Network for Studies on Pensions, Aging and Retirement

NETSPAR ACADEMIC SERIES

Automatic Emotion Recognition from Mandarin Speech

Yu Gu

PhD 08/2018-011

Automatic Emotion Recognition from Mandarin Speech

Yu Gu

SIKS Dissertation series No. 2018-30

The research reported in this thesis has been carried out under the auspices of SIKS, the Dutch Research School for Information and Knowledge Systems.

This research was approved by the Chinese Scholarship Council (CSC) under No.201206660009.

TiCC Ph.D. Series No.

ISBN 978-00-000-000-0

Cover design:
Printed & Lay Out by:
Published by:

Automatic Emotion Recognition from Mandarin Speech

PROEFSCHRIFT

ter verkrijging van de graad van doctor
aan Tilburg University
op gezag van de rector magnificus,
prof. dr. E.H.L. Aarts,
in het openbaar te verdedigen ten overstaan van
een
door het college voor promoties aangewezen
commissie
in de room van de Universiteit
op XXdag xx 2018 om 10.00 uur

door
Yu Gu,
geboren op 7 January 1987 te Xi'An, China

Promotores:

Prof. dr. E. O. Postma

Prof. dr. H. J. van den Herik

Prof. dr. H. X. Lin

Overige leden van de promotiecommissie:

Prof. dr. ir. P.H.M. Spronck

Prof. dr. H.C. Bunt

Prof. dr. F.J. Verbeek

Prof. dr. J.N. Kok

Prof. dr. S. Jiande

PREFACE

When I was young, I was really fascinated by the Chinese novel *Journey to the West (Monkey)*. It describes the story of Monk Xuan Zang who travels to the Western Regions to obtain Buddhist scriptures. He succeeded after many dangers and much suffering. For me, pursuing a Ph.D. study has also been a kind of "Journey to the West". There were many wonderful times as well as upsets and pains. I thought of giving up my study several times, but each time I defeated the disappointment and continued my journey.

The start of my research goes back to 2012, when I took up the topic of speech emotion recognition. Speech was such an intriguing topic for me, since it always poses a platform to express a personal purpose, attitude, and emotion. This led me to becoming a speech researcher. When I look back at my Ph.D. time, I feel so happy that there were many people in my 'neighborhood' who were prepared to help me. Without their support, I would not have been able to achieve the final completion.

First of all, I would like to express my special appreciation and thanks to my supervisors, Eric Postma, Jaap van den Herik, and Hai-Xiang Lin for their tremendous support and guidance, both in research and local culture. In the last five years of my Ph.D. journey, Eric was always patient and optimistic, which influenced me quite deeply. During the time working with him, I was time and time again inspired by his point of view on machine learning. No matter whether the experiment results were good or bad, he encouraged me to be more ambitious and perseverant. His attitude will accompany me into my later life. I should say many thanks to Jaap, too. He was very strict in research and academic writing. In the end, I noticed that I had benefited so much from his advice on how to write precisely that I now feel there is always a "small Japie" in my own neighborhood. Finally, the same feelings of gratitude go to Hai-Xiang Lin. We met each other for the first time in China in 2011. Without his support, I would not even have had the opportunity to study at Tilburg University. I could not expect any better professors than these three supervisors for guiding me in my Ph.D. journey.

Second, I would like to thank all my colleagues at TiCC. I felt privileged to work with them. It was wonderful. In particular, I mention Nanne van Noord who taught me so much in my first year when I knew so little about the deep-learning technology and the Dutch culture. I would also like to thank Tiago. Each time I was stuck in research or programming, his comments and suggestions were extremely helpful. I also received excellent support from the staff members, more specifically from Eva, Jachinta, and Joke. They are very smart and also nice to work with.

Third, I would like to thank my friends Sen Zhou, Yan Gu, Nie Hua, Kun-Ming Li, Cai-Xia Liu, Cai-Xia Du, Liang Tang, and many other people with whom I spent so much time in Tilburg. All of you gave me numerous useful suggestions on my career and life.

Moreover, I would like to attribute special thanks to my family. First of all, I am grateful to my wife He Zhang, for all her support during my Ph.D. journey. She has made numerous sacrifices and showed me that she unconditionally accepted me and my work. She spent days and nights on proofreading to make my thesis easy to read. She was always at my side no matter what situation I was in. Second, my father was the one who encouraged me to go to the Netherlands when I was given the opportunity to study abroad. Without his encouragement, I would not have made the brave choice. Third, my mother showed incredible tolerance to me. She always patiently set aside time to listen to my complaints, my troubles, and my unhappy times until I felt happy again.

Finally, I would like to thank the Chinese Scholarship Council (CSC). This research was funded by the Chinese Ph.D. Scholarship (No.201206660009) from the CSC. I gratefully acknowledge the support of the CSC for the full four years of funding; without it the completion of this research would have been impossible.

Tilburg, August 2018

Yu Gu

DEDICATION

The thesis is dedicated to my love, He Zhang, in particular, for her help, devotion, and endless support in times that I was upset. The dedication is also meant for my parents, Ya-Cheng Gu and Fang-Mei Zhang. I offer all three of you my sincere thanks for always being at my side. Your encouragement was a source of inspiration and will be remembered for the rest of my life.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

AER	Automatic Emotion Recognition
AMS	Affective Mandarin Speech
CNN	Convolutional Neural Network
DBN	Deep Belief Network
DC	Direct Current
DCNN	Deep Convolutional Neural Network
DNN	Deep Neural Network
DL	Deep Learning
FFT	Fast Fourier Transform
Fo	Fundamental Frequency
HMM	Hidden Markov Model
HNR	Harmonics to Noise Ratio
LDA	Linear Discriminant Analysis
LOO	Leaving-One-Out
LPC	Linear Prediction Coefficients
LPCC	Linear Predictive Cepstral Coefficients
LR	Likelihood Ratio
MAS	Mandarin Affective Speech
MFCC	Mel Frequency Cepstral Coefficients
NN	Neural Network
PCA	Principal Component Analysis
PS	Problem Statement
RBF	Radial Basis Function
RQ	Research Question
RM	Research Methodology
RNN	Recurrent Neural Network
SD	Standard Deviation
SER	Speech Emotion Recognition
SVM	Support Vector Machine
TCEC	Top Chess Engine Championship
VAD	Voice Activity Detection
VSS	Voiced Segment Selection
ZCR	Zero Crossing Rate

1 INTRODUCTION

Whether a person is speaking privately with family members or giving a presentation at a conference, emotion is an inevitable element of speech and is presented in some form. Moreover, in many social interactions, thoughts, wishes, attitudes, and opinions cannot be fully expressed without emotion. One of the most important functions of emotion is to support interpersonal understanding. The appropriate use of emotional expression helps to achieve better communication, enhance friendship and mutual respect, and improve relationships.

Due to the significant impact of emotion on humans' exchange of information, the recognition and understanding of emotions in communication behavior has become a prominent multidisciplinary research topic. The earliest modern scientific studies on emotion trace back to the work by Charles Darwin. In *The Expression of the Emotions in Man and Animals*, Darwin claimed that (1) the voice works as the main carrier of emotion signals in communication, and (2) clear correlations exist between particular emotional states and the sound produced by the speaker (see Darwin, 1872). Following this seminal text, we see that emotion studies were dominated by behavioral psychologists for more than 100 years. In this field, William James established the research theory of emotion that is still prevalent today(cf. James, 1884). Since that point, the topic has spread to a variety of disciplines (see Tao & Tan, 2005).

In human communication, the speaker generally has two channels for delivering his[1]emotional information to the listener: verbal and non-verbal communication (cf. Koolagudi & Rao, 2012). First, emotional information can be conveyed verbally, which is of interest to linguists. When expressing an emotion through speech, a person can organize words in specific ways to send an emotional signal to others. For example, it is common to hear words that explicitly suggest a certain emotion, such as "I am so sad today." However, emotion can sometimes work against the verbal form in which it is cased.

The second way to express and receive emotional information is through non-verbal means. The fundamental non-verbal cues for emotion in human communication fall into three main categories: facial expression, vocalization, and body language (cf. Watzlawick, Bavelas, Jackson, & O'Hanlon, 2011). Of these types, vocalization is one of the most efficient vehicles for information transfer (Postma-Nilsenová, Postma, Tsoumani, & Gu, n.d.). As we speak, our voices convey information about us as individuals. The sound of one's voice can reveal if he is happy, sad, panicked, or in some other emotional state.

1 For brevity, only the pronouns *he* and *him* are used whenever *he* or *she* and *him* or *her* are meant.

Changing our voice sounds can notify the listener that our emotions are shifting to a new direction. Thus, the voice is a way for a speaker to demonstrate his emotional state.

Given the wide range of emotional information that a listener receives from speech, it is not surprising that researchers from a variety of disciplines are interested in studying speech emotion. The following section provides a summary of previous research that forms a basis of this study. We start in 1935 when Skinner attempted to study happy and sad emotional information through analyzing the pitch of speech. The non-verbal conveyance of emotion includes paralinguistic acoustic cues such as pitch and energy. Skinner's study revealed that a person's pitch is more likely to change if he is happy or sad than if he is experiencing another type of emotion (cf. Skinner, 1935). In their later work, Ortony et al. (1990) observed that a single sentence can express various emotions as the speaker changes the speaking rate and energy used. Nygaard and Queen (2008) subsequently demonstrated that a listener was able to repeat happy or sad words, such as *comedy* or *cancer*, more quickly when the words were spoken in a tone of voice that matched the emotional content; the repetition proceeded more slowly when the emotional tone of voice contradicted the affective meaning of the words used. Schirmer and Simpson (2007) also found that the emotional tone of speech can influence a listener's cognitive processing of words. Furthermore, more than 50 years ago, Kramer's studies established that in cross-cultural communication, a listener who does not know the cultural background or language of the speaker can still understand and recognize the emotional information via non-verbal communication (see Kramer, 1964).

The above studies have collectively agreed that non-verbal aspects of speech can independently demonstrate emotional information. Since the non-verbal aspects of speech can separately contain emotion, as such, understanding emotion can help to overcome the language and cultural barriers often present in cross-cultural and international communication.

In this thesis, we aim to create a novel method for a computer to recognize emotion through non-verbal speech cues in the Mandarin language. The intention is to thus enable the computer to detect a Mandarin speaker's different emotional states. Our goal is to find an alternative to the current methods, which accurately characterize non-verbal speech emotion in languages other than Mandarin. In this study, we disregard the verbal aspects of speech and focus solely on non-verbal aspects of speech in all experiments.

The remainder of this chapter is organized as follows: We first introduce speech emotion recognition (SER) in Section 1.1. The applications of SER are then described in Section 1.2, while Section 1.3 subsequently discusses SER in the Mandarin language. In Section 1.4, we formulate our problem statement (PS) and research questions (RQs). Afterwards, Section 1.5 describes the research methodology, and Section 1.6 offers an overview of our contributions. Finally, Section 1.7 outlines the structure of this thesis.

1.1 SPEECH EMOTION RECOGNITION

In the science-fiction film *Interstellar*[2] released in 2014, the robot TARS shows to be highly capable of processing emotion in the language spoken by the astronauts with whom it interacts. TARS understands and recognizes the emotional expressions of the spaceship's crew. TARS can therefore interact with the crew members in a human manner. Although *Interstellar* is a fictional movie set 50 years in the future, the prediction that an artificially intelligent robot may be able to spontaneously recognize emotion using an application for human-machine interaction is no longer a bold expectation (Tziolas, Morrison, & Armstrong, 2017). An example of the current possibilities is seen in Figure 1.1. It contains a telling representation of using speech emotion: a tutoring session between a supervisor and a student. An effective tutoring application should recognize the student's emotional state during the session. The supervisor can then accordingly change the teaching style.

