RECOGNITION OF HUMAN EMOTION IN SPEECH USING 
MODULATION SPECTRAL FEATURES AND SUPPORT 
VECTOR MACHINES

by

SIQING WU

A thesis submitted to the 
Department of Electrical and Computer Engineering 
in conformity with the requirements for 
the degree of Master of Science

Queen’s University 
Kingston, Ontario, Canada 

September 2009

Copyright © Siqing Wu, 2009
Abstract

Automatic recognition of human emotion in speech aims at recognizing the underlying emotional state of a speaker from the speech signal. The area has received rapidly increasing research interest over the past few years. However, designing powerful spectral features for high-performance speech emotion recognition (SER) remains an open challenge. Most spectral features employed in current SER techniques convey short-term spectral properties only while omitting useful long-term temporal modulation information.

In this thesis, modulation spectral features (MSFs) are proposed for SER, with support vector machines used for machine learning. By employing an auditory filterbank and a modulation filterbank for speech analysis, an auditory-inspired long-term spectro-temporal (ST) representation is obtained, which captures both acoustic frequency and temporal modulation frequency components. The MSFs are then extracted from the ST representation, thereby conveying information important for human speech perception but missing from conventional short-term spectral features (STSFs).
Experiments show that the proposed features outperform features based on mel-frequency cepstral coefficients and perceptual linear predictive coefficients, two commonly used STSFs. The MSFs further render a substantial improvement in recognition performance when used to augment the extensively used prosodic features, and recognition accuracy above 90% is accomplished for classifying seven emotion categories. Moreover, the proposed features in combination with prosodic features attain estimation performance comparable to human evaluation for recognizing continuous emotions.
Acknowledgments

First, I would like to thank God for giving me the two wonderful years at Queen’s. He is my shepherd; I shall not be in want.

I have been extremely fortunate to have had Dr. Wai-Yip Geoffrey Chan as my supervisor during my Master’s studies. I sincerely thank him for his precious time and his mentorship. His profound knowledge is invaluable, and his great passion in research has always been an inspiration. Without his guidance, this thesis would have been impossible.

I would like to thank all of my colleagues at Mc²L, especially Dr. Tiago Falk for his valuable suggestions and comments on my research. I am also grateful to Dr. Karen Rudie, Dr. Evelyn Morin, and Dr. Lola Cuddy for serving on my thesis committee. I would also like to express my gratitude to all the professors that have taught me during my study at Queen’s, and all ECE staff members for their kind help whenever needed. And of course, lots of thanks go to my friends. Thank you for making my stay at Kingston enjoyable.

Finally, I wish to thank my dear parents for their unfailing love and support.
Contents

Abstract i
Acknowledgments iii
Contents iv
List of Tables vii
List of Figures viii
List of Acronyms x

1 Introduction 1
  1.1 Motivation .................................................. 1
  1.2 Contributions .............................................. 3
  1.3 Thesis Outline .............................................. 4

2 Speech Emotion Recognition 6
  2.1 Introduction .................................................. 6
  2.2 The Nature of Emotion in Speech ............................ 7
    2.2.1 Describing emotions ..................................... 8
    2.2.2 Producing emotional speech ............................. 10
  2.3 Automatic Speech Emotion Recognition .................... 11
    2.3.1 Emotional speech data .................................. 11
    2.3.2 Features .................................................. 14
    2.3.3 Machine learning algorithms ........................... 20
    2.3.4 Performance benchmark .................................. 21
  2.4 Summary ..................................................... 25

3 Support Vector Machines 26
  3.1 Introduction .................................................. 26
# List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Statistical properties of prosodic features</td>
<td>15</td>
</tr>
<tr>
<td>2.2</td>
<td>Human and machine recognition results for continuous emotions</td>
<td>24</td>
</tr>
<tr>
<td>2.3</td>
<td>Comparison of machine learning algorithms for continuous emotion recognition on the VAM database</td>
<td>24</td>
</tr>
<tr>
<td>4.1</td>
<td>List of prosodic features</td>
<td>64</td>
</tr>
<tr>
<td>5.1</td>
<td>Recognition results with MSFs and STSFs</td>
<td>82</td>
</tr>
<tr>
<td>5.2</td>
<td>Recognition accuracies at selected SNR levels (training on clean speech and testing on noisy)</td>
<td>84</td>
</tr>
<tr>
<td>5.3</td>
<td>Recognition results with prosodic and combined features</td>
<td>89</td>
</tr>
<tr>
<td>5.4</td>
<td>Confusion matrix for using only prosodic features</td>
<td>89</td>
</tr>
<tr>
<td>5.5</td>
<td>Confusion matrix for using prosodic and proposed features</td>
<td>89</td>
</tr>
<tr>
<td>5.6</td>
<td>Continuous emotion recognition (RBF kernel)</td>
<td>92</td>
</tr>
<tr>
<td>5.7</td>
<td>Continuous emotion recognition (linear kernel)</td>
<td>92</td>
</tr>
<tr>
<td>5.8</td>
<td>Continuous emotion recognition (polynomial kernel)</td>
<td>92</td>
</tr>
<tr>
<td>5.9</td>
<td>Binary classification results using three primitive features</td>
<td>95</td>
</tr>
<tr>
<td>A.1</td>
<td>Top 10 prosodic features ranked by AFR</td>
<td>114</td>
</tr>
<tr>
<td>A.2</td>
<td>Top 10 proposed features ranked by AFR</td>
<td>114</td>
</tr>
<tr>
<td>A.3</td>
<td>Top 10 combined features ranked by AFR</td>
<td>114</td>
</tr>
</tbody>
</table>
## List of Figures

<table>
<thead>
<tr>
<th>Figure Number</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>The activation-evaluation emotion space</td>
<td>9</td>
</tr>
<tr>
<td>3.1</td>
<td>Linearly separable data with an infinite number of decision hyperplanes.</td>
<td>29</td>
</tr>
<tr>
<td>3.2</td>
<td>Decision hyperplanes with large and small margins.</td>
<td>30</td>
</tr>
<tr>
<td>3.3</td>
<td>A support vector machine with 5 support vectors.</td>
<td>31</td>
</tr>
<tr>
<td>3.4</td>
<td>Geometric margins of data points with respect to the hyperplane.</td>
<td>32</td>
</tr>
<tr>
<td>3.5</td>
<td>Learning in feature space.</td>
<td>40</td>
</tr>
<tr>
<td>4.1</td>
<td>Flowchart for deriving the ST representation.</td>
<td>45</td>
</tr>
<tr>
<td>4.2</td>
<td>Impulse response of a fourth-order gammatone filter.</td>
<td>47</td>
</tr>
<tr>
<td>4.3</td>
<td>Magnitude response of a 19-band auditory filterbank.</td>
<td>48</td>
</tr>
<tr>
<td>4.4</td>
<td>A critical-band speech signal and its Hilbert envelope.</td>
<td>49</td>
</tr>
<tr>
<td>4.5</td>
<td>Magnitude response of a 5-band modulation filterbank.</td>
<td>51</td>
</tr>
<tr>
<td>4.6</td>
<td>ST representation for a “neutral” speech frame.</td>
<td>52</td>
</tr>
<tr>
<td>4.7</td>
<td>Average ST representation for seven emotion categories.</td>
<td>53</td>
</tr>
<tr>
<td>4.8</td>
<td>Alternative algorithms for computing the ST representation.</td>
<td>54</td>
</tr>
<tr>
<td>4.9</td>
<td>Comparison of algorithms for deriving the ST representation.</td>
<td>55</td>
</tr>
<tr>
<td>4.10</td>
<td>Average FDR curves of three algorithms.</td>
<td>56</td>
</tr>
<tr>
<td>4.11</td>
<td>Generated ST representation before and after delay equalization.</td>
<td>57</td>
</tr>
<tr>
<td>4.12</td>
<td>Estimated PDFs of $\Phi_3(1)$ for three basic emotions.</td>
<td>60</td>
</tr>
<tr>
<td>5.1</td>
<td>Average FDR curves of MSFs.</td>
<td>72</td>
</tr>
<tr>
<td>5.2</td>
<td>Average FDR curves of MFCC features.</td>
<td>72</td>
</tr>
<tr>
<td>5.3</td>
<td>Average FDR curves of PLP features.</td>
<td>73</td>
</tr>
<tr>
<td>5.4</td>
<td>Average FDR curves of prosodic features.</td>
<td>73</td>
</tr>
<tr>
<td>5.5</td>
<td>Performance comparison of different SVM kernels (MSFs).</td>
<td>75</td>
</tr>
<tr>
<td>5.6</td>
<td>Performance comparison of different SVM kernels (MFCC features).</td>
<td>75</td>
</tr>
<tr>
<td>5.7</td>
<td>Performance comparison of different SVM kernels (PLP features).</td>
<td>76</td>
</tr>
<tr>
<td>5.8</td>
<td>Performance comparison of different SVM kernels (prosodic features).</td>
<td>76</td>
</tr>
<tr>
<td>5.9</td>
<td>Average FDR curves of MSFs and ST sample features.</td>
<td>79</td>
</tr>
<tr>
<td>5.10</td>
<td>Comparison between MSFs and ST sample features.</td>
<td>79</td>
</tr>
<tr>
<td>5.11</td>
<td>Average FDR curves of MSFs, MFCC features and PLP features.</td>
<td>81</td>
</tr>
</tbody>
</table>
5.12 Recognition results with MSFs, MFCC features, and PLP features. . 83
5.13 Comparison of different features combinations. . . . . . . . . . . . . . 85
5.14 Average recognition performance of different feature combinations. . 85
5.15 Average FDR curves of prosodic features, MSFs, and their combination. 86
5.16 Comparison between prosodic and combined features. . . . . . . . . . 86
## List of Acronyms

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>AFR</td>
<td>Average Feature Rank</td>
</tr>
<tr>
<td>ASR</td>
<td>Automatic Speech Recognition</td>
</tr>
<tr>
<td>DCT</td>
<td>Discrete Cosine Transform</td>
</tr>
<tr>
<td>DFT</td>
<td>Discrete Fourier Transform</td>
</tr>
<tr>
<td>EM</td>
<td>Expectation-Maximization</td>
</tr>
<tr>
<td>ERB</td>
<td>Equivalent Rectangular Bandwidth</td>
</tr>
<tr>
<td>ERM</td>
<td>Empirical Risk Minimization</td>
</tr>
<tr>
<td>FDR</td>
<td>Fisher Discriminant Ratio</td>
</tr>
<tr>
<td>FL</td>
<td>Frame-Level</td>
</tr>
<tr>
<td>GMM</td>
<td>Gaussian Mixture Model</td>
</tr>
<tr>
<td>HCI</td>
<td>Human-Computer Interaction</td>
</tr>
<tr>
<td>HMM</td>
<td>Hidden Markov Model</td>
</tr>
<tr>
<td>IDFT</td>
<td>Inverse Discrete Fourier Transform</td>
</tr>
<tr>
<td>IIR</td>
<td>Infinite Impulse Response</td>
</tr>
<tr>
<td>KKT</td>
<td>Karush-Kuhn-Tucker</td>
</tr>
<tr>
<td>KNN</td>
<td>K-Nearest Neighbor</td>
</tr>
<tr>
<td>LDA</td>
<td>Linear Discriminant Analysis</td>
</tr>
<tr>
<td>LOO</td>
<td>Leaving-One-Out</td>
</tr>
<tr>
<td>LP</td>
<td>Linear Prediction</td>
</tr>
<tr>
<td>LPCC</td>
<td>Linear Prediction Cepstral Coefficient</td>
</tr>
<tr>
<td>MFCC</td>
<td>Mel-Frequency Cepstral Coefficient</td>
</tr>
<tr>
<td>MSF</td>
<td>Modulation Spectral Feature</td>
</tr>
<tr>
<td>NB</td>
<td>Naive Bayes</td>
</tr>
<tr>
<td>NN</td>
<td>Neural Network</td>
</tr>
<tr>
<td>PDF</td>
<td>Probability Density Function</td>
</tr>
<tr>
<td>PLP</td>
<td>Perceptual Linear Prediction</td>
</tr>
<tr>
<td>PLPC</td>
<td>Perceptual Linear Predictive Coefficient</td>
</tr>
<tr>
<td>RBF</td>
<td>Radial Basis Function</td>
</tr>
<tr>
<td>RMSE</td>
<td>Root Mean Squared Error</td>
</tr>
<tr>
<td>SBA</td>
<td>Segment Based Approach</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Full Form</td>
</tr>
<tr>
<td>--------------</td>
<td>-----------</td>
</tr>
<tr>
<td>SER</td>
<td>Speech Emotion Recognition</td>
</tr>
<tr>
<td>SFFS</td>
<td>Sequential Floating Forward Selection</td>
</tr>
<tr>
<td>SFS</td>
<td>Sequential Forward Selection</td>
</tr>
<tr>
<td>SN</td>
<td>Speaker Normalization</td>
</tr>
<tr>
<td>SNR</td>
<td>Signal-to-Noise Ratio</td>
</tr>
<tr>
<td>SRM</td>
<td>Structural Risk Minimization</td>
</tr>
<tr>
<td>ST</td>
<td>Spectro-Temporal</td>
</tr>
<tr>
<td>STSF</td>
<td>Short-Term Spectral Features</td>
</tr>
<tr>
<td>SVC</td>
<td>Support Vector Classification</td>
</tr>
<tr>
<td>SVM</td>
<td>Support Vector Machine</td>
</tr>
<tr>
<td>SVR</td>
<td>Support Vector Regression</td>
</tr>
<tr>
<td>TEO</td>
<td>Teager Energy Operator</td>
</tr>
<tr>
<td>UL</td>
<td>Utterance-Level</td>
</tr>
<tr>
<td>VC</td>
<td>Vapnik-Chervonenkis</td>
</tr>
<tr>
<td>ZCR</td>
<td>Zero-Crossing Rate</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

1.1 Motivation

As human-computer interaction (HCI) technology evolves, it becomes more desirable for machines to interact affectively with humans. The field of affective computing [1], consequently, has gained considerable research interest in the past few years, and one of its research areas, speech emotion recognition (SER), is going through a phase of especially rapid development, marked as moving from a period of exploratory study into one with the prospect of many applications [2]. However, despite the substantial advances achieved in this area, recognizing human emotion in speech still faces a number of challenges. In particular, as fundamentally a machine learning task, high-performance SER strongly demands effective features.

Most acoustic features used for emotion recognition can be grouped into two categories: prosodic and spectral. Prosodic features have been shown to convey very important emotional cues about the speaker [3–6]. Although no agreement has been reached on the optimal set to use, prosodic features form the most widely employed
feature type for SER, and have been extensively studied in previous works (e.g. [3–9]). Spectral features, on the other hand, also play a significant role in SER as they are related to the frequency content of the speech signal, hence providing information complementary to prosodic features. However, comparatively limited research efforts have been put into designing more powerful spectral features.

Existing spectral features are mostly extracted from short-term windowed speech segments of 20–30 ms duration, due to the common assumption in speech processing that the speech signal is a short-time stationary process. However, the limitation of merely using short-term spectral features (STSFs) for automatic speech recognition (ASR) is considerable [10]. Even though information about longer temporal regions can be incorporated via the inclusion of local derivatives of STSFs, the fundamental character of the features still remains quite short-term. Such limited perspective is considered a key weakness in contemporary ASR techniques [10], and thus likely to hamper high-performance machine recognition of speech emotions.

Furthermore, recent advances in neuroscience [11–14] indicate the existence of spectro-temporal (ST) receptive fields in the human auditory cortex which can extend up to fairly long-term temporal spans of hundreds of milliseconds and respond to the ST modulations. These new insights further reveal the shortcoming of STSFs as they convey the signal’s short-term spectral properties only while discarding the important temporal cues used by the human auditory system, especially the long-term temporal modulation information. Therefore, features capturing both spectral properties and temporal modulation information are sought-after, to better model the nature of human auditory perception and alleviate the limitations of traditional spectral features. In this thesis, ST features in line with these findings are proposed,
namely modulation spectral features (MSFs), with applications to several SER tasks.

1.2 Contributions

The thesis mainly makes the following contributions:

1. Spectro-temporal processing is performed for SER. An auditory-inspired long-term ST representation of speech is computed, considering the conventional acoustic frequency jointly with the modulation frequency. To derive the ST representation, a modulation filterbank is employed in addition to the regular auditory filterbank, performing frequency analysis on temporal amplitude modulations (more specifically, Hilbert envelopes) at multiple acoustic frequency bins and thereby capturing both spectral and temporal properties of the speech signal. The spectra of the temporal modulation signals are called modulation spectra, and the features derived from the ST representation, consequently, are termed modulation spectral features.

2. The proposed MSFs are benchmarked by both spectral and prosodic features. Support vector machines (SVMs) are used for machine learning. We show that the MSFs outperform STSFs based on well-known mel-frequency cepstral coefficients (MFCCs) and perceptual linear predictive coefficients (PLPCs), and further render a substantial improvement in recognition performance when used as additions to prosodic features. By combining the MSFs and prosodic features, the proposed SVM-based SER system achieves an overall accuracy of 91.6% for classifying seven emotion categories, which compares very well with state-of-the-art works, especially given that a much more compact feature pool
is extracted in our study. Moreover, besides the conventional discrete emotion classification task, further efforts are made to deal with the estimation of continuous emotions, a topic investigated by fewer studies. Estimation performance comparable to human evaluation is obtained.

3. To our knowledge, the only previous research effort on using modulation spectral content for SER is reported in [15], where the modulation features are combined with several other feature types (e.g. loudness and PLPC features) and approximately 70% recognition rate is obtained on the so-called Berlin emotional speech database [16]. This present study is different in several ways, namely: (1) filterbanks are employed to perform spectral decomposition in both acoustic frequency and modulation frequency domains; (2) the proposed MSFs are designed by carefully exploiting the ST representation, shown to achieve significantly better performance on the Berlin database relative to the results reported in [15]; and (3) estimation of continuous emotions is also investigated in this thesis.

1.3 Thesis Outline

This thesis is organized as follows. Chapter 2 provides necessary background and literature review for SER. We start by introducing the nature of speech emotions, and then describe emotional speech databases, features, and machine learning algorithms used for SER. State-of-the-art performance is also introduced. Chapter 3 presents the fundamentals of SVMs, a powerful machine learning technique upon which our emotion recognition system is built. The basic SVM model and its extensions are
described. Chapter 4 is the heart of this thesis. It first details the algorithm for computing the ST representation, with several alternative derivation algorithms discussed and compared. A detailed description of the proposed MSFs extracted from the ST representation is then given. The comparison features are also introduced. Simulation results are presented and discussed in Chapter 5. Both discrete emotion classification and continuous emotion regression are performed. Finally, Chapter 6 gives concluding remarks and suggests possible future work.
Chapter 2

Speech Emotion Recognition

2.1 Introduction

Affective computing is an interdisciplinary research field concerned with the automatic recognition, interpretation, and synthesis of human affective content in speech, facial expressions, and/or any other human biological communication channel [1]. Among its areas of interest, automatic speech emotion recognition (SER) has been shown to be an especially dynamic research subject, aiming at recognizing the emotional state of the speaker from the speech signal. Even though emotions hardly alter the linguistic content of speech, such paralinguistic information serves an important role in vocal communication and has been found to be useful in multiple ways [3, 4], especially as an essential ingredient of emotional intelligence, thus giving birth to a broad range of commercially promising applications in HCI, multimedia retrieval, etc.

