neighbor (K-NN) (Ververidis, Kotropoulos, & Pitas, 2004), (Hammal et al., 2005), (Shami & Verhelst, 2007) classifiers.

2.3 Text Modality

Achievements in this domain can be used in next generation intelligent robotics, artificial intelligence, psychology, blogs, product reviews, and finally development of emotion-aware applications such as emotion-aware Text to Speech (TTS) engines for emotional reading of text. CRM and service oriented companies like Right Now Technologies and NICE Systems produces customer service software SmartSense™ and NICE Perform™ respectively which recognizes customer emotions using keyword spotting technique and prosodic features of speech then performs flagging, prioritizing and routing inquiries and customers based on emotional content.

One side of the problem is the selection of a qualified dataset for machine learning methods. In order to cover most of the words in a given language, a large-scale dataset is needed. In addition, this dataset should have variation of emotional content, independent emotional responses from different cultures to eliminate cultural affects of emotion.

Manual creation of large-scale datasets is difficult and time-consuming task. Blog based datasets provides large-scale lexicons as presented in (Mishne, 2005). They worked on large collection of blog posts (122,624 distinct web pages) for classifying blog text according to the mood reported by its author during the writing. According to their results, increasing the amount of training data leads an additional increase in classification performance. On the other hand, the quality of the dataset is important for better classification.

Over the last quarter-century, there is increasing body of research on understanding the human emotions. Many approaches have been proposed for HER from text. These approaches can be grouped into three main groups: keyword spotting, statistical NLP, and ontology based approaches. Each approach has its own advantages and disadvantages. In addition, there is no rigid line between these approaches.

Keyword spotting is easy to implement, and based on predetermined set of terms to classify the text into emotion categories. Despite its simplicity, creation of an effective lexicon is difficult too since only 4% of words used in texts have emotional value (Pennebaker, Francis, & Booth, 2001). For these reasons, it is not suitable for wide range of domains. The second group is based on statistical NLP approaches. This approach is similar to lexical affinity where affinities of words are still used but as a feed for a machine learning algorithm. In case of lexical affinity, words have some probabilistic value representing the affinity for a particular emotion class. However, it requires high quality, large-scale training dataset for a better classification. The third groups is based on ontologies, heavily uses semantic networks like WordNet-Affect (Strapparava & Valitutti, 2004) and ConceptNet (Liu & Singh, 2004) are linguistic resources for lexical representation of affective information using commonsense knowledge. ConceptNet is an integrated commonsense knowledgebase with a natural language processing toolkit MontyLingua that supports many practical textual reasoning tasks over real world documents without additional statistical training.

Other methods use large annotated corpus or linguistic corpus to train their machine learning algorithms such as lexical affinity and Statistical NLP.

Mishne (2005) used blog based emotion datasets, where authors already label every blog document. It seems that it is good for generating large-scale lexicon for a better representation for a given language. Blogs have more than 200-300 words per document on average. However, assigning a single emotional label to a document having many words is not very meaningful. Therefore, a better training set for each emotional class must consider sentences and words, not paragraphs. After preparing a proper training set and selecting good features, the next task is to classify a given text.

Shaikh, Prendinger, & Ishizuka (2006) proposed a formal model that can make emphatic response with respect to the emotional state detected in the text. Shaikh, Prendinger, & Ishizuka (2007a) developed a new aggregator to fetch news from different news resources and categorize the themes of the news into eight emotion types using semantic parsers and SenseNet (Shaikh, Prendinger, & Ishizuka, 2007b). Shaikh, et al. (2006) studied the natural language and affective information using cognitive structure of affective information. They developed ALICE chat-bot based on Artificial Intelligence Markup Language (AIML) script to improve interaction in a text based instant messaging system that uses emoticons or avatar that represents the sensed emotion to express the emotional state.

According to Alm, Roth, & Sproat (2005), emotion annotation for text is a hard problem and inter-annotator agreement value k=.24-.51.

Liu, Lieberman, & Selker (2003) employed a commonsense knowledgebase OMCS (Open Mind Common Sense) having 400,000 facts about everyday world to classify sentences into basic emotions (happy, sad, angry, fearful, disgusted, and surprised) categories.

Boucouvalas & Zhe (2002) developed an emotion extraction engine that can analyze the input text in a chat dialogue, extract the emotion and displays the

expressive image on the communicating users display. Their parser only considers sentences in present continuous tense, sentences without starting auxiliary verbs (No question sentences allowed), positive sentences, etc.

Sugimoto & Yoneyama (2006) considered the emotional expressions for text-to-speech engines and emotional reading. They partitioned the text into nouns adjectives and adverbs and used the frequency of words to determine the emotional class.

Chuang & Wu (2004) tried to detect emotion from both speech and textual data. They manually defined the emotional keywords and emotion modification words. They have used "very" and "not" as a modification word where the only difference between "very happy", "happy", and "not happy" is the emotional intensity. As they are using keyword-spotting technique (they have 500 words labeled as emotion words), they reported that textual recognition rate is lower than speech based recognition. According to their work, emotion recognition performance of multimodal system is better than performance of individual modalities.

Figure 2.22 shows EmpathyBuddy application developed by (Liu et al., 2003) that support six basic emotions defined by Ekman (1993).

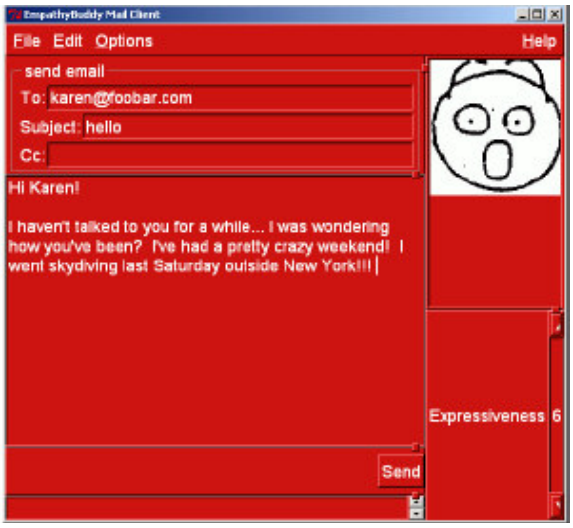

Figure 2.22 EmpathyBuddy email agent (Liu et al., 2003)

Figure 2.23 Example Scenario from (Liu et al., 2003)

2.4 Multimodal Emotion Recognition (MER)

Multimodal Emotion Recognition (MER) is a need for better classification on real world data. Current studies use primitive dataset for unimodal training and testing. For the audio modality, there exists limited number of speakers where usually one speaker speaks at a time and there is no background voice. Similarly, in visual modality, datasets with static images having single frontal upright faces recorded under studio environments are dominant. In addition, face pictures are taken when they are in silence so that mouth movements do not affect the classification for the video datasets, faces are in frontal upright position, and they do not speak during the emotional state changes. However, in case of video, it is not possible to find faces in frontal upright position or closed mouths all the time.

In spite of all these limitations, because of the semantic complexity of the emotion recognition task, reported state of the art performances for EER are very low. Therefore, researchers moved to study on bimodal and multimodal EER studies. Sebe, Bakker, Cohen, Gevers & Huang (2005) used visual and speech modalities, and Gunes & Piccardi (2007) used face and gesture properties. They reported that bimodal studies perform better than result of single modality. In case of fusing

different types of information, models that can handle incomplete and missing values must be used. Bayesian networks, Hidden Markov Models and TBM can be used for this purpose. Current integration methods use probabilistic classifiers such as Hidden Markov Models (HMM) and Bayesian Classifiers Alatan, Akansu, & Wolf (2001), Huang, Liu, Wang, Chen, & Wong (1999), Snoek, & Worring (2005).

De Silva & Ng (2000), studied the six basic emotions on a dataset having 144 image sequences and audio files from 2 subjects using Hidden Markov Models (HMM) and they used a rule based fusion scheme on video by considering both the facial expressions and emotional speech. On visual part they have used optical flow algorithm to find the displacements of facial feature points and on audio part they used pitch values with a HMM classifier. They have used rule based bimodal fusion where the same result is expected from visual and audio modalities, otherwise output of the dominant modality is selected as a final emotional class. Overall, bimodal recognition rate is 72%, which is better than unimodal video and audio results.

Sebe, Bakker, Cohen, Gevers, & Huang (2005), used Bayesian networks for bimodal fusion of audio-visual information containing a set of 38 subjects with 11 affective states. On visual part, they used a face tracker and 3D wireframe model to fit facial features to control points. On audio part, they used logarithm of energy, syllable rate and two pitch values. Their proposed model considers the speaking state of the speaker. If the subject speaks then recognition process is also affected by speech information. In addition, According to their results, average recognition accuracy is 56% for face-only classifier, 45% for the prosody-only classifier, and 90% for bimodal classifier.

Gunes & Piccardi (2007) automatically extracted the face and upper-body gesture features from video data of 4 subjects with 6 emotions. They selected to use BayesNet classifier for unimodal body and face features with the classification rate of 89.9% and 76.4% respectively. They employed both the feature level and decision level fusion schemes. According to their results, BayesNet classifier with feature level fusion scheme provides 94% accuracy, which is better than single modality results. They used posterior probabilities of unimodal results with sum, product, and

weight based criteria for the decision level fusion scheme and obtained 91.1%, 87.3%, 79.7% accuracy for each respectively. Their results showed that early fusion scheme gives better performance than both the late fusion and the unimodal schemes.

Wu, Oviatt, & Cohen (1999) studied on approximating the conditional density functions for speech and gesture modalities. Their approach can be further generalized for other modalities. They also proposed a method to predict the theoretical lower and upper performance bounds of the fused system.

Go, Kwak, Lee, & Chun (2003) studied on a dataset containing facial images and speech signals of 20 people (10 per gender) showing 6 emotions. They have used multi-resolution analysis of wavelets for speech signal and Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) for facial feature extraction. Their fusion mechanism considers the maximum of membership value, which is computed by comparing the input features with codebook features. Their bimodal recognition rate is 95% for male and 98% for female subjects.

Hammal, Couvreur, Caplier, & Rombaut (2007) has proposed fusion architecture based on Transferable Belief Model, TBM, where information from different sources can be combined to provide decisions that are more powerful. They showed that, humans are able to detect facial expressions by viewing only the contours of the facial features. In their study, emotions are represented by skeleton of the facial features generated from contour pixels of static and frontal viewed face images. TBM well suits for the fusion problem because TMB can handle both incomplete data and imprecise values.

CHAPTER THREE
EMOTION RECOGNITION in VISUAL MODALITY

Emotion recognition on visual modality tries to find emotional class of a given face or shot in a video. Researchers generally use still images and/or video for visual emotion recognition. Facial expression recognition is the most common research area but there are works based on detecting emotions by using cinematic features of video, another say emotion of the video. In case of video-based emotion recognition, first we need to find efficient and effective addressing of video using segmentation, indexing, and retrieval. Therefore, Indexing and retrieval of video has an important role in visual emotion recognition.

This chapter describes visual studies for facial expression recognition in video including shot boundary determination, face detection, visual query generation, and curve fitting based facial expression recognition respectively.

Facial expressions have a great role in face-to-face non-verbal human communication. Because, humans do not only communicate with words, they also use body language to support the focus of the subject. According to the Mehrabian, 55% of communicative message is transferred by facial expressions (Mehrabian, 1968, pp. 53-56).

In our facial expression recognition study, we presented a new method to recognize facial emotional expression that is all accepted globally that has a great role in human communication. The method uses basic image processing techniques and based on curve fitting on mouth region and able to detect happiness, surprise, and sadness emotions within the universally accepted emotional expressions (angry, disgust, fear, happy, sad and surprise). The proposed approach is tested on a test-bed containing 78 human face images of 13 different people with a different emotional expression and the experimentations resulted a satisfactory performance level.

Facial Expression Recognition based on polynomial curve fitting algorithm implemented using Matlab. The study focuses on finding a new approach for emotional expression recognition using basic image processing techniques.

We have used the matlab mcc compiler to get a standalone executable file that allows us to analyze a given Table of Contents (TOC) file of a video. We have written a service that gets the toc file and output file as input. It iterates over the TOC file and writes its output to another file in line based format. Table 3.1 shows sample output of the service. First column represents the videoId, second column represents the faceId, third column represents the emotional state of the face, and final column shows measured unnormalized emotion degree.

Table 3.1 Sample output from the emotion service

6 290 HAPPY 10
6 291 NEUTRAL 0
6 292 HAPPY 6
6 293 NEUTRAL 0
6 294 HAPPY 5
6 295 HAPPY 11
6 296 NEUTRAL 0
6 297 HAPPY 0
6 298 HAPPY 11
6 299 NEUTRAL 0

In video frames, faces usually come with different lighting conditions, so we have made modifications on the emotional expression recognition service to handle these variants. We applied sobel edge detection and implemented 8-connected region growth algorithm to increase the detection rate of the mouth region. In addition, we have used simple city-block distance based classification to map facial feature points to predetermined facemask. As a result, we classify five different emotional states namely "*Happy*", "*Sad*", "*Surprise*", "*Neutral*", and "*Unknown*".

Happy Emotion: We have a positive emotion value greater then 5

Sad Emotion: If negative emotion value less than 5

Surprise Emotion: We have either positive or negative emotion value and the mouth is classified as the minimum eccentricity region among others.

Neutral Emotion: Emotion value between ±5

Unknown Emotion: This emotion is return due to an error during the feature extraction process in mouth region. The most probably reason is that mouth region is not complete enough to detect.

3.1 Shot Boundary Detection

We have used a method based on histogram differences for cut and gradual transition detection on TRECVID2006 dataset. Both the details of our approach and effects of using different threshold & skip frame intervals on evaluation results are presented.

Our initial experiments based on shot detection has been implemented based on JPEG compressed frame size method which compares the frame size differences in order to detect cut points (Little et al., 1993, pp. 427-436).

In this study, we see that increasing the compression rate also increases detection rate of cut points. 50% compression gives the best results for cut detection. Figure 3.1 shows how cut points detected. Falls and raises in the file size curve are interpreted as cut points.

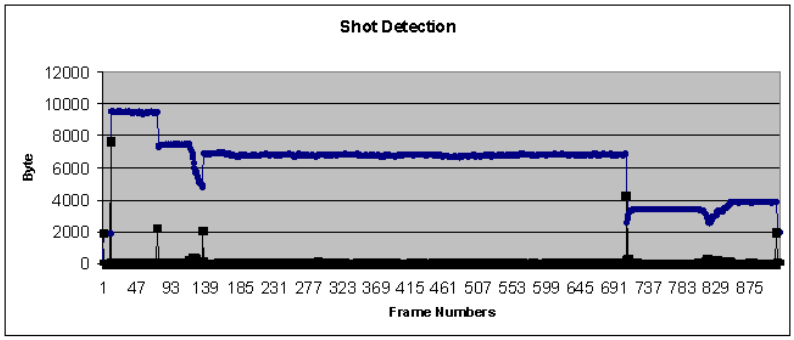

Figure 3.1 Shot detection

Our next approach for SBD task is based on color histogram differences in RGB color space for both cut and gradual transition detection. This method uses a threshold value for cut detection and a *skip frame interval* value for eliminating consecutive frames that have much redundant information (*TRECVID dataset and test videos are all MPEG files with 29,97fps ratio*) for faster processing. Instead of processing all consecutive frames, we skipped frames by a factor of *skip frame interval* value. The values used in our experiments changes from one to five frames and it is clear that it dramatically reduces the computation time with a relatively small change in the precision.

In our approach, histograms of pixels in RGB color space for each frame are used to detect shot boundaries. First, we have quantized RGB color space into 27 equal sub-spaces by dividing each axis of color space into three equal parts. In another say, each sub-space is a cube with a size of 85. Shortly, we have a 27-bin feature vector, which is easy to compute and process further. More formally, the histogram, H, can be defined as follows (5): where R_i, G_i, B_i, represents the i^{th} pixel values for red green and blue channels, respectively.

$$(\forall R_i, \forall G_i, \forall B_i)(r,g,b \ni \{1,2,3\}, p = 255 \div 3) \text{ and } H(r,g,b) = H(r,g,b) + 1 \quad (5)$$

where $(r = R_i \div p)$ & $(g = G_i \div p)$ & $(b = B_i \div p)$

Then, Euclidean distances of histograms belonging to two successive frames are calculated for the cut and gradual transition detection. It has three different thresholds. One for detection of abrupt changes T_C (for Cut Detection), the other is for detection of gradual transitions especially cross-fade effect T_G (for Gradual Transition Detection) and the last one T_{SFI} is to decrease temporal redundancy in video frames. If any frame-by-frame difference is greater than the maximum allowed threshold value, T_C, then a cut is said to be detected.

Consecutive frames in video usually have similar spatial information. If we skip only one frame then total number of comparisons halved. Therefore we use skip frame interval threshold T_{SFI} for better computing performance. We set the T_{SFI} value

to one through five in our experiments. However, our algorithm does not simply compare every T_{SFI}^{th} frames. It can compare frames in both directions (backward and forward) for better frame accuracy. Whenever it finds a distance that exceeds the T_C value, it turns back to the next frame of the previously compared frame and then continues its step-by-step comparison without considering T_{SFI} value. Pseudo algorithm is shown in Table 3.2;

Table 3.2 Pseudo code for shot change detection

```
While frames exists
 Compute Euclidean distance of current frame F_i to target frame F_{i+T_SFI}
 If distance exceeds T_C then
  If it's a first time shot then set target frame to F_{i+1-T_SFI}
  Else
    set shot.startFrame=i , shot.endFrame= TargetFrame
    If TargetFrame-i < T_G then mark this shot as 'SubGradual'
    else mark this shot as 'Cut'
    Set current frame with target frame
    Add T_SFI to target frame
    Else
     If it is not a first time shot then add T_SFI to target frame
     Else increase the target frame.
End
```

A good threshold value T_C should be selected for best accuracy on hard cuts and skip frame interval must be long enough to get better computing performance by eliminating the temporal dimension of video.

3.1.1 Gradual Transition Detection

Gradual transitions are detected on a second pass by computing the length of the consecutive cuts. We have used a second threshold that holds the minimum number of frames that a shot holds. If a shot region has frames, less than the threshold T_G then it is marked as a gradual transition. The value of T_G is fixed to 10 frames in our experiments. It is clear that selection of threshold values T_C and T_G critically effects the number of shots can be detected by the system. If T_C has a lower value, then

number of detected shot increases, otherwise decreases. If T_G has a lower value then number of detected gradual transitions decreases, otherwise it increases.

T_G is the minimum number of frames that a shot should holds. If a shot has less then T_G number of frames then it is marked as a SubGradual. Another say, shots that has short durations (compared with T_G value) are set as gradual transitions. Minimum and maximum boundaries of T_G can be set between 10-60 frames, which are the most commonly used average gradual cross-fade transition length. Its value is linearly proportional with the number of shots detected as cross-fade. Smaller T_G value also decreases the number of detected cross-fades. We can define the gradual cross-fades more formally;

Let T_G is the threshold value for detecting the cross-fade effect. S_i represents the i^{th} shot and represents the shot length in terms of # of frames. S_{CF_j} is a sub cross-fade region whose $L_{Si} \leq T_G$ and $CF_{i,j}$ is i^{th} ile j^{th} is the cross-fade between the i^{th} and j^{th} frames then S_{CF_i} and $CF_{i,j}$ is computed as shown in (6) and (7) respectively.

$$S_{CF_i} = \begin{cases} Cut & L_{S_i} \rangle T_G \\ SubGradual & L_{S_i} \leq T_G \end{cases} \quad \textbf{(6)}$$

$$CF_{i,j} = \begin{cases} Gradual & (\exists i, \exists j) S_{CF_i}, S_{CF_j} \in \{SubGradual\}, j = i+1 \\ null & else \end{cases} \quad \textbf{(7)}$$

Figure 3.2 shows cross-fade detection example. In the first line, there is a fade out, last line shows a fade-in effect, and they produce the middle line the actual dissolve effect.

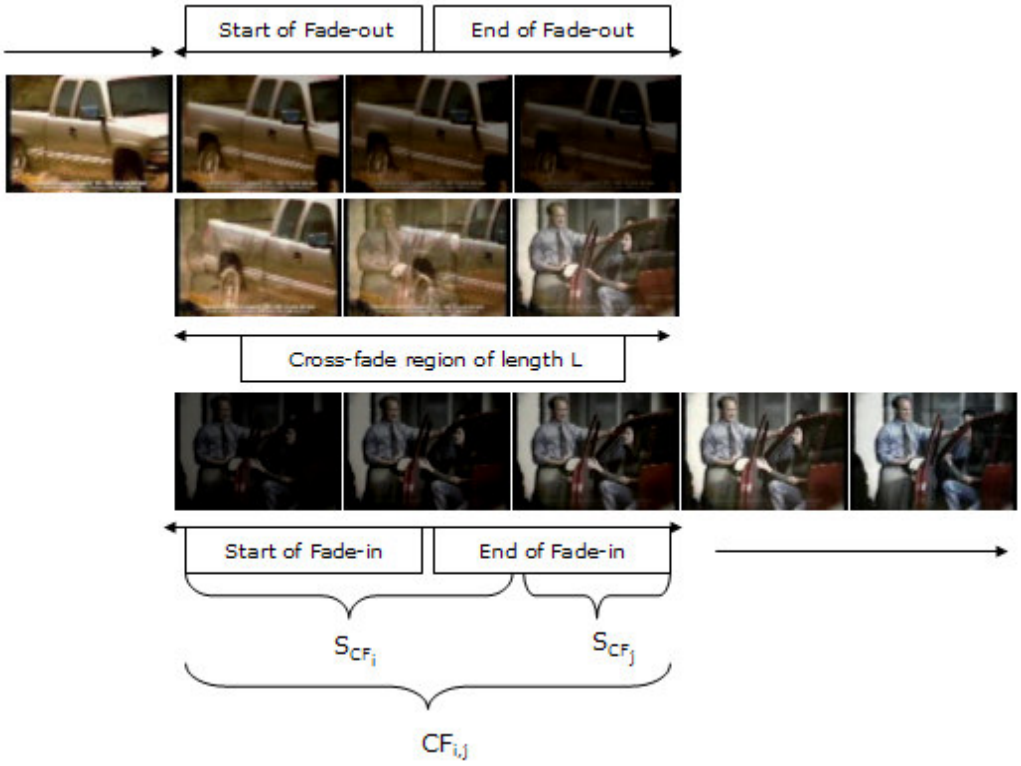

Figure 3.2 Example cross-fade detection

3.1.2 Keyframe Selection

Our strategy more like to the combination of first and last strategy. We select key frames from shots of type 'Cuts' having a human face. In this case, the middle frame of the emotion sequence is selected. Another redundancy can be provided by not to consider dissolve regions because that they do not have a semantic meaning except morphing as they are just a video effect. More formally;

Let S_i represents the i^{th} shot, $T(S_i)$ represents the type of shot, $E(F_j)$ represents emotion of the frame F_j, KF_{S_i} represents the key frames of S_i, n is the length of the S_i in terms of number of frames then;

$$(\exists i \exists j) KF_{S_i} = \left\{ F_{\frac{\min(j)+\max(j)}{2}} \right\} where\left(T(S_i) = 'Cut'\right) and\left(\left(E(F_j) = E(F_{j+1})\right) and\left(E(F_j) \neq 'null'\right) and(j \leq n)\right)$$

3.2 Face Detection

We have used the face detection algorithm provided by the OpenCV library. The algorithm has one profile detector and four frontal detectors. According to Acar, Atlas, Cevik, Olmez, & Unlu et al. (2007) OpenCV face detection method is said to be the most successful approach on TRECVID2006 dataset. The research of Isabel, Xavier, Jérôme & Vincent (2005) on 160 images having 540 frontal faces Stump-based 20×20 gentle adaboost frontal face classifiers outperforms the other methods with 90% detection rate but its false alarm rate is 67 over these 540 faces. In order to get better performance from face detection subsystem we added an additional layer on the face detector to decrease the false alarm rate by using simple threshold based skin color approach.

TRECVID2006 dataset has 79,484 shot in 260 videos. Face detection algorithm applied on 146,328 representative and non-representative key frames from 79,484 shots and 63,359 faces detected within 36,779 shots using OpenCV built-in face detection function. Results are stored on MySQL database considering E/R diagram shown in Figure 3.3.