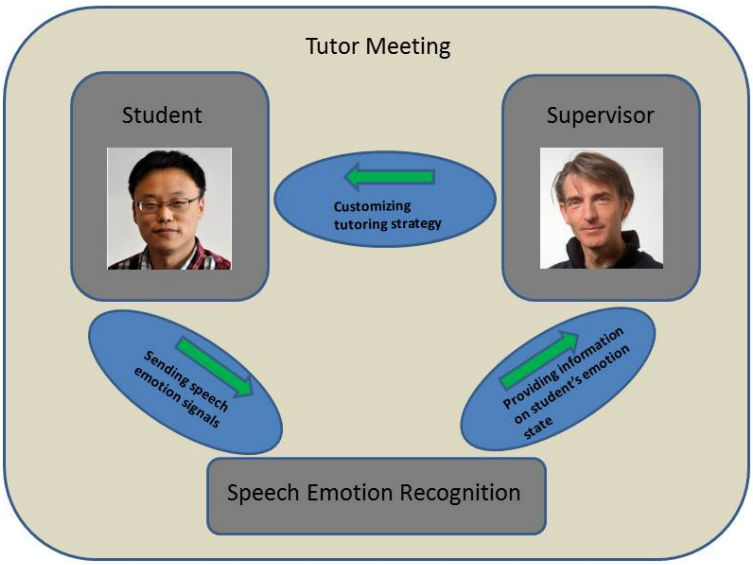


Figure 1.1: A model of how SER works for tutoring sessions

The following question should be answered is that: How should an intelligent tutoring application be designed and deployed in practice? The first step in designing and using an intelligent tutoring application in the real world is developing computer intelligence that simulates the human brain's ability of learning to recognize emotion expressions (cf. Picard & Picard, 1997). The second step is training the application to recognize verbal and non-verbal emotional expression (cf. Gupta, Raviv, & Raskar, 2018). Recent research efforts

2 https://www.imdb.com/title/tt0816692/

aim at enabling a computer program to detect, interpret, and create emotional behavior via so-called automatic emotion recognition (AER). Many research activities developing AER algorithms for facial expression and body language are nowadays ongoing (cf. Piana, Stagliano, Odone, Verri, & Camurri, 2014). Vocalization is also a crucial subject for research on emotion recognition (cf. Mirsamadi, Barsoum, & Zhang, n.d.). Due to all these research activities on vocalization, speech emotion recognition (SER) has become an indispensable branch of AER.

In brief, SER seeks to recognize emotion in human speech communication. Figure 1.2 illustrates a general flowchart of the structure of SER. The identification of speech emotion occurs to happen into three stages: (1) feature extraction, which consists of extracting a set of features containing emotion information from speech signals; (2) feature selection, i.e., selecting a subset of features for use in classification; and (3) classification, which entails separating features into classes of emotional states.

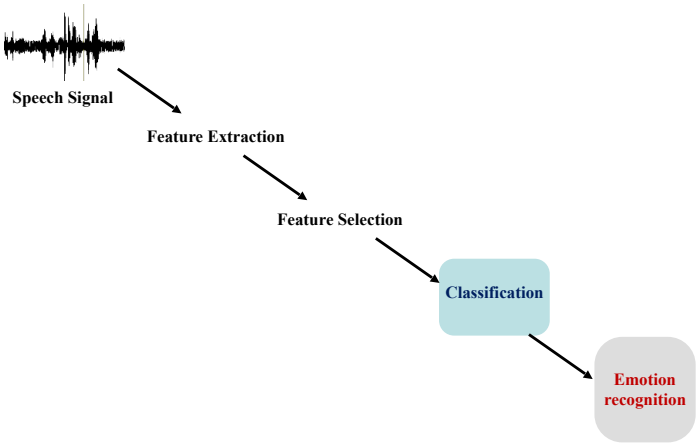


Figure 1.2: Three stages in the identification of speech emotion

1.2 APPLICATIONS OF SPEECH EMOTION RECOGNITION

At present, SER is playing an increasingly useful role in people's daily lives because of the considerable progress in human-machine interaction. In particular, we observe that the use of SER in social networking is expanding to a large variety of applications. As we see it, a SER application will be a tool (i.e., a system or device) designed to detect, differentiate, and recognize a human emotional state. Below we provide four examples of prevalent applications of SER.

(1) A SER system that is used in the automated service of call centers detects a customer's negative emotional expression during the automated conversation. Negative emotion can then be immediately remedied by changing from an automated conversation to a conversation with a human telephone receptionist, who may improve the service by helping the customer in a pleasant manner (Ramakrishnan & El Emary, 2013).

(2) A SER system has a wide range of applications in medical health services. Based on the development of both signal processing and medical science, a growing number of SER applications are being used as medical tools that aid in diagnosis and treatment. SER can be a helpful medical tool, especially for making diagnosis via behavioral analysis and depression detection analysis. A draw-back of this application may be a lack of accuracy that influences a doctor's decision (cf. Luneski, Konstantinidis, & Bamidis, 2010).

(3) A SER system that is also applied during a criminal investigation can automatically detect a suspect's mental emotional state. Suspects typically attempt to hide their true feelings (cf. Suzuki et al., 2002), but a SER system can detect and recognize their authentic emotional state. It may suggest that the suspect's real behavior is different from his apparent behavior. When that is the case, there is a chance that the suspect is lying or concealing facts (see Anagnostopoulos, Iliou, & Giannoukos, 2015).

(4) The most recent approaches to speech recognition have been established by major smartphone and software producers. For the most recent Windows smartphone operating system, Microsoft developed a machine learning translation application for laptops and tablets: Skype Translator. Skype Translator currently supports real-time voice-to-voice translation in English, French, German, Italian, and Mandarin[3]. Software developers have made important advances in speech recognition, as reflected by Google Translate, which has been available for Android systems since late 2013. While its earlier version could translate only one phrase at a time, the Android system is now capable of real-time translation in different languages[4].

As detailed above, the SER algorithm can be implemented in a wide range of applications in industrial fields. Researcher tried to improve the accuracy of SER over the past decade to get as closer as possible to human performance. Thus, what we need to answer is what kinds of the SER algorithms are currently used in research? In addition, if we want to improve SER accuracy, what is the state-of-the-art[5] of SER?

Here we would like to provide a brief specification of the mainstream research on SER and also the state-of-the-art of SER performance. A large number of studies have been performed in SER during the past two decades. Starting with the previous research in SER, there are two mainstream approaches

3 https://www.skype.com/en/features/skype-translator/

4 https://support.google.com/translate/

5 State-of-the-art refers to such a level of development reached at any particular time as a result of the common methodologies employed at the time.

used to obtain the adequate representation of emotional information from speech signals. Current SER algorithms commonly follow the procedure of feature extraction to feature selection to classification (from left to right), in order to encode the emotional information to be used in an application as shown in Figure 1.2.

The first approach, also known as the traditional approach, is to manually build the model on the acoustic representation (L. Chen, Mao, Xue, & Cheng, 2012). The resulting acoustic features (also known as low-level descriptor) are used to feed into the learning algorithm (Koolagudi & Rao, 2012). They commonly include pitch, formant, energies, and intensity (details are given in Subsection 3.1.1). The traditional approach has reached a performance of approximately 85% (Anagnostopoulos et al., 2015). The two main limitations of the traditional approach are (1) the acoustic features are not always optimally tuned to the task at hand, and (2) the features need to be manually constructed. The trend for this approach is to increase the number of features or to find new good representation features. Currently, the state-of-the-art for the traditional approach can be found in the Interspeech emotion challenges (REF to such a challenge), which was proposed in this field. The Interspeech competition provides the standard defined acoustic feature for SER. A recent development in the traditional approach is the use of the spectrogram of a speech signal as an image-like representation for SER. For example, Sun et al. used the local energy distribution of spectrograms for SER while (2015) extracted local texture features from spectrograms to achieve SER.

Due to significant learning and processing ability, a new approach has unsurprisingly attracted attention from researchers in recent years, which is deep learning for SER. Since Stuhlsatz et al., (2011) tried to use a Boltzman machine-based deep learning on SER, several researchers followed this approach to explore automatic learning feature representation. For example, Mao et al. (2014) employed Convolutional Neural Networks (CNN) to learn feature representation. Recently, Trigeorgis et al., (2016) presented an end-to-end learning system for SER that achieved an impressive accuracy which is 85% better than the traditional approach. However, The superior performance obtained through deep learning algorithms comes at the price of a large amount of datasets and computational resources for training.

1.3 SPEECH EMOTION RECOGNITION IN MANDARIN

Mandarin is the official language of China. In 2017, China's state news agency reported that about 70% of the Chinese population spoke Mandarin in 2015 (Cen et al., 2017), which means that 8.9 billion people speak Mandarin. With such a large population using this language, the development of Mandarin-oriented speech recognition technologies should be anticipated. The potential social implications and commercial values are expected to be large.

In fact, in 2016, the Chinese technology conglomerate Baidu established an online platform for speech recognition: Baidu Voice[6]. Each day Baidu Voice receives 140 million speech requests[7]. With the support of deep-learning algorithms trained on a huge volume of data, Baidu Voice can achieve a 97% accuracy rate under quiet conditions (cf. Collobert, Puhrsch, & Synnaeve, 2016).

Emotion is expressed by linguistic content, but also by other components of one's voice such as speed rate, tone, and volume. The voice, with its accompanying cues, has become an increasingly used carrier of information in social networking and messaging applications. As vocal (audio) data is less data-intenstive than audiovisual (video) data, but more enriched with emotional information than written forms, we claim that it makes the information exchange faster and more accurate. The growing user base of voice transmission applications, such as WeChat in China and WhatsApp in the West, provides strong evidence in support of this claim. The demand for a better SER method is more urgent than ever in view of the flourishing social networking and messaging applications. These are becoming increasingly voice-focused and will stimulate the acceleration of information exchange. However, despite this progress in speech recognition for smartphones and laptops, a SER application for this area has yet to be developed. Due to the limited accuracy of SER, major smartphone producers are currently only focusing on the recognition of speech content rather than on emotion (Longé, Eyraud, & Hullfish, 2017).

Currently, China has the largest number of users in the world of mobile devices. These devices are increasingly installed with speech recognition applications, such as Siri on the iPhone. Therefore, our research on SER algorithms aims to enrich existing knowledge of SER's potential social implications and commercial value for the immense Chinese society and market. The large amount of data generated from Chinese users exploiting these new information technologies may also benefit research in public and private sectors worldwide.

Furthermore, there is one crucial issue to be resolved by current and future SER application. The accuracy of SER is nowadays far from adequate, even when considering the recently advanced technology and algorithms. Bridging this gap is the focus of our research, as further detailed in the problem statement (PS) of this study (see Section 1.4).

1.4 PROBLEM STATEMENT AND RESEARCH QUESTIONS

Based on the above review, SER can be an important application in people's daily lives. However, due to its limited accuracy, SER performance needs to

6 http://http://yuyin.baidu.com/

7 https://github.com/baidu-research/warp-ctc.

be significantly improved. The Problem Statement (PS) for this study reads therefore as follows.

PS: To what extent can we improve SER accuracy using spectrogram information?

To ensure adequate comprehension of speech emotion, we need an algorithm capable of providing a precise performance. To address the PS, we formulate four research questions (RQs) for investigation by this study. We focus on the following three concepts: performance, features, and deep learning. The study addresses and answers the RQs by means of a well-balanced and well-selected scientific research methodology (see Section 1.5).

Formulation of RQ1

Section 1.1 has briefly reviewed the crucial emotional information contained in both voiced and unvoiced aspects of speech. The most important features of speech are the voiced aspects. However, there are no clear boundaries between voiced and unvoiced aspects of speech. Current methods to discern the boundary between voiced and unvoiced aspects of speech primarily exploit the intensity of speech signals for SER. Such techniques have produced unsatisfactory results, and researchers are calling for higher precision in determining the boundary between the voiced and unvoiced aspects of speech (Germain, Sun, & Mysore, 2013). Thus, if we could improve the performance of voice activity detection (VAD), we could consequently influence and enhance feature extraction for SER. Therefore we formulate the following RQ1.

RQ1: *Is it possible to design a new algorithm that improves the accuracy of detecting the voiced part activity in speech?*

To answer RQ1, our study proposes a new algorithm, the voiced segment selection (VSS) algorithm, which can produce an accurate segmentation of speech signals by using log-Gabor filters to detect voiced aspects of speech on a spectrogram. The VSS algorithm is evaluated by (1) a comparison with the current leading voiced activity detection algorithm, and (2) a comparison of SER performance with and without applying the VSS algorithm.

Formulation of RQ2

In previous studies (see, e.g., Jin, Li, Chen, & Wu, 2015), a large number of acoustic features were extracted from speech signals. Moreover, statistical descriptors were calculated using these features. A large portion of the features did not contain useful emotional information and were redundant in recognizing information. There are two prevailing drawbacks that are associated with the acoustic measurements: (1) shortcomings in the time domain and (2) shortcomings in the frequency domain. Spectrogram representations offer a means to deal with both shortcomings at once. Hence, we formulate the following RQ2,

RQ2: *How can we use two-dimensional features to analyze the spectrogram representation of speech?*

In the time domain, we can have the following observations. From the measurements, we can adequately calculate the information on durations or rates of speech emotion events, but we cannot identify different frequency signals in the speech. Similarly, the frequency domain can provide us with the details of the amplitude of the formant, but this is achieved at the expense of time. This implies that there is a limitation for us if we want to simultaneously measure both frequency and temporal location. If the temporal resolution is improved in time domain, it may lead to a less adequate estimation of the frequency, vice versa. This is analogous to the well-known as the Heisenberg's uncertainty principle. Interestingly, Gabor filters offer the optimal trade-off for dealing with both drawbacks. The Gabor function provides the best combination of temporal and frequency resolution. Filters designed according to Gabor's function are called Gabor filters. When applied to a temporal signal, the designed Gabor filters perform a localized measurement of the signal's frequency. The traditional Gabor filters are one-dimensional referred to as temporal Gabor filters. A spatiotemporal Gabor filter (SGFs) is an extension of the two spatial dimensions Gabor filters with a temporal component. A spectrogram (see Section 4.3) is the outcome of transferring sound signals into a two-dimensional visual representation. The resulting spectra (frequency histograms) form the columns of the spectrogram, where each column represents the spectrum of a temporal sample. Thus, Spatiotemporal Gabor filters provide good models for analyzing the combination temporal and spectra information in spectrogram. The visual representation can be detected with filters of certain time and directions.