Several inherent advantages make speech signals a good source for affective computing. First, compared to many other biological signals (e.g. electrocardiogram), speech signals usually can be acquired more readily and economically. Moreover,
since a fairly broad range of emotions can be faithfully delivered in a telephone conversation in which only auditory information is exchanged, it is promising to build high-performance SER systems, using speech signals as the sole input. Such speech based systems can function either independently or as modules of more sophisticated techniques that combine other emotion-bearing information sources such as facial expression [17], body gestures [18], and electrocardiogram [19].

Three key issues need to be addressed for successful SER, namely (1) constructing good emotional speech databases, (2) designing reliable classifiers or regressors using machine learning algorithms, and (3) designing effective features. This thesis mainly contributes to the last issue by proposing a novel and powerful feature set. The remainder of this chapter is organized as follows. Section 2.2 introduces the nature of emotions in speech. Section 2.3 presents contemporary SER techniques in four aspects: data (Section 2.3.1), features (Section 2.3.2), machine learning algorithms (Section 2.3.3), and state-of-the-art performance (Section 2.3.4). Lastly, Section 2.4 summarizes the chapter.

2.2 The Nature of Emotion in Speech

Compared to machine recognition of human emotion, research on the nature of emotion has a much longer history. The term “emotion” can refer to an extremely complex state associated with a wide variety of mental, physiological and physical events. However, its scope is limited to, in the context of SER, the part of this state conveyed in speech. Both the taxonomy used to describe emotions and the way emotional speech is produced shape the nature of speech emotion, as introduced in this section.
2.2.1 Describing emotions

There have been two frameworks for describing speech emotions, using *categorical* and *dimensional* descriptors (termed *discrete* and *continuous* emotions in this thesis), respectively. The categorical framework, dominantly used in SER [4], characterizes emotions using everyday emotion words such as *neutral* and *anger*. It offers instinctive and straightforward emotion descriptions and is considered to be “self-evidently” prescriptive by the great majority of the research community [2, 4, 20]. Moreover, the combination of some basic (or prototypical) discrete emotions such as the “big six” [21] (*anger, disgust, fear, joy, sadness, and surprise*) can serve as a good sampling of the universal emotion space [22].

However, the categorical description does suffer from several limitations. Most notably, it is unfeasible to achieve a comprehensive emotion representation under this framework. A categorical emotion model covering most of everyday emotions would require hundreds of keywords, an intractable scale for research [20]. Consequently, even though the categorical approach has been used in a number of practical applications, most of the applications are interested in detecting the presence of one specific emotion only [23–27]. Also, the categorical descriptors leave the relationships between different emotions undefined. Neither gradual changes within the same emotion nor transitions between different emotional states are characterized. Furthermore, everyday emotions may be mixtures of several prototypical emotions and hence cannot be unambiguously classified into specific categories [28, 29].

Such drawbacks can be overcome by the use of the dimensional description framework that furnishes continuous emotions. This psychologically-based method [30, 31] provides a more refined and systematic description of emotions by representing them
through a few dimensions. The term “dimension” actually refers to a set of primary attributes of emotions that form the bases of an emotion space, where categorical descriptors are points in the space and can be located by coordinates. A two-dimensional activation-evaluation emotion space is depicted in Fig. 2.1. *Activation* represents the intensity of emotions and *evaluation* (or *valance*) is a measure of an emotion characterized as having positive or negative significance according to human appraisal and experience [20]. As illustrated in the figure, a large amount of daily emotions can be described using this model of simply two dimensions, with their relationships clearly displayed. While a considerably greater number of emotion primitives may be needed to fully constitute a specific emotion, it is hoped that a small number of essential primitives would suffice to build machine-based SER systems.

![Figure 2.1: The 2-D activation-evaluation emotion space, with the approximate positions of some categorical descriptors shown in the plane [20].](image-url)
2.2.2 Producing emotional speech

The common practice to obtain emotional speech is to employ some professional actors to deliberately express several target emotions. The emotions produced in this manner, usually associated with categorical descriptors, are referred to as simulated (or acted) emotions. Simulated emotions usually exhibit high emotion arousal (i.e. emotion strength) and can be reliably recognized by both human and machines [4]. Corpora of simulated speech are also satisfactory resources for studies on emotional speech synthesis [32].

However, emotions simulated by actors are often exaggerated and biased compared to those expressed naturally in vocal communication, and few works exist to systematically analyze the relationship between acted and everyday emotional speech. In addition, the recording set up of acted emotional speech is typically under a monologue (instead of dialog) environment with the text material being “read” rather than “spoken” [2]. A way of partially compensating for the above drawbacks is to use emotion-triggering texts and/or real-life simulations to evoke the subject’s emotions. The emotional speech produced in such cases is usually called elicited emotional speech.

In contrast to artificial speech data, materials with more natural speech communication (e.g. TV news) provide an alternative source of emotional speech. The corresponding emotions are known as spontaneous (or natural) emotions. A number of human evaluators are employed to either label the spontaneous speech by one of the candidate emotion categories (in the discrete emotion case) or assign subjectively assessed values along the emotion dimensions (in the continuous emotion case).

Ideally, true natural emotional speech is more suitable for research on SER, as
it is the ultimate object that acted speech attempts to imitate. However, natural
emotions are unpredictable and potentially mixed, with little control applicable to
the speakers. As a result, it is usually difficult and expensive to gather sufficient
data for even a few basic emotions [16]. Also, copyright issues may arise as well (e.g.
when using materials from television or radio) [2,4], impeding data collection and
distribution. For these reasons, simulated emotions are favored in existing emotional
speech databases [4].

2.3 Automatic Speech Emotion Recognition

2.3.1 Emotional speech data

Emotional speech databases are valuable for research on SER. An excellent compre-
hensive review of data collections is available in [4]. An overview of existing databases
(mostly in German or English) indicates that speech emotions are dominantly de-
scribed using the categorical framework, and simulated emotions are more prevalent
relative to spontaneous or elicited emotions. In addition, recognition is shown to be
the major purpose for database creation, with synthesis of secondary interest.

However, given that a large number of databases exist, few of them are publicly
accessible to the whole research community, a main barrier to advances in SER. More-
over, relative to ASR tasks, no standardized speech corpora and test conditions exist
for SER [33], making performance comparison under the same settings difficult. In
this thesis, two publicly available databases, the Berlin emotional speech database [16]
and the Vera am Mittag (VAM) database [34], are employed, to perform experiments
assessing classification of discrete emotions and regression of continuous emotions, respectively. The two databases are described below, with state-of-the-art performance achieved on them introduced in Section 2.3.4.

### 2.3.1.1 Berlin database

The Berlin emotional speech database [16] is employed for discrete emotion classification in this thesis\(^1\). As a publicly available database, it has become one of the most popular databases used by researchers on SER, thus facilitating performance comparisons with other studies. Ten actors (5 male and 5 female) each uttered ten everyday sentences in German (5 short and 5 long, typically with durations between 1.5 and 4 seconds). The sentences are chosen to be semantically neutral and hence can be readily interpreted in all of the seven emotions simulated. Speech is recorded with 16 bit precision and 48 kHz sampling rate (later downsampled to 16 kHz) in an anechoic chamber.

The raw database contains approximately 800 sentences (7 emotions \(\times\) 10 sentences \(\times\) 10 actors + some second versions). The speech files are further evaluated by a subjective perception test with 20 listeners to guarantee the recognizability and naturalness of the emotions. Only utterances scoring higher than 80% emotion recognition rate and considered natural by more than 60% listeners are retained. The final numbers of utterances for the seven emotion categories in the Berlin database are: anger (127), boredom (81), disgust (46), fear (69), joy (71), neutral (79) and sadness (62).

\(^1\)The database can be downloaded at http://www.expressive-speech.net/emodb/.
2.3.1.2 VAM Database

The VAM database [34] is a very recent database created for studying spontaneous emotions\(^2\). It is recorded from natural human communication in a German TV talk-show “Vera am Mittag”. There are three individual modules in the complete VAM database: VAM-Video, VAM-Audio and VAM-Faces, freely available and containing audio+visual signals, audio signal only, and face images, respectively. In this thesis, only the VAM-Audio part is employed and hereinafter it is simply referred to as the VAM database. The speech signal is recorded with 16 bit precision and 44.1 kHz sampling rate (later downsampled to 16 kHz), and manually segmented at the utterance level.

The VAM database furnishes spontaneous emotions described in an emotion space consisting of three emotion primitives: *valence* (i.e. *evaluation*, negative to positive), *activation* (with levels from low to high, i.e. calm to excited) and *dominance* (the apparent strength of the speaker, i.e. ability to handle a situation, ranging from weak to strong) [35]. The continuous values on the primitive scale are subjectively assessed by a number of human evaluators. The VAM database contains two parts: (1) VAM I with 478 utterances from 19 speakers (4 male and 15 female) that are characterized by a high level of activity with a wide variety of emotional states and 17 human evaluators assessing the primitives, and (2) VAM II with 469 utterances from 28 speakers (7 male and 21 female) that possess a high level of activity but a smaller scope of emotions (e.g. only *anger*) and assessed by 6 evaluators. The joint database VAM I+II, therefore, includes 947 utterances from 47 speakers (11 male and 36 female).

\(^2\)The database can be accessed through the HUMAINE network at http://emotion-research.net/download/vam/.
2.3.2 Features

In machine learning, features are measurable attributes of samples. Extracting discriminating features is always essential for any pattern recognition application, including emotion recognition. As research on SER moves towards the exploitation of an increasingly complex feature space, specific features used by different works vary significantly [36]. Nevertheless, there are two most widely used feature groups: prosodic and spectral, as aforementioned in Chapter 1. They are introduced here.

2.3.2.1 Prosodic features

Prosody is the suprasegmental phonology of speech sounds, involving (but not limited to) aspects such as pitch, loudness, tempo, and stress. The importance of prosody for emotion prediction was noticed more than a century ago [37] – “Even monkeys express strong feelings in different tones, anger and impatience by low, fear and pain by high notes.” Today, researchers working on SER have put substantial effort into studying prosodic features and made them the most extensively used feature type, e.g. [3–9].

Common prosodic features are based on pitch, intensity, and speaking rate [4]. Pitch, often referred to as fundamental frequency, is the vibration rate of the vocal folds. It is closely related to glottal and vocal behavior and has been recognized as an effective emotion indicator. Speech intensity is measured as the short-term speech power. It impacts the perceptual loudness of speech and indicates the arousal level of emotion [4]. Speaking rate is usually calculated quantitatively as the number of syllables or words per unit time, and measures how fast speech is produced.

Prosodic features are typically estimated on a short-term frame basis. Although
the frame-level features can be used directly for machine learning, a more common practice in SER is to employ their contours to derive features (e.g. statistics) at the utterance-level. The statistical behaviors of prosodic features have been shown to convey important emotional cues [3–6]. Table 2.1 lists the statistical properties of prosodic features (pitch, intensity and speaking rate) for five basic emotions, with neutral being the reference point for comparison. Interestingly, it can be seen from the table that the statistical behaviors of features from males may be different or even opposed to those from females for the same emotion. For example, males tend to produce a decreased speaking rate when they are angry whereas females increase their speaking rate.

Table 2.1: Statistical properties of prosodic features for selected emotions [4].

<table>
<thead>
<tr>
<th></th>
<th>Pitch</th>
<th></th>
<th>Intensity</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Average</td>
<td>Range</td>
<td>Variance</td>
<td>Contour</td>
</tr>
<tr>
<td>Anger</td>
<td>&gt;&gt;</td>
<td>&gt;</td>
<td>&gt;&gt;</td>
<td></td>
</tr>
<tr>
<td>Disgust</td>
<td>&lt;</td>
<td>&gt;M, &lt;F</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fear</td>
<td>&gt;&gt;</td>
<td>&gt;</td>
<td>&gt;</td>
<td>/&gt;</td>
</tr>
<tr>
<td>Joy</td>
<td>&gt;</td>
<td>&gt;</td>
<td>&gt;</td>
<td>\</td>
</tr>
<tr>
<td>Sadness</td>
<td>&lt;</td>
<td>&lt;</td>
<td>&lt;</td>
<td>\</td>
</tr>
</tbody>
</table>

Explanation of symbols: ‘>’, ‘<’, and ‘=’ mean an increase, a decrease, and no change compared to the neutral emotional state, respectively; double symbol indicates a considerable change; ‘/>’ and ‘\’ stand for upward and downward inclines, respectively; subscripts M denotes male and F denotes female.

2.3.2.2 Spectral features

Spectral features characterize signal properties in the frequency domain, thus providing useful additions to prosodic features. The frequency domain representation can be obtained by taking the discrete Fourier transform (DFT) of the discrete-time
signal. However, the DFT of a signal usually results in a high-dimensional representation, in addition, with fine spectral details that may turn out to impair recognition performance. Therefore, the DFT spectrum is usually transformed into some more compact feature representations for speech-related recognition tasks, such as the mel-frequency cepstral coefficients (MFCCs) and the perceptual linear predictive coefficients (PLPCs) that are often used in current SER techniques (e.g. [36,38–41]).

The MFCCs\(^3\), introduced in [42], form one of the most successful feature representations in speech recognition [43]. They are derived through the following steps. First, the speech signal is blocked into short-term overlapping frames. Each speech frame \(x(n)\) is multiplied by an analysis window \(w(n)\) and the DFT is computed:

\[
X(k) = \sum_{n=0}^{N-1} x(n)w(n)e^{-j\frac{2\pi}{N}kn},
\]

where \(0 \leq k \leq N - 1\) with \(N\) being the DFT length. The magnitude spectrum \(|X(k)|\) is then weighted by a filterbank with \(L\) overlapping triangular filters equally spaced on the mel scale. A common frequency mapping between Hz \((f_{Hz})\) and mel \((f_{mel})\) is given by:

\[
f_{mel} = 1127.01048 \log_e(1 + \frac{f_{Hz}}{700}).
\]

The energy of the weighted spectrum is calculated for each filter, and is denoted as \(E(l)\) \((1 \leq l \leq L)\) where \(l\) refers to the \(l\)th mel-scale filter. Finally, discrete cosine transform (DCT) is applied to the log-energy values, and the \(m\)th MFCC is calculated as:

\[
\text{MFCC}(m) = \sqrt{\frac{2}{L}} \sum_{l=1}^{L} \log_e(E(l)) \cos(m\frac{\pi}{L}(l - \frac{1}{2})).
\]

The PLPCs [44] are obtained by perceptual linear prediction (PLP), an extension

\(^3\)In this thesis, cepstral features are considered a kind of spectral features, while some works may distinguish between spectral and cepstral features.
of traditional linear prediction. Motivated by psychoacoustics, PLP analysis is more consistent with human hearing than regular linear predictive analysis. To obtain the coefficients, the power spectrum $P(\omega)$ of the windowed speech signal is first calculated and warped into the Bark scale using the formula:

$$\Omega = 6 \log_e \left( \frac{\omega}{1200\pi} + \sqrt{\left( \frac{\omega}{1200\pi} \right)^2 + 1} \right).$$  \hspace{1cm} (2.4)

Then $P(\Omega)$ is convolved with the frequency response of the critical band filter $\Psi(\Omega)$ to obtain $\Theta(\Omega)$, where $\Psi(\Omega)$ equals: (1) $10^{2.5(\Omega+0.5)}$ for $-1.3 \leq \Omega \leq -0.5$, (2) 1 for $-0.5 \leq \Omega \leq 0.5$, (3) $10^{-1.0(\Omega-0.5)}$ for $0.5 \leq \Omega \leq 2.5$, and (4) 0 otherwise. $\Theta(\Omega)$ is downsampled at approximately 1 Bark intervals, and emphasized by an equal-loudness curve $W(\omega)$:

$$W(\omega) = \frac{(\omega^2 + 5.68 \times 10^7)\omega^4}{(\omega^2 + 6.3 \times 10^6)^2(\omega^2 + 3.8 \times 10^8)}. \hspace{1cm} (2.5)$$

Lastly, the emphasized spectrum $W(\omega)\Theta(\omega)$ is processed by cubic root compression, and approximated by an all-pole model. The PLPCs are the resulting autoregressive coefficients.

Besides the MFCCs and the PLPCs, several novel spectral feature representations have also been proposed for SER in the literature, all reported to be useful yet far less employed relative to the MFCCs or PLPCs. In [45], log-frequency power coefficients are proposed for emotion recognition. The features are derived by first separating the short-term spectrum of speech into 12 bands whose center frequencies and bandwidths are determined to match the critical bands of the human auditory system [46], and then grouping the energy within each band. In [40], similarly, the outputs of the mel-scale filterbank are used directly as features without transforming into the MFCCs. In [47], a set of voice quality parameters are described, parameterizing the voice quality in the frequency domain in terms of spectral gradients and capturing the
properties of glottal excitation.

Other spectral features that have been employed for SER include speech formants and spectral measures. Formants are the resonances of the vocal tract in the frequency domain, dominating the frequency response of the vocal tract and thus shaping the spectrum. Usually position (peak frequency), amplitude (peak magnitude) and bandwidth of the first two or three formants are extracted as features [4, 7, 38]. Spectral measures are parameters measuring certain spectral characteristics, such as band energy, centroid, bandwidth, flux, and roll-off frequency [36, 38, 48].

However, all the spectral features described above are short-term features. Long-term temporal information is not captured in feature extraction. As an effort to incorporate temporal modulations, long-term modulation features are proposed in [49]. The features are used for emotion recognition in [15], and achieve approximately 60% and 70% recognition rates on the Berlin database, by using the features solely and in combination with loudness and PLP features, respectively. This thesis also presents a novel set of modulation spectral features. Obtained by exploiting an auditory-inspired long-term ST representation, the features substantially differ from the ones proposed in [49]. Additionally, as shown in the simulation study (Chapter 5), our features yield considerably higher recognition accuracy.

2.3.2.3 Feature levels

Features for speech recognition are typically extracted on a frame basis. However, only a small portion of studies use frame-level (FL) features directly for emotion recognition in the literature. Moreover, such studies are usually set up to investigate specific machine learning algorithms that require FL features as the input (e.g. [23,
The standard approach in contemporary SER systems computes features at utterance-level (UL) [33], by applying a number of descriptive functions (typically statistical) to the contours of FL features, and often to their derivatives as well to extract local dynamic cues. The commonly used UL features calculated from the FL feature trajectories include mean, standard deviation, maximum, minimum, range, quartiles, difference between quartiles, central moments, etc.

Several advantages of the UL approach make it much more prevalent than the FL approach. Most importantly, the UL features capture the global properties and behaviors of their FL counterparts. Capturing such suprasegmental characteristics of emotional speech is desirable, as they convey information attributed to emotions rather than specific spoken-content [33]. Therefore, compared to FL modeling, the UL approach effectively avoids problems arising from spoken-content over-modeling [50]. Furthermore, the UL approach alleviates the limitation of local short-term features by extracting global information from them. Comparisons between the two approaches also indicate that using UL features delivers better recognition performance for SER [50, 51].

On the other hand, the temporal information omitted by the UL method may still contain useful emotional cues [4]. Efforts compensating for such information have also been witnessed. The segment based approach (SBA) [52, 53] is a variant of conventional UL modeling by incorporating different temporal scales. It measures an identical battery of statistics over the regular UL as well as the voiced-segment-level, and small improvement over the UL method is reported. Notice that the standard version of SBA directly doubles the size of the feature vector fed into the recognition
algorithm, since the same statistics have to be calculated at both UL and voiced-segment-level. Modified versions may calculate fewer features for a certain level. However, the learning complexity is still likely to be increased notably. In [50], joint FL and UL modeling is proposed. This joint modeling approach includes the temporal information by aggregating the FL feature sequences to UL scores through statistical modeling. The scores are then concatenated to the regular UL feature vector to perform UL recognition. The method is shown to outperform the UL method, but yielding additional computational expense coming from modeling FL features.