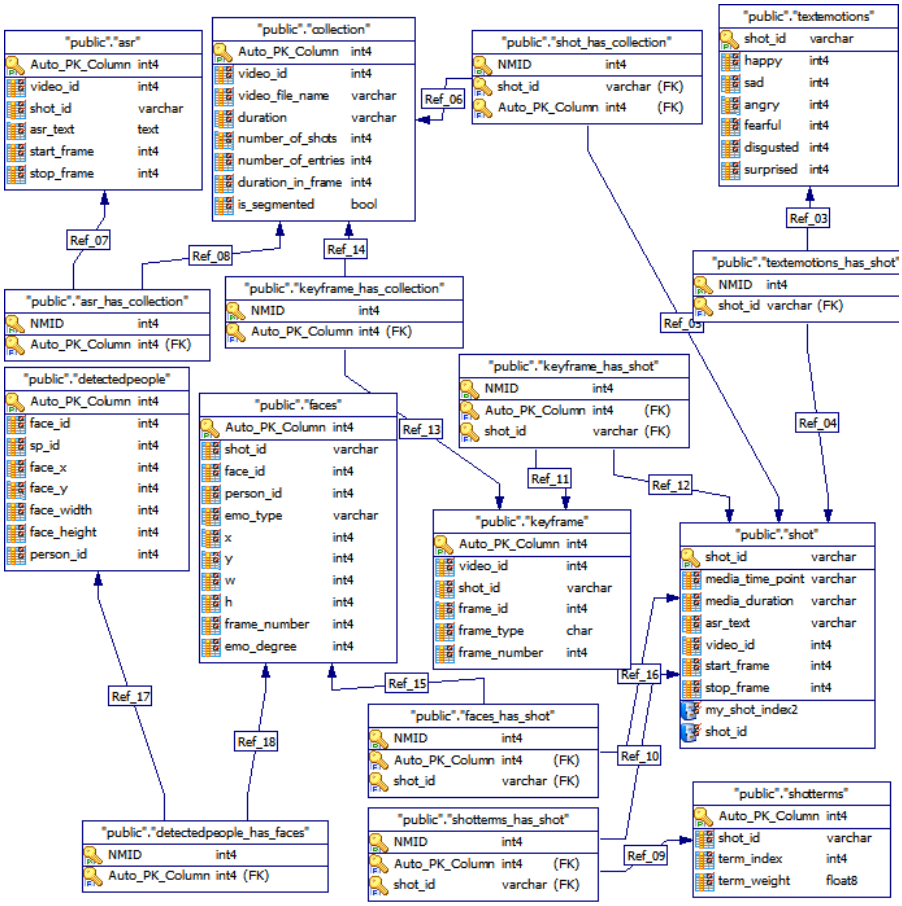

Figure 3.3 E/R diagram

3.2.1 Refinements on Face Detection

After applying standard face detection operation, we have collected a set of face from each frame of the video. Some of the faces (~15%) are false alarms that are not a face actually. Initially we do not have any false positive results. If a face is a false positive then we moved this face information to the set called non-face regions and modified the threshold value of skin color accordingly. Figure 3.4 shows example non-face regions from the dataset.

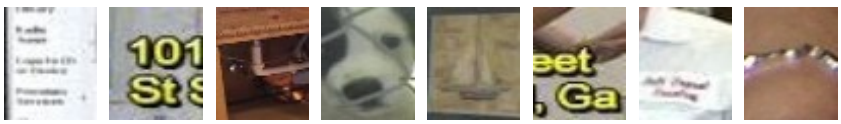

Figure 3.4 Non-face regions

We used cvMean function of OpenCV, which calculates the means value of the region of interest in a given image. In order to eliminate too bright face candidates, we set the threshold value to 200; if it exceeds the maximum threshold value then we delete the face with cvSeqRemove function. Without using this heuristic, we have 320 false alarms in selected 1000 faces and when we enabled the heuristic, it reduces to 275 false alarms. As a result, we eliminated ~15% of the false alarms in average.

3.3 Visual Semantic Query Generator

We have developed a visual query generator, which satisfies positional, existence and emotional queries by performing face detection on ADVIS2004 and TRECVID 2006 dataset.

We have developed a visual query generator that satisfies three types of queries namely *existence queries*, *positional queries*, and *emotional queries*. Emotion classification has done manually by the indexer. We have tested our algorithm in a database containing randomly chosen 100 still images from Advis 2004 Conference. Experimentation showed that our approach for retrieval of images success on human oriented image databases.

Simple interface for query formulation is shown in Figure 3.5. Queries are generated visually on this query generator and then translated into traditional SQL syntax. Our query types as follows:

Existence Query: This type of query is used to find a specific person, or it can be used to retrieve images having at least *p* persons where *p* is the number of generic query faces inserted on query generator. Examples can be:

- "Show me the list of images where *a* exists and *a* is 'Taner'
- "Show me the list of images where *a* exists with someone else and *a* is 'Taner'"
- "Show me the list of images where there exists at least *p* person and *p* is 3"

- "Show me the list of images where there exists at least *p* person and *p* is 3 and one of them let p_0 is 'Taner'"

Figure 3.5 shows sample existence query to search two people.

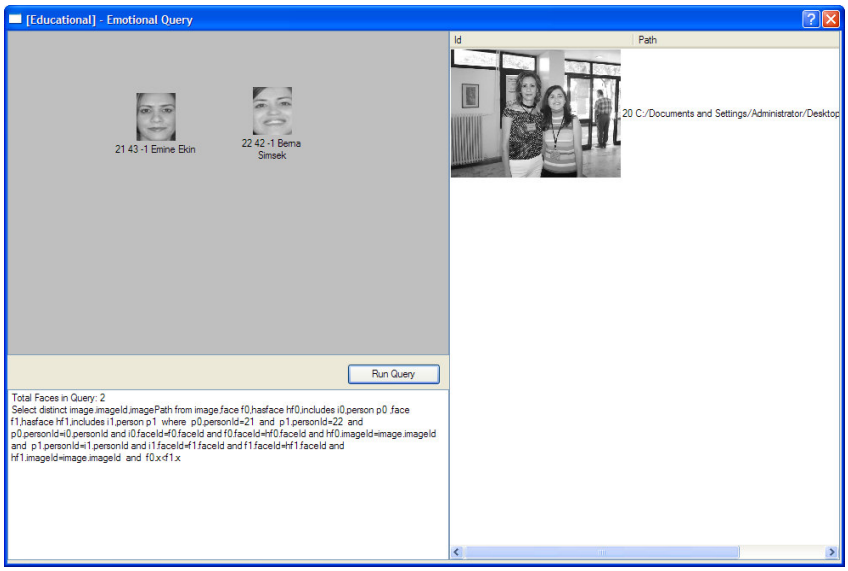

Figure 3.5 Existence query example

Positional Query: This type of query considers relative face locations on images. Using this type of query allows us to define constraints such as Leftof (*a*, *b*) and Rightof (*a*, *b*) where *a* and *b* are face objects. Examples can be:

- "Show me the list of images where *a* is left of *b* and *a* is 'Emine' and *b* is someone else"

- "Show me the list of images where person *b* is in middle of person *a* and person *c* and *b* is 'Emine'"

Emotional Query: This type of query considers emotional expressions of face as a constraint. Thus, it allows us to retrieve a set of "Happy" persons. Examples can be:

- "Show me the list of images where persons smiles"

- "Show me the list of images where a is left of b and a is smiling and b is sad and a is 'Ali' and b is anyone"

- "Show me the list of images where someone is happy while someone is sad"

3.4 Curve Fitting Based Facial Expression Recognition

Our approach uses, known image processing techniques to detect three emotional expressions *happy*, *sad,* and *surprised* using facial features on mouth region. Result of the processed images usually has discrete points, which is not suitable to process further. In this case, curve-fitting technique is used to find smooth geometric regions to find facial feature sets. Details of our approach and dataset are explained in following sections.

3.4.1 Dataset and Input Images

The Carnegie Mellon University Advanced Multimedia Processing Laboratory (AMP) face expression database is used as a test bed. The database includes 13 subject's face image with 75 expressions. Total number of images is 975. Each expression has a different intensity. They collected the face images in the same lighting condition using CCD camera. Face images have been well-registrated by the eyes location. The following example shows some expression images of one subject. Example input images from the database are shown in the Figure 3.6.

Figure 3.6 Example input images

Our first assumption is that, the input image contains only the face region, so no face detection is required. Appendix A shows the complete matlab source code for the application and Figure 3.7 shows basic steps for facial expression recognition.

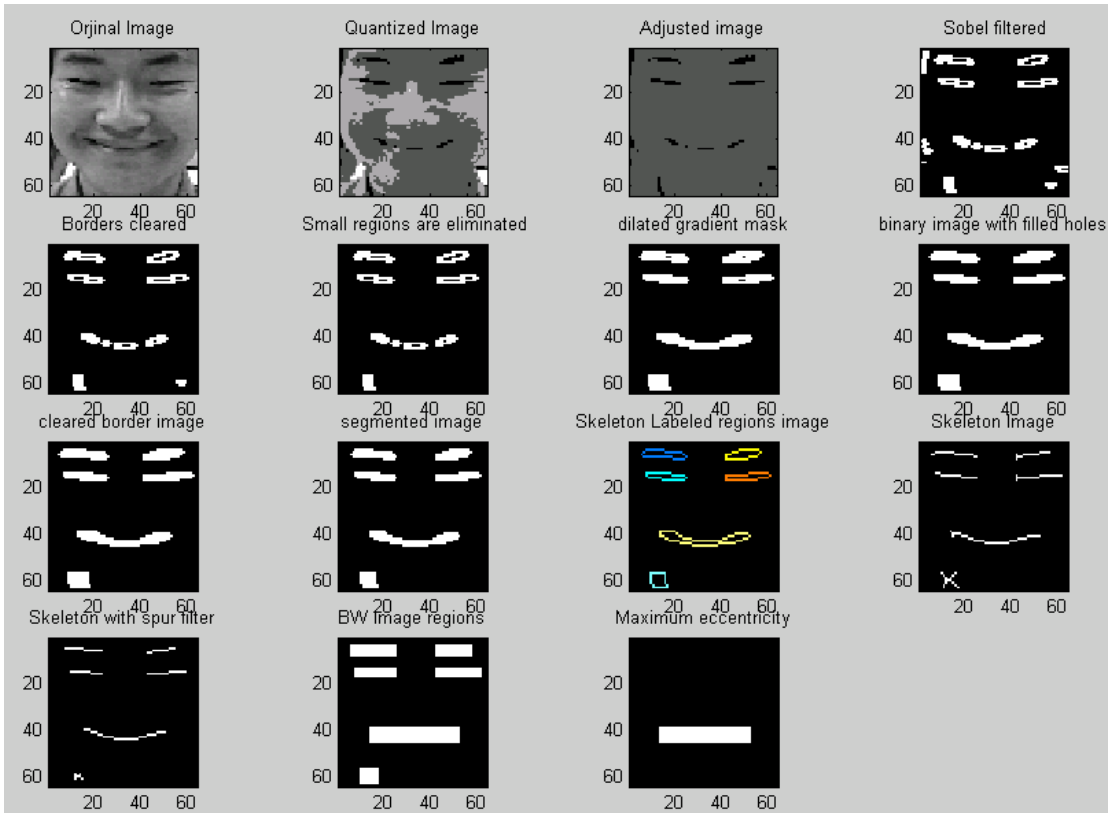

Figure 3.7 Step by step image processing

First, all the input image is resized to 64×64 pixel (Line 1 in Appendix A). This step is required in order to use images out of the database. An example image is show in Figure 3.8.

Figure 3.8 Example input image

In this step RGB color space is converted into grayscale image (Line 2-7 in Appendix A). First, the color component number, *Sz*, is checked and then if it is greater than one, which also means that, the image contains more than one color component than it is converted into grayscale image.

After that, nonuniform brightness in the image is eliminated by first converting image into double precision image and then coarse estimate of the background illumination is achieved by using blkproc function. After that, the coarse estimate is expanded in size using the function imresize so that it has the same size as the original image (Line 8-13 in Appendix A).

After the illumination correction step, the grayscale image is converted into indexed image. After that gray2ind scales, then rounds, the intensity image to produce an equivalent indexed image with four index entries. In addition, we further segment the image using thresholding (Line 14-19 in Appendix A) as shown in Figure 3.9.

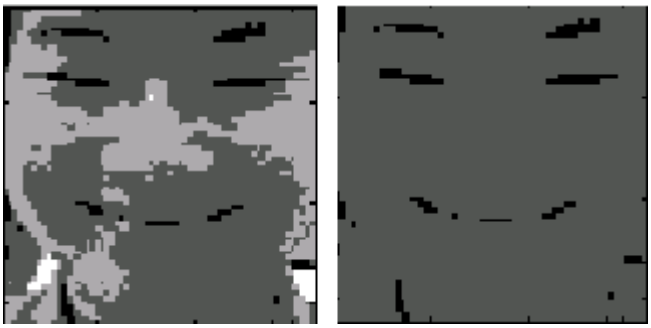

Figure 3.9 Quantized and threshold image

The Sobel method finds edges using the Sobel approximation to the derivative in both horizontal and vertical directions. It returns edges at those points where the gradient of *I* is maximum. The edge function ignores all edges that are not stronger than given threshold (Line 20-21 in Appendix A). Result is shown in Figure 3.10.

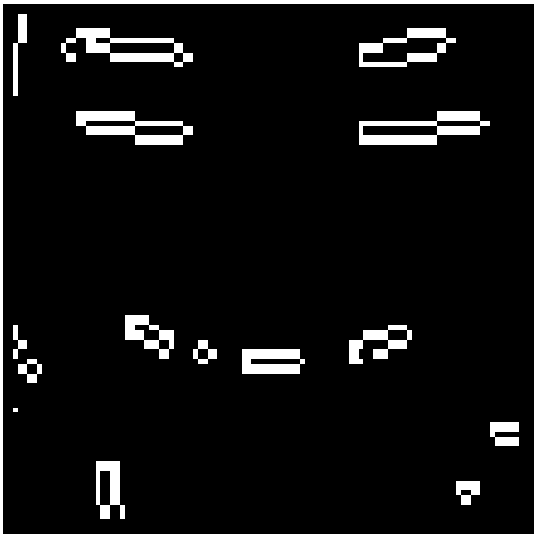

Figure 3.10 Sobel filtered image

The *imdilate* function dilates the image with a defined structuring element. The strel function creates the structuring element. In this case, we have used horizontal lines of width 1 and 2. It allows us to connect the related points in horizontal direction. The most advantage of using the horizontal lines is on mouth region of the face. It connects broken lines previously detected by the sobel filter (Line 22-24 in Appendix A) as shown in Figure 3.11.

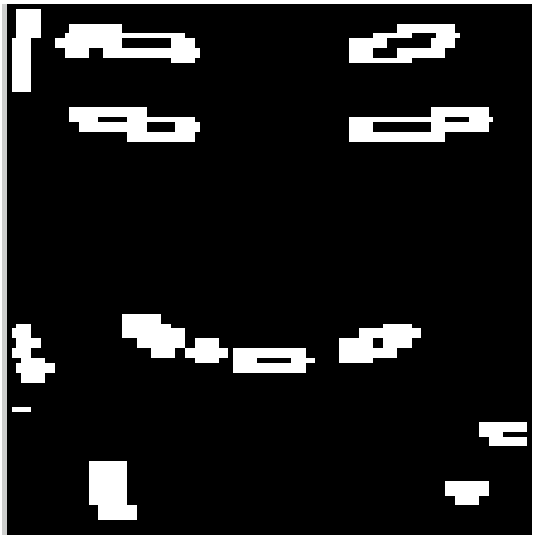

Figure 3.11 Dilated image with horizontal lines

We removed regions that connect to the borders of the image as shown in Figure 3.12. It is achieved by the *imclearborder* function with four connectivity. Any region having at least one pixel touching to the border is removed (Line 25 in Appendix A).

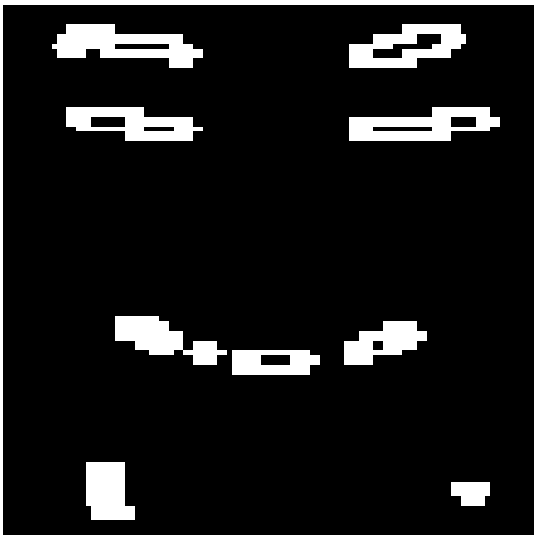

Figure 3.12 Border regions are cleared

After that eliminate small regions if they still not connect to any region. The minimum size of the region is determined by dividing the image size by 4 (Line 26-28 in Appendix A). Result is shown in Figure 3.13.

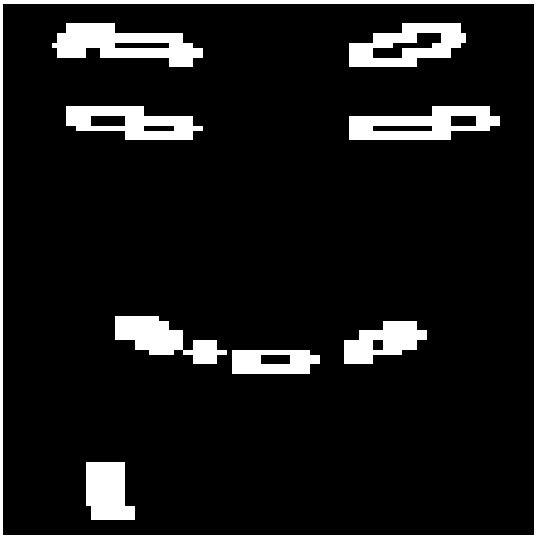

Figure 3.13 Eliminating small regions

After this step, three more steps applied in order to get better results; these are same as the previous steps. First dilated gradient mask is applied with horizontal line functions of having width length 1 and 5. After that step, holes in the regions are filled and borders are re cleared (Line 29-33 in Appendix A), as shown in Figure 3.14 and Figure 3.15.

Figure 3.14 Re-applying dilated gradient mask

Figure 3.15 Image with holes filled

Next is to find perimeter pixels. *imerode* function is used for thinning the holes filled image. The *bwperim* function finds the perimeter pixel values of the regions in the given image (Line 34-37 in Appendix A), Figure 3.16.

Figure 3.16 Perimeter pixels of the regions.

Then labeling allows us to distinct the regions from each other. The code segment provides labeling of human face regions (Line 38-40 in Appendix A) and Eccentricity values of the each region in the image. The eccentricity value of a region

is equal to the one if the region is completely looking like a line and zero if it is a circle. Then skeleton filter finds the skeleton of the image. However, only using the skeleton does not eliminates the branch outs on the extremity points. Therefore spur filter is applied as shown in Figure 3.17 and Figure 3.18. The Corner points holds x1, y1, x2, and y2 points of each of the region. The (x1, y1) pair is the top left corner of the region and (x2, y2) is the top right corner of the region (Line 41-56 in Appendix A).

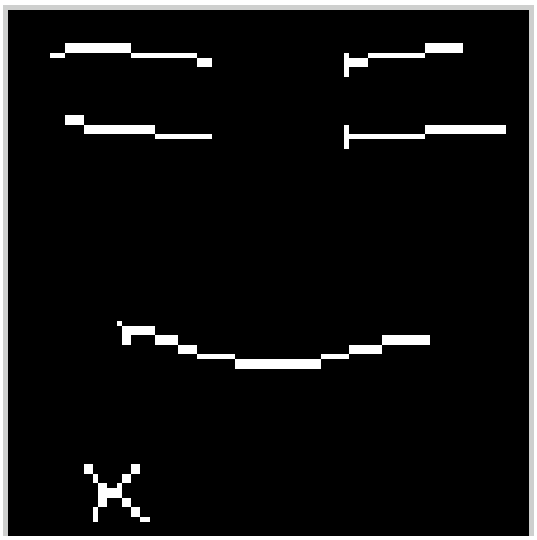

Figure 3.17 Skeleton filter applied image

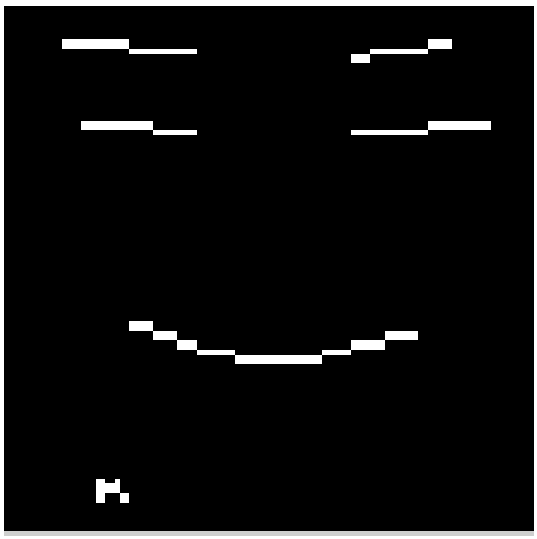

Figure 3.18 Spur filter applied image

Then we tried to find corner points of founded regions (upper left x: x1, upper right x: x2, upper left y: y1, upper right y: y2). After that, we have used a mask representing approximate positions of facial regions in a facial area of size 384×256 as shown in Figure 3.19. Then city-block distance measure is used to classify mouth, noise, eyes, and eyebrows.

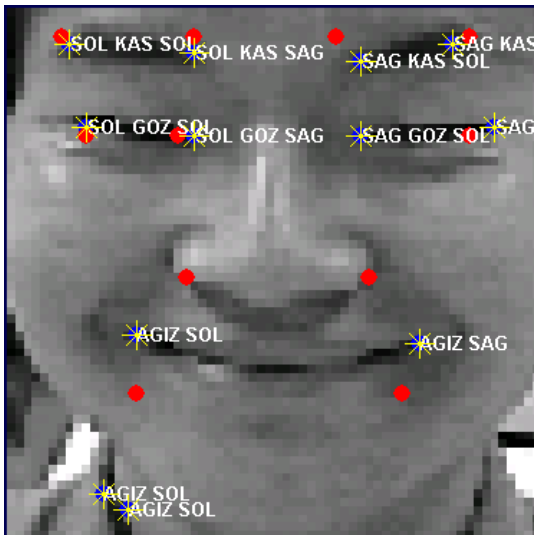

Figure 3.19 Classified regions

After classifying regions, the labeled regions fitted into a curve and a number of coefficients are provided. Using the string operations, (Line 57-84 in Appendix A) first finds the coefficients of the polygon that represents the labeled regions and then finds the mathematical formula of the polygon.

Emotion classes for this study is are happy, sad and surprise emotions. The happy and sad emotions can be detected by computing the integral of the mouth class. A line equation is computed by using the (x1, y1) and (x2, y2) points, then for each point between the x1 and x2 the cumulative difference between the evaluated y' points are computed and then normalized by |x1-x2|. If the result is greater than zero then the emotion is classified as happy as shown in Figure 3.20 and Figure 3.21, otherwise, it is classified as sad as in Figure 3.22 and Figure 3.23. It is the simplest constraint to detect the two emotions.