Formulation of RQ3

Research question two focuses on the primary feature pattern of acoustic speech. With the RQ3, we aim to further categorize emotional expressions in a spoken sentence into primary and subsequent feature patterns according to the intensiveness of the speech.

RQ3: *Can we extract additional, and likely less-intensive features via the composition of Gabor filters through a spectrogram?*

We seek to reveal the feature patterns of the less intensive emotional expressions via a spectrogram. For this purpose, we carry out a feature extraction using Gabor filters on the feature patterns of both primary emotional expressions and less intensive emotional expressions. Through experiments, we investigate the performance of using primary and subsequent feature patterns by comparing the algorithm to the state-of-the-art algorithms.

Formulation of RQ4

Most previous studies in the field (see, e.g., Dahake, Shaw, & Malathi, 2016) have manually carried out the three stages of feature extraction, selection, and classification. This is a core element that we cannot ignore. Up to this date, the majority of studies using these three steps have placed emphasis on (1) estimating and (2) manually optimizing certain parameters (cf. K. Wang, An, Li, Zhang, & Li, 2015). Hence, it is possible that an estimate, although optimal from one perspective, may not be optimal from another perspective. If all three steps can be automatically performed, the need for human involvement in decisions can be reduced and the best choice can be determined. Therefore RQ4 concerns how we can use a deep-learning algorithm to improve the accuracy of SER.

RQ4: *Can we apply the deep-learning method to the spectrogram outcomes to extract "visual" features to increase the accuracy of SER?*

1.5 RESEARCH METHODOLOGY

The investigation of the four RQs requires a scientific research methodology that integrates research and affective computing. The methodology in this study consists of five parts. The rough details of each parts are summarized as follows.

(1) *To investigate the scientific literature.* The scientific literature is reviewed and analyzed. We aim (1a) to identify relevant state-of-the-art achievements, (1b) to identify the algorithms that have been used in previous studies, and (1c) to design the experimental procedures to achieve a stronger experiment performance. The relevant literature contains the following domains: (1) speech signal processing, (2) affective computing, (3) machine learning, and (4) emotion recognition.

(2) *To experiment with traditional SER algorithms.* We will develop a deep understanding of traditional SER algorithms that focus on acoustic features and machine learning. Therefore, we will investigate the best known and most used SER algorithms with a rapid and accurate performance. One of the instruments historically employed for achieving a reliable analysis is the spectrogram. It can visually display a combination of time, frequency, and energy information in an image of a speech signal. In each spectrogram image, vertical and horizontal axes represent time and frequency, respectively, while colors signify the energy of the speech signal. As a spectrogram is able to illustrate a combination of signal indicators, we believe that it has the potential to produce new feature groups that have not been previously encountered. Thus, the second RQ aims at finding a new kind of feature that contains efficient emotional information that does not overlap with the existing feature group.

(3) *To perform comparative experiments.* Comparative experiments are executed to determine the optimal setting for feature extraction and to evaluate performance through cross-validation.

(4) *To analyze and interpret the results of the experiment.* The results of the experiment are analyzed for three purposes: (4a) to determine whether the selected algorithms work for SER, (4b) to compare their performance with other SER algorithms presented in the literature review, and (4c) to reveal the shortcomings of the algorithm.

(5) *To validate the performance of the algorithms.* Based on the results of the experiment, we provide an answer to the RQs and the PS formulated in Section 1.4.

1.6 OUR CONTRIBUTIONS

In searching for answers to the four RQs and the PS, our study seeks to offer four major contributions to the field of speech recognition. They are briefly described below.

Contribution 1 is the VSS algorithm. We introduce the VSS algorithm to improve the detection of voiced segments (aiming at better results than has thus far been possible) and to extract acoustic features for classification. The goal is to improve SER performance.

Acoustic features are the fundamental and indispensable components of the SER procedure. The more precise the acoustic features that can be extracted from the speech signal using voiced and unvoiced selection, the more accurate the SER performance. Details regarding contribution 1 are provided in Chapter 5.

Contribution 2 is the fact that the SER performance will be achieved by a log-Gabor filter algorithm. The algorithm is designed to detect and obtain the relevant features by using log-Gabor filters on a spectrogram.

The new features are named (1) primary log-Gabor features and (2) subsequent log-Gabor features. We typically tend to focus on the parts of a sentence that demonstrate intensive emotional expressions. The feature patterns can be extracted through Gabor filters and can improve the accuracy of emotion recognition. We conducted feature extraction using Gabor filters on the feature patterns of both primary and subsequent emotional expressions.

Contribution 3 is the development of SER with a convolutional neural network (CNN) algorithm. We apply convolutional neural networks to learn features from speech data and then evaluate the learned feature representations on several classification tasks.

Convolutional neural-network algorithms (see Neumann & Vu, 2017) attempt to learn straightforward features in the lower layers and more complex features in the higher layers. Convolutional deep neural networks have a strong ability to scale an algorithm to high-dimensional data.

Contribution 4 is the evaluation of the performance of existing methods in relation to our proposed methods regarding Mandarin speech from a Chinese database.

As far as we know, research in the field has dominantly focused on Western languages and has given less attention to Eastern language databases. This was an additional motivation for using a Chinese database in this study. The study demonstrates that the algorithm can also work well for the Mandarin language.

1.7 THESIS OUTLINE

The thesis comprises nine chapters. The structure is outlined below.

Chapter 1: Introduction

Chapter 1 provides an introduction to the thesis. We give an overview of SER and a rough description of the ideas of the algorithms. The chapter formulates the PS and four RQs. Our research methodology is then described, and the major contributions are listed. An outline of the structure of the thesis is also given.

Chapter 2: Speech Emotion Expression

Chapter 2 provides a review of emotion definitions and emotional state labelings. Three preliminary questions are addressed: (1) What is the definition of speech emotion? (2) How many emotional states are there? and (3) How does emotion affect the participant's expression in speech? The three questions are discussed, answered, and generalized in Chapter 2. In addition, the significance of studying SER is discussed.

Chapter 3: From Speech Signal to Emotion Recognition

Chapter 3 presents an overview of the three stages of SER. First, we review feature extraction and the most commonly used acoustic features in SER. Subsequently, we provide details regarding the feature selection method used in our research. Third, the classification algorithms are analyzed.

Chapter 4: Tools and Techniques

Chapter 4 provides a brief explanation of the tools and techniques used in this study. We first describe the databases chosen for our experiments. The spectrogram and log-Gabor filters are then introduced as the key tools in our research. They are used extensively in the experiments in Chapters 5 to 7.

Chapter 5: The Voiced Segment Selection Algorithm

In Chapter 5, we propose a new algorithm, the VSS algorithm, which produces a more accurate segmentation of speech signals by dealing with the voiced signal segments as image processing. The VSS algorithm significantly differs from the traditional methods. Moreover, we use log-Gabor filters to extract the voiced and unvoiced features from a spectrogram to classify the features. RQ1 is divided into RQ 1A and RQ 1B. Both questions are answered in this chapter. So, is RQ1.

Chapter 6: The Basis of Primary Feature

Chapter 6 describes the primary log-Gabor filter algorithm, which uses log-Gabor filters to extract the spectro-temporal features of emotional information from a spectrogram. The unique pattern that we design for each type of emotion in the spectrogram is illustrated. The recognition performance demonstrates that the new features are efficient for the feature group. Chapter 6 concludes with the answer to RQ2.

Chapter 7: Less-Intensive Features in Spectrograms

Chapter 7 proposes a further study seeking to categorize emotional expressions in a sentence into primary and less-intensive emotional expressions according to their intensiveness. This chapter reveals the feature patterns of the subsequent emotional expressions in a spectrogram. We use log-Gabor filters to identify and extract the subsequent feature patterns for different emotions in the spectrogram. Whereas Chapter 6 concentrated on the parts of a sentence that demonstrate intensive expression of emotion, Chapter 7 conducts feature extraction using subsequent Gabor filters on the feature patterns of less-intensive emotional expressions. Finally, the RQ3 is answered.

Chapter 8: Deep Learning of Mandarin Feature

Chapter 8 briefly describes the convolutional neural network (CNN) algorithm and the reasons that it is a necessary asset for our work. First, the architecture of the neural network is illustrated. We then explain the details of feature learning from a spectrogram using the CNN. The classification of data is subsequently demonstrated for each type of emotion. Finally, a comparison of algorithmic performances is given and analyzed. Then RQ4 is answered.

Chapter 9: Conclusion and Future Research

Chapter 9 provides an answer to the PS. The chapter begins by summarizing the answers to RQ1, RQ2, RQ3, and RQ4. We then review the PS, formulate our conclusions and offer recommendations for future research aimed at further improving SER.

2 SPEECH EMOTION EXPRESSION

In a natural environment, speech emotion has been found to attract more human attention than any other source of expression (see Belin, Zatorre, & Ahad, 2002). At the start of our research, there are three prevailing questions requiring clarification: (1) What is the definition of speech emotion? (2) How many emotional states are there? and (3) How does emotion affect the speaker's expression in speech? These are important questions because they affect the manner in which we approach the study of SER (Speech Emotion Recognition). These questions define our research and its relation to behavioral changes. They are answered in three sections.

The course of the chapter is as follows. In Section 2.1, we provide an overview of the existing definitions of emotion and describe our decision regarding a definition to be used in this study. In Section 2.2, the categories of different types of emotions are outlined. Section 2.3 discusses the question of how emotional expression affects speech. Finally, a brief chapter summary is provided in Section 2.4.

2.1 DEFINITION OF EMOTION

The topic of this dissertation, SER, reflects one of the key components of this study: *emotion*. Therefore, we begin by answering the following question: What is our definition of emotion?

For more than a century, scientists have been attempting to formulate a universal definition of the term. Moreover, they have sought to separate emotion from other affective states (cf. Cabanac, 2002). Thus far, there have been many debates on emotion, and researchers have not achieved a consensus regarding a shared or common definition (Plutchik & Kellerman, 2013b).

So, emotion is difficult to define, given that scholars from different disciplines have speculated for years regarding a proper definition of emotion to employ within their own methodologies. Psychologists believe that emotion is a psychological reaction that attempts to send or receive affective attention to a particular event or person (cf. Ketai, 1975). Izard stated that "a complete definition of emotion must take into account [...] the experience or conscious feeling of emotion [...], the processes that occur in the brain and nervous system and the observable expressive patterns of emotion" (see Izard, 2013). Neurologists have attempted to define emotions in two prevalent physiological reactions: (1) the experience of feeling and (2) bodily behavior (cf. Heilman & Gilmore, 1998).

Buck defined emotion as direct feelings and desires, derived from neurochemical systems (see Buck, 2000). Moreover, emotion can be explained using a range of psychological features, including personality, temperament, motivation, and mood (cf. Myers, 2004).

As we are focusing on the non-verbal manifestation of emotion in speech, which can only be grasped with an understanding of both psychology and neurology, we require a broader definition of emotion. Taking this into consideration and seeking to employ explicit and consistent definition in our research, we use the definition proposed by Barrett, Dunbar, and Lycett (see Definition 2.1).

Definition 2.1 Emotion (adapted from Barrett, Dunbar, and Lycett, 2002)
Emotion is defined as a complex and spontaneous mental reactionary phenomenon based on an individual person's response to a specific event in a particular environment.

2.2 EMOTIONAL STATE

One of the additional major problems encountered when investigating emotion is the lack of commonly recognized categories of emotional states. Thus we cannot straightforwardly answer the question: how many emotional states do exist? This is crucial for SER because the algorithm must engage in classification and recognition based on a clear categorization of emotional states. Therefore, we search for answers to the following questions: (1) which emotional states do exist? (2) Which emotional states should we select for experimentation in our study?

During the last half century, researchers have investigated the categorization of emotional states, but an explicit or unified categorization has not yet emerged. Psychologists have attempted to answer the question of "how many kinds of emotion do exist in the world?". The debate on kinds and numbers has ignited many arguments over the last several decades (cf. Gendron & Barrett, 2009). Earlier research on these topics (kinds and numbers) has been performed by Steunebrink. In his Ph.D. thesis, the logical structure of emotion has been analyzed and a new number of categories of emotional states has been proposed (Steunebrink, 2010).