2.3.3 Machine learning algorithms

Machine learning is a core subfield of artificial intelligence. Broadly speaking, it is concerned with the design and development of algorithms that allow systems to learn from data and experience, in a manner that continuously improves system performance, e.g. making more accurate predictions. For recognition tasks, the goal of the algorithms is to learn a mapping from the training samples in certain feature space to some labels, which can be either discrete (in classification cases) or real-valued (in regression cases).

Many machine learning algorithms have been used for discrete emotion classification, including support vector machine (SVM) [8, 48, 50, 52, 54–58], neural network (NN) [8,55–59], k-nearest neighbor (KNN) [15,35,52,55,57,58,60], discriminant analysis [7,25], decision tree [8,36,55,58], naive Bayes (NB) [47,58], Gaussian mixture model (GMM) [6,24,41,50], and hidden Markov model (HMM) [40,45,51]. For continuous emotion regression, however, comparatively fewer studies have been done, and the use of fuzzy logic [35,61], SVM [61] and KNN [28,61] has been witnessed.
There is no definitive answer to the choice of learning algorithm, for every technique has its own strength but none can provide the best recognition performance under all situations. Hence the selection criterion should be based on the concrete task. As an example, the simple NB classifier, despite its naive assumption of class conditional independence, is computationally efficient and may work well in many real-world situations, especially for high dimensional inputs and for features with normal distributions. On the other hand, SVM is a powerful technique capable of learning very complex decision boundaries that NB classifiers may fail to find, and provides good generalization performance. However, it can be very slow in some applications.

Moreover, the level of features to be modeled also influences algorithm selection for SER. For example, SVMs are often employed to handle UL feature inputs, while techniques such as GMMs and HMMs are popular choices if FL features are being modeled. In this present study, SVMs are employed to build emotion recognition systems, as they have been shown to give superior performance in many SER works [52, 55–58, 61]. The fundamentals of SVMs will be introduced in Chapter 3.

### 2.3.4 Performance benchmark

#### 2.3.4.1 Berlin database

As aforementioned, unlike regular ASR tasks, up to now no standardized speech corpora and test settings exist for SER to enable performance comparison under identical conditions. For the popular Berlin database, studies on it differ significantly in terms of features extracted, machine learning algorithms employed, and the way the data are partitioned, making it difficult to pinpoint state-of-the-art performance.
However, even though the performance figures cannot be compared directly, it would still be useful to briefly review good recognition results reported in the literature to serve as an indirect benchmark.

Unless otherwise specified, all the results cited here are achieved for classifying seven emotions in the Berlin database by using UL features. In [54], an overall recognition rate of 86.7% is obtained under 10-fold cross-validation, by using approximately 4000 features with SVMs employed as classifiers. This huge feature set is generated by applying a large number of functions (e.g. mean, standard deviation, skewness, kurtosis, centroid, quartiles, ranges, regression coefficients) to the contours of FL features (e.g. pitch, energy, MFCCs, formants, spectral measures) as detailed in [38]. The recognition accuracy is slightly improved to 86.9% after feature space optimization, but the final number of features used is not reported. Sadness and joy are the best and the worst recognized emotions with 94.3% and 69.0% recognition rates, respectively.

In [47], it is noticed that features optimal for recognizing certain emotions may not be discriminating for recognizing others. Therefore, multi-stage classification is proposed and shown to improve SER performance. Over 300 statistics are calculated from contours of pitch, intensity, duration, formants, and voice quality parameters, and fed into NB classifiers. Recognition is performed in a cascaded style with features being optimized for each stage. By designing two- and three-stage schemes, classification rates of 83.5% and 88.8% are reported, corresponding to 9% and 14% performance improvement relative to the direct (single-stage) classification, respectively. However, [47] only recognizes six emotions (no disgust).

In [50], joint utterance- and frame-level modeling is investigated. Approximately
1400 acoustic features are calculated for data mining, but no information is given about the number of features finally selected. The effect of speaker normalization (SN) is also studied, which removes the mean of features and normalizes them to unit variance. Experiments are performed under a speaker-independent condition and the SVM technique is used for recognition. It is shown that without feature selection, the SN procedure improves the recognition accuracy from 74.9% to 79.6%. The 79.6% accuracy is further raised to 83.2% after optimizing the feature space. Additionally, the FL features (a 39-dimensional vector consisting of 13 MFCCs and their delta and double-deltas) are modeled by GMMs, and the final GMM scores are concatenated to the UL features. This joint modeling yields a 89.9% recognition rate.

2.3.4.2 VAM database

Compared to the Berlin database, significantly fewer studies exist on the VAM database for recognizing continuous emotions. In [35], a fuzzy logic estimator is used for regression, and 46 UL features are extracted, based on pitch, intensity, speaking rate and MFCCs. Principal component analysis (PCA) is used for dimensionality reduction. The recognition results are summarized in Table 2.2, where $\overline{r}$ and $\overline{e}$ stand for correlation and mean absolute error averaged over the three emotion primitives in the database, respectively. It can be seen from the table that machine recognition generally gives correlation comparable to human evaluation, yet with a larger estimation error.

In [61], SVMs are used for regression (support vector regression, SVR) on a slightly reduced version of VAM I+II (with 893 utterances). The FL features extracted are similar to the ones used in [35]. However, a larger feature pool of 137 UL features is
Table 2.2: Human and machine recognition results for continuous emotions on VAM I, VAM II, and VAM I+II; correlation and mean absolute error values are averaged over the three primitives [35].

<table>
<thead>
<tr>
<th>Dataset</th>
<th># of speakers</th>
<th>human evaluation</th>
<th>machine recognition</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>$\tau$</td>
<td>$\epsilon$</td>
</tr>
<tr>
<td>VAM I</td>
<td>19</td>
<td>0.65</td>
<td>0.21</td>
</tr>
<tr>
<td>VAM II</td>
<td>28</td>
<td>0.56</td>
<td>0.15</td>
</tr>
<tr>
<td>VAM I+II</td>
<td>47</td>
<td>0.61</td>
<td>0.18</td>
</tr>
</tbody>
</table>

generated by computing more statistics. Results are summarized in Table 2.3. Three SVR kernels are tested: (1) radial basis function (RBF), (2) linear (Lin), and (3) polynomial (Poly), and the best outcomes are obtained using the RBF kernel. KNN estimators also furnish good performance comparable to RBF-SVRs. Both Tables 2.2 and 2.3 indicate that *activation* and *valence* are primitives best and worst estimated, a trend consistent with human evaluation results.

Table 2.3: Comparison of different machine learning algorithms for continuous emotion recognition on VAM I+II for three primitives [61].

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>valence</th>
<th>activation</th>
<th>dominance</th>
<th>average</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$r$</td>
<td>$\epsilon$</td>
<td>$r$</td>
<td>$\epsilon$</td>
</tr>
<tr>
<td>SVR-RBF</td>
<td>0.46</td>
<td>0.13</td>
<td>0.82</td>
<td>0.15</td>
</tr>
<tr>
<td>SVR-Poly</td>
<td>0.39</td>
<td>0.14</td>
<td>0.80</td>
<td>0.16</td>
</tr>
<tr>
<td>SVR-Lin</td>
<td>0.37</td>
<td>0.13</td>
<td>0.80</td>
<td>0.16</td>
</tr>
<tr>
<td>Fuzzy Logic</td>
<td>0.28</td>
<td>0.27</td>
<td>0.75</td>
<td>0.17</td>
</tr>
<tr>
<td>KNN</td>
<td>0.46</td>
<td>0.13</td>
<td>0.80</td>
<td>0.16</td>
</tr>
<tr>
<td>Human</td>
<td>0.49</td>
<td>0.14</td>
<td>0.72</td>
<td>0.22</td>
</tr>
</tbody>
</table>
2.4 Summary

This chapter has presented the background of SER. The categorical and dimensional frameworks for emotion description have been introduced. Simulated and spontaneous emotions are also described. Three important topics in automatic SER have been reviewed, namely (1) data, (2) feature, and (3) learning algorithm. Emotional speech data serves as a starting point for research on SER. Two emotional speech databases, the Berlin database and the VAM database, have been introduced. Then common prosodic and spectral features employed in current SER techniques are described. In particular, conventional spectral features focus on the signal’s short-term spectral content only and neglect long-term temporal cues. This thesis proposes novel spectral features for emotion recognition (detailed in Chapter 4), that capture both spectral properties and long-term temporal information of speech signals. Many machine learning methods have been proposed for emotion recognition in the literature. This present study employs the SVM technique (introduced in Chapter 3) to build recognition systems, due to its attractive properties as well as superior performance demonstrated in other SER works. Lastly, state-of-the-art recognition performance achieved on the Berlin and the VAM databases has also been reviewed, serving as a benchmark for further comparisons in Chapter 5.
Chapter 3

Support Vector Machines

3.1 Introduction

Support vector machines (SVMs) [62, 63] are a relatively new machine learning technique. They have gained interest in a broad range of recognition tasks (e.g. handwritten digit recognition, speaker identification, image face detection, time series prediction) [64]. And in many of these applications, SVMs have demonstrated better generalization performance compared to competing machine learning algorithms [64].

Although hidden Markov models undoubtedly remain the most popular method in speech recognition, SVMs are gaining increasing research attention [65, 66]. In particular, due to the widespread use of utterance-level (instead of frame-level) features for SER, SVMs have been widely employed in literature for emotion recognition [8, 48, 50, 52, 54–58], and are often shown to outperform other recognition techniques such as k-nearest neighbor [55, 57, 58], decision trees [55, 58], neural network [56–58], and naive Bayes [58].

In this thesis, SVMs are employed for both classification of discrete emotions and
regression of continuous emotions. Therefore, this chapter is dedicated to introduce the SVM technique. In Section 3.2, structural risk minimization (SRM) is described, a principal equipping SVM with good generalization performance. Section 3.3 then formulates the SVM model for the basic linearly separable case (Section 3.3.1) and discusses the corresponding optimization problem (Section 3.3.2). Extensions to the basic SVM model are presented in Section 3.4, including soft margin SVM (Section 3.4.1), multiclass SVM (Section 3.4.2), and support vector regression (SVR, Section 3.4.3). Lastly, section 3.5 describes nonlinear SVMs and the kernel trick.

### 3.2 Structural Risk Minimization

Suppose we have a set of \( k \) data points \( \mathcal{D} = \{(x_1, y_1), (x_2, y_2), \ldots, (x_k, y_k)\} \), where \( x_i \in X \subseteq \mathbb{R}^n \) and its label \( y_i \in Y \subseteq \mathbb{R}^1 \), \( 1 \leq i \leq k \). The objective is to train a function \( f \) with parameters \( \alpha \) that maps an input \((x, y)\) to \((f(x, \alpha), y)\) and ideally, minimizes the expected risk \( R(\alpha) \):

\[
R(\alpha) = \int L(f(x, \alpha), y) dP(x, y),
\]

where \( L(\hat{y}, y) \) is the zero-one loss function such that \( L(\hat{y}, y) = 0 \) if \( \hat{y} = y \) and 1 otherwise, and \( P(x, y) \) is the joint distribution function of \( x \) and \( y \). Unfortunately, \( P(x, y) \) is usually unknown. The empirical risk \( R_{\text{emp}}(\alpha) \), therefore, is calculated instead of \( R(\alpha) \) from the \( k \) samples:

\[
R_{\text{emp}}(\alpha) = \frac{1}{k} \sum_{i=1}^{k} L(f(x_i, \alpha), y_i).
\]

Notice that for a chosen function \( f \), \( R_{\text{emp}}(\alpha) \) is completely determined with fixed \( \alpha \) and given \( \mathcal{D} \). Minimizing \( R_{\text{emp}}(\alpha) \) is now doable and known as empirical risk minimization (ERM). However, ERM is often not a good criterion because for many
learning techniques, low (or even zero) \( R_{\text{emp}}(\alpha) \) can be easily achieved via overfitting the training data, making the generalization performance on unseen data poor.

In contrast with traditional ERM, structural risk minimization (SRM) \([67]\) minimizes an upper bound on \( R(\alpha) \), a principle superior to ERM \([68]\) and endowing SVM with good generalization performance. It has been shown \([62]\) that the following bound holds for \( R(\alpha) \) with probability \( 1 - \eta \) (\( 0 \leq \eta \leq 1 \)):

\[
R(\alpha) \leq R_{\text{emp}}(\alpha) + \sqrt{\frac{h(\log_e(2k/h) + 1) - \log_e(\eta/4)}{k}},
\]

where \( h \) is an integer called the Vapnik-Chervonenkis (VC) dimension. The VC dimension for a set of functions \( \{f(\alpha)\} \) (where a fixed \( \alpha \) specifies a particular function of the set) is defined as the maximum number \( h \) of points that can be separated into two classes in all \( 2^h \) possible ways using functions of the set \( \{f(\alpha)\} \) \([62]\). The VC dimension of the set of hyperplanes in \( \mathbb{R}^n \) is \( n + 1 \) \([64]\). Remarkably, while \( R(\alpha) \) is unknown, the right hand of equation 3.3, commonly referred to as the risk bound, can be readily computed given the VC dimension \( h \). SRM is then performed by minimizing the risk bound.

### 3.3 Basic SVM Model

#### 3.3.1 Formulation

In order to appreciate the basic principles of SVM, we begin with the case of binary linearly separable data, i.e. the two classes of data in the \( n \)-dimensional space can be completely separated by a hyperplane. Denote the two classes as \( \{+1, -1\} \). The objective is to learn a binary classifier \( y(x) = \text{sign}(f(x)) \), such that \( y(x_i) = y_i \), where \( f(x) \) is a linear function of \( x = (x_1, x_2, \ldots, x_n) \) fully specified by the weighting vector

\[
f(x) = \sum_{i=1}^{n} \alpha_i y_i x_i - b
\]
Figure 3.1: Two linearly separable classes with an infinite number of decision hyperplanes.

\[ \mathbf{w} = (w_1, w_2, \ldots, w_n) \text{ and the intercept term } b: \]

\[ f(\mathbf{x}) = \langle \mathbf{w}, \mathbf{x} \rangle + b = \sum_{i=1}^{n} w_i x_i + b, \quad (3.4) \]

and \( \langle \mathbf{w}, \mathbf{x} \rangle + b = 0 \) is the decision hyperplane perpendicular to the normal vector \( \mathbf{w} \).

As shown in Fig. 3.1, however, there exist an infinite number of decision hyperplanes separating the data, all resulting in the same empirical error \( (R_{\text{emp}}(\alpha) = 0) \). For SVM, the optimum hyperplane is the one that correctly separates all data while the distance (called margin) from it to the nearest sample is maximized, thereby commonly referred to as the maximum-margin hyperplane. Fig. 3.2 shows two decision hyperplanes in a two-dimensional space with maximum and smaller margins. Intuitively, larger margin offers greater confidence when making classification decisions. Actually, as shown later, maximizing margin is equivalent to implementing SRM.

\[ ^{1} \text{In conformity with the notations in Section 3.2, } f(\mathbf{x}) \text{ should be written as } f(\mathbf{x}, \{\mathbf{w}, b\}). \text{ The parameters } \{\mathbf{w}, b\}, \text{ however, are dropped for simplicity.} \]
Moreover, the solution of the optimum (i.e. maximum-margin) hyperplane only depends on a subset (usually small) of the data, known as support vectors. Fig. 3.3 illustrates an example of SVM where only the five support vectors play a role in determining the maximum-margin hyperplane.

For quantitative formulation of the problem, the intuitive functional margin $\rho_i$ of the $i$th data point $(x_i, y_i)$ with respect to the decision hyperplane $\langle w, x \rangle + b = 0$ is introduced:

$$\rho_i = y_i (\langle w, x_i \rangle + b)$$  \hspace{1cm} (3.5)

Clearly, correct classification of point $(x_i, y_i)$ would give positive $\rho_i$. However, the margin measure defined above is unconstrained, as the value can be made arbitrarily large by simply scaling up $w$ and $b$. Therefore, it would be useful to normalize $\rho_i$ by
the Euclidean norm of vector $\mathbf{w}$ to make it invariant to scaling:

$$r_i = \frac{\rho_i}{\|\mathbf{w}\|} = \frac{y_i \langle \mathbf{w}, \mathbf{x}_i \rangle + b}{\|\mathbf{w}\|}. \quad (3.6)$$

The geometrical interpretation of $r_i$ is depicted in Fig. 3.4. Data points $\mathbf{x}_1$ and $\mathbf{x}_2$ are taken from classes +1 and −1, respectively, whose Euclidean distances to the hyperplane $\langle \mathbf{w}, \mathbf{x} \rangle + b = 0$ are $r_1$ and $r_2$; $\mathbf{x}_1'$ and $\mathbf{x}_2'$ are their projections on it. In addition, notice that $\mathbf{w}$ is perpendicular to the hyperplane. The derivation of the following equations then becomes straightforward:

$$\mathbf{x}_1' = \mathbf{x}_1 - r_1 \frac{\mathbf{w}}{\|\mathbf{w}\|}, \quad (3.7)$$

$$\mathbf{x}_2' = \mathbf{x}_2 + r_2 \frac{\mathbf{w}}{\|\mathbf{w}\|}. \quad (3.8)$$

Hence for arbitrary data point $(\mathbf{x}_i, y_i)$, we can write $\mathbf{x}_i'$ as:

$$\mathbf{x}_i' = \mathbf{x}_i - y_i r_i \frac{\mathbf{w}}{\|\mathbf{w}\|}. \quad (3.9)$$
Since $x'_i$ lies on the hyperplane, it satisfies $\langle w, x'_i \rangle + b = 0$:

$$\langle w, x_i - y_i r_i \frac{w}{\|w\|} \rangle + b = 0.$$  \hfill (3.10)

Solving for $r_i$ gives:

$$r_i = y_i \frac{\langle w, x_i \rangle + b}{\|w\|},$$  \hfill (3.11)

a form identical to equation 3.6, and $r_i$ is therefore called geometric margin. The margin of the classifier $r_c$ can then be defined as:

$$r_c = \min_{i: y_i = -1} r_i + \min_{i: y_i = +1} r_i$$

$$= \min_{i: y_i = -1} \frac{y_i \langle w, x_i \rangle + b}{\|w\|} + \min_{i: y_i = +1} \frac{y_i \langle w, x_i \rangle + b}{\|w\|}$$  \hfill (3.12)

Recall that the functional margin $y_i \langle (w, x_i) + b \rangle$ can be arbitrarily scaled. Hence to facilitate problem formulation, $w$ and $b$ can always be constrained so that the functional margin satisfies:

$$y_i \langle (w, x_i) + b \rangle \geq 1, \quad i = 1, 2, \ldots, k.$$  \hfill (3.13)
CHAPTER 3. SUPPORT VECTOR MACHINES

The resulting hyperplane is called the *canonical hyperplane* [62]. Therefore, the classifier margin defined in equation 3.12 is now given by:

\[ r_c = \frac{2}{\|w\|}. \] (3.14)

Moreover, it has been shown [68] that the VC dimension of the set of hyperplanes in \( R^n \) satisfying \( \|w\| \leq A \) (i.e. \( r_c \geq 2/A \)) is bounded by:

\[ h \leq \min(n, R_s^2 A^2) + 1, \] (3.15)

where \( R_s \) is the radius of the smallest hypersphere containing all the samples. Maximizing \( r_c \) is equivalent to minimizing \( \|w\| \), which in turn, minimizes the upper bound of the VC dimension \( h \). Since there is no training error, i.e. \( R_{emp}(\alpha) = 0 \), this minimizes the risk bound in equation 3.3, in other words, becoming an implementation of SRM.