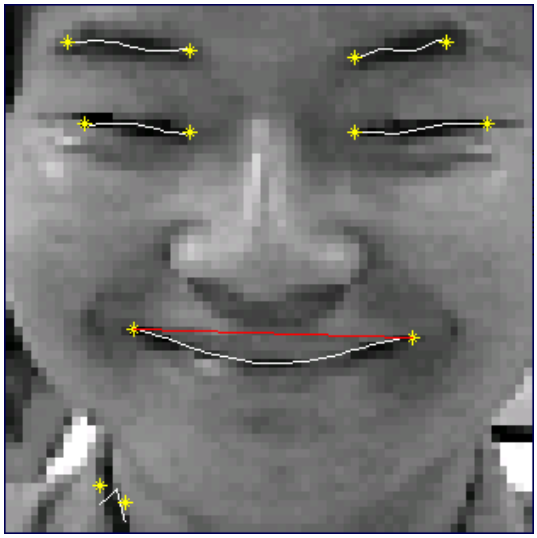

Figure 3.20 Happy emotion with +68.5

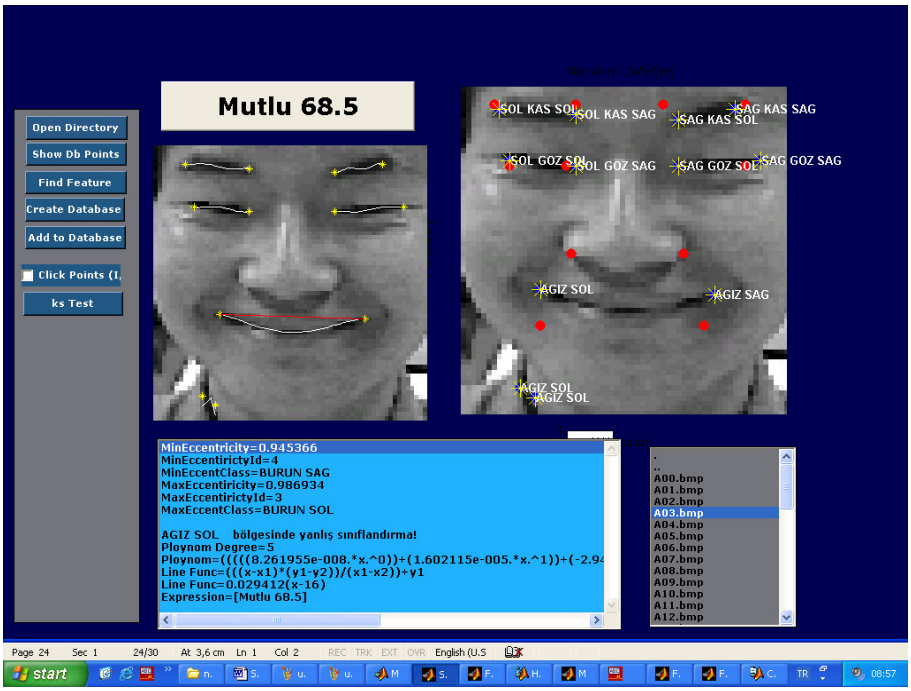

Figure 3.21 Happy emotion

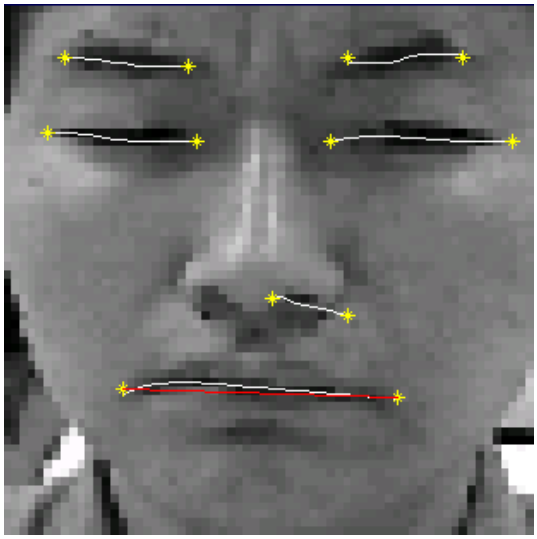

Figure 3.22 Sad emotion with -33.84

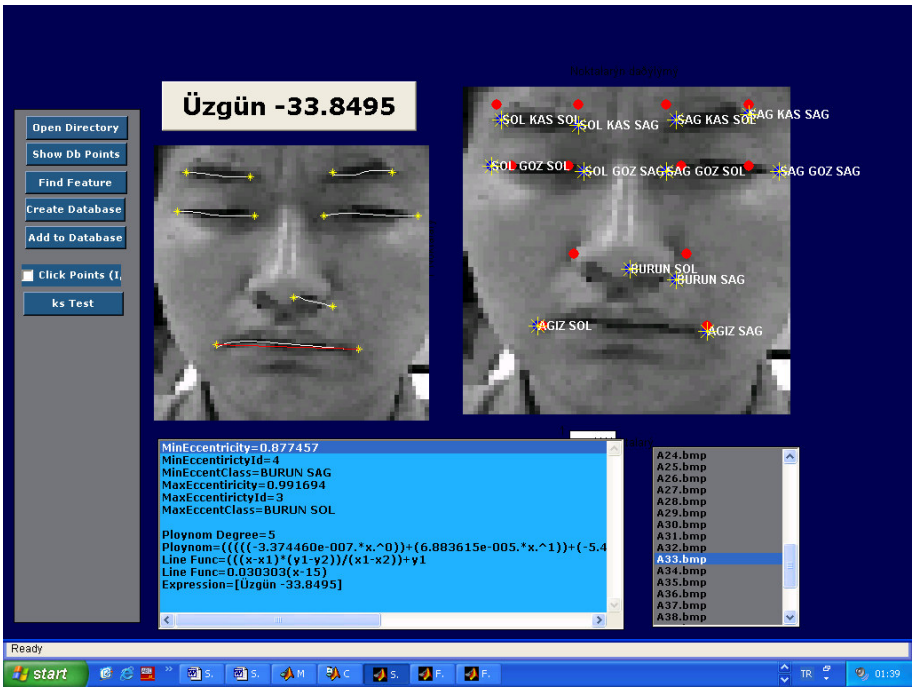

Figure 3.23 Sad emotion

The surprise emotion is detected when the mouth has the property of “Maximum area and minimum eccentricity among the other facial feature classes” as in Figure 3.24. Curve fitting with a line passing through the two corner points of the mouth is one of the simplest ways of understanding the happy and sad emotions. But in some circumstances the performance of the algorithm depends on the performance of the

image segmentation step, if the results of the image segmentation steps are not good enough then future steps gives low performance.

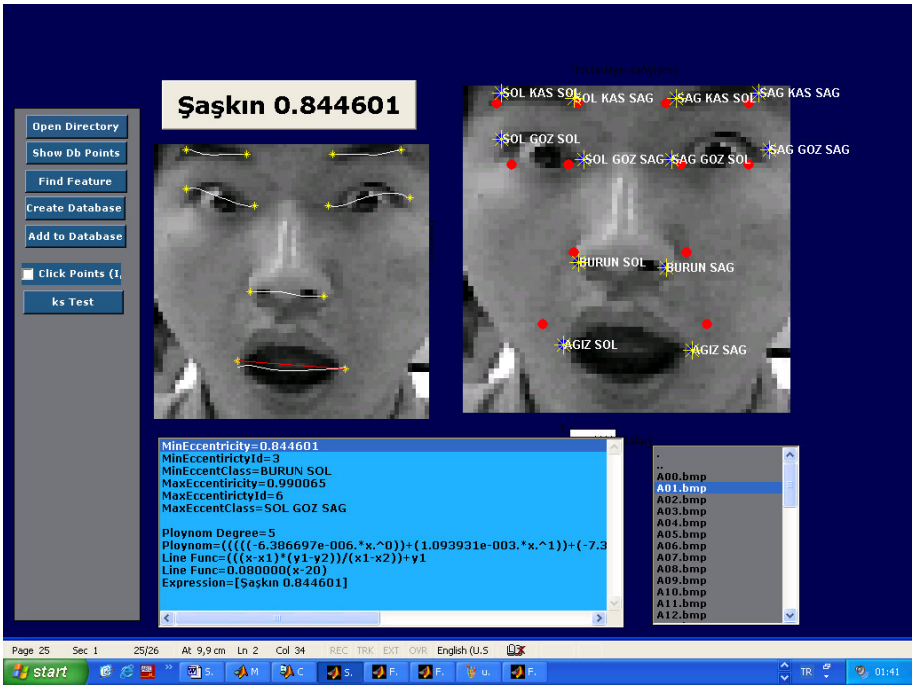

Figure 3.24 Surprise emotion

3.5 Experimentations

3.5.1 Shot Boundary Determination on TRECVID2006 Dataset

We have presented our approach to shot boundary determination problem using TRECVID 2006 dataset and master shot reference provided by (Petersohn, 2004) in our experiments.

For the experiments we have rescaled the size of each video frame to 80×60 for efficient processing. Figure 3.25 show the scaling and normalization of video frames. It also provides us to use a single threshold value for all different videos without normalization. Our maximum threshold value is $\frac{(80\times 60)}{20}$.

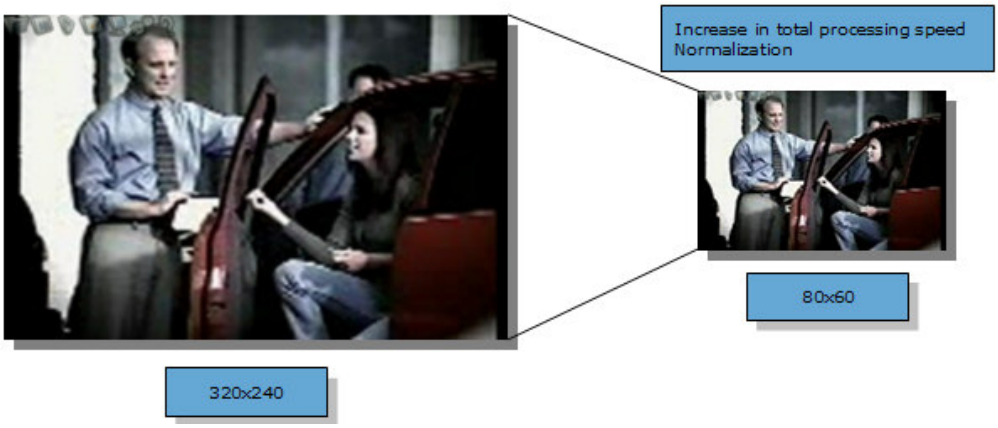

Figure 3.25 Scaling of video frames & Normalization

Figure 3.26 and Figure 3.27 represent the detected shot from three video files having 3, 4, and 9 shots. Each shot region has its own start frame, end frame, and shot type ∈ {'C','F'} where 'C' represents hard cuts (also known as cut with abrupt change in color histogram), 'F' represents the cross-fade effect.

Go back | Next | Refresh

```
SELECT * FROM `openface`.`shot` order by videoId, shotId;
```

Resultset 1

videoId	shotId	startFrame	endFrame	shotType
13	0	1	59	C
13	1	60	150	C
13	2	151	189	C
14	0	1	47	C
14	1	48	210	C
14	2	211	220	F
14	3	221	234	C
15	0	1	200	C
15	1	201	203	F
15	2	204	509	C
15	3	510	824	C
15	4	825	827	F
15	5	828	936	C
15	6	937	941	F
15	7	942	994	C
15	8	995	1146	C

Figure 3.26 Metadata database view for storing video shots

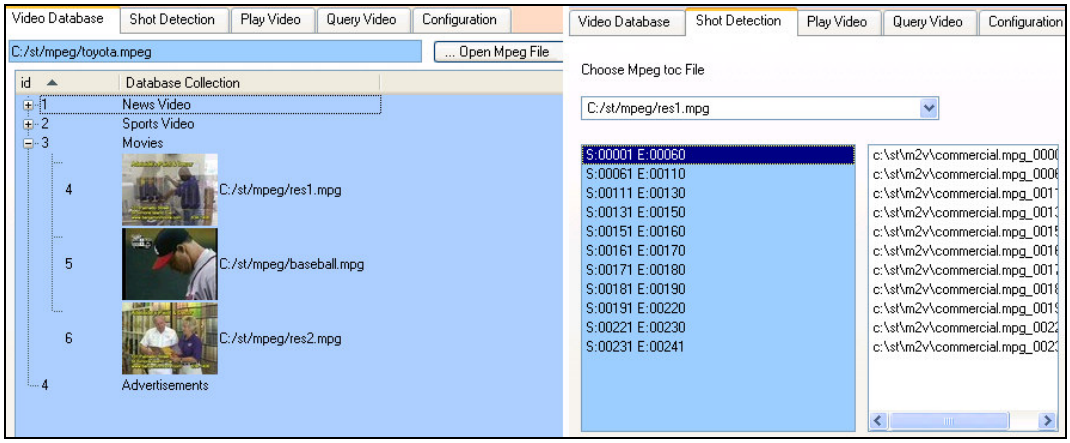

Figure 3.27 Shot detection

Table 3.3 shows the parameters used in each submission.

Table 3.3 Summary of each DEU run

System ID (runID)	T_C	T_G	T_{SFI}
EU_1150_1	1150	10	1
EU_1150_5	1150	10	5
EU_1250_1	1250	10	1
EU_1250_5	1250	10	5
EU_550_1	550	10	1
EU_550_5	550	10	5
EU_750_1	750	10	1
EU_750_5	750	10	5
EU_950_1	950	10	1
EU_950_5	950	10	5

Table 3.4 TRECVID 2006 shot boundary determination results

	ALL		CUTS		GRADUAL			
							Frame	
sysid	Recall	Prec	Recall	Prec	Recall	Prec	Recall	Prec
EU_1150_1	0.273	0.160	0.324	0.147	0.138	0.383	0.397	0.771
EU_1150_5	0.267	0.168	0.320	0.154	0.123	0.423	0.378	0.790
EU_1250_1	0.260	0.167	0.307	0.152	0.131	0.419	0.393	0.763
EU_1250_5	0.249	0.171	0.298	0.157	0.115	0.463	0.364	0.771
EU_550_1	0.425	0.143	0.446	0.122	0.367	0.322	0.523	0.729
EU_550_5	0.422	0.147	0.445	0.126	0.359	0.343	0.502	0.756
EU_750_1	0.372	0.153	0.414	0.136	0.260	0.330	0.471	0.768
EU_750_5	0.362	0.156	0.405	0.139	0.244	0.354	0.441	0.806
EU_950_1	0.329	0.163	0.377	0.147	0.199	0.354	0.405	0.795
EU_950_5	0.319	0.167	0.367	0.151	0.190	0.387	0.395	0.817

According to the results, we gained a low performance on both gradual and cut detection. Although gradual transition detection problem is more complex than cut detection problem, precision of gradual detection is quite better than precision on cuts because of the low T_G value.

Results showed that selecting a low TC value leads us to better cut detection result as in case EU_550_1 and EU_550_5. However, at the same time it makes the lowest frame precision value on gradual transition detection.

We made our experiments on a single Pentium IV 3Ghz processor having 1GB of memory. Our timing results shows that *skip frame interval* T_{SFI} value decreases the total processing time as shown in Table 3. However, the most significant time belongs to decoding phase. Effect of T_{SFI} dominates on feature extraction time. In this table, *decoding time* refers to the time takes to decode each mpeg file to corresponding jpeg files. The *segmentation Time* is the time to compute feature vector and time to find transition type. In addition, *total run time* is the sum of the Decoding Time and Segmentation time.

Table 3.5 Timing results in seconds for TRECVID 2006 test set.

Run Id	Decode Time (A)	Segmentation Time			Total Run Time A+B+C
		Feature Extraction Time (B)	Transition Detection Time (C)	Total Segmentation Time (B+C)	
EU_550_1	20,274.73	15,479.93	286.82	15,766.75	36,041.48
EU_750_1	20,274.73	15,479.92	276.25	15,756.17	36,030.90
EU_950_1	20,274.73	15,479.92	274.12	15,754.04	36,028.78
EU_1150_1	20,274.73	15,479.92	269.21	15,749.13	36,023.87
EU_1250_1	20,274.73	15,479.92	271.89	15,751.81	36,026.54
EU_550_5	20,274.73	3,382.27	244.82	3,627.09	23,901.82
EU_750_5	20,274.73	3,336.96	238.26	3,575.22	23,849.95
EU_950_5	20,274.73	3,286.78	250.54	3,537.32	23,812.06
EU_1150_5	20,274.73	3,112.32	223.22	3,335.54	23,610.28
EU_1250_5	20,274.73	3,329.27	253.51	3,582.78	23,857.51

It is clear that use of T_{SFI} decreases the total segmentation time with a small change in detection performance. Our gradual detection performance is quite better than our cut detection performance because of the low T_G value.

Main problem in our model is to find a universal threshold T_C and T_G to address all types of video having different characteristics. Adaptive thresholds can solve this problem. Another problem in this area is varying frame-numbering problem of different decoders on same video document. During our experiments frames decoded by mplayer (Gereffy, 2005) produced different frame numbers than frames in master shot reference.

Finally, for this approach, decoding time has the majority over others. After extracting the feature vectors, it takes about 258 seconds in average to complete the segmentation of all 13 videos in test set having 597,043 frames.

3.5.2 Facial Expression Recognition Results on CMU AMP Lab dataset

Using curve-fitting algorithm on mouth area in emotional expression recognition allows us better performance on smile and sad emotions, because both of the emotions are almost completely depends on shape of the mouth.

The surprise emotion is detected by considering the mouth eccentricity value, its area and height. If any area on the face satisfies the three constraints then the emotion is said to be surprise. The following table shows experimental results for three emotional classes.

Table 3.6 Facial expression detection results

	TP	FP	FN	TN	Precision	Recall	Accuracy
Happy	8	5	4	61	%66	%92	%88
Sad	6	7	2	63	%75	%90	%88
Surprised	8	5	1	64	%88	%92	%92

3.5.3 Facial Expression Recognition on TRECVID2006 Dataset

We have used Haar-like features for face detection on each RKF and NRKF keyframes of shots. Face regions are cropped and stored in different location for further processing. In addition, we have developed a web based manual video annotation system that is used to evaluate the face detection performance as shown in Figure 3.28.

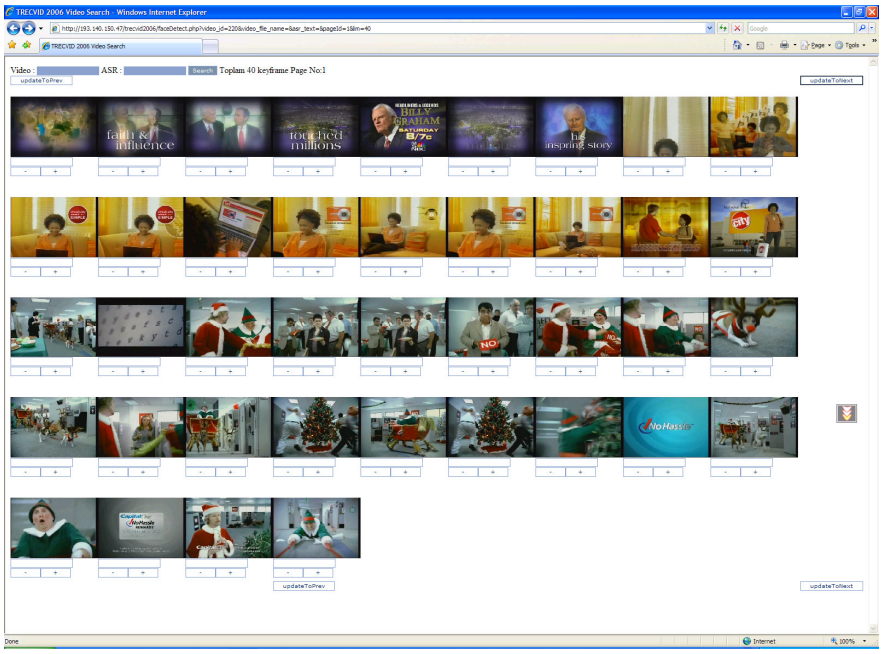

Figure 3.28 Web based manual face detection system

After that, we have used curve fitting base emotional expression recognition method to find the emotion class and intensity as shown in Figure 3.29.

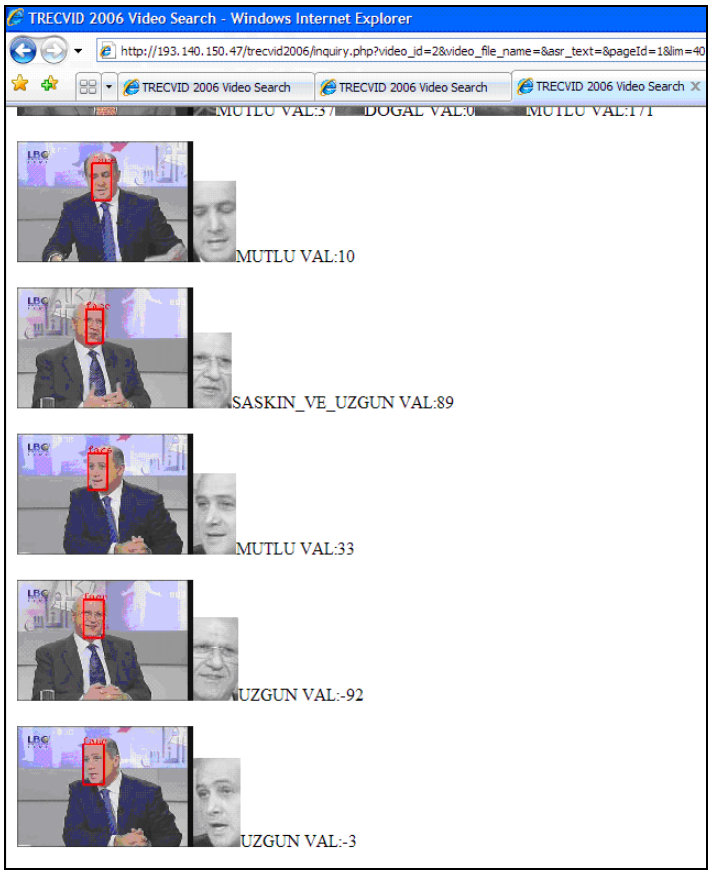

Figure 3.29 Automatic face detection & emotional expression recognition

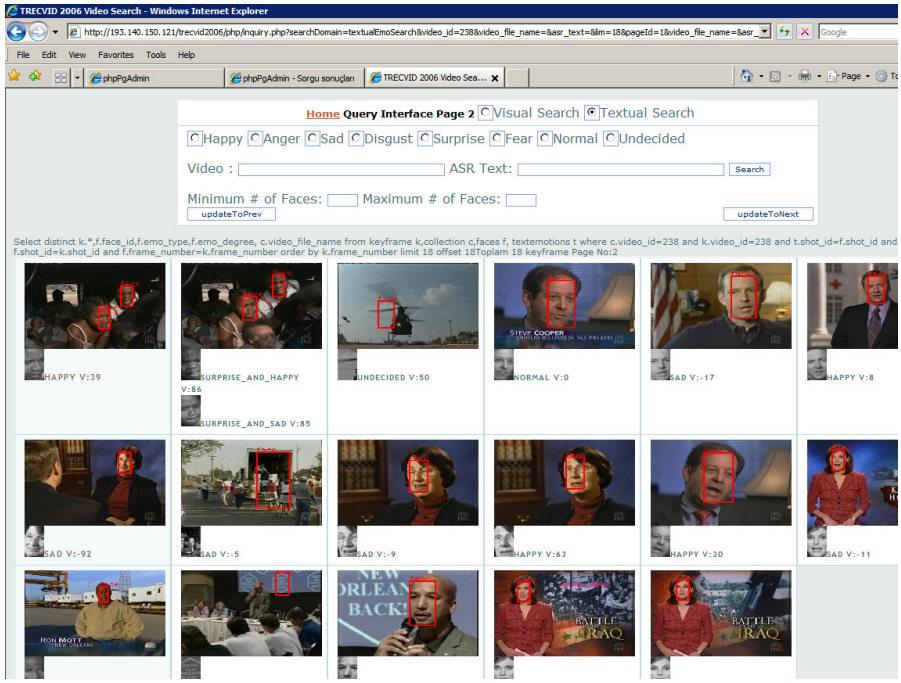

Figure 3.30 Video browsing & searching

3.6 Summary

In this section, we presented a new facial expression recognition algorithm based on curve fitting method for frontal upright faces in still images. Proposed algorithm considers the shape of mouth region to recognize happy, sad and surprise emotions. According to our experiments, our method achieves 89% average accuracy for AMP dataset as shown in Table 3.6.

In addition to still images, we studied on facial expression recognition in video. First, we studied for SBD on TRECVID2006 dataset. We have used RKF and NRKF frames to perform face detection for a given shot. However, facial expression recognition experiments on TRECVID2006 dataset showed that curve fitting based approach highly dependent to pose of face.

Finally, we also presented a visual query generator for performing existence, positional and emotional queries in still images. Then we implement a web-based interface to search faces within TRECVID2006 videos having specific emotions as seen in Figure 3.30.

CHAPTER FOUR
SPEECH BASED EMOTION RECOGNITION

Emotional speech recognition is classifying speech segment into a set of predetermined emotional classes. A variety of computer systems can use emotional speech classification including call center applications, psychology and emotion enabled Text to Speech (TTS) engines. Current studies on emotion recognition mainly concentrate on visual modalities, including facial expressions, muscle movements, action units, body movements, etc. However, emotion itself is a multimodal concept and emotion recognition task requires interdisciplinary studies including visual, textual, acoustic, and physiological signal domains.

Speech Based Emotion Recognition (SBER) aims to find predetermined set of emotions in a given speech utterance. Many of the studies in audio domain concentrated on "Who said what?" type of questions. Therefore there are limited number of research exists on understanding the emotions in speech signal. Initial studies on this area started with lie detectors and then moved to call center applications to detect angry customers, safe driver systems to detect angry drivers and emotional Text to Speech Engines (TTS) for emotional reading of text. Auditory modality includes speech, musical, and all other sounds available in video documents. Therefore, another research area is emotion recognition in musical sounds. Because, researches showed that music has influence on listener's emotional state.

We have started with Voice Activity Detection (VAD) and then used Ensemble of Support Vector Machines (SVM) to detect emotional classes of speech utterances.

In this chapter, fully automatic classifications of basic emotions (Angry, Happy, Neutral, Sad, Fear, and Surprise) in speech signal have been studied on EFN, DES and EmoDB dataset. For the classification part, a set of SVM (Vapnik, 1995), classifiers have been used.

One-vs.-all method is used in two different approaches. For the first approach, we have used biased sampling method and for the latter we have divided the training set into equal number of sub-training sets also having equal number of positive and negative samples for training purposes. For each positive set, we have provided non-overlapped set of negative examples. We tested this approach against traditional one-vs.-all method and experiments showed that using ensemble of SVM classifiers learned from small sets of training examples gives better results than traditional one-vs.-all method.