Part of the difficulty on kinds and numbers derives from the fact that our emotional experience is complex and involves numerous factors. Thus, distinguishing one emotion from another is like drawing lines of sand in the desert. It is similarly difficult to determine the boundaries between emotions in human speech (Steunebrink, 2010).

Despite these difficulties, many researchers are still attempting to distinguish among emotional states. However, most theories on emotion are highly subtle and focus on a certain aspect of emotion. Moreover, there are countless scenarios for each emotional state. In brief, no model is entirely satisfactory for all emotions (Steunebrink, 2010).

The following section introduces two mainstream categorization methods. The first method, fundamental emotion classification, is based on the observation that emotion is separable and can be distinguished into coarse categories. The second method is called multi-dimensional emotion classification. It is constructed on a dimensional space representation using a variety of emotional attributes (e.g., valence, arousal, and control). The details of these methods are explained in Subsection 2.2.1 and Subsection 2.2.2.

2.2.1 Fundamental Emotion Classification

Our literature review on the classification starts with two different lists (Table 2.1 and Table 2.2). Together they list 88 names for emotional states. The lists which are based on Plutchik & Kellerman (2013a) and on Anagnostopoulos et al (2015). Anagnostopoulos et al obtained the lists by reviewing scientific literatures over a period of 11 years (from 2000 to 2011). The numbers after the name of emotions indicate their frequency in research studies. Table 2.1 lists the emotions with a frequency greater than one (48 in total), while Table 2.2 catalogs those emotions with a frequency of exactly one (40 in total). To adequately address emotional states, one must identify clusters of fundamental emotional states. This approach reduces the numbers of variables and makes the data fit for our research.

In the 1970s, Paul Ekman was the first researcher to propose and identify six emotional states that could be universally recognized. He announced the six types of emotion as the *fundamental* emotional states, and called them the *"big six"* (Ekman, Sorenson, & Friesen, 1969). This set of emotions comprises: (1) anger, (2) disgust, (3) fear, (4) happiness, (5) sadness, and (6) surprise. Some scholars have rejected this list because the enumerated emotions do not cover the whole range of emotion. However, despite this limitation, the six basic emotions are generally viewed as measurable and separable according to a universal standard. Even when speakers have different cultural backgrounds, researchers are able to arrive at the same conclusions about their speech.

Whatever the case, Ekman's fundamental emotional states are widely used in automatic emotion recognition studies because they provide an explicit separation of emotional states (cf. Frijda, Kuipers, & Ter Schure, 1989; Sebe, Cohen, Gevers, & Huang, 2006; Izard, 1992). We use the fundamental emotional states as classification labels in our machine learning experiments. However, here we remark that in other cultures, the six fundamental emotions that Ekman defined for the Western world cannot always be traced. For example, the

Table 2.1: Frequency of occurrence of 88 emotional states in speech emotion expression part one
[adapted from Plutchik (2013) and Anagnostopoulos et al. (2015)]

Emotion	No.	Emotion	No.	Emotion	No.
Angry	85	Satisfaction	5	Irony	3
Fear	65	Pain	4	Coquetry	2
Sad	65	Tenderness	4	Disbelief	2
Happy	44	Admiration	4	Objectivity	2
Joy	31	Determination	3	Pleading	2
Disgust	26	Scornfulness	3	Hate	2
Surprise	24	Affection	3	Pomposity	2
Boredom	17	Cheerfulness	3	Threatened	2
Contempt	15	Longing	3	Relief	2
Love	10	Impatience	3	Reproach	2
Grief	9	Enthusiasm	3	Sarcasm	2
Interest	7	Uncertainty	3	Reverence	2
Anxiety	6	Contentment	3	Timidity	2
Doubt	6	Sorrow	3	Gladness	2
Elation	5	Shame	3	Comfort	2
Sympathy	5	Laughter	3	Confidence	2

corpus used in this research, the Mandarin Affective Speech (MAS) corpus (T. Wu, Yang, Wu, & Li, 2006), only categorizes emotions into five emotional states: angry, panic, happy, sad, and neutral. Notably, three of them, viz. angry, happy, and sad are mentioned in Ekman's list of the six basic emotions. To conveniently address this mismatch, we treat panic as equal to fear. The neutral emotion signifies a state that is neither strongly positive nor strongly negative. More details on the MAS corpus are given in Section 4.2. The set of five emotions is used to categorize the basic emotional states in this study's experiments on the Mandarin Affective Speech in the Chapters 5 to 8.

2.2.2 Multi-Dimensional Emotion Classification

Apart from efforts to complete the list of fundamental emotional states, few scholars have attempted to distinguish more emotional states. Instead, scholars have primarily examined the level of a speaker's emotional response, or arousal. Arousal is the key component that serves as a predictable and detectable change in speech production to be assessed.

Table 2.2: Frequency of occurrence of 88 emotional states in speech emotion expression part two
[adapted from Plutchik (2013) and Anagnostopoulos et al. (2015)]

Emotion	No.	Emotion	No.	Emotion	No.
Solemnity	1	Relaxation	1	Desire	1
Irritation	1	Calm	1	Delight	1
Kindness	1	Rage	1	Accommodation	1
Aversion	1	Fury	1	Tension	1
Insistence	1	Amusement	1	Dominance	1
Seductiveness	1	Disdain	1	Excitement	1
Pleasure	1	Friendliness	1	Dislike	1
Approval	1	Disappointment	1	Complaint	1
Nervousness	1	Grim	1	Terror	1
Worry	1	Hostility	1	Aggression	1
Panic	1	Humor	1	Boldness	1
Shyness	1	Jealousy	1	Startled	1
Lust	1	Indignation	1	Pedantry	1
Astonishment	1				

Many researchers have attempted to structure emotional states within a multi-dimensional space, in contrast to the basic models described above. The idea of creating an emotional space leads to a representation of emotional states on axes. The merit of a dimensional approach is that it allows researchers to avoid distinguishing each emotional state arbitrarily. Moreover, researchers are continuously able to place additional emotions in the multi-dimensional space.

Definition 2.2 Arousal (adapted from Lewis, Haviland, and Jeannette 2010) Arousal is normally defined as the experience of restlessness, excitation, and agitation. It manifests itself in heightened overt and covert bodily activities that create a readiness for action. Emotions can be classified in terms of how arousing they are.

Definition 2.3 Valence (adapted from Lewis, Haviland, and Jeannette 2010) Valence is used in discussing emotion. It measures an emotion based on the intrinsic attractiveness or unattractiveness of an event, object, or situation.

The most widely used theory in emotion classification is the two dimensional space structure. The most well-known structure is the arousal–valence model, also known as the "circumplex" model (cf. Russell, 1980). As illustrated in Figure 2.1, it uses arousal and valence as the attributes for the axes. Arousal (see Definition 2.2) represents the energy used for an emotional experience using a circular scale from excited to calm. Valence (see Definition 2.3) represents how the emotional experience feels using a circular scale from positive/pleasant, to negative/unpleasant.

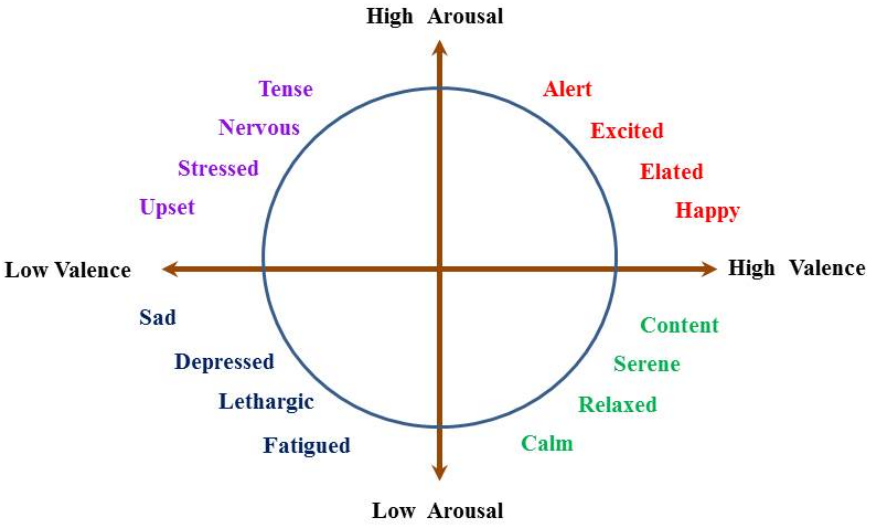


Figure 2.1: The two-dimensional emotional space (arousal - valence) [adapted from Lewis, Haviland and Jeannette (2010)]

As Figure 2.1 reveals, the two elements, arousal and valence, expose the relation between a speaker's emotional state which may be related to the acoustic characteristics of his voice (see Davitz, 1964)(The details of acoustic characteristics are introduced in Section 2.3). For example, high arousal is associated with high average vocal pitch values (Apple, Streeter, & Krauss, 1979) or with a fast speech rate and short pauses (Breitenstein, Lancker, & Daum, 2001). Moreover, numerous studies have demonstrated that emotions associated with a positive valence generate a low mean pitch value, shorter pauses, and a low voice intensity (Scherer, 1972).

These acoustic characteristics have close ties with SER, because they are the crucial signifiers of emotion in speech. Hence, the characteristics are widely

used as the feature group for the SER algorithms. This connection will help us to understand and use the acoustic features in this study.

2.3 THE EFFECT OF EMOTIONAL EXPRESSION ON SPEECH

Our study investigates non-verbal, or paralinguistic, expressions in speech. It means that we do not examine *what* a person says, but *how* that person says it. With this in mind, the section answers the question of how different emotional states shape the non-verbal aspects of a person's speech.

Since the 1950s, many physiologists and neurologists have been studying the topic. They have used physiology and neurology to analyze the body and nervous system. Their findings have illustrated that the stimulation caused by emotion can influence the body's nervous system, which consequently affects the speaker's style of speaking (Scherer & Zei, 1988). For example, the vocal cord's muscles can change under different emotional states affecting a speaker's vocal characteristics (Scherer, 1995). Acoustic characteristics of speech can vary widely according to different movements of the vocal cord muscles in the context of different emotions.

Table 2.3 lists four well-known acoustic characteristics (pitch, F0 contour, speaking rate, and intensity) and their definitions (column 3) in the literature of SER (Kiktova-Vozarikova, Juhar, & Cizmar, 2015). The fundamental frequency, also known as pitch (see row 2), is defined as the lowest frequency of a periodic waveform. In music, the fundamental frequency (also called simply the fundamental) is the musical pitch of a note that is perceived as the lowest partial present [8]. In speech, it is defined as the inverse of the signal period of a periodic signal. When we begin to speak, it is typically natural for our F0 contour (see row 3) to individually vary within a range of frequencies (Postma-Nilsenová, Postma, & Gu, 2014). The variation of F0 within the contour can usually be associated with his current emotional state. The speaking rate (see row 4) is defined as the number of speech units that can be produced within a standard amount of time. The most well-used measurement for the speaking rate is syllables per second. Speaking rate is believed to vary within the speech of one person according to his emotional state. Intensity (see row 5) is defined as the power carried by sound waves per unit area. It is commonly understood that various emotional states can influence the intensity of a person's speech.

Table 2.4 provides an overview of the associations between acoustic characteristics and the emotional expression in the speech signal (cf. Hammerschmidt & Jürgens, 2007; Forsell, 2007). Here, we aim to shed light on the correlations between acoustic characteristics and five fundamental emotional states (angry, happy, neutral, panic, and sad). Current measurement methods can be cate-

8 The fundamental frequency is usually abbreviated as f0, or F0 or FF. This refers to the lowest frequency counting from zero

Table 2.3: Acoustic characteristics and their definition

Acoustic characteristic	Perceived correlation	Definition
Pitch	F0	The inverse of the signal period of a periodic signal.
F0 contour	Pitch contour	Sequence of F0 values across an utterance. In addition to changes in pitch, the F0 contour contains temporal information.
Speaking rate	Speaking tempo	The number of speech units of a given type produced within a given amount of time.
Intensity	Volume of speech	The power carried by sound waves per unit area.

gorized into two groups: (1) assessing the speech signal based on the time (temporal) domain measurement and (2) assessing the speech signal based on the frequency (spectral) domain measurement (Busso, Lee, & Narayanan, 2009). As the names indicate, the time domain measurement is typically used according to the progression of acoustic features over time, while the frequency domain measurement makes an evaluation according to the signal response of different frequencies.