### 3.3.2 Optimization

The objective now is to maximize \( r_c \), subject to the linear constraints 3.13:

\[
\text{maximize } 2 \frac{\|w\|}{\|w\|} \quad \text{subject to } y_i (\langle w, x_i \rangle + b) \geq 1, \quad i = 1, 2, \ldots, k.
\] (3.16)

Maximizing \( 2/\|w\| \) is equivalent to minimizing \( \|w\|/2 \). Moreover, it would be easier to minimize \( \|w\|^2/2 \) instead of \( \|w\|/2 \) without altering the solution. Accordingly, the optimization problem 3.16 can be reformulated as:

\[
\text{minimize } \frac{\|w\|^2}{2} \quad \text{subject to } y_i (\langle w, x_i \rangle + b) \geq 1, \quad i = 1, 2, \ldots, k.
\] (3.17)
CHAPTER 3. SUPPORT VECTOR MACHINES

To solve it, we construct its Lagrangian primal form, and then find the corresponding dual form which is easier to solve. For the optimization problem:

\[
\begin{align*}
\text{minimize} & \quad f(x) \\
\text{subject to} & \quad g_i(x) \leq 0, \quad i = 1, 2, \ldots, k \\
& \quad h_j(x) = 0, \quad j = 1, 2, \ldots, l
\end{align*}
\] (3.18)

its Lagrangian primal form is given by:

\[
\Lambda(x, \mu, \lambda) = f(x) + \sum_{i=1}^{k} \mu_i g_i(x) + \sum_{j=1}^{l} \lambda_j h_j(x),
\] (3.19)

where \( \mu = (\mu_1, \ldots, \mu_k) \) and \( \lambda = (\lambda_1, \ldots, \lambda_l) \). For problem 3.17, consequently, the primal form is:

\[
\Lambda(w, b, \mu) = \frac{\|w\|^2}{2} - \sum_{i=1}^{k} \mu_i [y_i (\langle w, x_i \rangle + b) - 1],
\] (3.20)

where variables \( w \) and \( b \) correspond to \( x \) in equation 3.19.

The Karush-Kuhn-Tucker (KKT) theorem [69] states the necessary and sufficient conditions for an optimum solution to problem 3.18, given that the objective function \( f(x) \) is a continuously differentiable convex function, and \( g_i(x) \) \((i = 1, 2, \ldots, k)\) and \( h_j(x) \) \((j = 1, 2, \ldots, l)\) are affine functions:

\[
\begin{align*}
\nabla \Lambda(x, \mu, \lambda) &= 0 \\
g_i(x) &\leq 0, \quad i = 1, 2, \ldots, k \\
h_j(x) &= 0, \quad j = 1, 2, \ldots, l \\
\mu_i g_i(x) &= 0, \quad i = 1, 2, \ldots, k \\
\mu_i &\geq 0, \quad i = 1, 2, \ldots, k
\end{align*}
\] (3.21)
Substituting equation 3.20 into the first KKT condition yields:

\[ w = \sum_{i=1}^{k} \mu_i y_i x_i \]  

(3.22)

\[ \sum_{i=1}^{k} \mu_i y_i = 0 \]  

(3.23)

Combining equations 3.20, 3.22 and 3.23, we obtain:

\[ \Lambda(w, b, \mu) = \sum_{i=1}^{k} \mu_i - \frac{1}{2} \sum_{i,j=1}^{k} y_i y_j \mu_i \mu_j \langle x_i, x_j \rangle. \]  

(3.24)

Remarkably, computing the inner product \( \langle x_i, x_j \rangle \) is the only way the data is used in SVMs.

Finally, the dual form is derived:

\[
\begin{align*}
\text{maximize} & \quad \sum_{i=1}^{k} \mu_i - \frac{1}{2} \sum_{i,j=1}^{k} y_i y_j \mu_i \mu_j \langle x_i, x_j \rangle \\
\text{subject to} & \quad \sum_{i=1}^{k} \mu_i y_i = 0, \quad \mu_i \geq 0, \quad i = 1, 2, \ldots, k
\end{align*}
\]  

(3.25)

Solving the dual problem gives the Lagrangian multipliers \( \mu \), and it follows from equation 3.22 that:

\[ w = \sum_{i=1}^{k} \mu_i y_i x_i \]  

(3.26)

Nonzero \( \mu_i \) indicates \( x_i \) is a support vector of the classifier satisfying the equality sign in the linear constraints 3.13, and \( w \) can thereby be interpreted as a linear combination of the support vectors.

To obtain \( b \), we apply the constraints given by the fourth KKT condition:

\[ \mu_i [1 - y_i (\langle w, x_i \rangle + b)] = 0. \]  

(3.27)

Hence \( \bar{b} \) is given by:

\[ \bar{b} = -\frac{1}{2} \langle \bar{w}, \bar{x}_+ + \bar{x}_- \rangle, \]  

(3.28)
where $x_+$ and $x_-$ are any two support vectors from the two classes $y_+ = 1$ and $y_- = -1$, respectively\(^2\). The trained binary classifier is then:

$$y(x) = \text{sign}(\langle \sum_{i=1}^{k} \mu_i y_i x_i, x \rangle + \overline{b}).$$

(3.29)

### 3.4 Extended Model

#### 3.4.1 Soft margin SVM

Generally, the data points are not linearly separable, which means that mislabeled examples must be handled in order to extend the applications of SVMs to non-separable data. Moreover, even if linear separation is achievable, a solution better separating the great majority of the data while making a few exceptions (e.g. outliers or noisy samples) may still be preferred sometimes. SVMs allowing for misclassification on training data are called *soft margin* SVMs [70].

To implement a soft margin, the constraints 3.13 are relaxed by introducing non-negative slack variables $\zeta_i$:

$$y_i \left( \langle w, x_i \rangle + b \right) \geq 1 - \zeta_i, \quad i = 1, 2, \ldots, k. \quad (3.30)$$

For a mislabeled point $x_i$, $\zeta_i$ has to exceed 1. Cost $C\zeta_i$ is then assigned to each slack variable $\zeta_i$, where $C$ is a positive constant fixed globally. The minimization problem 3.17 is now revised to:

$$\begin{align*}
\text{minimize} & \quad \frac{\|w\|^2}{2} + C \sum_{i=1}^{k} \zeta_i \\
\text{subject to} & \quad y_i \left( \langle w, x_i \rangle + b \right) \geq 1 - \zeta_i \\
& \quad \zeta_i \geq 0, \quad i = 1, 2, \ldots, k
\end{align*} \quad (3.31)$$

\(^2\)Alternatively, $\overline{b}$ can be expressed as: $\overline{b} = y_i - \langle w, x_i \rangle$, where $x_i$ can be any support vector.
whose Lagrangian primal form is:

\[
\Lambda(w, b, \zeta, \mu, \lambda) = \frac{||w||^2}{2} + C \sum_{i=1}^{k} \zeta_i - \sum_{i=1}^{k} \mu_i[y_i((w, x_i) + b) - 1 + \zeta_i] - \sum_{i=1}^{k} \lambda_i \zeta_i. \tag{3.32}
\]

Applying the KKT conditions, we obtain:

\[
w - \sum_{i=1}^{k} \mu_i y_i x_i = 0 \tag{3.33}
\]

\[
\sum_{i=1}^{k} \mu_i y_i = 0 \tag{3.34}
\]

\[
C - (\mu_i + \lambda_i) = 0 \tag{3.35}
\]

\[
y_i((w, x_i) + b) - 1 + \zeta_i \geq 0 \tag{3.36}
\]

\[
\zeta_i \geq 0 \tag{3.37}
\]

\[
\mu_i \geq 0 \tag{3.38}
\]

\[
\lambda_i \geq 0 \tag{3.39}
\]

\[
\mu_i[y_i((w, x_i) + b) - 1 + \zeta_i] = 0 \tag{3.40}
\]

\[
\lambda_i \zeta_i = 0 \tag{3.41}
\]

Therefore, the dual problem is

\[
\text{maximize} \quad \sum_{i=1}^{k} \mu_i - \frac{1}{2} \sum_{i,j=1}^{k} y_i y_j \mu_i \mu_j \langle x_i, x_j \rangle
\]

subject to

\[
\sum_{i=1}^{k} \mu_i y_i = 0,
\]

\[
0 \leq \mu_i \leq C, \quad i = 1, 2, \ldots, k
\]

a form similar to problem 3.25 except for an upper bound term \(C\) placed on every Lagrange multiplier \(\mu_i\). Moreover, neither the slack variables \(\zeta_i\) nor their Lagrange multipliers \(\lambda_i\) appear in the dual. Again, \(\overline{w}\) can be obtained using the first KKT condition (equation 3.33), and \(\overline{b}\) can be found using the support vectors.
3.4.2 Multiclass SVM

Originally designed for binary classification, the SVM paradigm can further be extended to deal with multiclass data. There have been several ways to perform multiclass SVM classification [71]. Here we introduce two common practices that decompose the multiclass problem into multiple binary problems.

Probably the earliest technique used for multiclass SVM classification is the one-versus-all method. For a $C$-class problem, this strategy builds $C$ SVMs, where the $i$th classifier is constructed using examples from the $i$th class as the positive class and examples from all the other $C-1$ classes as the negative class. The test sample is then labeled by the SVM producing the largest functional margin.

Another popular approach is the one-versus-one (piecewise) method. It constructs $C(C-1)/2$ binary SVMs, each trained using two (a pair) of the $C$ classes. The “max wins” voting strategy is employed where a test sample is assigned the class labeled with the most votes. Although the number of classifiers to be trained increases considerably, the time for training each classifier may also decrease substantially, as the size of training data involved becomes much smaller.

3.4.3 Support vector regression

Besides the more often used support vector classification (SVC), SVMs can also be employed for regression, i.e. support vector regression (SVR). While SVC finds the hyperplane that maximizes the margin between two classes, SVR approximates most data points in $D = \{(x_1, y_1), (x_2, y_2), \ldots, (x_k, y_k)\}$ within $\epsilon$ precision (or $\epsilon$-margin), using a linear regression function $f(x) = \langle w, x \rangle + b$. The standard formulation of
CHAPTER 3. SUPPORT VECTOR MACHINES

soft-margin SVR is given by [63]:

\[
\text{minimize} \quad \frac{\|w\|^2}{2} + C \sum_{i=1}^{k} (\zeta_i + \zeta_i^*) \\
\text{subject to} \quad y_i - \langle w, x_i \rangle - b \leq \epsilon + \zeta_i \\
\quad \langle w, x_i \rangle + b - y_i \leq \epsilon + \zeta_i^* \\
\quad \zeta_i, \zeta_i^* \geq 0, \quad i = 1, 2, \ldots, k
\] (3.43)

where \(C\) is a positive constant as used in problem 3.30; \(\zeta_i\) and \(\zeta_i^*\) are upper and lower slack variables, respectively.

The Lagrangian primal of the above optimization problem is given by:

\[
\Lambda(w, b, \zeta, \zeta^*, \mu, \mu^*, \lambda, \lambda^*) = \frac{\|w\|^2}{2} + C \sum_{i=1}^{k} (\zeta_i + \zeta_i^*) - \sum_{i=1}^{k} \mu_i (\epsilon + \zeta_i - y_i + \langle w, x_i \rangle + b) - \sum_{i=1}^{k} \mu_i^* (\epsilon + \zeta_i^* + y_i - \langle w, x_i \rangle - b) - \sum_{i=1}^{k} \lambda_i \zeta_i - \sum_{i=1}^{k} \lambda_i^* \zeta_i^*
\] (3.44)

Hence the dual optimization problem is:

\[
\text{maximize} \quad -\frac{1}{2} \sum_{i,j=1}^{k} (\mu_i - \mu_i^*) (\mu_j - \mu_j^*) \langle x_i, x_j \rangle + \sum_{i=1}^{k} y_i (\mu_i - \mu_i^*) - \epsilon \sum_{i=1}^{k} (\mu_i + \mu_i^*)
\] (3.45)

subject to \(\sum_{i=1}^{k} (\mu_i - \mu_i^*) = 0,\)

\(0 \leq \mu_i, \mu_i^* \leq C, \quad i = 1, 2, \ldots, k\)

and the linear function \(f(x)\) is given by:

\[
f(x) = \langle \sum_{i=1}^{k} (\vec{P}_i - \vec{P}_i^*) x_i, x \rangle + \vec{b}
\] (3.46)

It can be shown that any support vector \(x_i\) has exactly one nonzero Lagrange multiplier [68].
3.5 Learning in Feature Space

So far, the discussion is restricted to linear functions only. On the other hand, real-world data sometimes may have complex boundaries, thereby requiring nonlinear separators. Such separators, while nonlinear in the original feature space, may still be linear in some transformed space. Therefore, we can non-linearly map the linearly non-separable data from the original input space to a new feature space where linear separation is possible, so that the classification task may be simplified. A simple illustration of this idea is depicted in Fig. 3.5, where the data clearly cannot be linearly separated in the original one-dimensional space. However, linear separation is attained after mapping the data into a two-dimensional space.

Denote the mapping as $\Phi$. Then the decision function given in equation 3.4 is
changed to:

\[ f(x) = \langle w, \Phi(x) \rangle + b, \quad (3.47) \]

and all the inner product terms \( \langle x_i, x_j \rangle \) appeared in previous sections are replaced with \( \langle \Phi(x_i), \Phi(x_j) \rangle \) accordingly. Nonlinear separators can then be learned via essentially the same procedures described above. However, since the new feature space is typically of higher dimension than the input space, the computational cost for training nonlinear SVMs is usually more expensive. Fortunately, the \textit{kernel trick} [62] provides an efficient way of performing the nonlinear mapping, by replacing the inner product \( \langle \Phi(x_i), \Phi(x_j) \rangle \) with a kernel function \( K(x_i, x_j) \) satisfying Mercer’s condition [62].

The idea exploits the fact that SVMs use only the inner product of data points for computation. Hence we can directly calculate the inner product in the new feature space as a function of the coordinates of the original data points instead of really doing the mapping. In this way, the amount of computation for nonlinear SVMs is still comparable to that of linear ones.

Learning in feature space and the kernel trick make the SVM technique much more flexible and powerful, capable of learning very complex separating boundaries. Some popular SVM kernels include:

- Linear (simply inner product):
  \[
  K(x_i, x_j) = \langle x_i, x_j \rangle, \quad (3.48)
  \]

- Polynomial:
  \[
  K(x_i, x_j) = (\langle x_i, x_j \rangle + c_0)^d, \quad (3.49)
  \]

  where \( d \) is the degree of the polynomial. In general, nonzero \( c_0 \) is preferred [68].
• Radial basis function (RBF):

\[ K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2), \quad (3.50) \]

where \( \gamma \geq 0 \) is called width parameter. Generally, the RBF kernel can be a reasonable first choice of kernel [68, 72].

• Sigmoid:

\[ K(x_i, x_j) = \tanh(\alpha \langle x_i, x_j \rangle + \beta). \quad (3.51) \]

The sigmoid kernel is also a commonly used kernel type as it is closely related to neural networks.

By using the kernel function \( K(x_i, x_j) \), the binary SVM classifier in equation 3.29 is now of the form:

\[ y(x) = \text{sign} \left( \sum_{i=1}^{k} \mu_i y_i K(x_i, x) + \bar{b} \right), \quad (3.52) \]

and the SVM regression function in equation 3.46 becomes:

\[ f(x) = \sum_{i=1}^{k} (\bar{\mu}_i - \mu_i) K(x_i, x) + \bar{b} \quad (3.53) \]

### 3.6 Summary

This chapter has introduced the SVM technique, which is used to build our emotion recognition system. The SRM principle that endows SVMs with good generalization performance has been described. The SVM model is first formulated for two-class linearly separable data, with the optimization problem discussed. The SVM paradigm has further been generalized to handle non-separable and multiclass data. SVR has also been presented. Lastly, nonlinear SVMs are described, and the kernel trick for efficient nonlinear learning has been introduced.
Chapter 4

Modulation Spectral Features

4.1 Motivation

Studies have shown that the modulators of speech signals are extremely important for speech perception [73, 74]. Dudley described in his work [75] on speech synthesis many decades ago that the intelligence content of speech is impressed on audible sound streams by “modulation processes of the true message-bearing waves which, by themselves, are inaudible”. In other words, the modulation signals of speech (i.e., the “message-bearing waves”) generated by the slowly-varying articulatory motions during speech production, though themselves varying at low and inaudible rates, modulate high-bandwidth carriers and dominate the intelligence content of the audible speech.

Moreover, recent research [11–14] uncovers the spectro-temporal (ST) processing in the human auditory system and thereby offers a finer picture about how speech modulations are interpreted by human listeners. In addition to the well-known auditory spectral decomposition [76], long-term modulation spectral content is further
extracted from the auditory spectrogram in the human auditory cortex [13, 14]. The importance of the modulation spectrum of speech, i.e. the frequency content of the modulation signal, has been evident in a number of areas, including auditory physiology, psychoacoustics, speech perception, as well as signal analysis and synthesis, as described in [74].

However, the modulation spectrum is not exploited by conventional spectral features based on short-term spectral representations such as MFCCs and PLPCs. In this chapter, long-term modulation spectral features (MSFs) are proposed. These more perceptually motivated features are derived from a long-term ST representation consistent with human ST processing of speech, hence capturing the modulation spectral content of speech ignored by traditional STSFs. The remainder of this chapter is organized as follows. Section 4.2 introduces the speech representation obtained via ST processing. Then in Section 4.3, several alternative algorithms for deriving the ST representation are compared. The novel MSFs are detailed in Section 4.4. Lastly, description of comparison features is given in Section 4.5.

4.2 Spectro-Temporal Representation

In order to extract the MSFs, an auditory-inspired long-term speech representation is first obtained, by emulating the ST processing performed in the human auditory system. The steps for computing the ST representation are illustrated in Fig. 4.1. The initial pre-processing module performs three tasks:

1. The speech signal is resampled to 8 kHz. The original speech files used in our experiment are sampled at 16 kHz. However, since emotions can be reliably
conveyed through band-limited telephone speech sampled at 8 kHz, the 8 kHz sampling rate is considered adequate for SER in this study.

2. The active speech level is normalized to -26 dBov using the P.56 speech voltmeter [77]. Normalization of energy level is usually beneficial for speech recognition, as it compensates for unwanted energy variations due to factors such as different microphone gains and different speakers.

3. The G.729 voice activity detection (VAD) algorithm [78] is applied to the speech signal, labeling speech regions as either active or inactive. Only active speech parts are retained.

The pre-processed speech signal, denoted as \( s(n) \), is framed into long-term windowed segments \( s_k(n) \), where \( k \) is the frame index. Windowing is done by multiplying an \( L \)-point Hamming window given by:

\[
w(n) = 0.54 - 0.46 \cos \left( \frac{2\pi n}{L-1} \right), \quad n = 0, 1, \ldots, L-1.
\]  

(4.1)

In this work, the frame length is set to 256 ms (i.e. \( L = 2048 \)) with 64 ms frame shift. The 256 ms duration is much longer relative to typical values used in traditional speech processing (around 20-30 ms). However, as described below, the first subband filter in the modulation filterbank performs frequency analysis at frequency
contents around 4 Hz. Thus this relatively long temporal span is necessary in order to acquire appropriate frequency resolution for such low modulation frequencies. Moreover, analyzing long-term temporal regions is in accordance with recent neuroscience findings [11, 12] that a typical ST receptive field in the human auditory cortex can extend up to several hundreds of milliseconds, and thereby helps to capture long-term temporal information used by human speech perception.