In addition, we have created a new multimodal emotional speech corpus called EFN (Emotional Finding Nemo). This dataset has prepared using 2054 utterances from the famous animation movie "Finding Nemo" as a participation of six person's common sense. For the creation of the corpus, we have developed a speech and emotion annotation tool in Matlab, which processes a subtitle file and finds speech segments. Experimenters are allowed to select one of the seven distinct emotional classes (Happy, Anger, Sad, Disgust, Surprised, Fear, Normal and Undecided) and emotional intensity of a given speech signal.

After that, we have used ensemble of Support Vector Machine classifiers on speech data where a set of classifiers using different training sets and then we have combined the predictions of multiple classifiers to decide on the final decision. By using this approach, we also eliminated the problem of biasing towards the class having large training size in case of unbalanced set of training examples. In addition, ensembles of classifiers may able to learn more complex concepts than a single classifier does. Experimental studies showed that we achieved 69.5% accuracy on five classes (Anger, Happy, Neutral, Sad, and Fear) emotion classification on this new corpus.

Although it seems to be easy to understand for a human to detect the emotional class of an audio signal, researches showed that, average score of identifying five different emotional classes (neutral state, surprise, happiness, sadness and anger) is between 56-85%, (global average is 67% and kappa statistic is 0.59) (Engberg & Hansen, 1996). On the other hand, without emotional clues, it is difficult to

understand exact meaning of spoken words. Words are followed by punctuation characters like “?” “!” “...” in textual domain which makes easy to understand the meaning of the text. On the other hand understanding the context from linguistic information is limited in some cases. In this case prosodic features of speech signal carries paralinguistic clues about the physical and emotional state of the human.

Emotional Speech Classification is not a trivial task and requires a set of successive operations such as VAD (Voice Activity Detection), feature extraction, training, and finally classification. Aim of the Emotions has a great role in human-to-human communication.

The contribution in this chapter, about emotional speech classification is two-fold: First, we present an approach for emotion classification of speech utterances based on ensemble of support vector machines. It considers feature level fusion of the MFCC, total energy and F0 as input feature vectors, and uses bagging method to ensemble of SVM classifiers.

The second, we present a new emotional dataset based on a popular animation film, Finding Nemo where emotions are much emphasized to attract attention of spectators. We concentrated on perceived emotion in video therefore 2054 utterances from 24 speakers were annotated by a group of volunteers based on seven emotion categories, and we selected 250 utterances each for training and test sets. Our approach is tested on both newly developed dataset as well as two publically available data sets, and the results are promising with respect to current state-of-the-art.

4.1 Feature Extraction

The feature vector we used to represent the emotional speech in our approach, aims to preserve the information needed to determine the emotional content of such a signal. First, each speech utterance was segmented into ~46ms frames (512samples) with a ~23ms overlap area (256 samples) with next frame. As a feature vector, we have used 30 MFCC, total energy, and F0 formant values calculated from each frame

and combined them as seen in equation (8). We assume that each frame from an utterance represents the same emotional state. In order to calculate MFCC coefficients, we have used Matlab Audio Toolbox (Pampalk, 2004), with 30-bin Mel Filter Bank. Then, a set of MFCC vectors, K_{MFCC} shown in (8) , is prepared for each utterance.

$$K_{MFCC} = \begin{bmatrix} C_{1,1} & C_{2,1} & . & . & . & C_{n-1,1} & C_{n,1} \\ C_{1,2} & C_{2,2} & . & . & . & C_{n-1,2} & C_{n,1} \\ . & . & . & . & . & . & . \\ C_{1,m} & C_{2,m} & . & . & . & C_{n-1,m} & C_{n,m} \end{bmatrix} \quad (8)$$

Similarly, F0 and total energy values are computed. At the end of the feature extraction phase, each speech frame was represented with 32-bin feature vector, containing *MFCC*, total energy and F0 value.

4.1.1 Voice Activity Detection

We studied speech-vs.-nonspeech segmentation of audio signals in video. In addition, we proposed an automatic method to create large-scale datasets for the need of supervised learning algorithms.

The purpose of this section is two-fold: First, we obtained 90.33% accuracy for non-speech classification and the second is propose of an automatic method, which uses existing subtitle information to find the approximate position of speech utterances in a given audio signal.

In this section, we proposed a new approach for speech vs. nonspeech segmentation of audio signals and achieved an overall accuracy of 87.77% and 90.33% recall value for speech and non-speech classes.

We presented a speech vs. non-speech segmentation of audio signals extracted from video. We used 4330 seconds of audio signal extracted from "Lost[1]" TV series

[1] 'Lost' is a registered trademark of MCA / Universal Studios. The 'Lost' logo and all images from the television series are copyright MCA / Universal Studios unless otherwise stated; music is copyright the original composers and producers;

for training and 7545 seconds of audio signal from “Lost” and “How I Met Your Mother” TV Series. Our training set is automatically build by using timestamp information exists in subtitles. After that, silence areas within those speech areas are discarded with a further study. Then, standard deviation of MFCC feature vectors of size 20 has been obtained. Finally, Support Vector Machines (SVM) is used with one-vs.-all method for the classification.

For VAD we have used a SVM classifier trained on standard deviation of MFCC feature vectors extracted from 400ms intervals. Training dataset is provided by extracting and labeling audio signal from popular TV series by using the subtitle information as described in (Danisman & Alpkocak, 2007).

We have used different combinations of parameters and features for our SVM classifiers in order to measure the positive or negative effect of these parameters on the classifier performance.

4.1.1.1 Automatic Creation of Large-Scale Dataset

Audio, stored in video has many different formats. One of the best known video format is MPEG-2. Because of the wide use of DVD (MPEG-2) format, there exists corresponding increase in the number of studies on this format. MPEG-2 standard have a DVD Video Object (VOB) carrier, which have subtitle data, consists of speech dialogues and temporal information. The standard that is less costly than dubbing technique therefore it brings us together the wide use of subtitles in video.

Subtitles stored as bitmap format in VOB carrier is translated into textual format for easy access by using Optical Character Recognition (OCR) systems. Table 4.1 shows an example subtitle file content.

Table 4.1 Sample subtitle content

6 00:00:32,799 --> 00:00:35,632 So... is there anything? 7 00:00:35,702 --> 00:00:38,398 - No. - Okay.

Subtitle information shows spoken content in textual format on screen for an appropriate time to allow watchers reading the text. In order to create training sets, we used season 1 episodes 2 & 3 from "Lost" TV series. For the test set, we selected episodes 4,5, and 6 from the Lost and season 2 episode 1 from How I Met Your Mother TV series. First, mplayer used to extract audio signal from video files (Gereffy, 2005), then, speech segments automatically annotated by considering start and end points in subtitle information as show in Figure 4.1 where two speech utterance is surrounded by three silence parts.

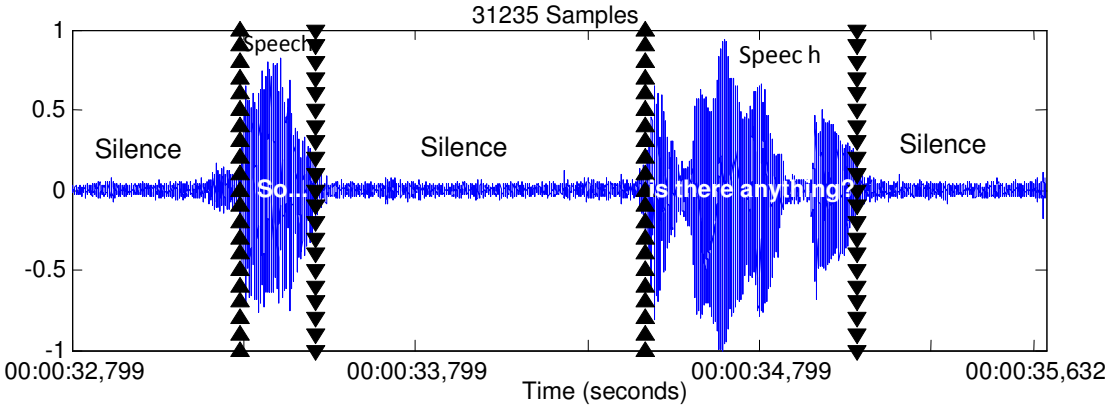

Figure 4.1 Voice activity detection (VAD)

T_s used as a threshold value for silence parts and, T_d used for determining the maximum temporal distance between two discrete utterances. In our experiments fs represents the sampling frequency where fs=11,025Hz, T_d=fs/2, and T_s =±0.15. After that, boundaries of the "Speech", "Non-Speech", and "Silence" parts created automatically as shown in Figure 4.2.

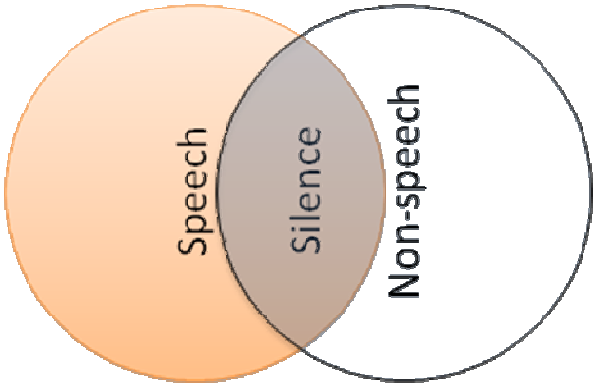

Figure 4.2 Speech, silence, and nonspeech audio classes

4.1.1.2 Feature Extraction for VAD

For emotion recognition, we have used standard deviation of a set of MFCC values extracted from 500ms of windows using the SVM classifier (Joachims, 1999). We have used 30-bin Mel Filter Bank using Matlab Audio Toolbox (Pampalk, 2004) for MFCC coefficient computations. Here each window generates a set of MFCC vectors KMFCC (9) consisting of $C_{i,j}$ column vectors. In addition to MFCC values, three statistical values of F0 formant values (minF0, maxF0, and meanF0) and energy of the signal from these windows are used.

$$K_{MFCC} = \begin{bmatrix} C_{1,1} & C_{2,1} & \cdot & \cdot & \cdot & \cdot & C_{n-1,1} & C_{n,1} \\ C_{1,2} & C_{2,2} & \cdot & \cdot & \cdot & \cdot & C_{n-1,2} & C_{n,1} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ C_{1,m} & C_{2,m} & \cdot & \cdot & \cdot & \cdot & C_{n-1,m} & C_{n,m} \end{bmatrix} \quad (9)$$

After computing mean of each row of the K_{MFCC} matrix, we obtained the $\overline{C_i}$ (10) vector.

$$\overline{C_i} = \frac{1}{n}\sum_{j=1}^{n} C_{j,i} \quad (10)$$

Then we computed the standard deviation of K_{MFCC} matrix represented by $MFCC_{std}$ using (11).

$$MFCC_{std} = \left(\frac{1}{n-1} \sum_{j=1}^{n} \left(C_{j,i} - \overline{C_i} \right)^2 \right)^{\frac{1}{2}} \quad (11)$$

For each emotion class, corresponding $MFCC_{std}$ vectors are provided to the SVM classifier for training. Each number in Table 4.2 shows the number of MFCCstd feature vectors extracted from 500ms windows in training set. As a result, total length of the training set is 1,243 seconds for speech and 3,087 seconds for non-speech classes.

Table 4.2 VAD training set

	Number of Feature Vectors in Training Set	
Audio file	Speech (vector)	Non-speech (vector)
Lost 1x2	1055	3058
Lost 1x3	1431	3116
Total	2486	6174

Corresponding test set shown in Table 4.3 where the total length is 2,527 seconds for speech and 5,018 seconds for non-speech.

Table 4.3 VAD test set

	Number of Feature Vectors in Test Set	
Audio File	Speech (vector)	Non-Speech (vector)
LOST S1×E4	1675	2620
LOST S1×E5	1083	3288
LOST S1×E6	1120	3414
HIMYM S2xE1	1176	714
Total	5054	10036

4.1.1.3 SVM Classification

We used support vector machines SVM (Vapnik, 1995), which is a supervised learning algorithm that tries to map the input feature space having known positive and negative samples into high dimensional space using kernel functions where a hyperplane maximizes the data separation as shown in Figure 4.3. Points labeled with 1,2,3,4,5 are members of positive class while points 6,7,8,9,10,11 are members

of negative class. Horizontal and vertical axises represent corresponding feature vector values. In this figure, color filled points create the support vectors and parallel lines create the hyperplane.

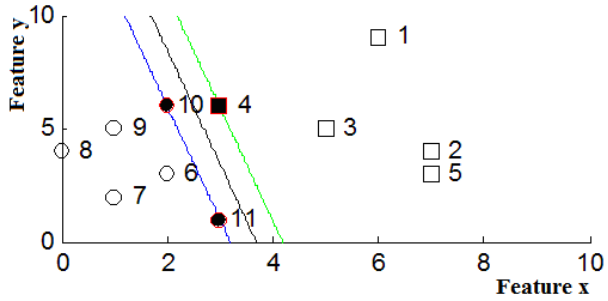

Figure 4.3 SVM classification using linear kernel

In our VAD experiments we selected the linear kernel for its fast learning speed with cost value, C, as 0.004.

4.1.1.4 VAD Experimental Results

In case of VAD, comparison of the results is difficult as there is no certain definition of the minimum addressable unit. Usually small windows of length 10-50ms called frames are used for the comparison. On the other hand, we used frames of length 500ms for fast and accurate classification.

We tested our method on a dataset shown in Table 4.3 of length 7545 seconds. According to our results, we achieved 87.77% accuracy for speech and 90.33% accuracy for non-speech classes.

We considered accuracy, precision, recall, F-measure and ROC values for the evaluation of the system. Table 4.4 shows the confusion matrix for VAD task and Table 4.5 shows evaluation results of the study.

Table 4.4 Confusion matrix for VAD task

		PREDICTED	
		Speech (+1)	Non-Speech(-1)
ACTUAL	Speech(+1)	4179 (TP)	875 (FN)
	Non-Speech (-1)	970 (FP)	9066 (TN)

In case of F-measure (12), α value is selected as 0.5

$$F-measure_{\alpha}=\frac{precision\times recall}{(1-\alpha)\times precision+(\alpha\times recall)} \tag{12}$$

Table 4.5 Evaluation criteria for VAD task

	Precision	Recall	F-measure	Accuracy
Speech	%81.16	%82.68	%81.92	%87.77
Non-Speech	%91.19	%90.33	%90.75	

Receiver Operating Characteristics (ROC) value found as (0.096, 0.82) which is very close to the perfect classifier point (0,1) as shown in Figure 4.4.

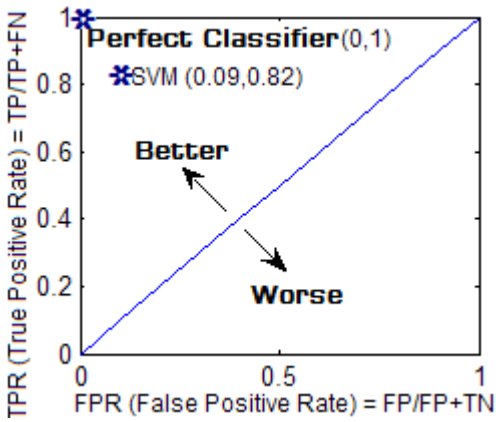

Figure 4.4 SVM classifier performance on ROC space

4.2 Ensemble of Support Vector Machines

In training phase, we used support vector machines (SVM). Since SVM is primarily a dichotomy classifier, we have used one-vs.-all method where the numbers of positive and negative samples are not equal. However, having such a distribution made the classifier biasing to negative class, as expected. Moreover, for an *m*-class classifier, it is a general problem since there exist *m*-1 negative samples for each positive sample. Consequently, the results biased toward to the majority class.

To overcome the biasing problem, first, we divided the negative samples into smaller parts equal to the size of the positive samples. However this fragmentations arises another problem of ensemble of classifiers. On the other hand, literature (Zhou, Wu, & Tang, 2002) shows that generalization ability of ensemble of classifiers has a better performance than a single learner. In literature, many solutions have been suggested for this problem, such as boosting, bagging, or *k*-fold partitioning. We choose bagging, Bootstrap aggregating, method (Breiman, 1996) to overcome aforementioned difficulties. Bagging is useful especially when the classifier gives unstable results in response to small changes such as speaker changes in the training data. In order to do this, we first manipulate our training samples, where we have provided a non-overlapping set of negative examples for each positive set, as seen in Figure 4.5.

Let us assume that we have equal size of samples in our training set for *m* classes. For a given positive class, C_i^+, normally there exist *m*-1 negative classes. In order to create sub-training sets including equal positive and negative samples, we have divided each negative class samples, C_i^-, into sub sets ($S_{i,j}$, where $\forall i,j, 1 \le i \le m, S_{i,j}=m-1$ for balanced sets) from $C_{i,1}^-$ to $C_{i,m-1}^-$. For each negative emotion class, C_i^-, there is a corresponding negative subsets, $C_{i,j}^-$, that need to be merged to create $'C_i^-$ with the same size as positive samples. This finalizes the creation of final training sets, as seen in Table 4.10, Table 4.11, and Table 4.12. These sets are then used for the creation of emotion model $\underline{EM}_{ij}$ where *i* represent emotion and *j* represents the sub

model of respective emotional class. Therefore total number of emotion models EM_{ij} in this approach is $m \times (m-1)$ for balanced sets.

If the number of samples for each emotion class is not equal then, the value of $S_{i,j}$ depends on the size of the C_i^+ and C_i^- and can be computed by (13).

$$S_{i,j} = \left(\sum_{i=1, i \in C^-}^{m} size(C_i^-) \right) / size(C_i^+) \tag{13}$$

After preparing a set of Emotional Model, EM_{ij}, we performed a cumulative addition on the floating SVM predictions. In other words, we sum up the classification results (i.e., prediction values) of the frame belonging to a specific utterance.

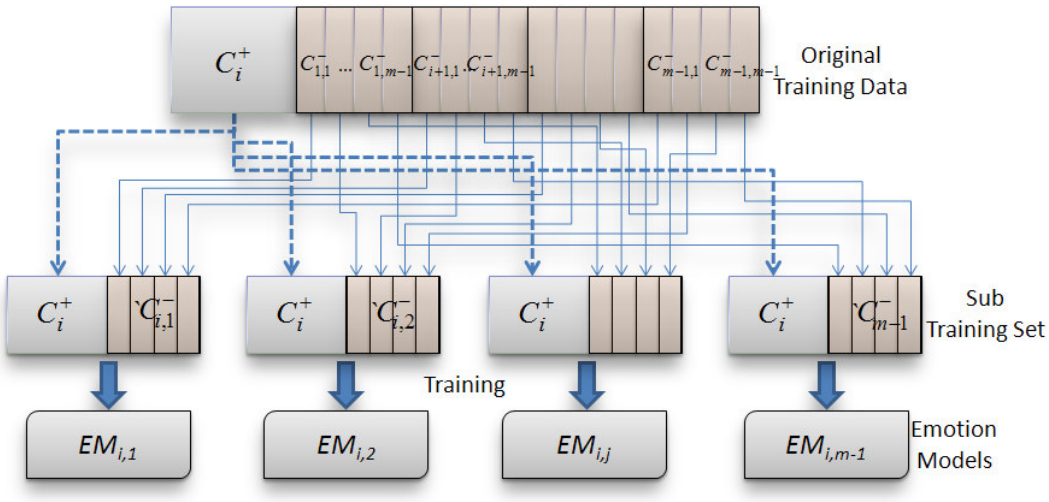

Figure 4.5 Partitioning data into equal positive and negative subsets for m=5

Considering a supervised learning algorithm, it receives a set of training samples, TS={$(x_1,y_1),\ldots,(x_n,y_n)$} where n is the number of samples in the training set and each x_i represents the feature vector in a form of $<x_{i,1},x_{i,2},\ldots,x_{i,k}>$, where each $x_{i,j}$ is a real valued component of x_i. Similarly, our training set at utterance level is represented by a set of samples, TSU={$(U_1, y_x), (U_2, y_z),\ldots,(U_n,y_s)$} where $y_i \in y=\{1,2,\ldots,m\}$ is its multiclass emotion label.

As the original implementation of SVM proposes a dichotomy classifier, we defined a function $f : U \to Y$ which maps an utterance U to an emotion label $f(U)$. Each utterance U_i consists of a set of features vectors x_i, represented by $U_{i,j}$, the number of feature vectors for a given utterance U_i is represented by size(U_i) and the number of models for a given emotion, and EM_i is represented by size(EM_i). Each EM_i has equal weights and for a given test sample U, the binary SVM classifier outputs an m-vector $f(U)=(f_1(U), f_2(U),\ldots, f_m(U))$ as shown in (14).

$$f_i(U) = \sum_{j=1}^{size(EM_i)} \sum_{k=1}^{size(U_i)} EM_{ij}(U_{i,k}) \tag{14}$$

Finally, classifier selects the maximum of $f_i(U)$ as result, and assigns the corresponding class label using (15).

$$f(U) = \arg\max_i f_i(U) \tag{15}$$

4.3 Experimentations

In this section we discuss the details and experiences on constructing a new emotional dataset activity and the details of experimentations we conducted in order to evaluate our approach. We have used SVM^{Light} (Joachims, 1999) as our classifier with linear and RBF kernels are selected for their fast learning speeds and efficiency respectively.

We have used 260 utterances from DES dataset, 535 utterances from EmoDB dataset, and 500 utterances from EFN dataset. Our experiments on EFN and other datasets showed that SVM classifiers gives acceptable performance on basic (five-class) emotion classification.

4.3.1 Emotional Speech Datasets

There are many different emotion sets exist in literature covering basic emotions, universal emotions, primary and secondary emotions, and neutral vs. emotional. According to the latest review by (Ververidis & Kotropoulos, 2006), there are 64 emotion related datasets exists. Many of the datasets (54%) are simulated, 51% percent are in English, and 20% are in German Language. In addition, 73% percent of the datasets are compiled for emotion recognition purposes while 25% percent is for speech synthesis.

There are several emotional speech datasets publically available for experimental studies such as DES (Danish Emotional Speech Database) and EmoDB (Berlin Database of Emotional Speech). Average length of the utterances in EmoDB and DES dataset is about 2.77 seconds and 3.9 seconds respectively. If we ignore long passages from DES dataset, these numbers decreases to 1.08 seconds. These datasets does not have enough utterance samples for efficient training and testing purposes. Studies on those datasets usually measured using cross validation technique. In addition, these datasets are recorded under silent and noise-free conditions and there is only one speaker speaks at a time. These assumptions increase the classifier performances only on similar datasets having similar properties. On the other hand, in real world and even in simple video files these conditions are not available most of the time. Therefore, it is a necessity for us to create a new emotional speech database having multiple speakers and background noise. For this purpose, we selected to use animation movies as a starting point. We found that animation movies are usually targeted the children which makes them having speech segments with high emotional intensity. Therefore, we selected to use the animation movie Finding Nemo[2].

In addition, a new emotional multi-speaker speech database EFN (Emotional Finding Nemo) has been created using the original English audio stream from the famous animation movie “Finding Nemo”. The difference between the EFN and other emotional speech datasets is that annotated utterances in EFN include natural background noise to meet real world situations. EFN consists of 2054 utterances

[2] © Finding Nemo is a Copyright owned by Walt Disney and Pixar Entertainment

automatically extracted from movie's audio stream using subtitle information. Total of 6 person have been annotated each utterance in EFN by considering seven emotional states (basic emotions plus Fear and undecided states) with a degree between 1 to 5. We considered feature level fusion of standard deviation of the MFCC coefficients and F0 values (min, max, mean) as input feature vectors to the ensemble of SVM classifiers.

4.3.1.1 Danish Emotional Speech Database (DES)

Danish Emotional Speech Database DES (Engberg & Hansen, 1996) is in Danish Language, and it consists of 260 emotional utterances, including 2 single words ('Yes',' No'), 9 sentences and 2 long passages, recorded from two female and two male actors. Each actor speaks under five different emotional states including anger, happiness, neutral state, sadness, and surprise.

Utterances were recorded under silent condition in mono channel, sampled at 20 KHz with 16-bit, and only one person speaks at a time. Average length of the utterances in DES dataset is about 3.9 seconds, 1.08 seconds when long passages ignored. Table 4.6 shows the details of DES, such as the number of utterances and total length per emotion for both training and test set.

Table 4.6 Properties of DES dataset, where #U and #FN indicates, number of utterances and number of feature vectors, respectively.

	Positive Samples			Negative Samples		Number of
Emotion	#U	Length(sec.)	#FV	#U	#FV	Subsets
Anger	52	192.9	2222	208	9423	4
Happiness	52	207.4	2494	208	9151	3
Neutral	52	207.3	2370	208	9275	3
Sadness	52	223.3	2196	208	9449	4
Surprise	52	205.1	2363	208	9282	3
TOTAL	260	1036.0	11645	1040	46580	17

To date, many of studies on this subject employed on DES dataset, and Table 4.7 provides a quick snapshot of them. Zervas et al. (2006) and Datcu & Rothkrantz (2005) achieved better accuracy than human based evaluation (Engberg & Hansen,

1996) using Instance Based Learning (IBL) and GentleBoost algorithms respectively. Baseline accuracy is computed by classifying all the utterances as the major emotional class in test set. According to Engberg & Hansen (1996), 67% of the emotions are correctly identified by humans on average on DES dataset. Sedaaghi et al. (2007) used sequential floating feature selection (SFFS) for optimizing correct classification rate of Bayes Classifier on DES dataset and get 48.91% accuracy, in average. Le et al. (2004) achieved 55% accuracy for speaker independent study. Their speaker dependent result is between 70% and 80%.