Table 2.4: Commonly reported associations between acoustic characteristics and a speaker's emotions
[adapted from Hammerschmidt et al. (2007) and Forsell (2007)]

Emotion	Angry	Happy	Neutral	Panic	Sad
Pitch	Extremely higher	Much higher	Normal	Extremely higher	Slightly lower
F0 contour	Much wider	Much wider	Normal	Narrower	More monotone
Speaking rate	Slightly faster	Faster	Slower	Much faster	Slightly slower
Intensity	Louder	Louder	Quieter	Normal	Quieter

Scholars have used both measurement methods to assess speech signals and identify the acoustic characteristics that are distinct to each of the five emotions. Numerous experiments on emotional states have been conducted, and their results have been consistent in Western language.

The verified results are the following: (1) Angry speech commonly evokes a slightly faster speaking rate, louder volume, and a much higher average F0 value (see Williams & Stevens, 1972). (2) Happy speech has a wider range of F0 values and features a fast speaking rate, as well as higher energy usage (see Hammerschmidt & Jürgens, 2007). (3) Neutral speech is characterized by an average F0 value that is lower than that seen when one is angry, happy, or in panic but higher than that seen when one is sad. For neutral speech, the range of frequency values is narrower than for angry and happy speech but wider than that seen that seen during the vocal expression of panic. Therefore, the frequency of neutral is in the mid-range among the five fundamental emotions. Moreover, neutral speech is associated with a mild energy and a very slow speaking rate (see Pittam & Scherer, 1993). (4) The vocal expression of panic is associated with a similar pronunciation style as anger, but with certain differences. The frequency of speech changes more quickly than that of angry speech and thus generates a sharp energy contour if drawn as a curve in a time-frequency spectrogram. In addition, when panic is expressed, the speaking rate is the fastest among all emotional states (see Van Lancker, 1991). (5) Finally, sad speech invokes a lower vocal frequency, but the speech duration is longer than for the other types of emotions (see Sidtis & Van Lancker Sidtis, 2003). In the scholars' investigations and findings highlighted above, the five emotional states in terms of F0 values, the range of frequency in speech, as well as in speaking duration and intensity. The above findings suggest that emotions can be detected through acoustic characteristics. That is, acoustic characteristics can effectively represent and thus help to distinguish emotions.

To summarize, emotional expression is an essential component of speech. Acoustic characteristics can be analyzed in a time and frequency domain representation of speech signals so that we may categorize them and correlate them with respective emotional states. This is how we intend to identify speech emotion through acoustic characteristics, specifically through visual speech-signal representations called spectrograms.

2.4 CHAPTER SUMMARY

In this chapter we discussed a variety of definitions of emotion and presented the definition utilized in our study. We define emotion as "a complex and spontaneous mental reactionary phenomenon based on an individual person's response to a specific event in a particular environment" (see Definition 2.1). Different categorizations of emotional states were then introduced and the *big six* emotional states proposed by Ekman (anger, disgust, fear, happiness, sadness, and surprise) were described. We then outlined a set of five fundamental emotional states derived from the Mandarin database used in this research. The set consists of angry, happy, neutral, panic, and sad. Finally, we presented the details of commonly used acoustic characteristics and discussed previous

approaches that other studies have employed to examine the link between emotional states and acoustic characteristics. The correlation between emotions and acoustic characteristics serves as the foundation for our research design.

3 FROM SPEECH SIGNAL TO EMOTION RECOGNITION

In this chapter, we review (1) the relevant methods used in the field of speech emotion recognition (SER) and (2) the technologies used in the research described in the subsequent chapters. In Section 3.1 we deal with two methods used for feature extraction. In Section 3.2, two different approaches to feature selection are introduced. Section 3.3 subsequently describes four classification algorithms employed in this area. Finally, Section 3.4 provides a chapter summary.

3.1 FEATURE EXTRACTION

In this section, we investigate which commonalities among emotional speech signals are suitable candidates for use in an algorithmic model. Feature [9] extraction (see Definition 3.1) is a method for extracting commonalities from raw data. Such commonalities indicate that a feature can be beneficially incorporated into a machine-learning algorithm (K. Wang et al., 2015). For example, as mentioned in Chapter 2, different emotional expressions in speech are associated with (or related to) certain acoustic characteristics (cf. Wu, Parsons, & Narayanan, 2010).

Definition 3.1 Feature Extraction (adapted from Ethem Alpaydin, 2014)
Feature extraction is defined as a process for measuring and building derived features to be used in subsequent learning and generalization algorithms.

Thus, the first step in SER is composing a suitable and informative feature set that efficiently represents emotional states. Most scholars in this field believe that an algorithm with a proper feature set significantly influences SER performance (Mencattini et al., 2014). The better the feature-based representation we achieve, the stronger the performance we may obtain. Thus far, previous research in this area has identified two methods for detecting and extracting a feature set. We call them *feature construction* (see Subsection 3.1.1) and *feature learning* (see Subsection 3.1.2).

9 In machine learning, a feature is an individual measurable property or characteristic of an observed phenomenon.

3.1.1 Feature Construction

Feature construction (see Definition 3.2) means manually building and extracting features and then transforming them into statistical features for classification. Acoustic characteristics primarily include the fundamental frequency (F0), speaking rate, and intensity (energy), as explained in Section 2.3. All these acoustic characteristics can be extracted from speech and are typically suitable representative features for the speech signal in general. They are called *acoustic features*. Acoustic features are the most economical, objective, and commonplace means of representing acoustic characteristics in SER (Scherer & Ekman, 1982). In addition to the feature mentioned before, there are other acoustic features such as the formants F1, F2 and F3, the zero-crossing rate (ZCR), linear prediction coefficients (LPC), and mel-frequency cepstral coefficients (MFCC). In detail, the first three formant frequencies, F1, F2 and F3, are estimated as the resonant frequencies of the vocal tract using linear predictive analysis (see Low, Maddage, Lech, & Allen, 2009). The MFCCs are calculated from a bank of auditory filters for the outputs. The filters are equally spaced on the logarithmic frequency scale, called the mel scale (Pao, Chien, et al., 2007). From the acoustic features, statistical features are derived using utterance-level mathematical statistics (C. Lee, Mower, Busso, Lee, & Narayanan, 2011). Statistical features primarily include the maximum (max) value, minimum (min) value, and average value of acoustic features (Li & Akagi, 2016).

Definition 3.2 Feature Construction (adapted from Alpaydin, 2014)
Feature Construction is the process of using domain knowledge of data to create features that make machine learning algorithms work.

Thinking beyond the existing group of acoustic features, we wondered whether there are still remaining features that can be manually detected and extracted. Of course, they should improve the performance of SER. As such, we began to investigate feature construction thoroughly. We found that the existing SER prior to this study had not paid much attention to the information advantage provided by spectrograms (see Chapter 4). Inspired by these observations from the literature (Bouvrie, Ezzat, & Poggio, 2008), we focus on developing and assessing novel algorithms that use spectrograms with the aim to extract features that are distinct from the acoustic features. By accurately identifying novel features, which means ruling out those that are irrelevant, we intend to reduce the computational expenses of features extractions significantly, and to improve the accuracy of the classification. The new feature construction process is described in Chapter 6 and Chapter 7.

3.1.2 Feature Learning

In contrast to the feature construction method, which entails manually designing and extracting features from the speech signal, an alternative method called feature learning (see Definition 3.3), also known as representation learning aided by deep-learning technology, is currently popular for feature extraction.

Definition 3.3 Feature Learning (adapted from Ng Andrew, 2009)
Feature learning is defined as a set of techniques that allows a system to discover the representations needed for feature detection or classification from raw data automatically.

Hinton et al. (2012) reported on their breakthrough achieved with deep learning on the task of phonetic classification for automatic speech recognition. The paper contained the shared views of four research groups. To the best of our knowledge, it was the first major industrial application of deep learning. In the same year, another study by Mao et al. used convolutional neural networks (CNNs) to learn the acoustic features of speech. In their study, a competitive performance was obtained for a Mandarin corpus (Mao et al., 2014). Their work marked the start of an interesting development. In the following year, Lee and Tashev attempted to learn the acoustic features of speech based on a long short-term memory (LSTM) recurrent neural-network algorithm. They found that the most useful features in an utterance are concentrated in a few frames. They obtained approximately 60% performance based on the IEMOCAP database (J. Lee & Tashev, 2015). In 2016, a DNN algorithm was applied to learning a mapping from Fourier-transform based filter banks. Soft labels generated from multiple annotators were used to model the subjectivity of emotion recognition (Fayek, Lech, & Cavedon, 2016). In their more recent study, Fayek, Lech, and Cavedon (2017) used recurrent neural network architectures and the feed-forward method to work on SER. In their work, the features were learned from spectrograms instead of speech signals.

In our study, in addition to relying on feature construction, we attempt to engage in feature learning using a CNN, which has the ability to automatically learn features from a spectrogram. This approach consists of feature learning with no domain knowledge for the features, which are in some sense "predesigned". If the manually constructed features are also detected and extracted by the feature-learning algorithm, that outcome would serve as confirmation that our feature construction was effective. Our feature-learning algorithm is described in Chapter 8.

3.2 FEATURE SELECTION

The second stage of SER is feature selection (see Definition 3.4). In the first stage of feature extraction, the extracted feature group may contain a large number of features. Among the many extracted features, a large proportion may be either redundant or irrelevant. Therefore, they can be removed without incurring a significant loss of information. In fact, this elimination is the central premise when using a feature selection technique. There are several reasons for eliminating a portion of the extracted features, and the two most important reasons are the following:

(1) This approach makes emotion recognition easier by selecting the more useful features of speech. There are many acoustic features involved in emotional expression, but they are not all equally relevant to, or informative of, speech emotion. Therefore, a selection algorithm is needed to choose the most useful acoustic features to be extracted so that the redundant features can be removed without losing significant information.

(2) This approach improves the overall accuracy of classification by reducing overfitting[10] (cf Bermingham et al., 2015). Previous studies have found that redundant features can hamper the accuracy of a classification or the robustness of the classifier (Witten & Frank, 2005). Thus, an adequate feature selection algorithm is needed to optimize the feature group.

Moreover, it is important to note that a small correlated feature group can generate a more accurate performance. The computational time consumed by the learning process may also be significantly reduced by filtering irrelevant features (Liu & Motoda, 2012).

Definition 3.4 Feature Selection (adapted from Van der Maaten, Postma, Van den Herik, 2009)
Feature selection is defined as the process of choosing a subset of relevant features for classification.

There are two major types of algorithms used during the feature selection stage: (1) feature selection algorithms and (2) dimensionality reduction algorithms. A feature selection algorithm selects the best features from the feature group. It aims to choose the most valuable sub-features from the entire extracted feature group. Chastagnol and Devillers used forward selection in their study (Chastagnol & Devillers, 2012), as well as Schuller et al. and Wu et al. applied backward elimination (Schuller, Vlasenko, Eyben, Rigoll, & Wendemuth, 2009; S. Wu, Falk, & Chan, 2009).

10 Overfitting refers to a machine learning method that is fitted too much to the training data at the expense of its fit to unseen (test) data.

Prior to feature selection, dimensionality reduction is used to reduce the feature group from many dimensions to fewer dimensions (see Van der Maaten & Hinton, 2008). After dimensionality reduction, the feature group may have a lower level of complexity and fewer dimensions. This type of algorithm does not directly choose a feature from the original feature group, but rather obtains a small set of values through the calculation of possibly correlated variables. For example, principal component analysis (PCA) is one of the more notable algorithms employed in dimensionality reduction (see Van der Maaten, 2009).

Principal component analysis is a statistical procedure that converts the N correlated variables of the whole data set into a set of N linearly uncorrelated variables called *principal components* (cf. Van der Maaten, Postma, & Van den Herik, 2009b). The number of M ($M < N$) of retained principal components is (considerably) less than the number of original variables.

In our experiments, PCA is chosen for acoustic feature selection procedures because of its ability to reduce the number of dimensions of the features. Our findings reveal that emotion recognition performance improves when using PCA (see Chapter 5 to 7). The feature selection does not apply to CNNs, because they incorporate both feature (representation) learning and selection together. Therefore, the feature selection is not necessary for the CNN.

3.3 FEATURE CLASSIFICATION

To investigate the effectiveness of our feature extraction and feature selection approach, we need to make feature classifications (see Definition 3.5) to recognize the emotional state of the speech signal on the basis of those features. Researchers have tried to utilize a number of pattern recognition algorithms over the last two decades (El Ayadi, Kamel, & Karray, 2011).

The four most prevalent algorithms are Hidden Markov Models (Pao, Liao, et al., 2007), Decision Trees (C. Lee et al., 2011), Support Vector Machines (Qi, Tian, & Shi, 2013) and Artificial Neural Networks (Trigeorgis et al., 2016). Actually, there is no agreement on which classification is the most suitable for SER. Each classifier has its advantages but also associates with limitations. All four classifiers will be detailed in the following.