It is well-known that the human auditory system (more precisely, the front-end stage of it) can be modeled as a series of overlapping band-pass frequency channels [76], namely auditory filters with critical bandwidths increasing with filter center frequencies. The output signal of the $i$th critical-band filter at frame $k$ is given by:

$$s_k(i, n) = s_k(n) * h(i, n),$$

(4.2)

where $h(i, n)$ is the impulse response of the $i$th auditory channel, and $*$ denotes convolution. Here, a critical-band gammatone filterbank [79] is used. It is a commonly employed auditory model consisting of a group of overlapping gammatone filters. A gammatone filter in the time domain has an impulse response $g(t)$ characterized by:

$$g(t) = at^{n-1} \cos(2\pi f_c t + \phi)e^{-2\pi bt}, t \geq 0, n \geq 1$$

(4.3)

where $a$ is the amplitude; $n$ is the order of the gammatone filter; $f_c$ is the filter center frequency (in Hz); $\phi$ is the phase term; and $b$ is the filter bandwidth (in Hz) that directly impacts the duration of the impulse response. Fig. 4.2 shows the impulse response of a gammatone filter, where $a = 1, n = 4, f_c = 1000, \phi = 0, b = 150$. The center frequencies of the auditory filters, henceforth called *acoustic* frequency to distinguish from *modulation* frequency analyzed by the modulation filterbank, are proportional to their bandwidths, which in turn, are characterized by the equivalent
rectangular bandwidth (ERB) \[80\]:

\[
ERB_i = \frac{F_i}{Q_{ear}} + B_{min},
\]

where \(F_i\) is the center frequency (in Hz) of the \(i\)th critical-band filter, and \(Q_{ear}\) and \(B_{min}\) are constants set to 9.26449 and 24.7, respectively.

In this thesis, the gammatone filterbank realization in \[81\] is adopted. It implements the fourth-order gammatone filter (the case of the most interest according to \[82\]) as a cascade of four second-order infinite impulse response (IIR) filters. Denote the sampling rate as \(F_s\). For a gammatone filterbank containing \(N\) channels, the center frequency of the \(i\)th filter is given by \[81\]:

\[
F_i = (F_1 + QB) \exp\left(\frac{1}{N} \log\left(\frac{F_1 + QB}{F_s/2 + QB}\right)\right) - QB
\]

where \(1 \leq i \leq N\) and \(QB = Q_{ear} \cdot B_{min}\). It is easy to verify that the \(N\) filters are uniformly spaced on a log-ERB scale. In our experiment, a gammatone filterbank with \(N = 19\) filters is used, with \(F_1\) set to 125 Hz. Consequently, the first and the
last filters are centered at 125 Hz and 3.5 kHz, with bandwidths of 38 Hz and 400 Hz, respectively. The magnitude response of the filterbank is given in Fig. 4.3.

Using the gammatone filterbank for signal decomposition, $N = 19$ outputs $s_k(i, n)$ are obtained for frame $k$. The Hilbert envelopes $H_k(i, n)$ are then computed from the decomposed signals, corresponding to temporal amplitude modulations at different acoustic frequencies. In the time domain, the analytic signal $\hat{x}(n)$ (i.e. a signal with no negative-frequency components) of a real-valued signal $x(n)$ is complex with its real part being the original signal and its imaginary part being the Hilbert transform ($\mathbb{H}\{\cdot\}$) of it [83]:

$$\hat{x}(n) = x(n) + j\mathbb{H}\{x(n)\}. \quad (4.6)$$

Applying the Hilbert transform to a signal is equivalent to shifting the phase of the negative and the positive frequency components of the signal by $\pi/2$ and $-\pi/2$, respectively. Therefore, $\hat{x}(n)$ doubles the positive frequency components of $x(n)$ and
zeros the negative frequency components. Then the Hilbert envelope $\mathcal{H}(n)$ of $x(n)$ is simply the magnitude of the complex analytic signal $\hat{x}(n)$. For the critical-band signal $s_k(i, n)$, its Hilbert envelope $\mathcal{H}_k(i, n)$ is hence expressed as:

$$\mathcal{H}_k(i, n) = |\hat{s}_k(i, n)| = \sqrt{s_k^2(i, n) + \mathbb{H}^2\{s_k(i, n)\}}, \quad (4.7)$$

and the instantaneous phase $\phi_k(i, n)$ is given by:

$$\phi_k(i, n) = \arctan \frac{\mathbb{H}\{s_k(i, n)\}}{s_k(i, n)}. \quad (4.8)$$

Now $s_k(i, n)$ can be readily represented in terms of its modulator and carrier as:

$$s_k(i, n) = \mathcal{H}_k(i, n) \cos(\phi_k(i, n)). \quad (4.9)$$

Fig. 4.4 shows an example of a bandpassed speech segment from a filter centered at 650 Hz (subplot a) and its Hilbert envelope (subplot b).

The critical-band filterbank modeling has been frequently used in speech processing and is also involved in the computation of some STSFs. However, such auditory
spectral decomposition only comprises the early stage of the signal transformation
performed in the human auditory system. It has been shown that the output of this
early processing (spectrogram) is further interpreted by the neurons in the auditory
cortex to extract ST modulation patterns [13, 14]. Therefore, merely emulating the
early stage, as done during computing some STSFs, is inadequate. To this end, an
$M$-band modulation filterbank is employed in addition to the gammatone filterbank,
to model the functionality of the auditory cortex and incorporate the modulation
spectral information.

By applying the modulation filterbank to each $\mathcal{H}_k(i, n)$, $M$ outputs $\mathcal{H}_k(i, j, n)$ are
generated where $j$ denotes the $j$th modulation filter, $1 \leq j \leq M$. The modulation
filters used in this thesis are second-order bandpass filters with quality factor set to
2, as described in [84]. Alternative filter structures may be used as well. An example
can be found in [85] where modulation filters with triangular magnitude responses
are used for speech quality assessment. The number of modulation filters has been
varied from 4 to 8, and the finally used filterbank consists of $M = 5$ filters, whose
center frequencies are equally spaced on logarithm scale from 4 Hz to 64 Hz. The
filterbank is chosen as it is shown in a preliminary experiment to strike a good balance
between performance and model complexity. The magnitude response of the modu-
lation filterbank is depicted in Fig. 4.5. Also, note that by using longer frame length,
even lower modulation frequency content such as 2 Hz can be analyzed. However,
longer frame length (and accordingly larger frame shift) would reduce the number of
frames for each speech file, unfavorable to both UL and FL modeling. Considering
the file durations in the Berlin and the VAM databases, the center frequency of the
first modulation filter is set to 4 Hz.
Lastly, the ST representation $E_k(i, j)$ of the $k$th frame is obtained by measuring the energy of the decomposed modulation signals $H_k(i, j, n)$, given by:

$$E_k(i, j) = \sum_{n=1}^{L} |H_k(i, j, n)|^2,$$

where $1 \leq k \leq T$; $L$ is the number of samples in one frame and $T$ is the total number of frames. For a fixed $j = j^*$, $E_k(i, j^*)$ represents the auditory spectral samples in modulation channel $j^*$ after critical-band grouping. For a fixed $i = i^*$, an analogous interpretation can be made for $E_k(i^*, j)$, except that the energy grouping is carried out in the modulation frequency domain. An example of $E_k(i, j)$ from a neutral speech segment is illustrated in Fig. 4.6.

By incorporating the additional modulation filterbank, a richer two-dimensional frequency representation is enabled. This joint representation allows for analysis of modulation spectral content across different acoustic frequency channels. Fig. 4.7 shows the ST representation $E(i, j)$ for the seven emotions in the Berlin database,
Figure 4.6: $E_k(i, j)$ for one frame of a “neutral” speech file: low channel index indicates low frequency.

where every $E(i, j)$ is the average over all frames (over all speakers available in the database) of the corresponding emotion. As shown in the figure, the average ST patterns are similar for some emotions (e.g. anger and joy), suggesting they are likely to become notable confusion pairs, while very distinct for some others (e.g. anger and sadness), suggesting they are likely to be well discriminated from each other. Moreover, it can be seen from Figures 4.6 and 4.7 that most energy is concentrated in low modulation frequencies of the ST pattern, a phenomenon mainly due to the physical limitation and inertia of human vocal articulators [73].

4.3 Algorithm Comparison

In the algorithm described in Section 4.2, denoted as algorithm 1, the speech signal is first framed and then fed into the auditory filterbank. In this section, two alternative algorithms are further investigated, as depicted in Fig. 4.8, both of which process the whole speech signal using the auditory filterbank prior to framing and then calculate
the Hilbert envelopes from the critical-band signals. In algorithm 2, the modulation filterbank is applied to each temporal envelope; the outputs are framed and energy is calculated for each frame. In algorithm 3, however, the framing module is placed before the modulation filterbank.

Comparing the three algorithms, they differ in the target signal to be framed. For algorithms 2 and 3, however, due to the varying delay of the IIR filters in the two filterbanks (the lower center frequency, the larger delay), the output signals are misaligned without synchronization. In particular, the modulation filters have much larger delay compared to the auditory filters. As a result, inappropriate framing operation would potentially incur notable artifacts, as magnified by the following simple example where an active speech signal of 512 ms has been padded with a 512 ms silence segment. The frame length and shift are set to 256 ms and 64 ms, respectively, with framing done using a rectangular window for all the three algorithms.
Figure 4.8: Algorithms 2 and 3 for computing the ST representation.

to conduct a straightforward comparison.

The resulting ST patterns are shown in Fig. 4.9 for four successive frames (7th – 10th frames). Notice that the last two frames (frames #9 and #10) in the figure correspond to two completely silent segments in the time domain. As indicated in the figure, algorithms 1 and 3 result in fairly similar ST patterns, even though the latter displays slight energy residual for frame #9 due to the delay of the auditory filterbank, mostly in very low acoustic channels whose delay is relatively bigger. However, the energy leakage of algorithm 2 is much greater, as the ST energy residual is considerable for frame #9 and still observable for frame #10.

Such artifacts, and more essentially, the underlying misalignment of filter outputs, are shown to deteriorate SER performance. In Fig. 4.10, the FDR curves (a measure of feature discriminating power defined in Chapter 5, higher curve indicating better discriminating ability) are depicted for the MSFs (detailed in Section 4.4) extracted from the three ST representations derived using the three algorithms, where “SN” means speaker normalization (also defined in Chapter 5) has been applied to the
Figure 4.9: Comparison of three algorithms for deriving the ST representation.

features. As illustrated in the figure, the ST representation generated via algorithm 2 produces features with the lowest average FDR values, both with and without speaker normalization. On the other hand, FDR curves of algorithms 1 and 3 turn out to be very close. Such outcomes are consistent with the recognition results (not shown) that, algorithm 2 produces the poorest recognition accuracy while algorithms 1 and 3 convey comparable performance, implying that the small delay of the auditory filterbank hardly degrades the recognition performance but the delay due to the modulation filterbank is much more detrimental.

Comparing algorithms 1 and 3, however, some differences should still be pointed out. In algorithm 1, each speech frame is processed as an individual unit. The framed signal is used for critical-band decomposition and the envelopes are then calculated. On the other hand, the whole speech signal is used for critical-band decomposition and
temporal envelope computation in algorithm 3. Since there is no concrete evidence showing that either of the two sources, the framed speech segments or the whole speech signal, is clearly superior to the other for modulation frequency analysis, the two algorithms are considered comparable for SER in this thesis and, indeed, they give very close recognition performance. All the results achieved in Chapter 5 are produced using algorithm 1.

Moreover, as long as the filter delay in algorithm 2 has been handled properly, the algorithm may still be considered sound and hence adopted. The principal difficulties lay in the fact that the delay response of an IIR filter is not a constant function. A coarse way to approximately compensate for the large difference in delay between the modulation filters is to equalize the delay across different modulation frequency channels by the delay value at the filter center frequency of each filter where the magnitude response is maximum. Fig. 4.11 shows the ST patterns generated by
algorithm 2 after such rough delay equalization. As can be seen from the figure, the energy residual is substantially reduced after delay compensation. However, the recognition performance is still inferior to the other two algorithms according to simulation, indicating more sophisticated technique is required for delay handling which, though, goes beyond the scope of this study and will not be further discussed.

Figure 4.11: The ST representation derived using algorithm 2 before (left column) and after (right column) approximate delay equalization.
4.4 Modulation Spectral Features

Two types of MSFs are extracted from the ST representation by means of spectral measures and linear prediction parameters. We used these features instead of the direct energy samples, because they give considerably better discriminating power and higher recognition performance, as shown later in Chapter 5.

4.4.1 Spectral measures

For each frame, the ST representation $E_k(i, j)$ is scaled to unit energy before further computation, i.e. $\sum_{i,j} E_k(i, j) = 1$ for every $k$. The first set of MSFs consist of features given by six spectral measures $\Phi_1$–$\Phi_6$ calculated on a per-frame basis. For frame $k$, the first measure $\Phi_{1,k}(j)$ is defined as the mean of the energy samples belonging to the $j$th modulation channel ($1 \leq j \leq 5$):

$$\Phi_{1,k}(j) = \frac{\sum_{i=1}^{N} E_k(i, j)}{N}. \quad (4.11)$$

Parameter $\Phi_1$ captures the energy distribution of speech along the modulation frequency. The second spectral measure is the spectral flatness defined as the ratio of the geometric mean of a spectral energy measure to the arithmetic mean of it. For the ST pattern at frame $k$, the energy samples $E_k(i, j)$ are used as the spectral energy measure for modulation channel $j$, and $\Phi_2$ is thereby defined as:

$$\Phi_{2,k}(j) = \frac{\sqrt[\Phi_{1,k}(j)]{\prod_{i=1}^{N} E_k(i, j)}}{\Phi_{1,k}(j)}. \quad (4.12)$$

A spectral flatness value close to 1 indicates a flat spectrum, while a value close to 0 suggests a spectrum with widely different spectral amplitudes. The third spectral measure employed is the spectral centroid. For the $j$th modulation channel, $\Phi_3$ offers a measure of the “center of mass” of the spectrum in the channel and is computed
as:

\[ \Phi_{3,k}(j) = \frac{\sum_{i=1}^{N} f(i) E_k(i,j)}{\sum_{i=1}^{N} E_k(i,j)}, \]  

(4.13)

where \( f(i) \) is a frequency measure. Two forms of \( f(i) \) have been experimented: (1) \( f(i) \) being the center frequency (in Hz) of the \( i \)th critical-band filter in the auditory filterbank, and (2) \( f(i) \) being the index of the \( i \)th critical-band filter, i.e. \( f(i) = i \).

No remarkable difference in performance is observed between the two measures, and \( f(i) = i \) is thus chosen for simplicity. Moreover, it is noticed that adjacent modulation channels usually exhibit considerable correlation. Consequently, the flatness and the centroid parameters of adjacent modulation channels also display high correlation. In order to alleviate such information redundancy, \( \Phi_{2,k}(j) \) and \( \Phi_{3,k}(j) \) are computed for \( j \in \{1, 3, 5\} \) only.

In the three spectral measures described above, \( \Phi_3 \) is shown to be a particularly useful measure. Fig. 4.12 illustrates an example, where \( \overline{\Phi}_3(1) \) stands for the average of \( \Phi_{3,k}(1) \) computed over each utterance in the Berlin database belonging to one of the three basic emotions: anger, neutral, and sadness, and the probability density function (PDF) of \( \overline{\Phi}_3(1) \) is estimated as a unimodal Gaussian for each emotion. Using neutral as a reference point, anger and sadness display clear upward and downward shift of spectral centroid in acoustic frequency as shown in the figure, respectively. The result is consistent with the ST patterns of these three emotions depicted earlier in Fig. 4.7, where anger shows more notable high frequency energy than neutral and sadness and the ST energy for sadness is highly concentrated in very low acoustic channels. Even though the PDFs of sad and neutral overlap to some extent, good separation is shown for anger vs. neutral, and almost perfect discrimination is accomplished between anger and sadness, using only one spectral centroid feature.
Besides parameters that measure the spectral behavior of each individual modulation channel, additional spectral measures that measure the relationship of different modulation channels are computed. First, the 19 acoustic channels are grouped into four divisions, namely 1—4, 5—10, 11—15, and 16—19, denoted as $D_l$ ($1 \leq l \leq 4$). Energy is summed for each division: $E_k(l,j) = \sum_{i \in D_l} E_k(i,j)$. Then the modulation spectral centroid ($\Phi_4$) is calculated in a manner similar to Equation 4.13:

$$\Phi_{4,k}(l) = \frac{\sum_{j=1}^{M} jE_k(l,j)}{\sum_{j=1}^{M} E_k(l,j)}.$$  

(4.14)

While $\Phi_{3,k}(j)$ measures the spectral centroid in the acoustic frequency domain for modulation band $j$, $\Phi_{4,k}(l)$ measures the centroid in the modulation frequency domain for acoustic region $D_l$. The last two spectral measures $\Phi_{5,k}(l)$ and $\Phi_{6,k}(l)$ are the linear regression coefficient (slope) and the resulting regression error (in terms of root mean squared error, RMSE) obtained by fitting a first-degree polynomial to $E_k(l,j)$, in a least squares sense. By incorporating spectral measures $\Phi_4-\Phi_6$, information is extracted about the rate of change of the selected acoustic frequency regions, thereby
compactly capturing the temporal dynamic cues. In total, 23 features are obtained from the ST representation per frame in terms of spectral measures.

### 4.4.2 Linear prediction parameters

The second set of MSFs are obtained by applying linear predication (LP) analysis to selected modulation channels $j$ where $j \in \{1, 3, 5\}$. This selection of modulation channels is also for the purpose of reducing information redundancy caused by high correlation between adjacent channels. The spectral samples $E_k(i, j)$ in the $j$th modulation channel are padded to be symmetric, and the inverse discrete Fourier transform (IDFT) is applied to the padded sequence, yielding the autocorrelation function in the time domain. Then the autocorrelation method [86] is used to find the LP coefficients $a_k(n, j)$ in the $p$th-order all-pole model whose system function is given by:

$$ H(z) = \frac{1}{1 + \sum_{n=1}^{p} a_k(n, j) z^{-n}}. \quad (4.15) $$

In order to suppress local details but preserving the broad structure beneficial to recognition, a 5th-order all-pole model is used to approximate the spectral samples. The computational cost of this autoregressive modeling is negligible due to the low LP order ($p = 5$) and the small number of spectral samples per modulation channel ($N = 19$). The direct LP coefficients $a_k(n, j)$ are further converted into LP cepstral coefficients (LPCCs), and are denoted as $C_k(n, j)$ ($0 \leq n \leq 5$). The conversion is performed using the following recursive relationships [86]:

$$
C_k(0, j) = \log_e(\sigma^2)
$$

$$
C_k(n, j) = -a_k(n, j) - \frac{1}{n} \sum_{l=1}^{n-1} (n-l) a_k(l, j) C_k(n-l, j), \quad 1 \leq n \leq p
$$

(4.16)

where $\sigma^2$ is the prediction error variance. The LPCCs have been shown to be a more
robust and reliable feature set for speech recognition than the direct LP coefficients [46]. Together with the 23 aforementioned features, a total of 41 MSFs are calculated frame-by-frame.