Table 4.7 Performance of past studies on DES dataset in terms of accuracy.

Study	Classifier	# of Classes	Accuracy %
Baseline	-	5	20.0
Datcu & Rothkrantz (2005)	GentleBoost	5	72.0
Hammal et al. (2005)	Bayes Classifier	5	53.8
Human Eval. (Engberg & Hansen (1996)	-	5	67
Le et al. (2004)	Vector Quantification	5	55.0
Sedaaghi et al. (2007)	Bayes + SFFS + Genetic Alg.	5	48.9
Shami & Verhelst (2007)	ADA-C4.5+ AIBO approach	5	64.1
Shami & Verhelst (2007)	ADA-C4.5+ SBA approach	5	59.7
Ververidis, & Kotropoulos (2004)	Bayes+SFS	5	51.6
Zervas et al. (2006)	C4.5	5	66.0
Zervas et al. (2006)	Instance Based Learning	5	72.9

4.3.1.2 Berlin Database of Emotional Speech (EmoDB)

EmoDB (Burkhardt, Paeschke, Rolfes, Sendlmeier, & Weiss, 2005) dataset is another popular and publically available emotional dataset. It is in German Language, and consists of 535 emotional utterances recorded from 5 female and 5 male actors. Each actor speaks at most 10 different sentences in 7 different emotions anger, happiness, neutral, sadness, boredom, disgust, and fear. Files are 16-bit PCM, mono channel; sampled at 16Khz. Total length of the 535 utterances in the dataset is 1487 seconds and average utterance length is about 2.77 seconds. Table 4.8 shows properties of EmoDB training and test set in detail.

Table 4.8 Properties of EmoDB dataset, where #U and #FN indicates, number of utterances and number of feature vectors and number of subsets, respectively.

	Positive Samples			Negative Samples		Number of Subsets
Emotion	#U	Length (sec.)	#FV	#U	#FV	
Angry	127	335.3	6076	408	22178	3
Happiness	71	154.2	3385	464	24869	7
Neutral	79	154.1	3613	456	24641	6
Sadness	62	186.3	4896	473	23358	4
Boredom	81	180.6	4348	454	23906	5
Disgust	46	251.2	3013	489	25241	8
Fear	69	225.0	2923	466	25331	8
TOTAL	535	1487	28254	3210	169524	41

Table 4.9 presents squeezed comparison of studies held on EmoDB dataset in terms of classifier type, number of classes and accuracy. As in studies on DES, (Datcu & Rothkrantz, 2005) again used GentleBoost algorithm on EmoDB dataset for six emotion classes out of seven, and achieved 86.3% accuracy. (Altun & Polat, 2007) used SVM for four class emotion classification, (Lugger & Yang, 2007) used linear discriminant analyses for anger, happiness, sadness, and neutral emotions and they reported 81.8% accuracy. Additionally, they have tested Bayes classifier, and achieved 74.4% accuracy for six classes using leave-one-speaker-out method on short utterances. Gender dependent study from (Zhongzhe et al., 2006) achieved 77.3% accuracy for female subjects considering seven classes.

Table 4.9 Previous studies on EmoDB dataset in terms of accuracy %.

Study	**Classifier**	**# of Classes**	**Accuracy %**
Altun & Polat (2007)	SVM	4	85.5
Baseline		7	23.7
Datcu & Rothkrantz (2005)	GentleBoost	6	86.3
Human Eval. Burkhardt et al. (2005)		7	86.0
Lugger & Yang (2007)	Bayes Classifier	6	74.4
Lugger & Yang (2007)	Linear Discriminant Analyses	4	81.8
Shami & Verhelst (2007)	SVM+ AIBO approach	7	75.5
Shami & Verhelst (2007)	SVM+ SBA Approach	7	65.5
Zhongzhe et al. (2006)	Two-Stage NN	7	77.3

4.3.1.3 Emotional Finding Nemo Dataset (EFN)

We need to have sufficient large-scale, multi-speaker, multi-emotional, multi-utterance emotional speech databases to train the machine learning classifiers. Unfortunately there are limited number of freely available databases exists and they do not have many of these properties. For this reason, we have developed an emotional dataset directly extracted from video of popular animation film of Finding Nemo, and called EFN.

Main reason for selecting an animation movie is that, animation movies usually targets the children's attention by using music, dancing and high intensity of emotions which makes them help to understand the content. Firstly, EFN dataset is in English, and the utterances were extracted directly from video audio channel including all background music, noise etc, which make it closer to meet real world situations in terms of perceived emotions. It contains 2054 utterances from 24 speakers. Boundaries of utterances are determined considering continuous speech.

Publically available datasets, DES and EmoDB, includes utterances recorded under silent conditions, and only one person speaks at a time. This is mostly not a case for a real world application. For example, for a given video fragment, speech utterances rarely come with a silent background. Additionally, the number of utterance samples in both the DES and EmoDB is not enough for an efficient training and testing. Because of the lack of the small sized dataset, studies on those datasets usually measured using cross validation technique. Consequently, we need a more realistic dataset, which fulfils the real world requirements for video.

The database is constructed using the interface provided by the "Emotional Speech Annotator" application implemented in Matlab as seen in Figure 4.6.

4.3.1.4 Emotion Annotation Tool for Speech

Emotion annotation tool is implemented in matlab and the interface is as in Figure 4.6. The tool loads a movie audio file and processes it according to the timestamp information as described in previous section. Each of six emotion class has an

intensity level scale between 1 and 5. Experimenters are allowed to listen an utterance more than once and they are able to change the type and intensity level of their choices at any time. During this process they are not provided the textual information, therefore, they only listens the speech part and give decisions according to speech modality.

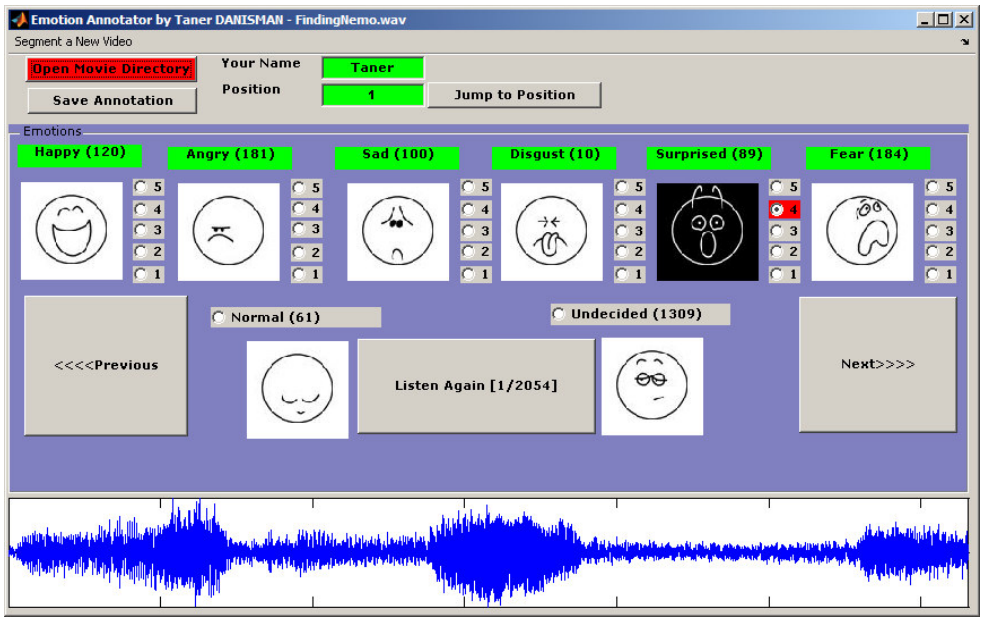

Figure 4.6 Emotional speech annotator

Output of the annotator is an m×n matrix representing an emotional state in each row where n=2. First column represents the emotion id (between 1-8) and second column represents the intensity of the corresponding emotion.

Boundaries of the utterances were extracted using the timestamp information exists in subtitles and voice activity detection (VAD) is used to find presence of speech signal as described in (Danisman & Alpkocak, 2007). Each utterance then further processed to trim silent parts from its beginning and ending. The average, minimum and maximum speech utterance length in EFN is 1.85, 0.5 and 6.1 seconds, respectively.

A total of seven persons, from our department, whose secondary language is English were participated in the experiment. Participants were instructed first to

classify each of 2054 utterance of length 3802 seconds (63.38 minutes) in a forced choice procedure choosing one among the seven emotion classes in addition to undecided class using the Emotional Speech Annotator. Default choice is set to the undecided class. For each utterance except normal and undecided classes, there are five different intensity levels exist. Level 1 represents the least intensity and level 5 represents the highest intensity for the emotion. Participants are able to listen to any utterance at any time. Correction in previous decisions is also possible.

In addition to the speech processing, original subtitle file, which holds the utterance texts, is processed. Because of this operation, we have automatically created a new subtitle file. Original file has 1956 utterances while the new file has 2054 utterances because we split some of the long utterances having long silent parts into two half.

Classifying utterances into a number of emotion classes is alone difficult task, as there is no clear cut between emotional classes. Assigning an emotion label to the utterances may change from annotator to annotator. In order to overcome this problem, we have selected best representative and consistent annotations having high intensity values. After that top ranked emotions for each emotion class are selected for training and testing. For experimental studies, we selected utterances having more than 71.4% accuracy only. In other words, we have chosen utterances, where at least five out of seven participants agreed. We did not include disgust emotion since there are too few samples in this class.

4.3.2 Training and Test Sets

Table 4.10 and Table 4.11 shows total number of feature vectors in each subset per emotion class used in training process for each of the datasets.

Table 4.10 Number of feature vectors in DES training and test sets

	Training Set		
Emotion	Positive	Negative	# of subsets
Angry	1963	9216	4
Happy	2385	8794	3
Neutral	2331	8848	3
Sad	2263	8916	3
Surprise	2237	8942	3

Table 4.11 Number of feature vectors in EmoDB training and test sets

	Training Set		
Emotion	Positive	Negative	# of subsets
Angry	4056	15182	3
Happy	2267	16971	7
Sad	3330	15908	4
Boredom	3083	16155	5
Disgust	1974	17264	8
Fear	2005	17233	8
Neutral	2523	16715	6

Table 4.12 shows the details of EFN dataset, including feature vectors in each emotion model used in training process. The total number of samples for both training and test set contains 250 different speech utterances. Table 5 also includes training and testing times of our experimentations.

Table 4.12 Number of utterances, subsets and corresponding feature vectors in EFN training and test set. #U=Number of Utterances, #FV=Number of feature vectors, #SS=Number of subsets

	Training Set					Test Set	
	(+)Samples		(-) Samples		#SS	Samples	
Emotion	#U	#FV	#U	#FV		#U	#FV
Angry	50	2009	200	7070	3	50	1964
Happiness	50	2178	200	6901	3	50	2342
Neutral	50	1174	200	7905	4	50	1728
Sadness	50	1877	200	7202	4	50	1952
Surprise	50	1841	200	7238	4	50	1603
TOTAL	250	9079	200	36316	18	250	9589

For all experimentations, we assumed that the smallest measurement unit is utterance and sampling rate of the all audio files is converted to 11025Hz and mono channel. Since the number of emotion classes is not equal in DES, EmoDB and EFN

datasets, we have performed different experimentations in terms of number of classes using SVM^{Light} with linear and RBF kernels. For the linear kernel we selected the cost factor *Cost*=0.001 and for the RBF kernel the *gamma* and *Cost* values are selected as 9.0-e005 and 6.0, respectively.

For the DES and EmoDB, we achieved 67.6% and 63.5% accuracy using RBF kernel as seen in Table 4.13 and Table 4.16, where reported human based evaluations are 67% Table 4.13 (Engberg &Hansen, 1996) for DES, and 86% (Burkhardt et al., 2005) for EmoDB respectively. Experiments show that Surprise-Happiness and Neutral-Sadness couples are most confused emotion classes as seen in Table 4.13, as in previous studies. In addition, in terms of computation time, RBF kernel method is more expensive as than linear kernel. However, its performance is better than the linear kernel for appropriate parameters.

4.3.3 Results on DES

We achieved an overall accuracy of 67.6% for five class emotional speech classification on DES dataset which is quite better than previous studies (51.6% (Ververidis & Kotropoulos, 2004), (53.8% Hammal et al. 2005).

Table 4.13 Confusion matrix using ensembles of SVM RBF kernel on DES, gamma=9.0e-5, Cost=6 vs. Human based evaluation (Engberg & Hansen,1996)

Predicted⇒ / ⇓Actual	Accuracy in %, Overall=67.6% using ensembles									
	Anger		Happiness		Neutral		Sadness		Surprise	
	RBF	Human	RBF	Human	RBF	Human	RBF	Human	RBF	Human
Anger	82	60.8	12	2.6	0	0.1	2	31.7	4	4.8
Happiness	20	10.0	70	59.1	0	28.7	2	1.0	8	1.3
Neutral	14	8.3	2	29.8	58	56.4	24	1.7	2	3.8
Sadness	4	12.6	0	1.8	16	0.1	78	85.2	2	0.3
Surprise	12	10.2	36	8.5	0	4.5	2	1.7	50	75.1

Table 4.14 Confusion matrix from (Hammal et a., 2005) for Bayes Classifier on DES, Avg= 53.8%

	Neutral	Surprise	Happy	Sad	Anger
Neutral	46.7	23.9	12.2	3,3	13.7
Surprise	20.1	51.6	6.5	5	16.6
Happy	7.1	5	56.6	24,6	65
Sad	4.5	3.1	28.7	61,8	1.6
Anger	12.5	29.1	4.2	1,8	52.2

Table 4.15 Confusion matrix from Ververidis & Kotropoulos (2004) for Bayes Classifier on DES, Avg=51.6%

	Neutral	Surprise	Happy	Sad	Anger
Neutral	51	15	2	28	4
Surprise	5	64	7	9	14
Happy	9	24	36	13	18
Sad	17	6	2	70	5
Anger	12	19	26	12	31

4.3.4 Results on EmoDB

For Emo-DB we achieved 47.17% for seven class emotional classification.

Table 4.16 Confusion matrix using ensembles of SVM on EmoDB, gamma=9.0e-5, C=6

Predicted⇒ / ⇓Actual	Accuracy in %, Overall=63.5% using ensembles						
	Anger	Happiness	Neutral	Sadness	Boredom	Disgust	Fear
Anger	90.0	5.8	0.0	0.0	0.0	0.8	3.3
Happiness	30.0	47.1	1.4	0.0	0.0	12.9	8.6
Neutral	0.0	0.0	60.0	1.4	32.9	0.0	5.7
Sadness	0.0	0.0	16.7	71.7	11.7	0.0	0.0
Boredom	1.3	0.0	43.8	8.8	40.0	3.8	2.5
Disgust	5.0	0.0	12.5	0.0	5.0	72.5	5.0
Fear	5.0	6.7	15.0	1.7	1.7	6.7	63.3

4.3.5 Results on EFN

We achieved 66.8% average accuracy on EFN dataset for five-class emotion classification and 77.5% for four-class emotion classification using RBF kernel. Table 4.17 shows the experimental results for different set of emotion classes.

Table 4.17 Emotional speech classification on EFN test set for different set of emotion class

	Emotion Classes						Performance		
	Anger	Happiness	Neutral	Sadness	Fear	Surprise	Linear Kernel	RBF Kernel	RBF Kappa
Four Classes Test	×	×		×	×		66.5%	77.5%	0.67
Four Classes Test	×	×	×	×			63.5%	69.0%	0.58
Five Classes Test	×	×	×	×	×		60.4%	66.8%	0.58
Six Classes Test	×	×	×	×	×	×	52.3%	61.3%	0.53

4.3.5.1 Four Class Classification Results (anger, happy, sadness, fear)

For four emotional classes (i.e., anger, happiness, sadness, and fear) emotion classification using RBF kernel we get 77.5% accuracy with kappa value of 0.67

substantial agreements as shown in Table 4.18. Results on Table 4.18 shows the confusion matrix using linear and RBF kernels trained on EFN training data and corresponding results on EFN test data.

Table 4.18 Confusion matrix using EFN trained linear kernel on EFN test set, Cost=5.0e-5 and RBF kernel gamma=9.0e-5, c=6, Anger, Happy, Sad, Fear

Prediction ⇒ Actual ⇓	RESPONSE of Linear kernel in % Average=66.5				RESPONSE of RBF kernel in % Average=77.5			
	Anger	Happy	Sad	Fear	Anger	Happy	Sad	Fear
Anger	64	12	12	12	74	12	6	8
Happy	22	52	18	8	20	70	4	6
Sad	6	16	78	0	8	8	84	0
Fear	16	10	2	72	6	18	2	74

4.3.5.2 Four Class Classification Results (anger, happy, neutral, sad)

Table 4.19 shows the confusion matrixes showing the performance of linear and RBF kernels on the test set for Anger, Happy, Neutral, and Fear emotions.

Table 4.19 Confusion matrix using EFN trained linear kernel on EFN test set, Cost=5.0e-5 and RBF kernel gamma=9.0e-5, c=6, Anger, Happy, Neutral, Sad

Prediction ⇒ Actual ⇓	RESPONSE of Linear kernel in % Average=63.5				RESPONSE of RBF kernel in % Average=69			
	Anger	Happy	Neutral	Sad	Anger	Happy	Neutral	Sad
Anger	56	26	16	2	62	12	22	4
Happy	8	72	12	8	12	78	8	2
Neutral	2	16	66	16	8	12	68	12
Sad	4	12	24	60	8	10	14	68

4.3.5.3 Five Class Classification Results (anger, happy, neutral, sadness, fear)

Table 4.20 shows the results we obtained from our experiments using RBF and linear kernels on EFN dataset for five emotional classes. Overall accuracy we achieved is 66.8% with kappa=0.58 for five emotional classes (i.e., anger, happiness, neutral state, sadness, and fear) using RBF kernel.

Table 4.20 Confusion matrix using EFN trained linear and RBF kernels on EFN test set, Cost=1.0e-3 for linear kernel, gamma=9.0-e005, Cost=6 for RBF kernel

Predicted⇒ / ⇓Actual	Accuracy in %, Overall=66.8% with RBF kernel									
	Anger		Happiness		Neutral		Sadness		Fear	
	Lin.	RBF	Lin.	RBF	Lin.	RBF	Lin.	RBF	Lin.	RBF
Anger	56	58	22	14	14	18	4	8	4	2
Happiness	10	12	58	72	12	8	12	2	8	6
Neutral	2	4	18	14	58	68	22	14	0	0
Sadness	4	4	12	8	18	14	66	74	0	0
Fear	22	18	12	18	0	0	2	2	64	62

4.3.5.4 Six Classes Classification Results (anger, happy, neutral, sadness, fear, surprise)

For six classes (i.e., anger, happiness, neutral state, sadness, fear, and surprise) we get 61.3% (kappa=0.53) and 52.3% (kappa=0.42) accuracy for RBF kernel and linear kernel, respectively. Table 4.21 shows the confusion matrixes showing the performance of linear and RBF kernels on the test set for anger, happy, neutral, sadness, fear, and surprise emotions.

Table 4.21 Confusion matrix using EFN trained linear kernel on EFN test set, Cost=5.0e-5 and RBF kernel gamma=9.0e-5, c=6, Anger, Happy, Neutral, Sad, Fear, and Surprise

Prediction ⇒ / Actual ⇓	RESPONSE of Linear kernel in % Average=52.33						RESPONSE of RBF kernel in % Average=61.33					
	Anger	Happy	Neutral	Sad	Fear	Surprise	Anger	Happy	Neutral	Sad	Fear	Surprise
Anger	52	20	10	2	14	2	64	10	14	0	8	4
Happy	12	56	10	4	10	8	14	62	6	2	6	10
Neutral	4	18	58	12	0	8	8	8	70	4	0	10
Sad	4	12	22	58	2	2	8	8	10	64	0	10
Fear	14	8	0	2	76	0	6	18	0	2	74	0
Surprise	28	30	14	12	2	14	22	18	14	10	2	34

4.4 Summary

We present an approach to emotion recognition of speech utterances that is based on ensembles of SVM classifiers. Since generalization ability of ensemble of classifiers has a better performance than a single learner, we choose bagging method for ensemble of SVM classifiers and considered feature level fusion of the MFCC, total energy, and F0 as input feature vectors.

Additionally, we also present a new emotional dataset based on a popular animation film, Finding Nemo. We choose this film because of utterances in cartoon films are especially exaggerated. We used original English version of film. However, annotated dataset can be easily transformed into other languages since many dubbed version of this film is available.

We tested our approach on our newly developed dataset EFN as well as publically available datasets of DES and EmoDB. Experiments showed that our approach achieved 77.5% and 66.8% overall accuracy for four and five class classification on EFN dataset respectively. In addition, we achieved 67.6% accuracy on DES (five classes) and 63.5% on EmoDB (seven classes) dataset using ensemble of SVM's with 10-fold cross-validation.

Our study showed that, different emotion sets have different classification results. Some emotions have higher recognition rates like anger, sadness, and fear. On the other hand, surprise is the least detected emotion. Furthermore, experiments on EFN, which is based on video audio channel, also showed that background and multi-speaker voices did not affect the performance of the classifiers. We reached up to 77.5% accuracy on four-class emotion classification in EFN dataset. The results we obtained will lead us to new studies on emotional classification of video fragments and can be further improved by using multimodality such as visual, musical, and textual attributes.

CHAPTER FIVE
TEXT BASED EMOTION RECOGNITION

Text seems to be the most studied modality since the text is relatively easier to process than other modalities. Human emotion recognition (HER) from text can be simply envisioned to be a classification problem of a given text according to predefined emotional classes. In this case, it first requires a preparation of proper training set for each emotional class and selection of good features. One of the solutions for this issue is Bag of Word (BoW). It is very similar to keyword spotting (Boucouvalas & Zhe, 2002) and lexical affinity (Valitutti, Strapparava, & Stock, 2004). BoW approach is widely used in information retrieval, and tries to generate a good lexicon for each emotional class and feature extraction. However, creation of emotional lexicon is both time consuming and labor-intensive task since usually requires manual annotations. On the other hand, the number of words in lexicons is very limited, and it is not desired for most classifiers using the BoW approach. Moreover, user's vocabulary may differ from the document vocabulary.

This chapter proposes a new method to recognize emotions in text using Vector Space Model (VSM). In addition, it presents experiments on automatic classification of anger, disgust, fear, joy, and sad emotions in text using International Survey on Emotion Antecedents and Reactions (ISEAR), WordNet-Affect and Wisconsin Perceptual Attribute Rating dataset (WPARD). Finally, describes the effect of the stemming and intensity of emotions on text based emotion classification.

5.1 Affect Sensing

Affect sensing is finding the cognitive structures of emotions from textual content. Before starting on a research on emotion classification, the first question is "Which emotions should be addressed?" There are many different emotion sets exists in the literature including basic emotions, universal emotions, primary and secondary emotions, neutral vs. emotional, and for some cases the problem is reduced to a two class classification problem (Sentiment Analysis) using the Positive and Negative

values as class labels. Simple classification sets give better performance than expanded sets of emotions, which require cognitive information, and deeper understanding of the subject. In our research, we have used five emotion classes (anger, disgust, fear, sad, and joy) that form the intersection between the ISEAR dataset and the SemEval test set. Therefore, the number of emotion classes s=5.

For the classification, we have used Vector Space Model with 801 news headlines provided by "Affective Task" in SemEval 2007 workshop that focuses on classification of emotions and valences in text. We have compared our results with ConceptNet and powerful text based classifiers including Naive Bayes and Support Vector Machines. Our experiments showed that VSM classification gives better performance than ConceptNet, Naive Bayes and SVM based classifiers for emotion recognition in sentences. We achieved an overall F-measure value of 32.22% and kappa value of 0.18 for five class emotional text classification on SemEval dataset which is better than Navie Bayes (28.52%), SVM (28.6%).

WEKA tool (Witten & Frank, 2005) is used for Naïve Bayes and SVM classification and TMG tool (Zeimpekis & Gallopoulos, 2005) is used for VSM classification.

We have tested and discussed the results of classification using cross-validation technique for emotion classification and sentiment analyses on both the ISEAR and SemEval datasets. In order to compare the performance of VSM and other classifiers, we have considered the mean F1-measure value and the kappa statistics that considers the inter-class agreements. In addition to the classification experiments we developed, an emotion enabled video player, which automatically detects the emotion from subtitle text of video and displays corresponding emoticon.

We have used sentences from ISEAR (Scherer & Wallbott, 1994) dataset, emotional words from Wordnet-Affect and polarity of words from WPARD datasets. Our approach uses Vector Space Model for HER. We measured the effect of stemming and emotional intensity on emotion classification in text.