Definition 3.5 Feature Classification
Feature Classification is defined as the process of identifying which new object belongs to which set of categories. It normally relies on a training set of data containing observations for which the category membership is known.

The first classifier to be discussed is the Hidden Markov Model (HMM). It is the most commonly used classifier in the literature for SER (Le & Provost, 2013). HMM is modeled via a Markov process which consists of a Markov

chain of which the states are unobserved (hidden). The states of the model are responded to capture the feature structure from the data.

The second classifier which has been used in SER is the decision tree (C. Lee et al., 2011). A decision tree is a predictive modeling approach to observe an object (represented in the branches) and then to conclude the object's target value (represented in the leaves).

The third classifier is the Support Vector Machine (SVM). The SVM has been used more frequently in recent studies (e.g., Nan, Sun, Chen, Lin, & Toh, 2017). These algorithms treat classification as a space boundary problem. In a multi-dimensional space, an SVM attempts to set a hyperplane that can be used to separate vectors into a binary class. This separation is achieved by evaluating the largest distance between hyperplanes containing the training vectors. Thus, in this study, we used an SVM as one of our classification algorithms to evaluate the performance of the feature construction method that we have proposed. Details are given in Chapter 5 to 7.

The fourth and nowadays most popular classifier used for SER is a variety of Neural Networks (NN). The NNs could be categorized into three types: Deep Neural Networks (DNN), Deep Belief Networks (DBN), and Recurrent Neural Networks (RNN), all of which are commonly used in SER. The advantage of an NN becomes more obvious in feature extraction, feature selection, and feature classification. NN achieves better classification performance than all the above classifiers, especially when the size of dataset is relatively large. However, an NN also has some disadvantages. First, it is highly demanding in regard to the computational cost. Second, in general, an NN has many parameter values (the weights and biases) and so-called hyperparameters (e.g., the number of the hidden layers, the number of the neurons in each layer) set in advance in order to achieve the best performance.

Apart from the SVM, Convolutional Neural Network (CNN), which falls into the category of DNN, is another classifier employed in this study. It has the capacity of sequentially processing feature learning and feature classification. A CNN consists of several layers of connected neural units. During the learning process at each layer, the weight values change through the connected units in real time. Details are given in Chapter 8.

3.4 CHAPTER SUMMARY

This chapter summarized the emerging studies according to the three stages of SER: feature extraction, feature selection, and feature classification. In Section 3.1, the process of feature extraction was briefly reviewed, including feature construction and feature learning. Because of the limitations of frequency and time domain measurements, we used spectrograms to visually combine the two with energy value in an attempt to extract novel features through feature construction in combination with feature learning.

Feature selection was then introduced in Section 3.2. We detailed the reasons why feature selection is crucial as one of the stages and illustrated the two mainstream approaches of feature selection and dimensionality reduction. We chose PCA, one of the dimensionality reduction algorithms, for feature selection. In support of this choice, we introduced PCA in detail and explained its advantage.

Furthermore, to investigate thoroughly feature extraction, i.e., the combination of the feature construction and feature learning, we need to have a deep understanding of feature classification for the final stage of SER. The definition of feature classification was given in Section 3.3, together with several commonly used classifiers which include HMM, decision trees, SVM and NN. The advantages and disadvantages of each classifier were outlined. Based on them, SVM and CNN were chosen for the experiments in this study.

4 TOOLS AND TECHNIQUES

To implement a newly designed algorithm and evaluate its performance, we require tools and techniques. They are detailed in this chapter and used in the research described in the following chapters. The structure of this chapter is as follows. In Section 4.1 we describe how to record emotional expressions. Then the corpus for affective speech is described in Section 4.2. Subsequently, spectrograms are explained in detail in Section 4.3. Section 4.4 explains the log-Gabor filters. Spectrograms and log-Gabor filters are both key tools that are used in the experiments described in the Chapters 5 to 7. Finally, Section 4.5 provides the chapter summary.

4.1 HOW TO RECORD EMOTIONAL EXPRESSION

In the literature on SER, common methods for collecting data include amassing experimental results and online recording (cf. Zanettin, Bernardini, & Stewart, 2014). Using a database created from social events in real-world situations is likewise convenient and practical (Uhrin, Chmelikova, Tovarek, Partila, & Voznak, 2016). However, there are many inherent difficulties in producing a natural speech database. For example, the influence of background noise is an issue. In addition, the collection of natural speech may raise legal and ethical issues that could prevent the database from being used for research purposes. Therefore, constructing a database of emotional speech is a delicate matter in the context of research. As such, existing databases have primarily been constructed in lab environments. In most cases of corpus collection, professional actors are asked to perform and simulate emotional expressions. Even then, several problems may arise from a database of acted emotions.

For instance, many researchers suspect that the differences between spontaneous real-world expression and acted expression can influence the objectivity of research. Here we mention two issues: (1) an actor can make mistakes (Kotani, Yoshimi, Nanjo, & Isahara, 2016) or (2) exaggerate emotion. these issues, a special procedure is commonly followed, which we consider to be a requirement. A listening test is therefore necessary after data collection. Researchers use the listening test to ensure that the actor can accurately convey emotional expression (Schuller et al., 2010). The test attempts to detect and remove under-expression, over-expression, and incorrect expression.

As mentioned in Chapter 1, most previous studies on SER are based on European languages. Here we observed that the Mandarin language remains

under-researched in this field. Given the fact that in 2017, China's general public used more new technologies than citizens in other parts of the world (Gong & Cortese, 2017), and given the social transformation that technological innovation has brought to many populations, testing the current understanding of Western methods and theories with the Chinese language is a highly valuable issue on the research agenda. Mandarin is therefore used as the language of this study's primary database. In the following section we describe the Mandarin Affective Speech (MAS) corpus (T. Wu et al., 2006), which is the major database used in all experiments in this study.

4.2 THE MANDARIN DATABASE

This section describes the database (corpus) that we use in our experiments. The corpus is a publicly available database.

Table 4.1 gives a brief overview of the corpus used in this study, which regard to the language, number of speakers, types of emotions, number of sentences, and number of repetitions. A Mandarin Affective Speech (MAS) corpus is a database of emotional speech that consists of audio recordings and corresponding transcripts collected at the Advance Computing and System Laboratory, College of Computer Science and Technology, Zhejiang University, China. It provides affective speech data that are essential for SER investigations.

Table 4.1: Brief overview of the Mandarin Affective Speech corpus

MAS features	Number
Number of speakers	68
Types of emotions	5
Number of sentences	20
Number of repetitions	3
Total number of utterances	20,400

There were more than 100 non-professional speakers who participated in the data collection. Wu et al. designed different scenarios to elicit expressions of various emotional states (e.g., a wonderful vacation for expressing happiness, a colleague's mistake for expressing anger) from the speakers. To avoid exaggerated expressions of emotions, an additional group of listeners reviewed the recordings during a second-round listening check. The records marked as overstated expressions of emotions were omitted from the corpus. The final database includes 68 speakers, of which 45 are male and 23 are female. Five types of emotional states were collected to build the corpus: *angry*, *happy*, *neutral*, *panic*, and *sad*. These five emotional states are used in all experiments in this study as the fundamental emotional states. Each speaker used the five

different emotions to characterize a total of 20 sentences. Each speaker read all of the sentences and repeated each sentence three times to establish the emotional state involved as adequate as possible. These sentences include all the phonemes and most common consonant clusters in Mandarin. The total number of recordings was equal to 20,400 (number of emotions $\times$ number of sentences $\times$ number of repetitions $\times$ number of speakers). Providing more information about the content of the Chinese sentences to the reader, Appendix A list all 20 sentences (both as Chinese source texts and as English translations) and relate them to the five emotions. All utterances were recorded at a sampling rate of 8 kHz at 16 bits. The corpus was obtained through the Linguistic Data Consortium [11].

4.3 SPECTROGRAMS

By the preparation of our research with respect to feature construction and feature learning, we attempt to construct a visualization of a speech signal. Such a procedure requires four steps. The first step is to answer the question of how to transfer speech into a visual image. The idea is to transfer a speech signal into a spectrogram as an image that can be analyzed using image-processing methods. An examination of an example spectrogram follows below. We start by giving a definition of a spectrogram (see Definition 4.1).

Definition 4.1 Spectrogram (adapted from Schroeder, 2013)
A spectrogram is a visual representation of the time-varying spectrum of frequencies of sound.

A spectrogram (see Definition 4.1) is the outcome of transferring sound signals into a two-dimensional visual representation. The spectrogram is obtained by applying the short-term Fourier transform to small (partially overlapping) temporal segments of the sound signal. The resulting spectra (frequency histograms) form the columns of the spectrogram, where each column represents the spectrum of a temporal sample. When treating a spectrogram as an image, each pixel value corresponds to an energy value associated with the time (column) and frequency (row) represented by the pixel location.

Several well-known speech-related phenomena can be displayed in a spectrogram (e.g., fundamental frequency, harmonicity, and formants). Many studies have used spectrogram-based approaches to extract perceptual cues. Past works in this area have included analyses of speech discrimination (Meyer & Kollmeier, 2011), environmental sound classification (Souli & Lachiri, 2011), automatic speech recognition (Gu, Postma, & Lin, 2015), and personality and likeability (Buisman & Postma, 2012).

11 http://catalog.ldc.upenn.edu/LDC2007S09/

Below we provide (A) an example of the spectrogram of an utterance together with selected part, and (B) the general composition of a spectrogram.

A: An example of a spectrogram of an utterance

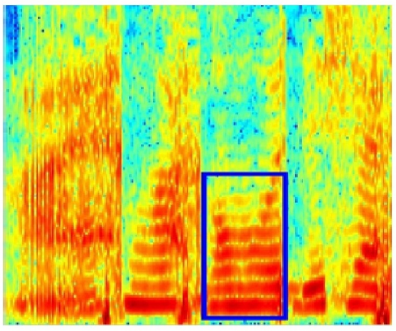


Figure 4.1: Example of a spectrogram of an utterance

An example of a spectrogram will shed light on the concept of treating a spectrogram as an image to be analyzed using image-processing methods. Figure 4.1 illustrates a speech spectrogram of the MAS database. The spectrogram depicts the utterance "他是个好人" in Chinese ("He is a good person" in English) expressed with a neutral emotion. The segment enclosed by the blue rectangle corresponds to the fragment "好人 (a good person)". Figure 4.2 zooms in on part of the spectrogram of the neutral utterance which shows a nearly horizontal orientation.

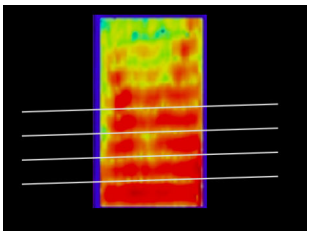


Figure 4.2: A selected part of the spectrogram, as indicated by the blue rectangle in figure 4.1. The four lines illustrate the near-horizontal orientations of the four energy bands

B: The general composition of a spectrogram

In brief, the general composition of a spectrogram is as follow. Spectrograms constitute a two-dimensional graph, with an extra third dimension which is represented by colors. The horizontal axis represents time and progresses from left (oldest) to right (most recent). The vertical axis signifies the frequency, with the lowest frequency value at the bottom and the highest frequency value at the top. The energy at a particular time and at a particular frequency is depicted by the intensity or color of a

pixel. In our case, colors are used that range from red to blue. Bright red indicates a stronger energy value while soft blue corresponds to a low energy value. For periodic vocal signals, the spectrogram contains parallel bands that correspond to the partials of the complex tone generated by the vocal chords. The blue rectangle in figure 4.1 provides an example of a periodic fragment with horizontal bands. The horizontal orientation of the energy bands reflects the constant frequency over the selected time period. Properly-tuned two-dimensional Gabor filters respond to the width (spatial frequency) and to the orientation of the bands in the spectrogram. For further reading we refer to Section 6.2.

4.4 LOG-GABOR FILTERS

We treat the spectrogram as an image by performing analyses of its local spectro-temporal structure. The analyses are carried out by using standard image processing, i.e., two-dimensional Gabor filters which are locally tuned to the orientations of energy bands in the spectrogram. Thus, log-Gabor filters are the second key technique that helps analyzing a speech spectrogram.

Definition 4.2 Gabor filter (adapted from Dennis Gabor, 1971)
A Gabor filter is a linear filter used for texture analysis. It essentially analyzes whether there is any specific frequency content in a localized region of an image in particular directions.