4.5 Comparison Features

The proposed MSFs are compared to features based on MFCCs and PLPCs, two popular STSFs used in speech recognition. Moreover, due to the widespread use of prosodic features in SER, it is also important to investigate whether the MSFs can serve as useful additions to prosodic features. Therefore, both STSFs and prosodic features are extracted as benchmarks. They are described in this section.

4.5.1 Short-term spectral features

4.5.1.1 MFCC features

The MFCCs are extracted as the first type of STSFs for comparison with the proposed long-term MSFs. The speech signal is first filtered by a high-pass filter $H(z)$ with a pre-emphasis coefficient of 0.97, i.e.

$$H(z) = 1 - 0.97z^{-1}. \quad (4.17)$$

The first 13 MFCCs (including the zeroth order log-energy coefficient) are extracted from 25 ms Hamming-windowed speech frames every 10 ms. As a common practice, delta and double-delta MFCCs (sometimes referred to as velocity and acceleration coefficients, respectively) are calculated as well to capture local dynamics, forming a 39-dimensional FL feature vector. Although the particular UL MFCC features extracted for SER vary from author to author, the most frequently used ones are
mean and standard deviation (or variance) of the first 13 MFCCs and their deltas
\[24, 35, 36, 47, 50, 61\]. In this work, mean, standard deviation, and 3rd–5th central
moments of the first 13 MFCCs and their delta and double-delta coefficients are
calculated, generating totally 195 MFCC features. This MFCC feature set is an
extension to the ones used in [35, 47], by further considering the delta coefficients.

4.5.1.2 PLP features

In addition to MFCCs, PLPCs are also extracted from speech, serving as an alter-
native choice of STSFs for comparison. Even though MFCCs are historically more
popular, PLPCs are still useful features exploiting human perception properties, more
consistent with human hearing than conventional LP coefficients. A 5th-order model
is employed for PLP analysis as suggested in [44]. The obtained coefficients are fur-
ther transformed to cepstral coefficients \(c(n)\) \((0 \leq n \leq 5)\). Delta and double-delta
coefficients are extracted too. The same statistical parameters as used for MFCCs
are then calculated for the PLPCs and their deltas, giving 90 UL PLP features.

4.5.2 Prosodic features

Prosodic features have become the most widely used feature type for SER. Hence
they are employed here as a benchmark, and more importantly, to verify whether
the MSFs can serve as useful additions to the extensively used prosodic features.
Totally 75 prosodic features are extracted as listed in Table 4.1. Pitch is computed
for voiced speech regions using the robust pitch tracking algorithm proposed in [87],
and intensity is measured for active speech in dB. Note that shimmer is computed for
the trajectories of pitch and intensity only, not for their deltas. For a discrete-time
sequence \( x_n \) of length \( N_x \), shimmer is calculated as:

\[
\text{shimmer} = \frac{1}{N_x-1} \sum_{n=1}^{N_x-1} |x_n - x_{n+1}| \sum_{n=1}^{N_x} x_n.
\] (4.18)

Features describing speaking rate are also extracted using syllabic and voicing information as shown in the table. Moreover, the zero-crossing rate (ZCR) and the Teager energy operator (TEO) [88] of the speech signal are calculated, though they are not directly related to speech prosody. TEO extracts useful information about the nonlinear airflow structure of speech production [89]. For \( x_n \), its TEO is defined as:

\[
\text{TEO}(x_n) = x_n^2 - x_{n+1}x_{n-1}.
\] (4.19)

The mean Teager energy of the speech signal is used as a feature, and is found to be an effective feature in simulation (cf. the Appendix).

<table>
<thead>
<tr>
<th>Table 4.1: List of prosodic features</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>pitch, intensity, delta-pitch, delta-intensity</strong></td>
</tr>
<tr>
<td>mean, std. dev., skewness, kurtosis, shimmer, maximum, minimum, median, quartiles, range, differences between quartiles, linear &amp; quadratic regression coefficients, regression error (RMSE)</td>
</tr>
<tr>
<td><strong>speaking rate</strong></td>
</tr>
<tr>
<td>mean and standard deviation of syllable durations ratio between the duration of voiced and unvoiced speech</td>
</tr>
<tr>
<td><strong>others</strong></td>
</tr>
<tr>
<td>zero-crossing rate (ZCR) mean Teager energy</td>
</tr>
</tbody>
</table>
4.6 Summary

This chapter has proposed a novel set of spectral features, known as modulation spectral features (MSFs). They aim to overcome the shortcomings of conventional STSFs. A long-term ST representation has been described, which is obtained by emulating the ST processing performed in the human auditory system and considers regular acoustic frequency jointly with modulation frequency. Alternative algorithms for deriving the ST representation have also been discussed. Extracted from the ST representation, the MSFs capture important modulation spectral content ignored by traditional spectral features. Two types of STSFs as well as prosodic features extracted for comparison purpose have also been described. The SVM recognition results with different feature types are presented in the following Chapter 5.
Chapter 5

Simulation Study

5.1 Experimental setup

In this chapter, simulation results are presented and discussed. The MSFs proposed in Chapter 4 are evaluated on the Berlin database and the VAM database. Description of the two databases can be found in Section 2.3.1. All results achieved on the Berlin database for discrete emotion classification (Section 5.2) are produced using 10-fold cross-validation. Cross-validation is a common practice used in performance analysis that randomly partitions the data into \( N \) complementary subsets, with \( N - 1 \) of them used for training in each validation and the remaining one used for testing. On the other hand, results for continuous emotion regression (Section 5.3) achieved on the VAM database are obtained using leave-one-out (LOO) cross-validation. The LOO setup is adopted in order to facilitate comparison with the machine recognition results reported in [35].

The SVM implementation in [90] is used for simulation. As suggested in [72], features from training data are linearly scaled to \([-1, 1]\) before applying SVM, with
features from test data scaled using the trained linear mapping function, i.e. if the training features are in the range \([f_{\text{min}}, f_{\text{max}}]\), a feature \(f\) is scaled to \(f_{\text{scale}}\) using the following transformation:

\[
f_{\text{scale}} = \frac{2}{f_{\text{max}} - f_{\text{min}}} f - \frac{f_{\text{max}} + f_{\text{min}}}{f_{\text{max}} - f_{\text{min}}}
\]  

(5.1)

Scaling features before SVM recognition is important, for it avoids features with large numeric values dominating features with small numeric values when computing the inner product and also reduces numeric difficulties during computation [72].

## 5.2 Discrete Emotion Classification

In this section, experimental evaluation is performed on the Berlin database to classify seven discrete emotions. All classification results to follow are obtained under 10-fold cross-validation. A two-stage scheme is first proposed in Section 5.2.1 for dimensionality reduction. The effect of taking a speaker normalization (SN) step prior to recognition is investigated, and several SN methods are compared in Section 5.2.2. Section 5.2.3 further draws comparison of different SVM kernels. The MSFs are then compared to features based on direct energy samples of the ST representation (Section 5.2.4), MFCC and PLP features (Section 5.2.5), and prosodic features (Section 5.2.6).

### 5.2.1 Dimensionality reduction

Training learning algorithms using all available features might not provide the best recognition performance due to the curse of dimensionality [91]. To this end, a two-stage scheme is proposed for dimensionality reduction. The first stage uses the Fisher
discriminant ratio (FDR) to rank each feature individually. The normalized $C$-class FDR for the $u$th feature is defined as:

$$ FDR(u) = \frac{2}{C(C-1)} \sum_{c_1} \sum_{c_2} \frac{(\mu_{c_1,u} - \mu_{c_2,u})^2}{\sigma_{c_1,u}^2 + \sigma_{c_2,u}^2}, \quad (5.2) $$

with $1 \leq c_1 < c_2 \leq C$, where $\mu_{c_1,u}$ ($\mu_{c_2,u}$) and $\sigma_{c_1,u}^2$ ($\sigma_{c_2,u}^2$) are mean and variance of the $u$th feature for the $c_1$th ($c_2$th) class, respectively. The normalization factor is set to the number of binary comparisons made between two classes.

The FDR measure defined in equation 5.2 borrows the criterion used in Fisher linear discriminant analysis (LDA), as it favors features with well-separated means across classes and small within-class variances. In addition, this FDR measure is invariant to scaling of features by any nonzero real number. Irrelevant features of little discriminating power (i.e. “noisy” features) can then be quickly eliminated by FDR thresholding. We used a threshold empirically set to 0.15, removing roughly 10% of the proposed features and 15% of prosodic features, but up to 50% of MFCC features and 40% of PLP features. Therefore, the FDR pre-screening step is particularly necessary for the two STSF pools, which would otherwise be rather noisy for feature mining.

In the second stage, two techniques are assessed to obtain good features from the pre-screened feature pools. The first technique is the sequential forward selection (SFS) [92]. The algorithm iteratively augments the selected feature subset and considers the combined effect of features and classifier during the evaluation process. Let $F = \{f_u | u = 1, 2, \ldots, N_f\}$ stand for the candidate feature pool of $N_f$ features, $X_k = \{x_u | u = 1, 2, \ldots, k, x_u \in F\}$ for the feature set selected by SFS at the $k$th iteration, $N_d$ ($N_d \leq N_f$) for the desired number of features, and $J$ for the objective function. The SFS algorithm is sketched below:
(0) initialization: \( X_0 \leftarrow \emptyset, k \leftarrow 0; \)

(1) select the best feature from the remaining feature set that maximizes \( J \):

\[
x^+ \leftarrow \arg \max_{f \in F-X_k} J(X_k + f);
\]

(2) augment the feature set: \( X_{k+1} \leftarrow X_k + x^+, k \leftarrow k + 1; \)

(3) if \( k < N_d \), goto (1); otherwise terminate.

In our experiment, the SFS algorithm is found to outperform the more sophisticated sequential floating forward selection (SFFS) algorithm [93]. Moreover, it offers a convenient way to visualize changes of recognition accuracy as the selected feature subset evolves, thus providing a straightforward performance comparison of different feature types.

The second technique is the well-known multi-class Fisher LDA. It finds the transformation optimizing the Fisher objective, which in turn maximizes the between-class distance and minimizes the within-class distance simultaneously. Consider a \( C \)-class problem where a batch of samples \( \mathbf{x} \) (each a column vector) are given. Define the “between-class scatter matrix” \( \mathbf{S}_b \) and the “within-class scatter matrix” \( \mathbf{S}_w \) as:

\[
\mathbf{S}_b = \sum_{c=1}^{C} N_c (\mathbf{\mu}_c - \bar{\mathbf{x}}) (\mathbf{\mu}_c - \bar{\mathbf{x}})^T, \tag{5.3}
\]

\[
\mathbf{S}_w = \sum_{c=1}^{C} \sum_{\mathbf{x} \in \text{class } c} (\mathbf{x} - \mathbf{\mu}_c) (\mathbf{x} - \mathbf{\mu}_c)^T, \tag{5.4}
\]

where \( N_c \) and \( \mathbf{\mu}_c \) are sample size and sample mean of the \( c \)th class, respectively; \( \bar{\mathbf{x}} \) is the mean of all the samples; and \( \{\cdot\}^T \) denotes taking the transpose. Then LDA maximizes the objective function \( J(\mathbf{w}) \) given by:

\[
J(\mathbf{w}) = \frac{\mathbf{w}^T \mathbf{S}_b \mathbf{w}}{\mathbf{w}^T \mathbf{S}_w \mathbf{w}}, \tag{5.5}
\]
It can be shown that the solutions maximizing $J(w)$ are the eigenvectors of the following generalized eigen-problem [94]:

$$S_b w = \lambda S_w w.$$  

(5.6)

The eigenvectors are then used to form the columns of the transformation matrix $W$ which would transform data point $x$ to $y = W^T x$. The components of $y$ constitute the new LDA-transformed features. Moreover, since the maximum rank of $S_b$ is $C - 1$, the number of features given by LDA is no more than $C - 1$. Therefore, LDA can render a drastic reduction in feature dimensionality for high-dimensional data, effectively alleviating the curse of dimensionality. It is hence particularly useful in practice if the size of the training data is limited relative to the number of features. The main limitation of LDA is that it cannot be applied to regression problems. In our seven-class SER case, the maximum number of meaningful features returned by LDA is six. All six LDA-transformed features are used to train SVM classifiers.

### 5.2.2 Speaker normalization

Speaker normalization (SN) is useful to compensate for the variations due to speaker diversity rather than change of emotional state. The effect of taking an SN step prior to SVM recognition is thus studied. Three different SN schemes are compared:

1. In Scheme 1 (SN 1), features are mean and variance normalized within the scope of each speaker. Let $f_{u,v}(n)$ ($1 \leq n \leq N_{u,v}$) stand for the $u$th feature from speaker $v$ where $N_{u,v}$ is its sample size. Then the new feature $f_{u,v}^{SN1}(n)$ after SN is given by:

$$f_{u,v}^{SN1}(n) = \frac{f_{u,v}(n) - \bar{f}_{u,v}}{\sqrt{\frac{1}{N_{u,v}-1} \sum_{m=1}^{N_{u,v}} (f_{u,v}(m) - \bar{f}_{u,v})^2}},$$

(5.7)
where \( f_{u,v} = \frac{1}{N_{u,v}} \sum_{n=1}^{N_{u,v}} f_{u,v}(n) \).

2. Scheme 2 (SN 2) is similar to Scheme 1, except that neutral is taken as a reference point for normalization. Let \( f_{neu}^{u,v}(n) (1 \leq n \leq N_{neu}^{u,v}) \) stand for the \( u \)th feature of neutral utterances from speaker \( v \) where \( N_{neu}^{u,v} \) is its sample size.

Then the new feature \( f_{SN2}^{u,v}(n) \) with SN applied is:

\[
f_{SN2}^{u,v}(n) = \frac{f_{u,v}(n) - f_{neu}^{u,v}}{\sqrt{\frac{1}{N_{neu}^{u,v}} \sum_{m=1}^{N_{neu}^{u,v}} (f_{neu}^{u,v}(m) - f_{neu}^{u,v})^2}},
\]

(5.8)

where \( f_{neu}^{u,v} = \frac{1}{N_{neu}^{u,v}} \sum_{n=1}^{N_{neu}^{u,v}} f_{neu}^{u,v}(n) \).

3. In Scheme 3 (SN 3), features from each speaker are linearly scaled from the original numerical range \([f_{min}, f_{max}]\) to \([-1, 1]\), i.e.

\[
f_{SN3}^{u,v}(n) = \frac{2}{f_{max}^{u,v} - f_{min}^{u,v}} f_{u,v}(n) - \frac{f_{max}^{u,v} + f_{min}^{u,v}}{f_{max}^{u,v} - f_{min}^{u,v}}.
\]

(5.9)

To compare the SN schemes, features are ranked by their FDR values (calculated using all data available in the Berlin database). Curves depicting the FDR values averaged over the top \( N_{fdr} \) FDR-ranked features are shown in Figures 5.1–5.4 as a function of \( N_{fdr} \), where the three SN schemes are applied to different feature types. These average FDR curves can be viewed as rough indicators of feature discriminating power independent of the specific machine learning algorithm used. Higher curves suggest better discriminating ability.

As can be seen from the figures, SN 1 consistently provides the best FDR performance for all the four feature types: the proposed MSFs, MFCC features, PLP features, and prosodic features (denoted as “PROS”), while SN 2 results in significantly inferior outcomes (even worse than the original features without SN). Recognition results (not shown) are also in agreement with FDR evaluation, as schemes 1 and 2
Figure 5.1: Average FDR curves of MSFs.

Figure 5.2: Average FDR curves of MFCC features.
CHAPTER 5. SIMULATION STUDY

Figure 5.3: Average FDR curves of PLP features.

Figure 5.4: Average FDR curves of prosodic features.
yield the best and the worst recognition performance, respectively. The low performance of Scheme 2 might be due to the insufficient number of neutral samples from each speaker incapable of obtaining reliable sample statistics. Thus SN 1 is adopted as the speaker normalization scheme, henceforth simply referred to as SN, to generate all the following SN results.

5.2.3 SVM kernel selection

As described in Section 3.5, nonlinear SVMs can be applied in an efficient way through the kernel trick that replaces the inner product computed in linear SVMs by a kernel function. Therefore, it would be useful to evaluate the SER performance produced by SVMs with different kernels, before drawing comparison of different feature types. Three kernel functions are investigated here:

- Linear (Lin) kernel:

  \[ K(x_i, x_j) = \langle x_i, x_j \rangle; \quad (5.10) \]

- Polynomial (Poly) kernel of degree 2:

  \[ K(x_i, x_j) = (\langle x_i, x_j \rangle + c_0)^2, \quad (5.11) \]

- Radial basis function (RBF) kernel:

  \[ K(x_i, x_j) = \exp(-\gamma\|x_i - x_j\|^2), \gamma \geq 0, \quad (5.12) \]

SVMs with the three kernels are then trained using different features. The design parameters of SVMs (i.e. the \( C \) value in soft margin SVM, the width parameter \( \gamma \) in the RBF kernel, and the coefficient \( c_0 \) in the polynomial kernel) are tuned using training data via a grid search on a base-2 logarithmic scale. Each feature type
CHAPTER 5. SIMULATION STUDY

Figure 5.5: Performance comparison of different SVM kernels (MSFs).

Figure 5.6: Performance comparison of different SVM kernels (MFCC features).
Figure 5.7: Performance comparison of different SVM kernels (PLP features).

Figure 5.8: Performance comparison of different SVM kernels (prosodic features).
has already been screened by FDR thresholding. Recognition results using SFS are shown in Figures 5.5–5.8, as a function of the number of features selected by SFS. The results have been averaged over the 10 cross-validation trials, and thus the accuracy trajectories in the figures depict the average recognition rate calculated as the number of samples from all emotions correctly recognized divided by the total number of samples.

As shown in the four figures, the RBF kernel furnishes slightly better accuracy than the other two kernels (though not always), especially for MFCC features (illustrated in Fig. 5.6), and it might also be concluded that the linear kernel often leads to the lowest accuracy. In [72], it is justified that the RBF kernel can be a good choice of kernel function in general, as it has a number of advantages over other kernels:

- it can model the nonlinear relation between attributes and target values well;
- the linear kernel is a special case of the RBF kernel;
- it has fewer parameters (only $\gamma$) than the polynomial kernel (both $c_0$ and degree);
- it has fewer numerical difficulties compared to the polynomial and the sigmoid kernels.

Therefore, the RBF kernel is used in all the following SVM classification experiments.

### 5.2.4 Comparison with ST sample features

Recall that the proposed MSFs are extracted from the ST representation by means of spectral measures and linear prediction parameters, instead of taking the energy
samples directly. Such exploitation of the ST representation, as indicated by the comparison below between the MSFs and the feature set consisting of features based on the direct ST energy samples (named ST sample features), considerably improves the recognition performance. To form the ST sample features, the 95 \((19 \times 5)\) ST energy samples of the ST representation are used as FL features, with their mean and standard deviation calculated as UL features.

The average FDR curves of the two feature types (UL MSFs and UL ST sample features) are depicted in Fig. 5.9. It is indicated in the figure that while the ST sample features give low discriminating power, the MSFs obtained after mining the ST representation show considerably better discriminating ability. Moreover, the SFS results using the two feature types are given in Fig. 5.10. Again, the advantage of the proposed MSFs is notable, as they consistently convey significantly better recognition accuracy than the ST sample features.