As an initial experiment, we have used the simple set theory to find the set differences of words in ISEAR corpus. Results give hope us to use tf-idf (Term Frequency- Inverse Document Frequency) values for emotion classification because these non-intersected words have very low term frequency values but have high inverse document frequency values. In addition, by using the tf-idf values we are also able to use the words in intersected areas. VSM is widely used in information retrieval especially in document-based retrieval systems where each document has large number of feature vectors representing usually the tfidf values of a given document.

5.2 Set Theory and Emotions

First, we have used set theory, which deals with collections of abstract objects to find the intersections and set differences of objects in a given set. For the graphical simplicity, we only show three emotional classes (anger, disgust, and fear) with a few words in Figure 5.1 where each circle represents an emotional class and entries represent the words.

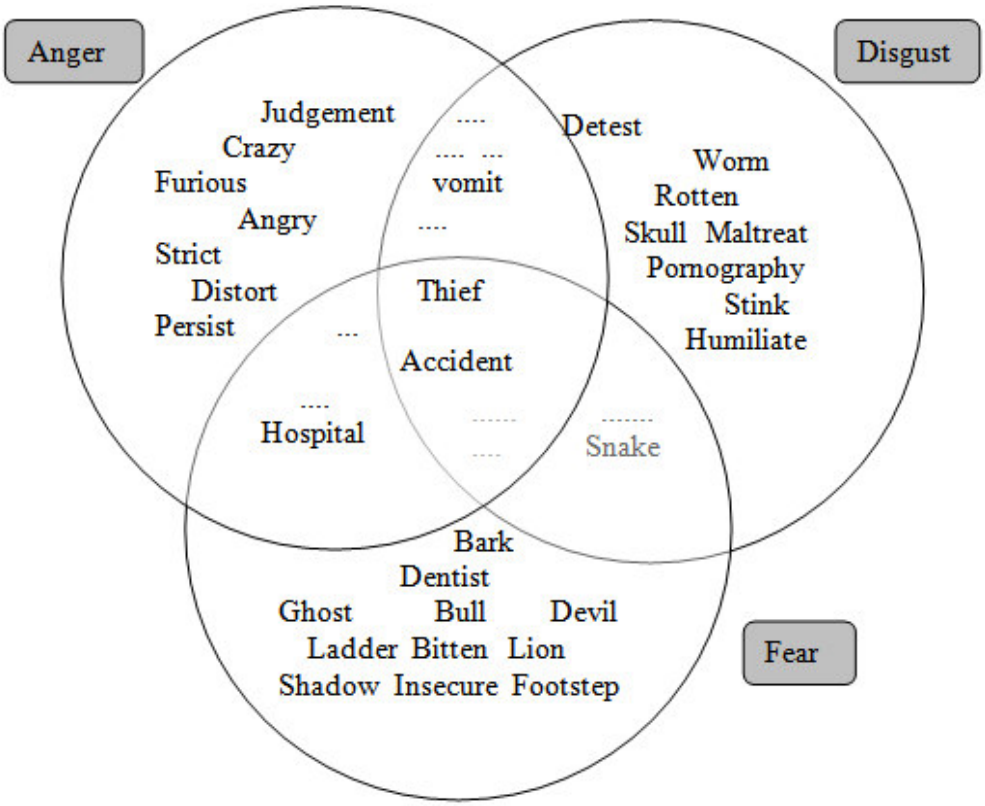

Figure 5.1 Words belonging to specific emotions in ISEAR dataset after finding set differences

Using the set difference words, the word "ghost, bark, dentist and lion" is appeared only in sentences representing the fear emotion whereas "successful, rejoice, sunshine, tremendous, accredit, and carnival" appeared only in joy sentences. The following table represents the automatically extracted emotion words, which are non-intersected words in ISEAR database.

Table 5.1 Emotion related words from ISEAR dataset using set difference for each emotion set.

Anger	Disgust	Fear	Joy	Sad	Shame	Guilt
rules	detest	ghost	happiest	weep	aloud	pinch
strict	worm	bark	joyous	sadden	zip	decline
disregard	rotten	dentist	envelop	coma	infidelity	feign
judgment	skull	devil	success	farewell	underwear	counselor
crazy	hypocrite	ladder	Rejoice	drought	smear	harsh
hatred	stink	bitten	Ecstatic	tragic	dialect	caretaker
persist	humiliate	lion	sunshine	saddest	spell	insinuate
…	…	…	…	…	…	…

5.3 Vector Space Model

Vector Space Model (VSM) is widely used in information retrieval where each document is represented as a vector, and each dimension corresponds to a separate term. If a term occurs in the document then its value in the vector is non-zero. Let us assume that we have *n* distinct terms in our lexicon. Then, lexicon ℓ is represented as a set of ordered terms, and more formally, it is defined as follows:

$$\ell = \{t_1, t_2, t_3, ..., t_n\}$$

Then, an arbitrary document vector, $\vec{d}_i$, is defined as follows:

$$\vec{d}_i = \langle w_{1i}, w_{2i}, ..., w_{ni} \rangle$$

where w_{ki} represents the weight of k^{th} term in document *i*. In literature, there several different ways of computing these weight values have been developed. One of the best known schemes is *tf-idf* weighting. In this scheme, an arbitrary normalized w_{ki} is defined as follows;

$$w_{ki} = c(t_k, d_i) = \frac{tf_{ik} \log(N / n_k)}{\sqrt{\sum_{k=1}^{n} (tf_{ik})^2 [\log(N / n_k)]^2}} \text{ where;}$$

$t_k = k^{\text{th}}$ term in document d_i

tf_{ik} = frequency of the word t_k in document d_i

$idf_k = \log\left(\frac{N}{n_k}\right)$ inverse document frequency of word t_k in entire dataset

n_k = number of documents containing the word t_k,

N = total number of document in the dataset.

More formally an emotion class, M_j, is represented by a set of documents, $M_j=\{d_1,d_2,\ldots,d_c\}$. Then, we have created a model vector for an arbitrary emotion, $\vec{E}_j$, by taking the mean of $\vec{d}_j$ for an arbitrary emotion class. More formally, each $\vec{E}_j$ is computed as follows (16):

$$\vec{E}_j = \frac{1}{\left|M_j\right|} \sum_{\vec{d}_i \in M_j}^{\left|M_j\right|} \vec{d}_i \tag{16}$$

where $\left|M_j\right|$ represents the number of documents in M_j. After preparing model vectors for each emotion class, the whole system is represented with a set of model vectors, $D=\{E_1, E2,..., Es\}$ where s represents the number of distinct emotional classes to be recognized.

In VSM, documents and queries are represented as vectors, and cosine angle between the two vectors used as similarity of them. Then normalized similarity between a given query text, Q, and emotional class, E_j, is defined as follows (17):

$$\text{sim}(Q, E_j) = \sum_{k=1}^{n} w_{kq} \times E_{kj} \tag{17}$$

In order to measure the similarity between a query text and the D matrix of size $s \times n$, first we convert the query text into another matrix $n \times 1$ similar to D where n is the size of the lexicon and s is the number of emotions. Then for each emotion (each row of D matrix), we make multiplication between the query matrix Q and one row of D matrix. After these multiplications, we have m scalar values representing the cosine similarity. The index of the maximum of these values is selected as the final emotional class. More formally, the classification result is then becomes as follows (18):

$$\text{VSM}(Q) = \arg\max_j(\text{sim}(Q, E_j)) \tag{18}$$

The basic hypothesis in using the VSM for classification is the contiguity hypothesis where documents in the same class form a contiguous region, and regions of different classes do not overlap.

5.4 Stop Word Removal Strategy

This study also showed that some words are only appeared in specific sentences belonging to a single emotion, so stop-word removal based on minimum term frequencies is not suitable for emotion recognition. Stop words are usually the most frequent words including articles (a, an, the), auxiliary verbs (be, am, is, are), prepositions (in, on, of, at), conjunctions (and, or, nor, when, while) that do not provide additional improvement for search engines but increase the computational complexity by increasing the size of the dictionary. The important aspect of stop-word removal in emotion recognition is the words, not their frequencies. There are several publically available stop-word lists consisting of approximately 400-500 most frequent words in a given language. However, public stop-word lists consider the information retrieval and they do not consider words carrying emotional content. Therefore we first need to remove some of the emotional words from the stop-word list including negative verbs (not, is not, does not, do not, should not, etc.). Table 5.2 shows the replaced words in order they applied to the text.

In addition, we replaced the word “very” with blank and the word “blank not blank” is replaced by “blank not”. In addition, Words in Table 5.2 are removed from the stop-word list to improve the classification rate. We ignored the part of speech tagging on input text because of its effect of reducing the classification accuracy as described in (Boiy, Hens, Deschacht, & Moens, 2007).

Table 5.2 Modification on input text

Original	Replaced by
very	
isn't	is not
aren't	are not
wasn't	was not
weren't	were not
don't	do not
doesn't	does not
shouldn't	should not
didn't	did not
haven't	have not
hadn't	had not
couldn't	could not
won't	will not
shouldn't	should not
not	NOT
!	XEXCLMARK
?	XQUESMARK

Since non-alpha tokens are automatically removed by TMG, the exclamation marks and question marks are replaced by descriptive new words “XXEXCLMARK” and “XXQUESMARK” respectively. Negative short forms are also replaced by negative long forms such that “doesn’t” is replaced by “does not”. After these replacements, the following sentences are changed as follows:

“I don’t love you!” => “I do not love you XXEXCLMARK” => “I do NOTlove you XXEXCLMARK”

“I am not very happy.” => “I am not happy.” => “I am NOThappy.”

As seen in the above examples, the word “happy” and “love” is used to create new words “NOTlove” and “NOThappy”. In this way, we can discriminate the word “love” having positive meaning and “NOTlove”. In the same way, the new word “NOThappy” has a negative meaning.

5.5 Experimentations

In order to build up the Document matrix for VSM, we used TMG toolbox. First, we made normalizations including limited stop-word elimination, term-length thresholds, which is 3 in our case. We did not consider global and local thresholds. Average number of terms per document before the normalization is 22.43 and after the normalization number of index terms per document is 16 and the dictionary size is 5,966 terms. This result leads us to a D matrix of size 7,466×5,966. As the size of average number of index term elements per document is 16, the D matrix is very sparse. After computing E_j vectors, the new size is 5×5,966.

After the normalization step, we have computed the term frequency and inverse document frequency (tf-idf) values that provide a level of information about the importance of words within the documents. The tf–idf weight is a weight often used in information retrieval and text mining. This weight is a statistical measure used to evaluate how important a word is to a document and how important the word is to all the documents in the collection.

5.5.1 Training and Test Sets

International Survey on Emotion Antecedents and Reactions (ISEAR) (Scherer & Wallbott, 1994) consists of 7666 sentences and snippets. 1096 participants from fields of psychology, social sciences, languages, fine arts, law, natural sciences, engineering and medical in 16 countries across five continents completed a questionnaire about the experiences and reactions to seven emotions in everyday life including joy, fear, anger, sadness, disgust, shame, and guilt. Participants also rated their emotional reactions using a 4-point scale where (1= Not Very, 2=Moderately, 3= Intense and 4= Very Intense). The nonverbal, paralinguistic, verbal, and expressive reactions to the experience is measured using the scales seen in Table 5.3.

Table 5.3 Verbal, non-verbal and paralinguistic activity measures in ISEAR dataset

Type	ISEAR Variable						
Nonverbal Activity	EXPRESS	Laughing Smiling	Crying Sobbing	their facial expression change	Screaming Yelling	other voice changes	changes in gesturing
Paralinguistic activity	PARAL	speech-melody change		speech disturbances		speech tempo change	
verbal activity	VERBAL	Silence		short utterance		one or two sentences	lengthy utterance

Surprisingly, ISEAR dataset is not studied yet for text based emotion classification. Previous studies using the ISEAR dataset try to find relationships among emotions and different cultures, genders, ages, and religions. On the other hand, this corpus is well suited to use for emotional text classification purposes. Table 5.4 shows samples from this dataset for the anger emotion. There are approximately 1100 sample sentences exists each of these emotion classes.

Table 5.4 ISEAR anger samples

"A close person lied to me".
"A colleague asked me for some advice and as he did not have enough confidence in me he asked a third person".
"A colleague asked me to study with her. I could not explain things as perfectly as she had expected. So she reacted in an aggressive manner." ....

For training, we have used combination of ISEAR, Wordnet-Affect and WPARD datasets. Testing is performed on SemEval Task 14 "Affective Text" test set.

Our main training dataset, ISEAR, is further expanded by adding emotional words from Wordnet-Affect (Strapparava & Valitutti, 2004) and Wisconsin Perceptual Attribute Rating Database (WPARD) (Medler, Arnoldussen, Binder, & Seidenberg, 2005) to improve the emotional classification of sentences. Each word in Wordnet-Affect and WPARD is replicated up to average number of terms per

document, which are 16 (as seen on Table 5.6) in our experiment to make ISEAR like sentences. In this case, the sentences are constructed using the same words.

WPARD is like a polarity dataset were collected from 342 undergraduate students using online form to rate how negative or positive were the emotions they associated with each word, using a scale from -6 (very negative feeling) to +6 (very positive feeling), with 0 being a neutral feeling. Table 5.5 shows samples from this dataset.

Table 5.5 Sample cross-section from WPARD (Medler et al., 2005)

Word	Value	Word	Value
rape	-5.60	hope	+4.43
killer	-5.55	honeymoon	+4.48
funeral	-5.47	home	+4.50
slavery	-5.41	sunset	+4.53
cancer	-5.38	beach	+4.58
corpse	-4.95	family	+4.58
slave	-4.84	friend	+4.60
war	-4.78	peace	+4.62
coffin	-4.73	kiss	+4.64
morgue	-4.72	holiday	+4.73
cigarette	-4.49	fun	+4.91

Before extracting the features, we have preprocessed the ISEAR dataset and manually eliminated some of the inconsistent and incomplete entries (such as “No response” lines). Normalization is performed using the TMG toolbox (Zeimpekis & Gallopoulos, 2005) and get the following distribution as seen in Table 5.6.

Table 5.6 Number of sentences per emotion in ISEAR dataset

Emotion	Number of sentences	# of words before stop word removal	Average # of terms before normalization	Average # of terms after normalization
Angry	1,072	26,3	24.8	17.7
Disgust	1,066	22,8	21.6	15.8
Fear	1,080	25,6	23.9	17.1
Joy	1,077	21,1	19.8	14.2
Sad	1,067	21,3	20.2	14.6
Shame	1,052	24,9	23.9	16.9
Surprise	1,053	23,5	22.6	15.9
Average	1,066	23,7	22.4	16.0

SemEval Task 14 “Affective text” test set is used for testing. Table 5.7 shows the sample cross-section in XML format and Table 5.8 shows corresponding ground truth for this test set.

Table 5.7 Sample cross-section from SemEval test set

```
<corpus task=”affective text”>
<instance id="500">Test to predict breast cancer relapse is approved</instance>
<instance id="501">Two Hussein allies are hanged, Iraqi official says</instance>
<instance id="502">Sights and sounds from CES</instance>
<instance id="503">Schuey sees Ferrari unveil new car</instance>
…
```

Table 5.8 Corresponding ground truth data for SemEval test set

Instance Id	Anger	Disgust	Fear	Joy	Sadness	Surprise
500	0	0	15	38	9	11
501	24	26	16	13	38	5
502	0	0	0	17	0	4
503	0	0	0	46	0	31
…	…	…	…	…	…	…

5.5.2 Experiment 1: Affect of Emotional Intensity on Emotion Classification

We studied the effect of emotional intensity to classification performance on the SemEval test set. In our experiment, we have selected emotions having either positive or negative valence value greater than a threshold *T* where *T* is between 0-70. According to Figure **5.2**, F1-Measure value increases proportionally with the *T* when *T* is between 30 to 70. It shows that increased emotional intensity also increases the classification performance.

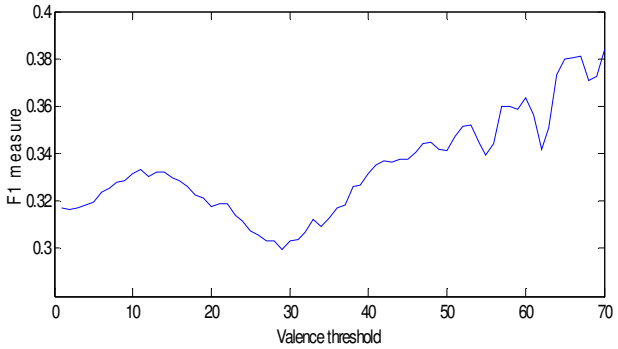

Figure 5.2 Valence threshold versus F1-measure on VSM classifier

5.5.3 Experiment 2: Affect of Stemming on Emotion Classification

A stemmer is an easy to implement algorithm which determines a stem (or morphological root) form of a given inflected (or, sometimes, derived) word form. In some cases, it is known as suffix remover. Stem in linguistics, is the combination of the basic form of a word (called the root) plus any derivational morphemes, but excluding inflectional elements. This means, alternatively, that the stem is the form of the word to which inflectional morphemes can be added, if applicable. A stemmer for English, for example, should identify the string "cats" (and possibly "catlike", "catty" etc.) as based on the root "cat", and "stemmer", "stemming", "stemmed" as based on "stem". Stemmers in linguistic are widely used in search engines and query based systems to improve the efficiency of these systems.

Initially we have used stemming to find morphological root of a given word. For emotion classification, stemming also removes the emotional meaning from the words. We found that some words having the same morphological root can have different emotional meaning. For example, the words “marry” and “love” is frequently shown in joy sentences while the words “married” and “loved” are appeared in sad sentences. So, the tense information also affects the emotional meaning of the words. In spite of this, those samples are very limited. Our experiments showed that use of the stemming algorithms still gives additional increase in classification accuracy as seen in Table 5.9, Table 5.10 and Table 5.11 for Naïve Bayes, SVM and VSM classifiers. Bold values represent the best scores considering three classifiers.

In our experiments, we have used 10-fold cross validation on the ISEAR dataset and test on unseen test set called SemEval.

Table 5.9 Naive Bayes results for five-class emotion classification

Training Set	Stemming	Naïve Bayes Classifier 5 Class Emotional Classification					
		10 Fold cross validation on the ISEAR dataset			Test on SemEval test set		
		Kappa	Mean F1	Accuracy	Kappa	Mean F1	Accuracy
ISEAR	Yes	.59	67.0	67.2	.14	27.1	31.3
	No	.59	67.2	67.4	.09	23.3	26.8
ISEAR+WPARD+ WORDNET_AFFECT	Yes	.51	61.2	60.8	.16	29.0	35.0
	No	.46	57.8	57.0	.12	25.3	30.8

Table 5.10 Support Vector Machine results for five class emotion classification

Training Set	Stemming	Support Vector Machine 5 Class Emotional Classification					
		10 Fold cross validation on the ISEAR dataset			Test on SemEval test set		
		Kappa	Mean F1	Accuracy	Kappa	Mean F1	Accuracy
ISEAR	Yes	.59	67.5	**67.4**	.11	24.5	27.2
	No	.58	67.0	66.9	.09	23.4	26.4
ISEAR+WPARD+ WORDNET_AFFECT	Yes	**.61**	**68.3**	70.2	.12	24.9	27.0
	No	.56	65.0	67.1	.09	23.7	28.0

Table 5.11 Vector Space Model results for five class emotion classification

Training Set	Stemming	Vector Space Model Classifier SemEval test set		
		Kappa	Mean F1	Accuracy
ISEAR	Yes	0.16	28.7	**36.0**
	No	0.11	26.1	32.0
ISEAR+WPARD+ WORDNET_AFFECT	Yes	**0.17**	28.5	34.8
	No	0.11	25.5	32.0

In addition to stemming experiment, we have considered the effect of adding emotional words from Wordnet-Affect and WPARD dataset into our training set. Results showed that, classification performance increased for Naïve Bayes and SVM classifiers but in case of VSM, the performance is reduced and there is only a small

increase in kappa. This is because; we only added the word itself not sentences in our training set. Therefore, during the normalization step, words come from Wordnet-Affect and WPARD behaved like a document, which results a decrease in accuracy as seen in Table 5.10.

5.5.4 Experiment 3: Polarity Test

In this experiment, we only considered positive and negative classes. Therefore, we combined the anger, disgust, fear, and sad emotions in Negative class while joy is the only member of the Positive class. Table 5.12 shows the results of this classification for different classifiers where the best performance for cross-validation comes from SVM classifier with 79.5% F-Measure value and 59.2% with VSM classifier.

Table 5.12 Experimental results for polarity in terms of F-Measure using cross-validation on the ISEAR dataset

Classifier	Test method/set	Positive	Negative	Overall F1
Naïve Bayes	10Fold Cross Validation /ISEAR	64.1	89.9	74.8
Naïve Bayes	SemEval	55.3	60.6	57.8
libSVM	10Fold Cross Validation /ISEAR	69.0	93.8	79.5
libSVM	SemEval	49.9	66.3	56.9
VSM	SemEval	59.1	59.4	59.2

Previous studies achieve up to 42.4% F1-measuse using coarse-grained evaluation for polarity detection on this dataset as reported in (Strapparava & Mihalcea, 2007) while VSM approach achieves 59.2% F1-measure.

For emotion classification, previous studies on this dataset achieves up to 30.3% F1-measure for single class and 11% on average for six-class emotion classification using coarse-grained evaluation. Evaluation criteria of these studies can be found in (Strapparava & Mihalcea, 2007). Our results achieve up to 49.6% F1-measure for single classes and 32.2% on average for five-class emotion classification as seen on Table 5.13.

For the stop word experiment, "English.stop" file from Porter stemmer and "common_words" file from TMG are used. As seen on Table 5.13, almost all best F-Measure (mean of precision and recall) scores come from our classifier with 32.22% value.

In case of ConceptNet, we have used XML-RPC based client to communicate with ConceptNet server. For the evaluation, ConceptNet outputs a prediction vector $P(S) = \langle p_1(S), p_2(S), ..., p_m(S) \rangle$ of size m where S represents a sentence or a snippet, $p_i(S)$ represents prediction value of i^{th} emotion class for the sentence S. Final classification result selects the maximum of $p_i(U)$ and assigns the corresponding class label using

$$P(S) = \arg\max_i \; p_i(S)$$

Table 5.13 Experimental results (in terms of F1-Measure) for emotions trained from ISEAR and tested on SemEval Test set

Classifier / Stop Word	Stop Word	Anger	Disgust	Fear	Joy	Sad	Overall F1
Naïve Bayes	Porter	20.2	5.2	41.9	39.6	32.6	27.9
Naïve Bayes	Tmg	21.5	5.4	42.7	40.5	32.5	28.5
libSVM	Porter	17.7	9.5	39.0	42.7	34.1	28.6
libSVM	Tmg	14.5	8.8	40.0	42.0	33.9	27.8
VSM	Porter	22.1	9.1	40.1	49.2	37.1	31.5
VSM	Tmg	24.2	9.3	41.1	49.6	36.7	32.2
ConceptNet	N/A	7.8	9.8	16.8	49.6	26.3	22.1

5.6 Summary

In this chapter, we presented a VSM approach for HER from text. We measured the effect of emotional intensity on emotion classification in text and showed that VSM based classification on short sentences can be as good as other well-known classifiers including Naïve Bayes, SVM, and ConceptNet.

According to our studies, VSM gives better recall, precision, and F-Measure values than state of the art classification results obtained in SemEval (Semantic Evaluations of News and Headlines Text, June 2007) (Strapparava & Mihalcea,

2007) conference where the test set consists of 800 headlines obtained from BBC and CNN headlines. We also compared our results with ConceptNet and the results of different machine learning algorithms including Naïve Bayes, KNN, and Support Vector Machines using different kernel (linear, polynomial, and radial basis) functions on Weka Data Mining tool (Frank, Hall, Holmes, Kirkby, & Pfahringer et al., 2005).

We also studied the effect of stemming to emotion classification problem. According to our experiments, use of stemming removes and decreases the emotional meaning from words. However, these examples are very rare in our test set therefore use of stemming still increases the classification performance for all classifiers.

CHAPTER SIX
MULTIMODAL EMOTION RECOGNITION MODEL

Significant research has been performed on emotion recognition using unimodal information during the last decade. Today, many of the studies for emotion recognition performed on unimodal features containing only text, speech, or visual information (e.g. face, body gestures). However, recent work targets to use multimodal information for emotion recognition. According to the famous 7% , 38% , & 55% rule of Mehrabian (1968), emotions are conveyed by verbal, vocal and facial features, respectively. Therefore, emotions itself are multimodal in nature.

6.1 Introduction

Multimodal Emotion Recognition (MER) is a collection of interdisciplinary methods for the recognition of human emotion. MER is a new research trend to investigate roles of multiple modalities for EER and obtain better results from the power of multimodality. However, multimodality introduces the fusion problem. When and how should the information combined by other resources? In literature, there exist two different approaches namely early and late fusion. The former merges features in low-level feature space whereas the latter uses semantic output of individual unimodal classifiers as inputs to a higher-level classifier.