The original (one-dimensional) Gabor filter (see Definition 4.2) was proposed by Dennis Gabor (Gabor, 1946), and was intended to deal with the inherent uncertainty in determining the temporal localization and frequency. In its one dimensional iteration, the Gabor filter corresponds to a sine wave weighted by a Gaussian envelope which combines the localization (the mean of the Gaussian) with frequency determination (the frequency of the sinusoid). This approach has two advantages. First, the Gabor filter can be implemented by means of the Fast Fourier Transform (FFT) which is fast and efficient. Second, it is able to visualize the output of the filter detection. Figure 4.3 shows a Gabor filter bank composed of the combinations of six orientations (the columns) and five spatial frequencies (the rows) from high (top row) to low (bottom row).

Meanwhile, it was observed that the Gabor filter has a limitation, that is, it can have a non-zero DC value (which is the mean value of a waveform) for certain bandwidths. Therefore, we choose to use the log-Gabor filter instead of the standard Gabor filter. Thus, we can ensure a zero DC value by defining the Gabor filter on a logarithmic frequency scale. Following this line of thinking, we try to detect the periodic patterns of varied speech phenomena as orien-

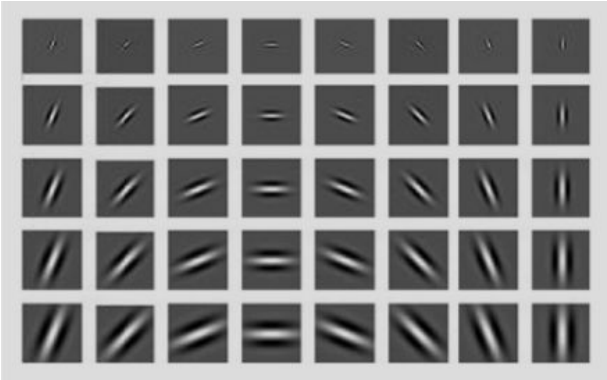


Figure 4.3: A Matlab-generated visualization of the Gabor filters

tations and (spatial) frequencies. Hence, we use log-Gabor filters as a tool for analyzing the periodic phenomena in the spectrogram.

4.5 CHAPTER SUMMARY

In this chapter, we remarked that European language corpora have attracted more attention in current studies than corpora in the Chinese language. Moreover, we also see that projects in terms of SER has been under-investigated with respect to the Chinese language. However, the number of Chinese users of new technologies is very large, which means that there is a considerable research market for Chinese researchers interested in speech emotion. For all these reasons, we will employ a Mandarin corpus in our experiments and discuss a variety of details in Chapters 5 to 8.

In Sections 4.3 and 4.4, two key components of this study's techniques were then explained in detail: spectrograms and log-Gabor filters. In Section 4.3 we introduced the spectrogram. In our experiments, we will detect and extract new features from the visualization of speech signals for the task of feature extraction (feature construction and feature learning). The key issue is that spectrograms transfer speech signals into "images". They are used in all experiments reported in this thesis.

Finally, in Section 4.4, log-Gabor filters will help analyzing the visualized speech signals. Gabor filters are flexible and can be tuned to the orientation of energy bands in a spectrogram. After analyzing the limitations of Gabor filters, we decided to use log-Gabor filters instead of standard Gabor filters for the experiments detailed in the Chapters 5 to 7.

5 THE VOICED SEGMENT SELECTION ALGORITHM

Our investigations are part of the research area Voiced Activity Detection (VAD). Currently, there are three main types of VAD methods, viz. the template feature method, the statistical method, and the deep-learning method. All these types have their advantages and disadvantages. They will be analyzed and discussed in this chapter. However, the main task of this chapter [12] is to address Research Question one (RQ1). We reiterate it below.

RQ1: Is it possible to design a new algorithm that improves the accuracy of detecting the voiced part activity in speech?

Here we would like to remark that it is not our aim to design a brand-new method (i.e., a fourth VAD method), but to design a new algorithm which may combine the strong points from three existing methods, augmented with some new ideas. Of course, the new ideas should be substantial and should aim, after translation in algorithmic form, at outperforming the existing algorithms. Here, we should discuss what 'outperform' means, since several measurements are possible. At this point we return to RQ1.

To properly address the difficulties involved in answering RQ1, we split RQ1 into Research Question 1A (RQ 1A) and Research Question 1B (RQ 1B).

RQ 1A: What characteristics should a new algorithm possess for being more accurately in detecting voiced part activity in speech?

RQ 1B: What is the gain in SER performance provided by the new voice detection algorithm as compared to the original algorithms?

To answer RQ 1A, we develop a novel voiced segment selection (VSS) algorithm (see Section 5.1 and Section 5.2). The performance of that VSS algorithm is compared with three other leading algorithms (one algorithm from each type). To answer RQ 1B, the SER performance for the acoustic features extracted from all available speech is compared with the performance of the VSS-filtered speech signals.

Speech always consists of voiced and unvoiced parts (Burnett, 2007). When the vocal cords vibrate to pronounce vowels, verbal segments are generated. In contrast to unvoiced segments, voiced segments show prosodic signals. Unvoiced segments show irregular signals, generated by the influence of a narrow vocal tract. Emotional information in speech is represented by a variety of

12 This chapter is based on work by Gu, Yu, Eric O. Postma, Hai-Xiang Lin, and H. Jaap van den Herik. Speech Emotion Recognition Using Voiced Segment Selection Algorithm. In Proceedings of European Conference on Artificial Intelligence, ECAI, 2016.

acoustic characteristics and is mainly contained in the voiced parts (Chaudhari & Kagalkar, 2014). Therefore, researchers have focused on the voiced aspects of speech in emotion recognition. Hence, separating the voiced and unvoiced aspects of speech is important for SER.

Different types of emotion are to be found in the voice of the speaker, and each type can be described by the acoustic features of the selected segments of the speech, such as intensity, pitch of the voice, and other spectral features. Many previous studies have revealed the relationship between acoustic features and types of emotion (cf. Goudbeek & Scherer, 2010). We present two different examples. First, arousal changes in emotion could lead to (1) a shorter duration of breath between speech utterances and (2) a faster speech rate. Second, continuing the first example (by speculation), a speaker who is experiencing fear or stress and may tend to use a fast and loud voice (cf. Jürgens, Hammerschmidt, & Fischer, 2011). After these two examples it should be clear that the VAD research area is a challenging area to which we would like to contribute.

The structure of this chapter is as follows. Section 5.1 reviews mainstream voice activity detection (VAD) methods. The conceptualizing of the VSS algorithm is described in Section 5.2. In Section 5.3, the set-up of the first VSS algorithm experiment and its results are outlined, and RQ 1A is answered. Then, Section 5.4 provides the results from the experiment using the VSS algorithm for SER performance, answering RQ 1B. A discussion of the findings is found in Section 5.5. Finally, Section 5.6 summarizes the answers to RQ 1A and RQ 1B, and respond to RQ1.

5.1 VOICED ACTIVITY DETECTION: LITERATURE REVIEW

Voiced activity detection (see Definition 5.1) is a research area that refers to the capability to identify the voiced regions of a given speech signal (cf. Sohn, Kim, & Sung, 1999). A typical VAD algorithm is comprised two core components: (1) feature extraction and (2) detection of voiced or unvoiced segments. During the last decade, researchers attempting to achieve a more accurate performance have proposed a remarkable number of VAD algorithms (cf. Germain et al., 2013; Zhang & Wang, 2014). A variety of features that could contain properties of voiced signals have been exploited, and segment detection has also been improved (see, e.g., Eyben, Weninger, Squartini, & Schuller, 2013).

Definition 5.1 Voiced Activity Detection (adapted from Sohn, Kim, and Sung, 1999)
Voiced activity detection is a research area used in speech processing to detect the presence or absence of human speech.

In previous studies, researchers have relied on three main VAD methods to detect voiced segments, namely (A) the template feature method, (B) the statistical method, and (C) the deep-learning method. In this section, we review these mainstream VAD methods.

A: The template feature method

The template feature method was widely used in the earliest research period in this area to distinguish voiced and unvoiced parts (and aspects) of speech. The researchers used acoustic parameters on which they based their calculations. In particular, these algorithms detected the voiced segments based on a comparison of the results from acoustic parameters used and a threshold value. The threshold value is determined by the acoustic features and is usually set according to the researcher's experience. A typical acoustic feature algorithm normally works on time-domain or spectral-domain features. In early studies, the prevalent acoustic features were the short-term energy (Benyassine, Shlomot, Su, & Yuen, 1997), the zero-crossing rate (ZCR) (Bachu, Kopparthi, Adapa, & Barkana, 2008), and the spectrum (Davis, Nordholm, & Togneri, 2006). The performances ranged from 60% to 80% for the correct detection of voiced activities. After that, more robust acoustic features, such as autocorrelation (Sadjadi & Hansen, 2013), pitch (Sharma & Rajpoot, 2013) and formant (Yoo, Lim, & Yook, 2015) were exploited.

The advantage of the template feature algorithm is that it is straightforward and easy to apply. However, difficulties arise when dealing with variable conditions. Two cases explain this point: (1) Different people have different voice characteristics, and (2) background noises vary in different situations (S.-H. Chen, Chang, & Truong, 2007), because the threshold is a preset fixed value with limited flexibility. In practice, the Fourier transform was used in most of the above-mentioned studies for acoustic feature extraction. However, in these cases, accuracy can degrade rapidly when the signal is a non-stationary one (Joseph & Babu, 2016).

B: The statistical method

The statistical method was proposed by researchers using statistical models such as the likelihood ratio test to distinguish voice activity parts (see, e.g., Suh & Kim, 2012). For instance, the statistical likelihood ratio (LR) has been employed as the main feature for the support vector machine (SVM) algorithm to be used as classifier (cf. Yu & Hansen, 2010). Their argument was that the spectral component of speech follows a complex Gaussian distribution. Hence, the advantage of this algorithm is that it can reach a high detection accuracy with a powerful mathematical framework. Nevertheless, there are at least three severe limitations associated with the statistical likelihood algorithm: (1) there is a large variety of spectral patterns in the input speech signal, (2) the LR approach only works well if it is assumed that the noise types are stationary, and

(3) prior knowledge of background noises is needed for this algorithm (Bach, Kollmeier, & Anemuller, 2010).

C: The deep belief network method

Hierarchical neural-network methods were already investigated by many researchers more than thirty years ago (Dechter, 1986; Fukushima, 1980). However, the modern neural-network methods used for VAD has only recently been designed and implemented. For example, Zhang and Wu proposed as applicable algorithm the deep-belief network algorithm (cf. Zhang & Wu, 2013). Compared to more traditional methods, this approach is strongly competitive due to its automatic detection and extraction of the features of voiced and unvoiced parts. It integrates them naturally into the VAD classification. The deep-belief network (DBN) can be considered a predecessor of the deep learning method (that we will study in Chapter 8). The DBN yields a better performance in terms of its detection of verbal activity. However, its weakness is that the computational time demand required for data training is high, and this is particularly true for very large databases. Additionally, the various parameter settings of deep-learning algorithms are difficult to optimize during initiation.

5.2 CONCEPTUALIZING THE VSS ALGORITHM

In this section, we propose a novel algorithm called the VSS algorithm. It is different from the methods discussed in Section 5.1 in that it is a combination of the statistical method and the deep-learning method. One of our ideas at this point in development is to handle VAD by extracting novel features from speech signals (Gu, Postma, Lin, & Van den Herik, 2016a).

To visualize and analyze a the speech signal, a researcher requires two key tools: (1) a visualization tool for signals (spectrogram) and (2) an image analysis tool for spectrogram outputs (log-Gabor filter). These instruments have been introduced in Sections 4.3 and 4.4, respectively.

In prior work, Ezzat, Bouvrie, and Poggio (2007) proposed an approach that combines a Gabor filter and spectrogram in response to speech phenomena such as formants, vertical edges, and background noise. They showed that these speech phenomena can be observed as different types of spectro-temporal modulations on a spectrogram and found that the two-dimensional Gabor filter can analyze and recognize these modulations.

We thoroughly investigated Ezzat et al.'s spectro-temporal modulation and found it useful for this research. In contrast to the above-described template feature and statistical algorithms for VAD (see Section 5.1) and based on Ezzat et al.'s findings, we propose our VSS algorithm as a new approach to detecting voice activity. As speech phenomena are always generated from a human voice, phenomena such as those mentioned by Ezzat et al. can be considered

to be activities in the voiced segments of speech. In this vein, we continue our line of thinking. If we could efficiently detect and analyze these phenomena, the corresponding voiced aspects of speech could then be recognized and segmented. Voiced segment selection is thus first and foremost intended to trace voice parts through the speech phenomena mentioned above. Therefore, we expect to use log-Gabor filters to extract the novel features from a spectrogram via a new feature construction algorithm that intends to improve the VAD performance. This algorithm may also provide a novel way of approaching other VAD topics.

5.3 EXPERIMENT ONE: THE VSS ALGORITHM

This section addresses RQ 1A. It details the experiment employing the VSS algorithms. The set-up of the experiment is outlined in Subsection 5.3.1. Thereafter, the implementation of the experiment is described in Subsection 5.3.2. Finally, in Subsection 5.3.3 the experiment results are discussed, and RQ 1A is answered.