Combing Figures 5.9 and 5.10, it is evident that the proposed MSFs form a better feature representation for emotion recognition than the direct ST sample features. This implies that though the ST representation itself bears valuable information for SER, a careful exploitation of it is necessary in order to extract discriminating features. Further investigation of the ST representation may yield even more powerful features. However, this is the goal of our future work. Moreover, as demonstrated by the remainder of this chapter, the current MSFs already give good recognition performance, outperforming conventional spectral features and serving as powerful additions to prosodic features.
CHAPTER 5. SIMULATION STUDY

Figure 5.9: Average FDR curves of MSFs and ST sample features.

Figure 5.10: Comparison between MSFs and ST sample features.
5.2.5 Comparison with short-term spectral features

The proposed long-term MSFs are compared to two representative STSFs in this section, namely MFCC features and PLP features (described in Section 4.5.1). The discriminating ability of the three feature types is first evaluated, in terms of average FDR curves as shown in Fig. 5.11 for cases with and without SN. As indicated in the figure, the MSFs consistently exhibit substantially better discriminating power than the other two STSFs, whose FDR curves are relatively close. Speaker normalization is beneficial to emotion recognition, boosting the average FDR curves of all feature sets. However, choosing features based solely on their FDR scores can be hazardous, because FDR only evaluates individual features rather than feature combinations. Effects such as feature correlation have not been taken into account. Moreover, it might also be useful to consider the classifier properties during feature selection. Therefore, a subsequent step after FDR pre-screening is necessary, in order to find better feature combinations, as performed by the second stage of our dimensionality reduction scheme.

The numeric results of applying SFS and LDA in the second stage are detailed in Table 5.1, where RBF-SVMs are employed for classification in the feature space. For SFS, the results shown in Table 5.1 are for the number of features that yields the best average performance. Notice that an upper limit of 50 has been put to the number of SFS-selected features, because the pre-screened PLP feature pool consists of 51 features only. Terminating the SFS algorithm at 50 features equalizes the maximum number of features that can be selected from each feature type, and thus forces a fair comparison. For LDA, the six LDA-transformed features are used, and the resulting performance is shown in the table.
Figure 5.11: Average FDR curves of MSFs, MFCC features and PLP features.

From Table 5.1, it can be concluded that the proposed features achieve the best accuracy for most emotion classes and in overall performance, with 85.6% average recognition rate reached. Similar to the FDR evaluation case, applying SN improves recognition results, for both SFS and LDA reach higher recognition accuracy after SN. The most notable performance gain by taking SN is achieved for the MSFs using LDA (from 78.9% to 85.6%). For the MSF set, the 85.6% accuracy furnished by six LDA features is even better than the 82.8% accuracy furnished by 40+ SFS-selected features. Such outcome is likely due to the effective reduction of feature dimensionality given by LDA that enables more sufficient and reliable training of classifiers. However, as further illustrated in comparison with prosodic features (Section 5.2.6), applying SN is observed to be indispensable for high-performance LDA.

The SFS accuracy trajectories are further pictured in Fig. 5.12 for the three feature types. The figure shows that the MSFs not only prevail at the best-performing number
Table 5.1: Recognition results with MSFs, MFCC features, and PLP features; “AVE.” denotes average recognition rate; $\sigma$ denotes standard deviation of the 10 cross-validation accuracies.

<table>
<thead>
<tr>
<th>Test</th>
<th>Feature</th>
<th>Method</th>
<th>SN</th>
<th>Recognition Rate (%)</th>
<th>AVE. ($\sigma$)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Anger</td>
<td>Boredom</td>
</tr>
<tr>
<td>#1</td>
<td>MSF</td>
<td>SFS</td>
<td>No</td>
<td>92.1</td>
<td>86.4</td>
</tr>
<tr>
<td></td>
<td>MFCC</td>
<td>SFS</td>
<td>No</td>
<td>91.3</td>
<td>80.3</td>
</tr>
<tr>
<td></td>
<td>PLP</td>
<td>SFS</td>
<td>No</td>
<td>89.8</td>
<td>79.0</td>
</tr>
<tr>
<td>#2</td>
<td>MSF</td>
<td>LDA</td>
<td>No</td>
<td>85.8</td>
<td>81.5</td>
</tr>
<tr>
<td></td>
<td>MFCC</td>
<td>LDA</td>
<td>No</td>
<td>83.5</td>
<td>85.2</td>
</tr>
<tr>
<td></td>
<td>PLP</td>
<td>LDA</td>
<td>No</td>
<td>88.2</td>
<td>74.1</td>
</tr>
<tr>
<td>#3</td>
<td>MSF</td>
<td>SFS</td>
<td>Yes</td>
<td>89.8</td>
<td>88.9</td>
</tr>
<tr>
<td></td>
<td>MFCC</td>
<td>SFS</td>
<td>Yes</td>
<td>88.2</td>
<td>82.7</td>
</tr>
<tr>
<td></td>
<td>PLP</td>
<td>SFS</td>
<td>Yes</td>
<td>90.6</td>
<td>72.8</td>
</tr>
<tr>
<td>#4</td>
<td>MSF</td>
<td>LDA</td>
<td>Yes</td>
<td>90.6</td>
<td>87.7</td>
</tr>
<tr>
<td></td>
<td>MFCC</td>
<td>LDA</td>
<td>Yes</td>
<td>85.0</td>
<td>79.0</td>
</tr>
<tr>
<td></td>
<td>PLP</td>
<td>LDA</td>
<td>Yes</td>
<td>90.6</td>
<td>80.3</td>
</tr>
</tbody>
</table>

of features as listed in Table 5.1, but consistently outperform MFCC and PLP features with feature number from 1 to 50, irrespective of the SN step. However, performing SN boosts the accuracy curves more notably for the two STSFs than for the proposed features. Moreover, it is evident from Fig. 5.12 that MFCC features are more suitable for emotion recognition relative to PLP features, despite the FDR results suggesting the two feature types to be comparable. Such observation resonates with the fact mentioned earlier that feature selection merely based on FDR scores is inadequate. Finer procedures are needed to obtain good feature combinations after FDR pre-screening.

Recognition is further performed using noisy speech data, where clean speech is corrupted by additive white noise. Even though real-world noise can be much more complex, this simple noise assumption already provides some useful insights. The recognition accuracies for different features (without SN) are listed in Table 5.2, with signal-to-noise ratio (SNR) levels varying from 40 dB (almost clean) to 0 dB (highly
Figure 5.12: Recognition results with MSFs, MFCC features, and PLP features using SFS.

noisy) at a 5 dB step. When varying the SNR level, it can be seen from the table that the two STSFs are highly sensitive to noise. A sharp drop in recognition rate is witnessed for them (especially MFCC features) even under slightly noisy conditions, and large SNR value is needed to maintain acceptable SER performance. On the other hand, the proposed features demonstrate clearly better robustness against noise than MFCC and PLP features, and still work reasonably well under moderate SNR (e.g. 20 dB).

5.2.6 Comparison with prosodic features

Due to the widespread use of prosodic features in current SER systems, it is important to study the contribution of the proposed MSFs as complementary features. However, before focusing on prosodic and proposed features, it is useful to consider the two
Table 5.2: Recognition accuracies at selected SNR levels (training on clean speech and testing on noisy).

<table>
<thead>
<tr>
<th>Test</th>
<th>Feature</th>
<th>Method</th>
<th>Recognition Rate (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>clean</td>
<td>40dB</td>
</tr>
<tr>
<td>#1</td>
<td>MSF</td>
<td>SFS</td>
<td>79.6</td>
</tr>
<tr>
<td></td>
<td>MFCC</td>
<td>SFS</td>
<td>76.5</td>
</tr>
<tr>
<td></td>
<td>PLP</td>
<td>SFS</td>
<td>71.2</td>
</tr>
<tr>
<td>#2</td>
<td>MSF</td>
<td>LDA</td>
<td>78.9</td>
</tr>
<tr>
<td></td>
<td>MFCC</td>
<td>LDA</td>
<td>77.9</td>
</tr>
<tr>
<td></td>
<td>PLP</td>
<td>LDA</td>
<td>71.4</td>
</tr>
</tbody>
</table>

STSFs first in accordance with the previous Section 5.2.5. Therefore, MFCC and PLP features are combined with prosodic features and compared to combined prosodic and proposed features. The corresponding SFS results are given in Fig. 5.13, where the proposed features achieve the highest recognition rate, either with SN (MSF: 87.7%, MFCC: 86.5%, PLP: 87.3%) or without SN (MSF: 85.4%, MFCC: 84.1%, PLP: 81.9%). Although the best performance of different feature combinations might seem close in Fig. 5.13, the advantage of the proposed features is more notable in Fig. 5.14. The label $N_{\text{best}}$ denotes that the $N_{\text{best}}$ ($1 \leq N_{\text{best}} \leq 50$) highest recognition rates (among the 50 recognition rates obtained by selecting 1 to 50 features using SFS) are chosen, whose average is calculated and depicted in the figure. As shown in Fig. 5.14, the MSFs always furnish the highest average recognition rate relative to MFCC and PLP features. Similarly, they also attain the best performance in the LDA case, either with SN (MSF: 91.6%, MFCC: 85.8%, PLP: 84.7%) or without SN (MSF: 79.3%, MFCC: 75.3%, PLP: 76.8%).

Having shown that the MSFs are better additions to prosodic features than the two STSFs, we now draw comparisons between the MSFs and prosodic features, using (1) only prosodic features, and (2) combined prosodic and proposed features. The
CHAPTER 5. SIMULATION STUDY

Figure 5.13: Comparison of different combinations of spectral features and prosodic features.

Figure 5.14: Average recognition performance of different combinations of spectral features and prosodic features using $N_{\text{best}}$. 
Figure 5.15: Average FDR curves of prosodic features, MSFs, and their combination.

Figure 5.16: Comparison between prosodic features and their combination with MSFs.
average FDR curves of the two feature types and their combination are depicted in Fig. 5.15. As can be seen from the figure, when evaluated individually, the proposed features are superior to prosodic features in terms of average FDR scores, and after the inclusion of the MSFs, the discrimination power of the prosodic feature pool can be substantially enhanced. Both feature types benefit considerably from SN. The SFS accuracy trajectories are further illustrated in Fig. 5.16 where the contribution of the MSFs is notable in the figure for cases both with and without SN. A list of top features selected by SFS from prosodic features, the MSFs, and their combination (with SN) can be found in the Appendix.

Recognition rate is shown for each emotion in Table 5.3. The LDA results are included as well. The row labeled “↓ %” indicates the percentage reduction of error rate obtained by adding the MSFs to prosodic features, calculated as:

\[ \downarrow \% = \frac{RR_{PROS+MSF} - RR_{PROS}}{1 - RR_{PROS}} \times 100\%, \]

where “RR” represents recognition rate. Notice that the importance of applying SN is particularly evident by comparing Tests #2 and #4 where LDA is used. Two confusion matrices are shown in Tables 5.4 and 5.5 (left-most column being the true emotions), for the best performance achieved by using prosodic features (85.2%, SN+LDA) and combined prosodic and proposed features (91.6%, SN+LDA), respectively. The Rate column and the Precision row list the Recall (taken as recognition rate) and Precision values for each emotion, respectively\(^1\). As shown in the confusion matrices, including the MSFs improves the recognition and precision rates of all emotion categories. Also, most emotions can be well recognized with accuracy

---

\(^1\)Recall for a class is the number of samples correctly classified divided by the total number of samples that actually belong to the class. Precision for a class is the number of samples correctly classified divided by the total number of samples classified to the class.
from 89.9% (fear) to 100% (sadness) except joy. It forms the most notable confusion pair with anger, though they are of opposite valence. This might be due to the fact that activation is more easily recognized by machines than valence, as suggested in the regression results for the emotion primitives on the VAM database presented in Section 5.3.

It would also be useful to briefly review good results achieved on the Berlin database by other works. More details on these works have been given in Section 2.3.4.1. Although the performance figures cannot be directly compared due to different experimental settings, the indirect comparison is still useful as a benchmark. For works that do not use SN, 86.7% recognition rate is achieved in [54], by using approximately 4000 features. In [47], 88.8% accuracy is achieved by employing a three-stage classification scheme with more than 300 features extracted, but based on recognition of six emotions only (no disgust). For works that employ SN, 83.2% recognition rate is obtained in a speaker-independent experiment [50] by extracting around 1400 features. The accuracy is further improved to 89.9% by joint modeling of UL and FL features. Roughly speaking, the proposed SVM-based SER system with combined prosodic and proposed features is comparable to state-of-the-art algorithms in terms of accuracy (91.6%). Moreover, we have used a much more compact feature set (100+ vs. 1000+) with six LDA features as final input to SVMs. Also, the performance attained by the proposed MSFs is significantly better than that reported in [15], where the modulation features proposed in [49] are used for SER and give approximately 60% and 70% recognition rates when used alone and combined with other feature types, respectively.
### Table 5.3: Recognition results with prosodic features and their combination with MSFs.

<table>
<thead>
<tr>
<th>Test</th>
<th>Feature</th>
<th>Method</th>
<th>SN</th>
<th>Recognition Rate (%)</th>
<th>AVE. (σ)</th>
</tr>
</thead>
<tbody>
<tr>
<td>#1</td>
<td>PROS</td>
<td>SFS</td>
<td>No</td>
<td>89.8 87.7 73.9 84.1 59.2 79.8 83.9</td>
<td>81.1 (5.6)</td>
</tr>
<tr>
<td>#2</td>
<td>PROS</td>
<td>LDA</td>
<td>No</td>
<td>89.8 87.7 73.9 84.1 59.2 79.8 83.9</td>
<td>81.1 (5.6)</td>
</tr>
<tr>
<td>#3</td>
<td>PROS</td>
<td>SFS</td>
<td>Yes</td>
<td>94.5 88.9 76.1 84.1 57.8 89.9 96.8</td>
<td>85.4 (4.3)</td>
</tr>
<tr>
<td>#4</td>
<td>PROS</td>
<td>LDA</td>
<td>Yes</td>
<td>94.5 88.9 76.1 84.1 57.8 89.9 96.8</td>
<td>85.4 (4.3)</td>
</tr>
</tbody>
</table>

### Table 5.4: Confusion matrix for using only prosodic features.

<table>
<thead>
<tr>
<th>Emotion</th>
<th>Anger</th>
<th>Boredom</th>
<th>Disgust</th>
<th>Fear</th>
<th>Joy</th>
<th>Neutral</th>
<th>Sadness</th>
<th>Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anger</td>
<td>114</td>
<td>0</td>
<td>1</td>
<td>3</td>
<td>8</td>
<td>1</td>
<td>0</td>
<td>89.8%</td>
</tr>
<tr>
<td>Boredom</td>
<td>0</td>
<td>69</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>8</td>
<td>3</td>
<td>85.2%</td>
</tr>
<tr>
<td>Disgust</td>
<td>0</td>
<td>1</td>
<td>37</td>
<td>2</td>
<td>0</td>
<td>5</td>
<td>1</td>
<td>80.4%</td>
</tr>
<tr>
<td>Fear</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>60</td>
<td>3</td>
<td>2</td>
<td>0</td>
<td>87.0%</td>
</tr>
<tr>
<td>Joy</td>
<td>19</td>
<td>0</td>
<td>1</td>
<td>2</td>
<td>44</td>
<td>5</td>
<td>0</td>
<td>62.0%</td>
</tr>
<tr>
<td>Neutral</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>0</td>
<td>74</td>
<td>1</td>
<td>93.7%</td>
<td></td>
</tr>
<tr>
<td>Sadness</td>
<td>0</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>58</td>
<td>93.6%</td>
<td></td>
</tr>
</tbody>
</table>

Precision **92.6% 90.2% 89.6% 80.0% 77.1% 92.1%**

### Table 5.5: Confusion matrix for using prosodic and proposed features.

<table>
<thead>
<tr>
<th>Emotion</th>
<th>Anger</th>
<th>Boredom</th>
<th>Disgust</th>
<th>Fear</th>
<th>Joy</th>
<th>Neutral</th>
<th>Sadness</th>
<th>Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Anger</td>
<td>119</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>6</td>
<td>0</td>
<td>0</td>
<td>93.7%</td>
</tr>
<tr>
<td>Boredom</td>
<td>0</td>
<td>78</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>3</td>
<td>0</td>
<td>96.3%</td>
</tr>
<tr>
<td>Disgust</td>
<td>0</td>
<td>0</td>
<td>42</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>91.3%</td>
</tr>
<tr>
<td>Fear</td>
<td>2</td>
<td>0</td>
<td>0</td>
<td>62</td>
<td>3</td>
<td>2</td>
<td>0</td>
<td>89.9%</td>
</tr>
<tr>
<td>Joy</td>
<td>12</td>
<td>0</td>
<td>1</td>
<td>3</td>
<td>52</td>
<td>3</td>
<td>0</td>
<td>73.2%</td>
</tr>
<tr>
<td>Neutral</td>
<td>0</td>
<td>3</td>
<td>1</td>
<td>0</td>
<td>75</td>
<td>0</td>
<td>94.9%</td>
<td></td>
</tr>
<tr>
<td>Sadness</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>62</td>
<td>0</td>
<td>100%</td>
<td></td>
</tr>
</tbody>
</table>

Precision **95.9% 96.3% 93.3% 92.5% 83.9% 88.2% 100%**
5.3 Continuous Emotion Regression

In this section, the combined prosodic and proposed features shown to furnish very good performance for discrete emotion classification are further assessed on the VAM database to recognize three continuous emotion primitives (Section 5.3.1), namely *valence*, *activation*, and *dominance*. Cross-database experiment is performed as well (Section 5.3.2), applying the continuous emotions to classify discrete emotions.

5.3.1 Emotion primitive recognition

Support vector regression (SVR) is used for emotion primitive regression. Recognition is performed using (1) prosodic features, and (2) combined prosodic and proposed features (without SN). Cross-validation is performed in a leaving-one-out (LOO) fashion to match the experimental setup in [35]. Moreover, instead of selecting features based on the VAM database, we use the features chosen on the Berlin database by SFS here as a means of testing feature robustness. The 10 cross-validation trials over the Berlin database produce 10 SFS feature sets. The union of the 10 feature sets is then taken to merge the feature sets into one for the regression experiment on the VAM database. To form the union, the top $N_{sfs}$ SFS-selected features are retained for each of the 10 sets. It is observed that using $N_{sfs} = 15$ is sufficient, as retaining more features hardly improves regression performance.

The three SVM kernels described in Section 5.2.3 are tested for SVR. Continuous emotion regression results are summarized in Tables 5.6-5.8, where $r$ and $e$ stand for correlation coefficient and mean absolute error, respectively. The machine recognition and human subjective evaluation results in [35] are also included in Table 5.6 for reference. For two sequences $x_n$ and $y_n$ of the same length $N_{xy}$, the correlation
Coefficient $r$ is calculated using Pearson’s formula:

$$r = \frac{\sum_{n=1}^{N_{xy}} (x_n - \bar{x})(y_n - \bar{y})}{\sqrt{\sum_{n=1}^{N_{xy}} (x_n - \bar{x})^2 \sum_{n=1}^{N_{xy}} (y_n - \bar{y})^2}}; \quad (5.14)$$

where $\bar{x}$ is the average of $x_n$; $\bar{y}$ is the average of $y_n$; and the mean absolute error $e$ is given by:

$$e = \frac{1}{N_{xy}} \sum_{n=1}^{N_{xy}} |x_n - y_n|. \quad (5.15)$$

Mean absolute error is used to match the error measure employed in [35]. Note that in [35], only the standard deviation of subjective scores is presented for each primitive. The error is the standard deviation minus a constant bias term depending on the number of evaluators, which can be inferred from the paper.