First, we compared advantages and disadvantages of early and late fusion scheme, and then explained experimental studies on emotion-aware video player application. In addition, we explained the experiments using late fusion scheme for MER on TRECVID2006 dataset.

6.2 Early vs. Late Fusion

Multimodal analysis and integration introduces feature fusion problem. Fusion is a need for multimodal systems where a decision is required based on outputs of two or more unimodal sources. Some researcher's use the terms early and late fusion

Gunes & Piccardi (2007) whereas others use data, feature and decision level fusion Dasarathy (1997), or semantic fusion Wu, Oviatt, & Cohen (1999).

Early fusion is nothing different from simple machine learning process. Since the fused feature vector size obtained from multiple modalities is usually quite large, it introduces the feature dimensionality problem. On the other hand, scalability of late fusion scheme is better than early fusion scheme because that fused number of classes is much less than fused number of features vectors. Since late fusion is much like interpretation of high-level semantics, the recognition accuracy directly depends on the performance of individual modalities.

Common belief is to use the hierarchical levels for the fusion process where output of a level is used as input for the next level in the hierarchy. This hierarchy is constructed so that semantic complexity increases proportionally with the higher levels. If the different modalities does not complement or support each other then accessing to relevant information requires well-defined assign of feature and fusion weights.

In case of MER, multimodal studies in emotion recognition are different from other domains such that, emotions do not exist for some circumstances or different emotional labels can be assigned for the same temporal section of video for different modalities. Handling of these two problems for such cases is difficult in early fusion schemes where the fusion of feature vectors from different modalities is required and absence of the feature vectors affects the classification performance. In addition, early fusion of features containing different classes of emotions is not suitable for machine learning purposes as the machine learning algorithms designed for classifying similar concepts. On the other hand, absence of features is not a problem for late fusion scheme because that outputs of individual unimodal results are used. At this point, each modality can result different emotion labels for the same video segment.

6.3 Experimentations

We used high level information comes from low level classifiers and employed late fusion technique which is known to be accurate in multimodal integration studies. Main reason for selecting the late fusion technique is its scalability and power on incomplete features. Compared with early fusion scheme, it is much more scalable for high-level fusion as the number of fused classes is much less than number of feature vectors in low-level fusion approaches like early fusion, which makes it more possible to study.

In our MER experiments, we used algorithms presented in Chapter 2, 3, and 4 for visual, aural and textual EER. We considered shots as the minimum emotion addressable unit in video. Emotional class of each shot region is classified considering multiple modalities.

6.3.1 Experiments on TRECVID2006 dataset

Late fusion scheme is used for MER in TRECVID2006 dataset. For each shot region, S_i, normalized classification results come from different modalities are fused in a consecutive manner where speech, face and text based results are represented by;

$$Speech(S_i) = \langle Sp_1(S_i), Sp_2(S_i), ..., Sp_m(S_i) \rangle$$

$$Face(S_i) = \langle Fp_1(S_i), Fp_2(S_i), ..., Fp_m(S_i) \rangle$$

$$Text(S_i) = \langle Tp_1(S_i), Tp_2(S_i), ..., Tp_m(S_i) \rangle$$

$Sp_j(S_i)$, $Fp_j(S_i)$ and $Tp_j(S_i)$ represents normalized classification results for each shot S_i which is between 0-1. Final decision, $D(S_i)$, is computed by using the sum rule as shown in (19):

$$D(S_i) = \langle (Sp_1(S_i) + Fp_1(S_i) + Tp_1(S_i))/3, (Sp_2(S_i) + Fp_2(S_i) + Tp_2(S_i))/3, ..., (Sp_m(S_i) + Fp_m(S_i) + Tp_m(S_i))/3 \rangle \quad (19)$$

After that, emotion having maximum $D(S_i)$ value is selected as final emotion class as shown in (20):

$$P(S) = \arg\max_i D(S_i) \quad (20)$$

According to the experiments, we also developed a web based multimodal emotion search interface as shown in Figure 6.1. In addition, Figure 6.2, Figure 6.3 and Figure 6.4 show the change in emotions in time.

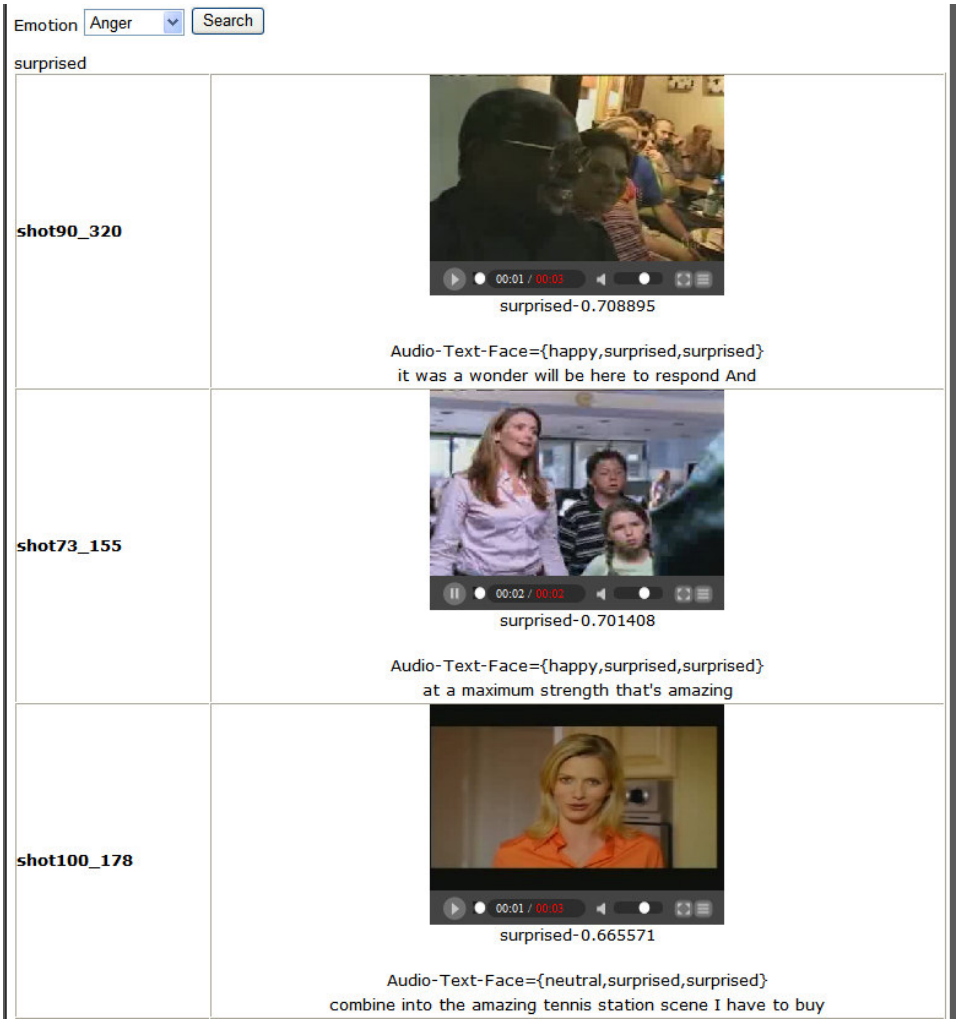

Figure 6.1 Emotion vs. time graph for 20051102_142800_LBC_NAHAR_ARB file from TRECVID2006 dataset

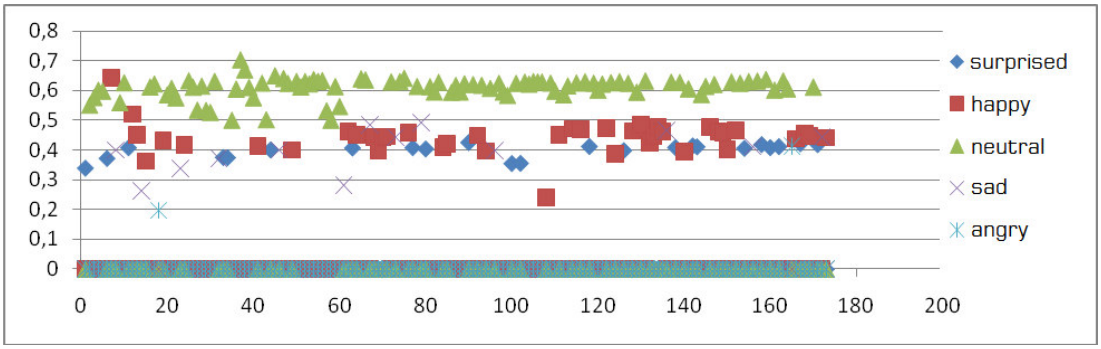

Figure 6.2 Emotion vs. time graph for 20051102_142800_LBC_NAHAR_ARB file from TRECVID2006 dataset

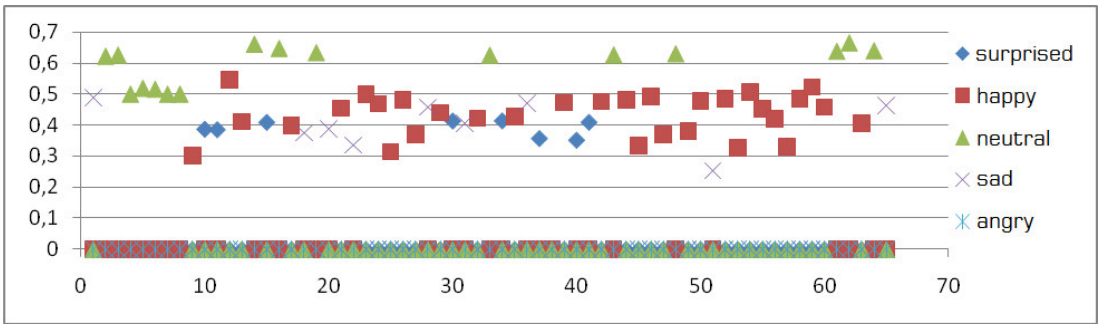

Figure 6.3 Emotion vs. time graph for 20051103_142800_LBC_NAHAR_ARB file from TRECVID2006 dataset

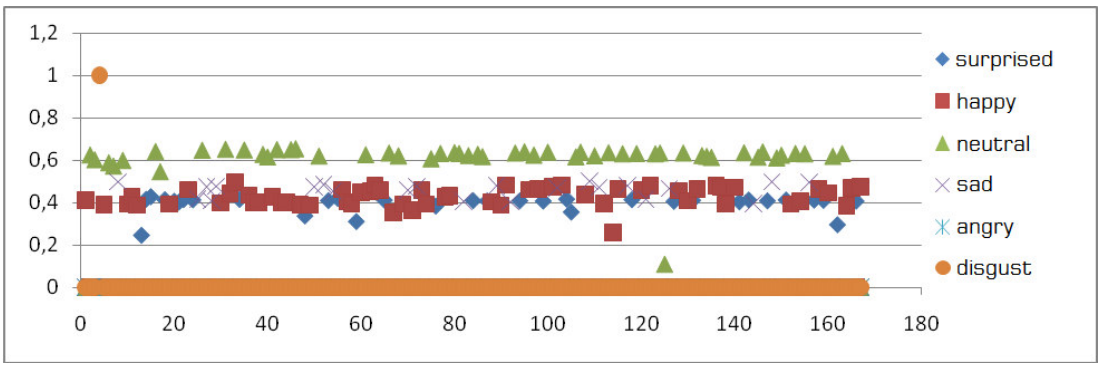

Figure 6.4 Emotion vs. time graph for 20051104_142800_LBC_NAHAR_ARB file from TRECVID2006 dataset

6.3.2 Emotion-aware Video Player

Video is necessary for real life multimodal emotion detection as it contains visual textual and audio information. A simple recorder is enough to obtain the video data and no other special hardware, sensors, wearable clothes are required.

We create a video player, which detects the emotion of subtitle texts, and speech signal using the VSM and SVM classifiers trained on the ISEAR and displays the corresponding emoticon as seen in Figure 6.5. Emotion recognition in speech signal is performed using ensemble of support vector machines. In addition to emotion recognition, it detects the polarity of emotions in a given shot. Positive emotions are represented by blocks in right whereas negative emotions are represented by block in left.

The video player uses a new subtitle format similar like SubRip (SRT) files where we add additional information lines per subtitle. Table 6.14 shows the structure of the new format and Table 6.15 shows an example cross section. First line is the subtitle ID, next line is time information, Speech based ground truth data, and intensity of emotion, text based prediction and its valence respectively. Finally, Figure 6.5 shows the screenshot from the emotion enabled video player.

Table 6.14 Multimodal subtitle structure

```
Subtitle number
Start time --> End time
Emotional Class modality1
Emotional Value modality1
Emotional Class modality2
Emotional Value modality2
Text of subtitle (one or more lines)
Blank line
```

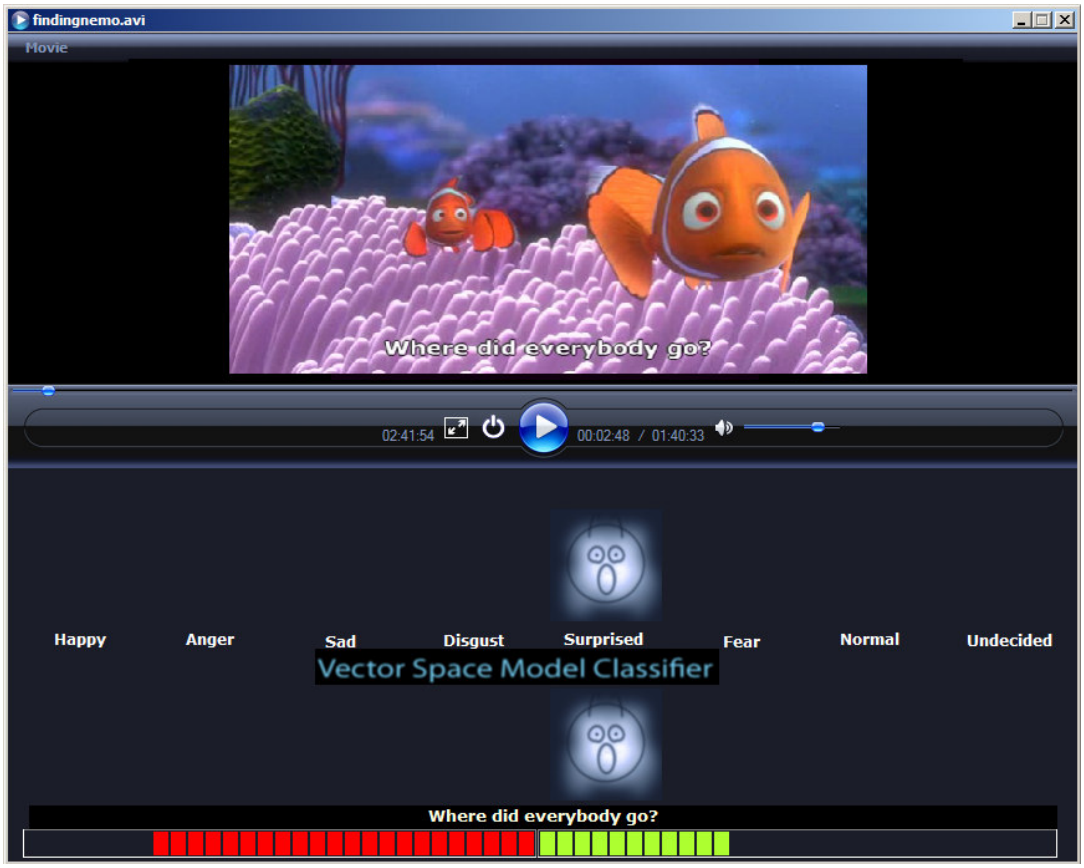

Figure 6.5 Emotion-aware video player screenshot from Finding Nemo[3]

Table 6.15 Sample processed subtitle for Finding Nemo

51 00:02:46,923 --> 00:02:47,667 Surprised 4 Surprised -64 Where did everybody go? 52 00:02:55,898 --> 00:02:57,367 Fear 3 Fear -90 Coral, get inside the house.

[3] © Finding Nemo is a Copyright owned by Walt Disney and Pixar Entertainment

6.4 Summary

Main question in MER is to find how the data coming from different modalities processed. Two approaches are early and late fusion schemes.

Early fusion is a preferred method for multimodal studies. However, early fusion requires complete set of features beforehand, which is not suitable for MER. Similarly, emotions do not exist in some shots. Therefore, we used late fusion scheme that is more suitable to handle situations where features are incomplete or absent. However, final classification results in late fusion scheme highly depend on the performance of individual classifiers. Failure of a single classifier affects the whole results.

In order to decrease the effect of false alarms we employed normalization over individual classification results. In this way, we are able to use single modal, bi-modal, and three-modal results together.

Finally, we developed an emotion-aware video player that is able to display recognized emotions come from different modalities of video during video playing. We are planning to enhance the player by developing intelligent HCI interface for emotional seeking and browsing of video.

CHAPTER SEVEN
CONCLUSIONS

In this thesis, we proposed new approaches to recognize emotions in video using visual, aural, and textual modalities. For each modality, we presented details of proposed methods and experimental results.

Experiments of visual modality showed that, curve fitting based facial expression recognition produces acceptable results for frontal upright images. Approximated shape of mouth is used to recognize happy, sad, and surprise emotions. On the other hand, proposed algorithm fails when applied to video frames where both the quality of image and pose of the face are not suitable with the suggested image processing techniques.

Experiments on video started with SBD task. We proposed a skip frame based approach for efficient SBD on TRECVID2006 dataset. After that, we developed a visual query generator for performing existence, positional and emotional queries in key frames. Finally, we implement a web-based interface to search faces within TRECVID2006 videos having specific emotions.

In aural modality, we present an approach to emotion recognition of speech utterances that based on ensembles of SVM classifiers trained on MFCC, total energy and F0 features. Lack of the availability of emotional speech datasets lead us to create a new emotional speech dataset based on a popular animation film, Finding Nemo. In our experiments, we used original English version of film. However, EFN dataset can be easily transformed into other languages since many dubbed version of this film is available. We tested our approach on EFN, DES and EmoDB datasets and our approach achieved 77.5% and 66.8% overall accuracy for four and five class classification on EFN dataset respectively. In addition, we achieved 67.6% accuracy on DES (five classes) and 63.5% on EmoDB (seven classes) dataset using ensemble of SVM's with 10-fold cross-validation.

According to speech-based experiments, type of emotions affects the classification performance. Some emotions have higher recognition rates than others. For example, anger, sadness, and fear emotions have higher recognition rate than surprise emotion. Furthermore, experiments on EFN showed that background and multi-speaker voices did not affect overall classification performance.

In textual modality, we proposed a VSM approach for HER from text. We measured the effect of emotional intensity and use of stemming on emotion classification in text. According to the experiments, VSM based classification on short sentences can be as good as other well-known classifiers including Naïve Bayes, SVM, and ConceptNet.

Finally, we developed an emotion-aware video player that is able to display recognized emotions come from different modalities of video. In addition, we employed late fusion scheme to combine results come from different modalities. A web based search interface is developed to demonstrate results of singular and combined modalities.

Since EER is a difficult problem, a number of problems remain to be solved. The most important problem is the effectiveness of existing approaches that are too far from human based evaluations. As a future work, we are planning to improve the effectiveness of proposed methods and widen the set of emotions with emotion related events like thinking, excitement, etc.

REFERENCES

Acar, C., Atlas, A., Cevik, K., Olmez, I, Unlu, M., Ozkan, D., & Duygulu, P. (2007). Yuz bulma yontemlerinin haber videolari icin sistematik karsilastirmasi. In Proceedings of IEEE 15. Sinyal İşleme ve İletişim Uygulamaları Kurultayı (SIU), Anadolu Üniversitesi, Eskişehir, Turkey, June 11-13.

Adams, B., Amir, A., Dorai, C., Ghosal, S., Iyengar, G., Jaimes, A., Lang, C., Lin, C., Natsev, A., Naphade, M., Neti, C., Nock, H. J., Permuter, H. H., Singh, R., Smith, C.R., Srinivasan, S., Tseng, B. L., Ashwin, T. V., & Zhang, D. (2002). IBM Research TREC-2002 Video Retrieval System.

Alatan, A.A., Akansu, A.N., & Wolf, W. (2001). Multi-modal dialogue scene detection using hidden markov models for content-based multimedia indexing. *Multimedia Tools and Applications, 14*(2): pp.137-151.

Albanese, M., Chianese, A., Moscato, V. & Sansone, L. (2004). A formal model for video shot segmentation and its application via animate vision. *Multimedia Tools and Applications*, Volume 24, N.3, pp. 253-272.

Alm, C. O., Roth, D., & Sproat, R. (2005). Emotions from text: machine learning for text-based emotion prediction. In Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing, pages 579-586, Vancouver, British Columbia, Canada, Association for Computational Linguistics.

Altun, H. & Polat, G. (2007). New frameworks to boost feature selection algorithms in emotion detection for improved human-computer interaction. (LNCS), Vol. 4729, pp. 533–541, Springer, Heidelberg.

Arijon, D. (1976). Grammar of the film language. Silman-James Press

Arman, F., Hsu, A., & Chiu, M.Y. (1994). Image processing on encoded video sequences. *Multimedia Systems* Vol. 1, No. 5, pp. 211-219.

Arslan, U., Donderler, M., Saykol, E., Ulusoy, O., & Gudukbay, U. (2002). A semi-automatic semantic annotation tool for video databases. In Proceedings of the Workshop on Multimedia Semantics (SOFSEM 2002). Milovy, Czech Republic.

Babaguchi, N., & Nitta, N. (2003). Intermodal Collaboration: a strategy for semantic content analysis for broadcasted sports video. Osaka University, ICIP 2003.

Byrne, W., Beyerlein, P., Huerta, J., Khudanpur, S., Marthi, B., Morgan, J., Peterek, N., Picone, J., Vergyri, D., & Wang, W. (2000). Towards language independent acoustic modeling. *In Proceedings of ICASSP, volume 2*, pages 1029 – 1032.

Boiy, E., Hens, P., Deschacht, K., & Moens, M.F. (2007). Automatic sentiment analysis in on-line text, ELPUB2007. Openness in Digital Publishing: Awareness, Discovery and Access. In Proceedings of the 11th International Conference on Electronic Publishing, Vienna, Austria.

Boreczky, J. S., & Rowe, L.A. (1996). Comparison of video shot boundary detection techniques. In Storage and Retrieval for Still Image and Video Databases IV, Proc. SPIE 2664, pp. 170-179.

Boucouvalas, A. C., & Zhe, X. (2002). Text-to-Emotion engine for real time internet communication. In. Proceedings of the 3rd International Symposium on CSNDSP, Staffordshire University, UK, pp. 164-168.

Breiman, L. (1996). Bagging predictors. *Machine Learning*, vol. 24, no. 2, 123–140.

Burkhardt, F., Paeschke, A., Rolfes, M., Sendlmeier, W., & Weiss, B. (2005). A database of german emotional speech. In Proceedings of INTERSPEECH 2005, ISCA, Lisbon, Portugal, pp.1517–1520.

Calic J., & Izquierdo, E. (2002). A Multiresolution Technique for Video Indexing and Retrieval. In Proceedings of ICIP 2002, Rochester, New York, USA.

Casagrande, N., Eck, D., & Kigl, B. (2005). Frame-level audio feature extraction using AdaBoost. In Proceedings of ISMIR 2005, pp.345-350, London, UK.

Chuang, Z.J., & Wu, H. (2004). Multi-Modal emotion recognition from speech and text. *International Journal of Computational Linguistics and Chinese Language Processing, 9*(2), pp.1-18.

Cohen, I. (2000). Automatic expression recognition from video sequences using temporal information. MSc. Thesis, University of Illions, pp. 8-30.

Colombo, C., Bimbo, A. D., & Pala, P. (1999). Semantics in visual information retrieval. University of Florence , Italy, *IEEE Multimedia*, pages 38-53.

Dailey, M. N., Cottrell, W. C., Padgett, C., & Adolphs, R. (2002). EMPATH: A Neural Network that Categorizes Facial Expressions. *Journal of Cognitive Science, 14*(8), 1158-1173.

Danisman, T., & Alpkocak, A. (2006a). Recognition of facial emotional expression in images. 14th IEEE Signal Processing and Communications Applications Conference 2006, Sabanci University, Antalya, Turkey.

Danisman, T., & Alpkocak, A. (2006b). Dokuz Eylül University video shot boundary detection at Trecvid 2006, In Proceedings of TRECVID 2006.

Danisman, T., & Alpkocak, A. (2007). Speech vs. nonspeech segmentation of audio signals using support vector machines. 15th Signal Processing and Communication Applications Conference 2007, Anadolu University, Eskisehir, Turkey.

Danisman, T., Alpkocak, A. (2008). Emotion Classification of Audio Signals Using Ensemble of Support Vector Machines. 4th IEEE Tutorial and Research Workshop, Perception and Interactive Technologies for Speech-Based Systems, Germany, *Lecture Notes in Artificial Intelligence LNAI*, *Vol: 5078*, pp.205-216, Springer-Verlag Berlin Heidelberg.

Danisman, T., & Alpkocak, A. (2008). Feeler: emotion classification of text using vector space model. In Proceedings of the AISB 2008 Symposium on Affective Language in Human and Machine, University of Aberdeen, Scotland, Volume 2, ISBN: 1-902956-61-3, pp. 53-59.

Dasarathy, B. (1997). Sensor fusion potential exploitation-innovative architectures and illustrative applications. *IEEE Proceedings 85*(1) pp. 24-38.

Datcu, D. & Rothkrantz, L.J.M. (2005). Facial expression recognition with relevance vector machines. IEEE International Conference on Multimedia & Expo (ICME '05), ISBN 0-7803-9332-5.

De Silva L.C. & Ng, P.C. (2000). Bimodal emotion recognition. In Proceedings of IEEE FG pp. 332-333.

Ekman, P. (1993). Facial expression of emotion. *American Psychologist, 48*, 384-392.

Ekman, P. & Friesen, W.V. (1978). Facial action coding system. *Consulting Psychologists Press Inc.*, 577 College Avenue, Palo Alto, California 94306.

Engberg, I. S. & Hansen, A.V. (1996). Documentation of the danish emotional speech database (DES). Internal AAU report, Center for Person Kommunikation, Denmark.

Fischer, S., Lienhart, R., & Effelsberg, W. (1995). Automatic recognition of film genres. Proc. ACM Multimedia 95, San Francisco, CA, pp. 295-304.

Foo, S.W., & Yap, T. (1997). HMM speech recognition with reduced training. International Conference on Information, Communication and Signal Processing ICICS'97, Singapore.

Franco, L., & Treves, A. (1997). A neural network facial expression recognition system using unsupervised local processing. Cognitive Neuroscience Sector, SISSA.

Frank, E., Hall, M. A., Holmes, G., Kirkby, R., Pfahringer, B., Witten, I. H. & Trigg, L. (2005). WEKA - A Machine Learning Workbench for Data Mining. InOded Maimon & Lior Rokach (Eds.), *The Data Mining and Knowledge Discovery Handbook*, pp. 1305-1314.

Fujisaki, H. & Hirose, K. (1984). Analysis of voice fundamental frequency contours for declarative sentences of Japanese. *Journal of the Acoustical Society of Japan 5*(4): 233–242.

Gabbouj, M., Kiranyaz, S., Caglar, K., Cramariuc, B., Cheikh, F.A., Guldogan O., & Karaoglu, E. (2001). MUVIS: A multimedia browsing, indexing and retrieval system, Signal Processing Laboratory, Tampere University of Technology, Tampere-Finland.

Garg, G., Sharma, P. K., Chaudhury, S., & Chowdhury, R. (2002). An appearance based approach for video object extraction and representation. ICPR 2002, pp. 536-539.

Gargi, U., Kasturi, R., & Strayer, S.H. (2000). Performance characterization of video-shot-change detection methods. *Circuits and Systems for Video Technology, IEEE Transactions on, 10*(1):1.

Gereffy, A., (2005). mplayer tool. Retrieved May 2005, from http://www.mplayerhq.hu, Version 1.0

Go, H., Kwak, K., Lee, D. & Chun, M. (2003). Emotion recognition from the facial image and speech signal. In Proceedings of SICE 2003 Annual Conference, Fukui, Japan, vol. 3, pp. 2890-2895.

Gunes H. & Piccardi, M. (2007). Bi-modal emotion recognition from expressive face and body gestures. *Journal of Network and Computer Applications, 30*:1334-1345, 2007.

Guo, J., JongWon, K., & Kuo, C.C.J. (2000). Fast and adaptive semantic object extraction from video. Image and Video Communications and Processing 2000, Proc. SPIE Vol. 3974, pp. 440-451.

Hammal, Z., Bozkurt, B., Couvreur, L., Unay, U., Caplier, A. & Dutoit, T. (2005). Passive versus active: vocal classification system. In Proc. XIII European Signal Processing Conference, Antalya, Turkey.

Hammal, Z., Couvreur, L., Caplier, A. & Rombaut, M. (2007). Facial expression classification: An approach based on the fusion of facial deformations using the transferable belief model. *International Journal of Approximate Reasoning, 46*:542-567.

Huang, J., Liu, Z., Wang, Y., Chen, Y., & Wong, E.K. (1999). Integration of multimodal features for video scene classification based on HMM. In IEEE Workshop on Multimedia Signal Processing, Copenhagen, Denmark.

Isabel, M.-P., Xavier, D., Jérôme, M., & Vincent, D. (2005). Robust human face hiding ensuring privacy. WIAMIS Montreux, Switzerland.

Izquierdo, E., Casas, J. R., Leonardi, R. , Migliorati, P., O'Connor, N. E. , Kompatsiaris I., & Strintzis, M.G. (2003). Advanced content-based semantic scene analysis and information retrieval: the schema project. 4th European Workshop on Image Analysis for Multimedia Interactive Services, WIAMIS 2003, London, UK Apr 2003 981-238-355-7.

Jaimes, A., & Smith, J.R. (2003). Semi-automatic data-driven construction of multimedia ontologies. In Proceedings IEEE Intl. Conf. on Multimedia and Expo. July 2003, Baltimore, MD, USA.

Joachims, T. (1999). *Making Large-Scale SVM Learning Practical. Advances in Kernel Methods - Support Vector Learning*. B. Schölkopf , C. Burges & A. Smola (Eds.), MIT-Press.

Le, X.H., Quenot, G. & Castelli, E. (2004). Speaker-Dependent emotion recognition for audio document indexing. In International Conference on Electronics, Information, and Communications (ICEIC'04).

Lienhart, R. (1999). Comparison of automatic shot boundary detection algorithms. *SPIE Storage and Retrieval for Still Image and Video Databases VII 1999*, Vol. 3656, pp. 290-301.

Lienhart, R. (2001a). Reliable Transition Detection in Videos: A Survey and Practitioner's Guide. *International Journal of Image and Graphics (IJIG)*, Vol.1, No.3, pp.469-486.

Lienhart, R. (2001b). Reliable Dissolve Detection. *In Storage and Retrieval for Media Databases 2001*, Proc. SPIE 4315, pp. 219-230.

Little, T.D.C, Ahanger, G., Folz, R.J., Gibbon, J.F., Reeve, F.W., Schelleng, D.H., & Venkatesh, D. (1993). A digital on- demand video service supporting content-based queries. Proc. ACM Multimedia 93, Anaheim, CA, pp. 427-436.

Liu, H., Lieberman, H., Selker, T. (2003). A model of textual affect sensing using real-world knowledge. In Proceedings International Conference on Intelligent User Interfaces (IUI-03) 125–132.

Liu, H., & Singh, P. (2004). ConceptNet: A Practical Commonsense Reasoning Toolkit. *BT Technology Journal 22*(4):211-226. Kluwer Academic Publishers.

Long, F., Feng, D., Peng, H., & Siu, W. (2001). Extracting Semantic Video Objects. *Computer Graphics and Applications, IEEE*, *21*(1) : 48-55.

Lugger, M. & Yang, B. (2006). Classification of different speaking groups by means of voice quality parameters. ITG-Sprach-Kommunikation.

Lugger, M. & Yang, B. (2007). An incremental analysis of different feature groups in speaker independent emotion recognition. In 16th Int. Congress of Phonetic Sciences.

Martínez, J. M. (Ed.) (2002). *Coding of Moving Pictures and Audio, MPEG-7 Overview. -ISO/IEC JTC1/SC29/WG11N4980*. MPEG Alliance Web Site. http://www.mpeg-industry.com/mp7a/w4980_mp7_Overview1.pdf

Medler, D. A., Arnoldussen, A., Binder, J.R., & Seidenberg, M.S. (2005). The wisconsin perceptual attribute ratings database. Retrieved, December 2007 from http://www.neuro.mcw.edu/ratings/

Meinedo, H. ve Neto, J. (2005). A stream-based audio segmentation, classification and clustering pre-processing system for broadcast news using ANN models. In INTERSPEECH-2005, 237-240.

Miller, G. A. (1995). Wordnet: A lexical database for english. *Communications of the ACM, 38*(11)3941.

Mishne, G. (2005). Experiments with mood classification in blog posts. In Style2005 – 1st Workshop on Stylistic Analysis of Text for Information Access, at SIGIR 2005.

Murray, I. R. & Arnott, J. L. (1993). Towards the Simulation of Emotion in Synthetic Speech: A Review of the Literature of Human Vocal Emotion. *Journal of Acoustic Society of America 93*(2):1097-1198.

Naphade, M.R., Krisljansson, T., Frey, B., & Huang, T. S. (1998). Probabilistic multimedia objects (multijects): a novel approach to video indexing and retrieval in multimedia systems. In proceedings of IEEE International Conference on Image Processing, Volume 3, pages 536-540, Chicago.

Naphade, M.R., Yeung, M. M., & Yeo, B.L. (2000). A novel scheme for fast and efficient video sequence matching using compact signatures. In Proceedings of SPIE Storage and Retrieval of Multimedia Databases, vol 3972, pp 564-572.

Naphade, M.R., Kozintsev, I. V., & Huang, T. S. (2002). A factor graph framework for semantic video indexing. IEEE Transactions on Circuits and Systems for Video Technology, Vol. 12, No. 1, pages 40-52.

Naphade, M.R., & Smith, J.R. (2004). On the detection of semantic concepts at trecvid. Proceedings of the 12th annual ACM international conference on Multimedia, ISBN: 1-58113-893-8, pp: 660- 667, New York.

Pampalk, E. (2004). A matlab toolbox to compute music similarity from audio. In Proceedings of the 5th International Conference on Music Information Retrieval, pp. 254–257.

Pasechke, A. & Sendlmeier, W.F. (2000). Prosodic characteristics of emotional speech: measurements of fundamental frequency movements. In Proceedings of ISCA Workshop on Speech and Emotion, Northern Ireland, pp.75–80.

Pei, S., & Chou, Y. (1999). Efficient MPEG compressed video analysis using macroblock type information. IEEE Transactions on Multimedia, vol.1, no.4, Dec. 1999, pp.321-33.

Pennebaker, J. W., Francis, M. E., & Booth, R. J. (2001). Linguistic Inquiry and Word Count: LIWC 2001. Mahwah, NJ: Lawrence Erlbaum.

Petersohn, C. (2004). Fraunhofer HHI at TRECVID 2004: shot boundary detection system. TREC Video Retrieval Evaluation Online Proceedings, TRECVID 2004.

Picard, R. (1997). *Affective Computing*. Cambridge, MA: MIT Press.

Rasheed, Z., Sheikh, Y., & Shah, M. (2003). Semantic film preview classification using low-level computable features. In 3rd International Workshop on Multimedia Data and Document Engineering (MDDE-2003), Berlin, Germany.

Rautiainen, M., Seppänen, T., Penttilä, J., Peltola, J. (2003). Detecting semantic concepts from video using temporal gradients and audio classification. *Lecture Notes in Computer Science, Volume 2728*, Jan 2003, pp. 260–270.

Salway, A., & Graham, M. (2003). Extracting information about emotions in films. Proceedings of 11th ACM conference on multimedia 2003, 4th-6th Nov. 2003, pp. 299-302. ISBN 1-58113-722-2.

Scherer, K. R., & Wallbott, H.G. (1994). Evidence for universality and cultural variation of differential emotion response patterning. *Journal of Personality and Social Psychology, 66*, 310-328.

Sebe, N., Bakker, E., Cohen, I., Gevers, T., & Huang, T.S. (2005). Bimodal emotion recognition. 5th International Conference on Methods and Techniques in Behavioral Research, Wageningen, The Netherlands.

Sedaaghi, M. H., Kotropoulos, C. & Ververidis, D. (2007). Using adaptive genetic algorithms to improve speech emotion recognition. In IEEE 9th Workshop on Multimedia Signal Processing, MMSP 2007, pp. 461–464.

Sethi, I.K., & Patel, N. (1995). A statistical approach to scene change detection. Proceedings of SPIE Storage and Retrieval for Image and Video Databases III, San Josè, CA, 2420.

Shafran, I. & Rose, R. C. (2003). Robust speech detection and segmentation for real-time asr applications. In Proceedings of International Conference on Acoustics, Speech, and Signal Processing, pp. 432-435.

Shaikh, M., Prendinger, H., & Ishizuka, M. (2006). A cognitively based approach to affect sensing from text. Proceedings of 10th International Conference on Intelligent User Interface (IUI 2006), Sydney, Australia, pp.349-351.

Shaikh, M., Prendinger, H., & Ishizuka, M. (2007a). Emotion sensitive news agent: An approach towards user centric emotion sensing from the news. The 2007 IEEE/WIC/ACM International Conference on Web Intelligence (WI-07), Silicon Valley, USA, pp. 614-620.

Shaikh, M., Prendinger, H., & Ishizuka, M., (2007b). SenseNet: A linguistic tool to visualize numerical-valence based sentiment of textual data (Poster). Proceedings 5th International Conference on Natural Language Processing (ICON-07), Hyderabad, India, pp. 147-152.

Shami, M. & Verhelst, W. (2007). An evaluation of the robustness of existing supervised machine learning approaches to the classification of emotions in speech. *Speech Communication, 49*, nr. 3, 201–212.

Shafran, I. & Rose, R.C. (2003). Robust speech detection and segmentation for real-time ASR applications. In Proceedings of International Conference on Acoustics, Speech, and Sig. Proc., pages 432-435.

Smeulders, A., Worring, M., Santini, S., Gupta, A., & Jain, R. (2000). Content based image retrieval at the end of the early years. IEEE Transactions on Pattern Analysis and Machine Intelligence 22, pages 1349-1380.

Snoek, G.M.C., & Worring, M. (2005). Multimodal video indexing: a review of the state of the art. *Multimedia Tools and Application 25*(1), pp. 5–35.

Strapparava, C. & Mihalcea, R. (2007). SemEval-2007 Task 14: affective text. In Proceedings of the 4th International Workshop on Semantic Evaluations (SemEval-2007), pages 70–74, Prague.

Strapparava C., & Valitutti., A. (2004). WordNet-Affect: an affective extension of WordNet. In Proceedings of the 4th International Conference on Language Resources and Evaluation (LREC 2004), Lisbon, pp. 1083-1086.

Sugimoto F., & Yoneyama, M. (2006). a method for classifying emotion of text based on emotional dictionaries for emotional reading, In Proceedings of the IASTED International Conference on Artificial Intelligence and Applications, Austria, pp. 91-96.

Teodorescu, H.N. & Feraru, M. (2007). A study on speech with manifest emotions. *In Proc. TSD 2007. (LNCS), Vol. 4629*, pp. 254–261, Springer, Heidelberg.

Tran, D. A., Hua, K. A., & Vu, K. (2000). Semantic reasoning based video database systems. Proceedings of the 11th International Conference on Database and Expert Systems Applications, pp. 41-50, September 4-8, 2000, London, England.

Truong, B. T., Dorai, C., & Venkatesh, S. (2000). New enhancements to cut, fade, and dissolve detection processes in video segmentation. *In Proceedings of the 8th ACM International Conference on Multimedia*, pages 219-227.

Valitutti, A., Strapparava, C., & Stock, O. (2004). Developing affective lexical resources. *PsychNology Journal, 2*(1):61-83.

Vandecatseye, A., & Martens, J. P. (2003). A fast, accurate and stream-based speaker segmentation and clustering algorithm. In Proceedings of Eurospeech 2003, Geneva, Switzerland.

Vandecatseye, A., Martens, J.P., & Neto, J. (2004). The cost278pan-european broadcast news database. In Proceedings of the International Conference on Language Resources and Evaluation (LREC '04), pp. 873–876, Lisbon, Portugal.

Vapnik, V. N. (1995). *The Nature of Statistical Learning Theory*. Springer.

Ververidis, D., & Kotropoulos, C. (2004). Automatic speech classification to five emotional states based on gender information. In Proceedings Eusipco, Vienna, pp.341-344.

Ververidis, D., Kotropoulos, C. & Pitas, I. (2004). Automatic emotional speech classification. In Proceedings of International Conference on Acoustics, Speech, and Signal Processing, pp.593–596, Montreal, Canada.

Ververidis, D. & Kotropoulos, C. (2006). Emotional peech recognition: resources, features, and methods. *Speech Communication*, *48*, nr.9:1162–1181.

Visser, R., Sebe, N., & Lew, M.S. (2002). Detecting automobiles and people for semantic video retrieval. In Proceeding of 16th International Conference on Pattern Recognition (ICPR '02),vol. 2, pp. 733–736, Quebec City, Canada.

Wang, P., Ma, Y., Zhang H., & Yang, S. (2000). A people similarity based approach to video indexing. Microsoft Research Asia

Wei-Ying, M., & HongJiang, Z. (2000). An indexing and browsing system for home video. *Hewlett-Packard Laboratories.*

Witten, I. H., & Frank, E. (2005). *Data Mining: Practical machine learning tools and techniques, (2nd Edition)*. Morgan Kaufmann, San Francisco.

Wu, L., Oviatt, S. L., & Cohen, P. R. (1999). multimodal integration – a statistical view. *IEEE Transactions on Multimedia, 1*(4):334-341.

Yeung, M., Yeo, B.L., & Liu, B. (1996). Extracting story units from long programs for video browsing and navigation. In Proceedings of IEEE Conference on Multimedia Computing and Systems.

Yongsheng, Y., & Ming, L. (n.d.) A survey on content based video retrieval. Hong Kong University of Science & Technology, Retrieved May 2008 from http://citeseer.ist.psu.edu/248151.html

Zeimpekis, D., & Gallopoulos, E. (2005). Tmg: A matlab toolbox for generating term-document matrices from text collections. Technical report, University of Patras, Greece.

Zervas, P., Mporas, I., Fakotakis, N. & Kokkinakis, G. (2006). Employing Fujisaki's intonation model parameters for emotion recognition. In Proceedings of 4th Hellinic Conf. Artificial Intelligence (SETN'06), Heraklion, Crete.

Zhang, H.J., Kankanhalli A. & Smoliar S.W. (1993). Automatic Partitioning of Full Motion Video. *Multimedia Systems, 1*, 10-28.

Zhong, D., Zhang, H.J. & Chang, S.F. (1996). Clustering methods for video browsing and annotation. Storage and Retrieval for Still Image and Video Databases IV, IS&T/SPIE's Electronic Imaging: Science & Technology 96 [2670-38].

Zhongzhe, X., Dellandrea, E., Dou, W. & Chen, L. (2006). Two-stage classification of emotional speech. International Conference on Digital Telecommunications, pp.32.

Zhou, Z.H., Wu, J. & Tang, W. (2002). Ensembling neural networks: many could be better than all. *Artificial Intelligence, 137*(1-2) 239–263.

APPENDICES

A. Facial Expression Recognition

```
F64 = imresize(Face,[64 64]);
[Sx Sy Sz] =size(F64);
if(Sz~=1) I=rgb2gray(F64);
else I=F64;
end
Face=[];
Face=I;
Faced=im2double(Face);
bg4=blkproc(Faced,[32 32],'min(x(:))');
bg64=imresize(bg4,[64 64],'bicubic');
H=Faced-bg64;
Face=[];
Face=H;
[FaceX,map] = gray2ind(Face,4);
for(i=1:1:64)
  for(j=1:1:64)
    if(I(i,j)>0) I(i,j)=1;   end
  end
end
DI=I;
BWs = edge(DI, 'sobel','both', (graythresh(DI) * .01));
se180 = strel('line', 1, 180);
se0 = strel('line', 2, 180);
BWlining = imdilate(BWs,[se0 se180]);
BWnobord1 = imclearborder(BWlining, 4);
[SizeX SizeY]=size(BWs);
MinSize=floor(SizeX/4);
BWs2=bwareaopen(BWnobord1,MinSize);
se180 = strel('line', 5, 180);
se0 = strel('line', 1, 0);
BWsdil = imdilate(BWs2,[se0 se180 ]);
BWdfill = imfill(BWsdil, 'holes');
BWnobord = imclearborder(BWdfill, 8);
BWnobord = imclearborder(BWdfill, 8);
seD = strel('line',3,180);
BWfinal = imerode(BWnobord,seD);
BWoutline = bwperim(BWfinal);
[BWLabeled,NumRegions]=bwlabeln(BWoutline);
display(NumRegions);
colored=ind2rgb(BWLabeled+1,[0 0 0 ;jet(NumRegions)]);
BWSkel = bwmorph(BWoutline,'skel',Inf);
BWSpur = bwmorph(BWSkel,'spur',2);
[BWLabeled,NumRegions]=bwlabeln(BWSpur);
for i=1:1:NumRegions
  [Hx,Hy,Hv]=find(BWLabeled==i);
  Ymax=max(Hy);
  Ymin=min(Hy);
```

```
48    XMaxIndex= find(Hy==Ymax(1));
49    XMinIndex= find(Hy==Ymin(1));
50    Xmax=Hx(XMaxIndex);
51    Xmin=Hx(XMinIndex);
52    CornerPoints(i,1)=Xmin(1);
53    CornerPoints(i,2)=Ymin(1);
54    CornerPoints(i,3)=Xmax(1);
55    CornerPoints(i,4)=Ymax(1);
56  end
57  PolDegree=5; Saskin=0;
58  for i=1:1:NumRegions
59    [Hy,Hx,Hv]=find(BWLabeled==i);
60    p = polyfit(Hx,Hy,PolDegree);
61    PolY2 = polyval(p,Hx);
62    plot(Hx,PolY2,'Color','w');   fun='';
63    for j=1:1:PolDegree
64    if j==1  fun=sprintf('(%d.*x.^%d)',double(p(j)),j-1);
65    else  fun=sprintf('(%s)+(%d.*x.^%d)',fun,double(p(j)),j-1);
66    end
67  end
68  [num1 num2]=size(Hx); x1=Hx(1); y1=Hy(1);
69  x2=Hx(num1);  y2=Hy(num1); x=0;
70  Lfunc='(((x-x1)*(y1-y2))/(x1-x2))+y1';
71  y=inline(Lfunc,'x','x1','x2','y1','y2');
72  if((Classes(i,1)==1)&(Classes(i,2)==2))
73   %Mouth area
74   if(i==MaxAreaInd)
75     if(i==MinEccentId) Saskin=1; end
76   end
77   polfunc=inline(fun); dif=0;
78   for(x=x1:1:x2)
79     LineVal=y(x,x1,x2,y1,y2);
80     PolVal=PolY2(x-x1+1);
81     dif=dif+PolVal-LineVal;
82   end
83   Start=Hy(1); Stop=Hy(size(Hx));
84  end
```

B. Abbreviations

AIML	Artificial Intelligence Markup Language
ANN	Artificial Neural Networks
ASR	Automatic Speech Recognition
AU	Action Units
BOW	Bag of Words
CRM	Customer Relationship Management
DCMI	Dublin Core Metadata Initiative
DCT	Discrete cosine transform
DES	Danish Emotional Speech Database
EER	Emotional Expression Recognition
EMODB	Berlin Database of Emotional Speech
EFN	Emotional Finding Nemo
EMG	Electro-Myo-Graphy
FACS	Facial Action Coding System
GMM	Gaussian Mixture Models
HMM	Hidden Markov Models
IBL	Instance Based Learning
ISEAR	International Survey on Emotion Antecedents and Reactions
ISFER	Integrated System for Facial Expression Recognition
KNN	K-Nearest Neighbor
LDA	Linear Discriminant Analysis
MCSVM	Multi-Class Support Vector Machines
MFCC	Mel-frequency Cepstral Coefficients

MLP	Multi Layer Perceptron
MPEG	Moving Picture Experts Group
NLP	Natural Language Processing
NN	Neural Networks
NRKF	Non-Representative Keyframe
OCR	Optical Character Recognition
OMC	Observer Motion Coherence
OMCS	Open Mind Common Sense
OPENCV	Open Computer Vision Library
PCM	Pulse Code Modulation
RBF	Radial Basis Functions
ROC	Receiver Operating Characteristics
PFA	Principal Feature Analysis
PBVD	Piecewise Bezier Volume Deformation
RDF	Resource Description Framework
RKF	Representative Keyframe
RPC	Remote Procedure Call
SBD	Shot Boundary Determination
SBER	Speech Based Emotion Recognition
SDR	Spoken Document Retrieval
SFFS	Sequential Floating Feature Selection
SRG	Semantic Relation Graphs
SVM	Support Vector Machines
SVO	Semantic Video Objects

TBM	Transferable Belief Model
TEO	Teager Energy Operator
TF-IDF	Term Frequency- Inverse Document Frequency
TMG	Text Matrix Generator
TOC	Table of Contents
TREC	Text REtrieval Conference
TTS	Text to Speech
VAD	Voice Activity Detection
VOB	DVD Video Object
VSM	Vector Space Model
WPARD	Wisconsin Perceptual Attribute Rating Database