From Tables 5.6-5.8, it can be concluded that RBF-SVRs consistently provide the best regression performance, and the polynomial kernel gives better correlations than the linear kernel. As a common trend observed for the three kernels, adding the proposed MSFs to prosodic features improves the correlations for the three primitives on all datasets, and also slightly reduces the estimation errors in general. By using combined features and RBF-SVRs, the primitive *activation* is best estimated with up to 0.88 correlation achieved on VAMI, followed by *dominance* with 0.84 correlation. For *valence*, however, the improved correlation values after including the MSFs are still quite low. Nevertheless, even human evaluation gives poor correlations for recognizing *valence* compared to the other two primitives.

Overall, the proposed recognition system yields higher correlations and smaller estimation errors compared to the machine recognition results in [35]. Moreover, the performance achieved by the proposed system is at times somewhat superior to human assessment. In [61], good estimation results are also achieved for *activation* and
Table 5.6: Recognition results for continuous emotions on the VAM database; the RBF kernel is used.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Feature</th>
<th>correlation (r)</th>
<th>absolute error (e)</th>
<th>average</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>valence</td>
<td>activation</td>
<td>dominance</td>
</tr>
<tr>
<td>VAM I</td>
<td>PROS only</td>
<td>0.54</td>
<td>0.84</td>
<td>0.81</td>
</tr>
<tr>
<td></td>
<td>PROS+MSF</td>
<td>0.60</td>
<td>0.88</td>
<td>0.84</td>
</tr>
<tr>
<td></td>
<td>[35] N/A</td>
<td>0.41</td>
<td>0.78</td>
<td>0.68</td>
</tr>
<tr>
<td></td>
<td>human</td>
<td>0.49</td>
<td>0.78</td>
<td>0.68</td>
</tr>
<tr>
<td>VAM II</td>
<td>PROS only</td>
<td>0.28</td>
<td>0.68</td>
<td>0.64</td>
</tr>
<tr>
<td></td>
<td>PROS+MSF</td>
<td>0.38</td>
<td>0.75</td>
<td>0.67</td>
</tr>
<tr>
<td></td>
<td>[35] N/A</td>
<td>0.48</td>
<td>0.66</td>
<td>0.54</td>
</tr>
<tr>
<td></td>
<td>human</td>
<td>0.49</td>
<td>0.72</td>
<td>0.61</td>
</tr>
<tr>
<td>VAM I+II</td>
<td>PROS only</td>
<td>0.45</td>
<td>0.77</td>
<td>0.75</td>
</tr>
<tr>
<td></td>
<td>PROS+MSF</td>
<td>0.49</td>
<td>0.82</td>
<td>0.78</td>
</tr>
<tr>
<td></td>
<td>[35] N/A</td>
<td>0.49</td>
<td>0.72</td>
<td>0.61</td>
</tr>
<tr>
<td></td>
<td>human</td>
<td>0.49</td>
<td>0.72</td>
<td>0.61</td>
</tr>
</tbody>
</table>

Table 5.7: Recognition results for continuous emotions on the VAM database; the linear kernel is used.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Feature</th>
<th>correlation (r)</th>
<th>absolute error (e)</th>
<th>average</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>valence</td>
<td>activation</td>
<td>dominance</td>
</tr>
<tr>
<td>VAM I</td>
<td>PROS only</td>
<td>0.54</td>
<td>0.83</td>
<td>0.80</td>
</tr>
<tr>
<td></td>
<td>PROS+MSF</td>
<td>0.54</td>
<td>0.86</td>
<td>0.81</td>
</tr>
<tr>
<td>VAM II</td>
<td>PROS only</td>
<td>0.25</td>
<td>0.66</td>
<td>0.61</td>
</tr>
<tr>
<td></td>
<td>PROS+MSF</td>
<td>0.25</td>
<td>0.73</td>
<td>0.67</td>
</tr>
<tr>
<td>VAM I+II</td>
<td>PROS only</td>
<td>0.40</td>
<td>0.75</td>
<td>0.74</td>
</tr>
<tr>
<td></td>
<td>PROS+MSF</td>
<td>0.42</td>
<td>0.78</td>
<td>0.76</td>
</tr>
</tbody>
</table>

Table 5.8: Recognition results for continuous emotions on the VAM database; the polynomial kernel is used.

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Feature</th>
<th>correlation (r)</th>
<th>absolute error (e)</th>
<th>average</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>valence</td>
<td>activation</td>
<td>dominance</td>
</tr>
<tr>
<td>VAM I</td>
<td>PROS only</td>
<td>0.53</td>
<td>0.84</td>
<td>0.80</td>
</tr>
<tr>
<td></td>
<td>PROS+MSF</td>
<td>0.57</td>
<td>0.88</td>
<td>0.83</td>
</tr>
<tr>
<td>VAM II</td>
<td>PROS only</td>
<td>0.21</td>
<td>0.69</td>
<td>0.63</td>
</tr>
<tr>
<td></td>
<td>PROS+MSF</td>
<td>0.32</td>
<td>0.74</td>
<td>0.67</td>
</tr>
<tr>
<td>VAM I+II</td>
<td>PROS only</td>
<td>0.44</td>
<td>0.77</td>
<td>0.75</td>
</tr>
<tr>
<td></td>
<td>PROS+MSF</td>
<td>0.48</td>
<td>0.80</td>
<td>0.78</td>
</tr>
</tbody>
</table>
dominance on VAM I+II, but valence is still poorly estimated with 0.46 correlation. This may be the reason for the poor joy vs. anger classification results in Tables 5.4 and 5.5, as the two emotions are of the same activation level, yet of opposite valence. On the other hand, since activation can be well recognized, anger and sadness, two emotions of opposite activation, should display good discrimination. This is also verified according to the confusion matrices given in Tables 5.4 and 5.5, where anger and sadness are never misclassified to the other.

Moreover, as might also be noticed from Tables 5.6-5.8, regression performance on VAM I is always superior to that on VAM II, irrespective of type of features and SVM kernels. It is because VAM I contains utterances from “very good” speakers characterized by a high level of activity with a wide variety of emotions, while VAM II consists of utterances from “good” speakers that, though possessing high activity as well, produce a smaller scope of emotions such as anger only [34]. Consequently, the combination of VAM I and VAM II (i.e. VAM I+II) yields moderate recognition results in the three datasets.

5.3.2 Cross-database evaluation

In this section, the continuous primitives are applied to discrete emotion classification. To implement this, three RBF-SVRs are first trained on the VAM I+II dataset corresponding to the three emotion primitives, using the combined prosodic and proposed features. Then each file in the Berlin database is assigned three predicted primitive values by the three regressors trained on the VAM database. The three estimated primitive values are used as features for discrete emotion classification, achieving an overall recognition rate of 53.6% (49.5%) with (without) SN for classifying the seven
emotions in the Berlin database under 10-fold cross-validation. Although the accuracy obtained is not good, it is still far beyond the 14% random chance, indicating that the continuous primitives do contain useful information for identifying discrete emotions. The low recognition rate could be due to inadequacy of using the three primitives and mismatch between databases. Moreover, no “mutual information” relating the emotion data of the two databases is available. Such information could have been produced by having the same human subjects rating both databases, on both the discrete and the primitive scales.

Since the regression performance varies with the three primitives as indicated in the regression experiment in Section 5.3.1, binary classification tasks where discrimination relies on the specific discriminating information conveyed by particular primitives might provide more insights than the comprehensive seven-class recognition task. Therefore, three two-class classification tasks are designed to test the three primitive features separately: (1) anger vs. joy, (2) anger vs. sadness, and (3) anger vs. fear, mainly involving (though not limited to) the recognition of valence, activation, and dominance, respectively. Therefore, if the three primitives were ideally recognized, valence, activation, and dominance predicted by the trained regressors would be the features providing the best recognition performance for tasks (1), (2), and (3), respectively.

Results of the binary tests are listed in Table 5.9 with SN applied. As can be seen from the table, perfect and good classification results are obtained for discriminating anger vs. sadness (100%) and anger vs. fear (84.2%) using only one primitive regressor, namely activation and dominance, respectively. Combining all three primitive features slightly improves the recognition performance for tasks (1) anger vs. joy
Table 5.9: Binary classification results using three primitive features.

<table>
<thead>
<tr>
<th></th>
<th>valence</th>
<th>activation</th>
<th>dominance</th>
<th>ALL</th>
</tr>
</thead>
<tbody>
<tr>
<td>anger vs. joy</td>
<td>61.6%</td>
<td>64.1%</td>
<td>63.6%</td>
<td>65.7%</td>
</tr>
<tr>
<td>anger vs. sadness</td>
<td>82.0%</td>
<td>100%</td>
<td>99.5%</td>
<td>99.5%</td>
</tr>
<tr>
<td>anger vs. fear</td>
<td>70.4%</td>
<td>77.0%</td>
<td>84.2%</td>
<td>85.2%</td>
</tr>
</tbody>
</table>

and (3) anger vs. fear. However, one notable difficulty arises when using the valence feature to classify anger and joy, resulting in only 61.6% recognition rate, even lower than that obtained with the activation or dominance feature. The valence regressor, therefore, needs to be refined substantially in order to attain practical performance figures. These results also resonate with our previous findings that activation and valence are primitives with the best and the worst estimation performance, respectively. Additionally, the seven-class FDR scores for the three primitive features are valence: 1.3 (0.9), activation: 3.9 (3.5), and dominance: 4.4 (3.6) with (without) SN, again indicating that the valence feature has the lowest discriminating power.

5.4 Summary

In this chapter, simulation studies performed on the MSFs proposed in Chapter 4 are reported. The main part of this chapter is dedicated to evaluating the proposed features on the Berlin database for discrete emotion classification. A two-stage scheme for dimensionality reduction has been introduced. Different SN schemes and SVM kernels have also been investigated. The results show that processing features by mean and variance normalization within the scope of each speaker is the most effective SN method, and RBF-SVMs give superior recognition performance.

The MSFs are initially compared to features based on direct energy samples of
the ST representation. Results of the comparison indicate that the exploitation of the ST representation performed for extracting the MSFs is worthwhile, as it results in more powerful features. Comparisons are then drawn between the proposed features and two STSFs, where the MSFs achieve better recognition performance than MFCC and PLP features in various tests. The proposed features are further benchmarked by prosodic features, and have been shown to be powerful additions. By using combined prosodic and proposed features, above 90% recognition rate is achieved for classifying seven emotion categories.

Finally, regression of three continuous emotion primitives has been performed on the VAM database. Again, RBF-SVRs yield the best regression performance. It has been shown that the recognition performance of prosodic features can be improved by including the MSFs. By using combined features, estimation performance comparable to human evaluation is accomplished. Several binary classification tests have also been designed to assess the three emotion primitives separately. Promising outcomes are obtained for estimating activation and dominance, but the valence parameter needs to be further refined to achieve practical recognition performance.
Chapter 6

Conclusions and Future Work

6.1 Conclusions

Speech emotion recognition (SER) has become an area of active research interest in recent years. SER plays an important role in building more anthropomorphic human-computer interfaces. As a machine learning task, successful SER requires both reliable machine learning techniques and emotion discriminating features. While various learning algorithms have been proposed for SER, constructing powerful features, specifically spectral features, remains an open challenge. In the literature, most spectral features used for SER convey merely short-term spectral properties of speech signals, while omitting long-term temporal modulation information critical for human speech perception.

This thesis, therefore, proposes a novel set of spectral features for the automatic recognition of human emotion in speech, aiming to overcome the limitations rooted in conventional short-term spectral features (STSFs). The features are derived from
an auditory-inspired spectro-temporal (ST) representation of speech. The representation incorporates long-term temporal modulation information in a way consistent with recent neuroscience findings that ST modulation patterns are extracted in the human auditory cortex. To obtain the ST representation, the speech signal is first decomposed by an auditory filterbank. The Hilbert envelopes of the critical-band outputs are computed to form the modulation signals. A modulation filterbank is further applied to the Hilbert envelopes to perform frequency analysis. The spectral contents of the modulation signals are referred to as modulation spectra, and the proposed features are thereby termed modulation spectral features (MSFs). Lastly, the ST representation is formed by measuring the energy of the decomposed envelope signals, as a function of regular acoustic frequency and modulation frequency.

The MSFs are then extracted from the ST representation by means of spectral measures and linear prediction parameters. It has been shown in the thesis that such exploitation of the ST representation yields much more effective features than the original ST energy samples. Recognition systems are built using powerful support vector machines (SVMs). Based on the structural risk minimization principle, SVMs exhibit better generalization performance relative to many other machine learning techniques.

Experimental evaluation is first carried out on the Berlin database to classify seven discrete emotions. Dimensionality reduction is performed in a two-stage fashion. The first stage quickly eliminates irrelevant features, and the second stage makes a finer exploitation to obtain good feature combinations. Moreover, applying speaker normalization prior to SVM recognition has been demonstrated to be beneficial. In a comparison of SVMs with linear, polynomial, and radial basis function (RBF) kernels,
RBF-SVMs are found to convey the best performance for emotion classification. Simulation results show that the MSFs outperform features based on mel-frequency cepstral coefficients and perceptual linear predictive coefficients, two commonly used STSFs. Moreover, the recognition accuracy furnished by prosodic features can be substantially improved after the inclusion of the proposed features. By using combined features, state-of-the-art recognition performance is accomplished, with a feature set more compact relative to many state-of-the-art works.

Recognition of three continuous emotion primitives is further performed on the VAM database, by means of support vector regression (SVR). Similar to the case of discrete emotion classification, the RBF kernel conveys superior regression results. However, for any SVM kernel type tested, improved regression performance is attained by adding the MSFs to prosodic features. By using RBF-SVRs and combined features, recognition performance comparable to subjective human evaluation is achieved. Promising regression results are obtained for estimating activation and dominance, but lower performance is obtained when recognizing valence, a trend also reported in other studies on the VAM database. Moreover, the continuous emotion primitives are applied to classify discrete emotions. Among the three primitive features, activation and dominance are shown to give satisfactory classification results, while further refinement is needed for valence in order to achieve good performance figures.

6.2 Future Work

In future work, we aim at a more in-depth investigation of the ST representation. Although the current MSFs already give very good emotion recognition performance,
a further exploitation may contribute to the development of even more powerful features. Moreover, the features will be tested under more complex real-world conditions (e.g. reverberant and noisy speech). Other works (e.g. [95–97]) have suggested that the modulation spectra are robust to reverberation and noise impairments. We have also demonstrated in a preliminary experiment that the MSFs have the potential to function better under such non-ideal conditions than traditional spectral features. Also, recognition performance of different machine learning algorithms will be studied.
Bibliography


Appendix A

Selected Features

Denote the feature set as $F$. The average feature rank (AFR) of the $u$th feature $f_u \in F$ given $N$ cross-validation trials is calculated as:

$$\text{AFR}(f_u) = \frac{1}{N} \sum_{n=1}^{N} \text{rank of } f_u \text{ in the } n\text{th trial.}$$  \hspace{1cm} (A.1)

If $f_u$ is not selected in a trial, its rank is replaced by a penalty value $P$. The top 10 features from prosodic features, the MSFs, and their combination are listed in Tables A.1–A.3 as ranked by AFR, respectively, with 10 features selected by SFS in each trial and the penalty value $P$ empirically set to 11. Notations $Q_1$, $Q_2$ and $Q_3$ in the tables stand for the 1st quartile, the 2nd quartile (median), and the 3rd quartile, respectively, and “std. dev.” denotes standard deviation. The notations of the proposed features are explained in Section 4.4.
### Table A.1: Top 10 prosodic features ranked by AFR.

<table>
<thead>
<tr>
<th>Rank</th>
<th>Feature</th>
<th>AFR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>TEO</td>
<td>3.7</td>
</tr>
<tr>
<td>2</td>
<td>mean syllable duration</td>
<td>3.8</td>
</tr>
<tr>
<td>3</td>
<td>$Q_3 - Q_2$ of delta-pitch</td>
<td>4.2</td>
</tr>
<tr>
<td>4</td>
<td>minimum of intensity</td>
<td>5.4</td>
</tr>
<tr>
<td>5</td>
<td>mean of pitch</td>
<td>5.5</td>
</tr>
<tr>
<td>6</td>
<td>std. dev. of pitch</td>
<td>6.1</td>
</tr>
<tr>
<td>7</td>
<td>minimum of pitch</td>
<td>8.3</td>
</tr>
<tr>
<td>8</td>
<td>$Q_1$ of delta-intensity</td>
<td>9.0</td>
</tr>
<tr>
<td>9</td>
<td>$Q_3$ of delta-intensity</td>
<td>9.1</td>
</tr>
<tr>
<td>10</td>
<td>$Q_3 - Q_2$ of pitch</td>
<td>9.3</td>
</tr>
</tbody>
</table>

### Table A.2: Top 10 proposed features ranked by AFR.

<table>
<thead>
<tr>
<th>Rank</th>
<th>Feature</th>
<th>AFR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>mean of $\Phi_{3,k}(1)$</td>
<td>2.0</td>
</tr>
<tr>
<td>2</td>
<td>mean of $\Phi_{1,k}(3)$</td>
<td>2.6</td>
</tr>
<tr>
<td>3</td>
<td>mean of $\Phi_{5,k}(2)$</td>
<td>4.9</td>
</tr>
<tr>
<td>4</td>
<td>mean of $\Phi_{1,k}(2)$</td>
<td>7.1</td>
</tr>
<tr>
<td>4</td>
<td>mean of $\Phi_{3,k}(1)$</td>
<td>7.1</td>
</tr>
<tr>
<td>6</td>
<td>mean of $\Phi_{4,k}(3)$</td>
<td>7.4</td>
</tr>
<tr>
<td>7</td>
<td>mean of $\Phi_{4,k}(4)$</td>
<td>8.5</td>
</tr>
<tr>
<td>8</td>
<td>mean of $\Phi_{5,k}(4)$</td>
<td>9.0</td>
</tr>
<tr>
<td>9</td>
<td>std. dev. of $\Phi_{6,k}(2)$</td>
<td>9.4</td>
</tr>
<tr>
<td>10</td>
<td>std. dev. of $\Phi_{6,k}(2)$</td>
<td>9.4</td>
</tr>
</tbody>
</table>

### Table A.3: Top 10 combined features ranked by AFR.

<table>
<thead>
<tr>
<th>Rank</th>
<th>Feature</th>
<th>AFR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>mean of $\Phi_{3,k}(1)$</td>
<td>2.0</td>
</tr>
<tr>
<td>2</td>
<td>mean of $\Phi_{1,k}(3)$</td>
<td>2.9</td>
</tr>
<tr>
<td>3</td>
<td>mean of $\Phi_{5,k}(2)$</td>
<td>6.6</td>
</tr>
<tr>
<td>4</td>
<td>mean of pitch</td>
<td>7.1</td>
</tr>
<tr>
<td>4</td>
<td>std. dev. of pitch</td>
<td>7.1</td>
</tr>
<tr>
<td>6</td>
<td>mean syllable duration</td>
<td>7.6</td>
</tr>
<tr>
<td>7</td>
<td>$Q_3$ of pitch</td>
<td>7.8</td>
</tr>
<tr>
<td>8</td>
<td>mean of $\Phi_{4,k}(3)$</td>
<td>9.4</td>
</tr>
<tr>
<td>9</td>
<td>mean of $\Phi_{5,k}(4)$</td>
<td>9.6</td>
</tr>
<tr>
<td>10</td>
<td>mean of $\Phi_{6,k}(2)$</td>
<td>9.7</td>
</tr>
</tbody>
</table>