5.3.1 Set-up of the VSS Experiment

The VSS experiment set-up consists of three steps: (A) spectrogram generation, (B) feature construction using log-Gabor filters and spectrogram , and (C) voiced and unvoiced segment classification. The details of each of these steps are outlined below.

A: Spectrogram generation

First, we need to transfer the speech fragments to spectrograms in order to visualize them. Two relevant questions then are: (1) how long are the fragments? and (2) is each fragment translated into one spectrogram? The length of each temporal segment is 0.04 seconds. The duration of an utterance in the MAS dataset is 1.6 second. Therefore each utterance consists of 40 fragments. The VSS algorithm has a short-time Fourier transform with an overlap of half a length. All the 40 fragments in each utterance will be fully transformed into a spectrogram using Matlab's spectral analysis function[13]. The values of two parameters: the length of fragment, time Δt are given in Table 5.1.

B: Feature construction

Each spectrogram is convolved by a log-Gabor filter with six parameters of scale and orientation. Their values are given in Table 5.2. The parameters of the log-Gabor filters are applied to achieve the convolution of

13 https://au.mathworks.com/help/signal/ref/spectrogram.html

Table 5.1: The parameter values for the spectrogram in the VSS algorithm

Name of parameter	Value of parameter
Length of fragment	0.04 s
Time Δt	20 ms

Table 5.2: The parameter values for log-Gabor filters in the VSS algorithm

Name of parameter	Value of parameter
N_s	12
N_o	6
MinWavelength	3
Mult	1.35
SigmaOnf	0.8

the spectrogram. N_s denotes the number of wavelet scales (i.e., spatial frequencies). The scale represents the width of the texture pattern. N_o denotes the number of orientations. They can be vertical, horizontal, or oblique. The scales ($N_s = 12$) are designed to match the different widths of the patterns that could arise. The orientations ($N_o = 6$) in equal steps cover 0°, from 0° to 180° to cover all orientations. *MinWavelength* is the wavelength of the smallest scale filter. In the spectrogram, its value is specified in pixels, here we have set $Minwavelength = 3$. *Mult* is the scaling factor between successive filters. *SigmaOnf* is the ratio of the standard deviation of the Gaussian describing the log-Gabor filter's transfer function in the frequency domain to the filter center frequency. We use the values of $Mult = 1.35$ and $SigmaOnf = 0.8$, respectively, to ensure that the change in filter is not too drastic. We focus on whether a local pattern appears and is detected. By convolving the spectrogram with a log-Gabor filter of a given spatial frequency and orientation, we obtain a convolved spectrogram representing spectro-temporal patterns with the associated spatial frequency (width) and orientation, respectively. Therefore, if the number of scales and orientations is equal to N_s and N_0, respectively, then, the $N_s \times N_0$ filter images are generated. Each combination of scales and orientations results in one filter image. We use Kovesi's log-Gabor functions[14] to perform the convolutions on each spectrogram.

C: Classification

From Step (B), we know that each combination of one scale and one orientation yields a single convolution image for each spectrogram. Then

14 http://www.csse.uwa.edu.au/ pk/research/matlabfns/

Table 5.3: Specification of the the two SVM parameters c and g that are optimised using grid search. The first column lists the parameters, the second columns shows their definition in terms of the SVM cost parameter C and kernel parameter γ. The last column specifies the range and step size (middle number) examined in the grid search.

Parameter	Definition	Range of values examined (minimum:stepsize:maximum)
c	$\log_2 C$	-2:0.1:10
g	$\log_2 \gamma$	-8:0.1:3

the N_s by N_o matrices of energy value features could be obtained. In order to maintain the time sequence, we do not use any feature selection or dimension reduction here. Thereafter, the convolution values within each fragment are averaged yielding a total number of Gabor energy values equal to the number of orientations multiplied by the number of scales. Then, the values of each fragment could be stored as a vector.

The vector-representations of each fragment form the inputs for the SVM classifier. The Waikato Environment for Knowledge Analysis (WEKA)[15] implementation for SVM is used to classify the voiced and unvoiced segments from the spectrogram[16]. Using a Radial Basis Function (RBF) kernel, two parameters define the SVM: the regularisation parameter C and the RBF kernel parameter γ . As suggested by Chang and Lin (2011), we express both parameters in logarithmic (base 2) units, i.e., $c = \log_2 C$ and $g = \log_2 \gamma$. Table 5.3 specifies the ranges examined for c and g.

5.3.2 Evaluation Procedure

In this study, a performance evaluation takes the form of a comparison of the accuracy of the voiced part detection between the VSS algorithm and the other three prevalent VAD algorithms. Three major algorithms (see Section 5.1) are duplicated: (1) the short-term energy and ZCR algorithm, (2) the statistical LR algorithms (Suh & Kim, 2012), and (3) the deep-belief network algorithms (Zhang & Wu, 2013).

- The short term energy (henceforth abbreviated to Energy) and ZCR algorithm is based on two acoustic feature values, the Energy and the ZCR. The comparison relies on two threshold sets, one for the Energy and one

15 http://www.cs.waikato.ac.nz/ml/weka/

16 The WEKA is a suite of machine learning software, developed at the University of Waikato. It has well-functioning algorithms for data analysis and machine learning modeling, alongside graphical user interfaces for easy employment of these functions.

for the ZCR value to assess whether the value of the parameter setting of the algorithm exceeds the threshold.

- With respect to the statistical likelihood ratio (LR) algorithm. Suh and Kim used techniques from the multiple acoustic model observation to optimize the weights of the likelihood ratios which could achieve a higher accuracy of voiced detection. Based on this experiment setting, the window length is set to 8 (Suh & Kim, 2012).
- The deep belief neural network (DBN) algorithm was originally proposed for voiced detection (cf. Zhang & Wu, 2013). DBN had three critical parameters which are (1) the number of the restricted Boltzmann layers layers, (2) the number of units in each layer, and (3) the learning rate. The number of the restricted Boltzmann layers was set to 3. The numbers of units used in each layer were 52, 7, 7 units, respectively.

Making a generalization about how these algorithms are generally evaluated, there are two standards which we use. The first evaluation standard is the classification performance (see Definition 5.2), and the second standard is the standard deviation.

Definition 5.2 Classification Performance (adapted from Schroeder, 2013) Classification performance is defined as the accuracy of the machine learning method in detecting objects.

In all the experiments mentioned above, all parameters are set strictly according to the values in the respective papers referred to by the authors. The evaluation of the VSS algorithms is performed on the MAS corpus (MAS, 2007). The details of the corpus were given in Section 4.2.

5.3.3 Results of the VSS Experiment

We used SVM with a RBF kernel for classification. To achieve the highest performance, the grid search is used to obtain the best combination of SVM parameters c and g using 10 fold cross validation. After extracting the features from dataset, we start the grid search with the sequence of c and g. Then, the parameters c and g associated with the highest cross-validation performance are provided. Figure 5.1 illustrates a contour plot of visualizing the results of the grid-search parameter optimization. As showed in Figure 5.1, the x-axis indicates c while the y-axis indicates the g. We can observe that the performance varies with different values of parameters.

The best performance, a classification accuracy of 91.7%, is obtained for the values $c = 6$ and $g = -3$ ($C = 64$ and $\gamma = 0.125$). The sensitivity of the performance to these parameter values is revealed in Figure 5.1, which

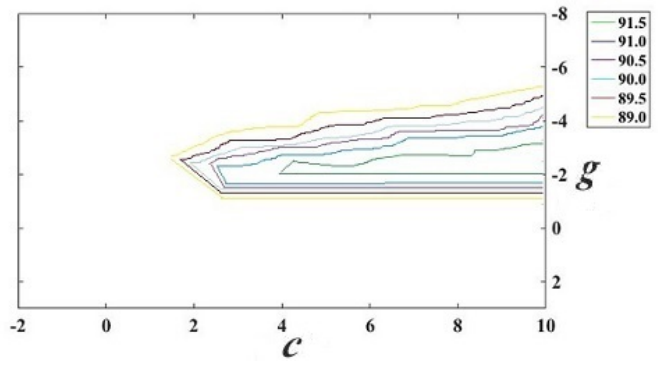


Figure 5.1: Contour plot illustrating the grid-search results for optimizing the SVM parameters c and g for the VSS algorithm.

shows a contour plot of the performances obtined for the examined range of parameter values. The region demarcated by the green contour represents the parameter values for which the best performance is obtained. As can be seen, a considerable range of parameter values leads to the best performance.

Table 5.4 shows the VAD accuracy results obtained through the experiment using the VSS algorithm. It is presented in comparison with the results through our duplication of three other major algorithms which include Energy and ZCR, Statistical LR, and DBN, in order to evaluate the performance of VSS. In the performance comparison table, the left column lists the algorithms and the right shows the corresponding VAD accuracy results. The following paragraphs discuss the performances of all four algorithms.

Table 5.4: A performance comparison between the VSS algorithm and three additional algorithms

Method	Classification Performance (SD%)
Energy + ZCR	76.8% (3.6%)
Statistical LR	85.0% (2.4%)
DBN	93.2% (1.1%)
VSS	91.7% (1.4%)

As we can see from Table 5.4, the VSS algorithm obtains a higher accuracy (91.7%) than the traditional Energy and ZCR and Statistical LR algorithms. Possible reasons are as follows. The first reason is that the VSS is efficient at using log-Gabor filters to extract features from a spectrogram. These features are quite different from acoustic features used by Energy and ZCR or statistical features used by Statistical LR. The VSS can treat the speech signal as an image and extract all features at once. The second reason is that log-Gabor filters have a strong ability to distinguish phonetic phenomena from background noise in a spectrogram. Even though the noise in speech can bring a large number of

corresponding peaks on the spectrogram, those peaks that are relevant for the formants and vertical edge phenomena, are still detectable. This is because the output of log-Gabor filters is achieved through integrating all of the two-dimensional spectrogram information, which makes the two-dimensional filter bank more robust to the noise than the one-dimensional spectrogram pattern counterparts.

Table 5.4 also reveals that the Energy and ZCR algorithm has the lowest performance rate, at 76.8%. This is similar to the result reported in the literature (see Section 5.2). Three primary factors are responsible for its poor result: (1) the Energy and ZCR algorithm has manually set threshold values, which are not always optimized, (2) the Energy and ZCR typically need to be calculated in a quiet environment and are sensitive to noise or lower energy levels, and (3) the specific thresholds have to be chosen using this algorithm, which reduces the flexibility of threshold selection.

As illustrated in Figure 5.2, the DBN algorithm outperforms the VSS algorithm. An explanation is that the DBN algorithm has a strong ability to learn features from speech signals, and each feature can reflect the voiced parts of speech. Therefore, DBN can effectively distinguish the voiced parts of speech from the unvoiced parts of speech. However, DBN does not achieve an overwhelmingly higher classification performance than VSS, and the DBN requires extensive computational resource consumption. Therefore, considering the balance between (1) accuracy of voice detection and (2) real-time efficiency, we still may conclude that the VSS algorithm is a (quite) useful and practical algorithm for VAD.

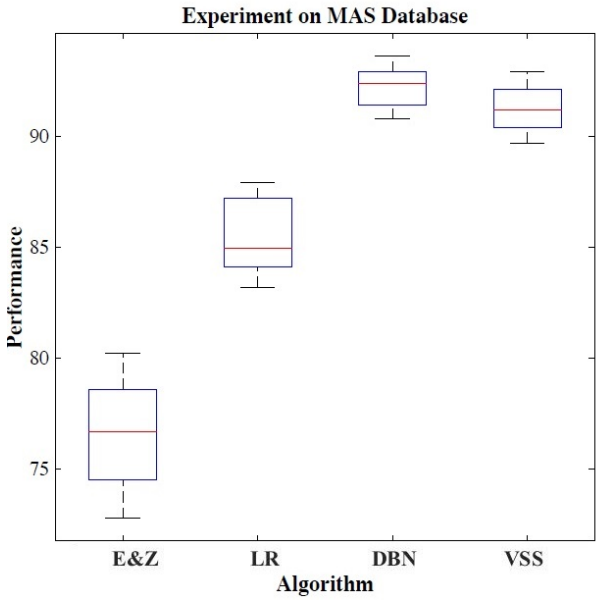


Figure 5.2: Comparison of the voiced part accuracy performances obtained on the MAS database.

Lastly, we provide a summarized answer to RQ 1A. According to the first experiment, we see a comparison of the performance between VSS and other algorithms. The VSS outperformed two other prevalent algorithms (Statistical LR and Energy + ZRC). Taking accuracy and real-time efficiency into consideration we may state that the results for VSS are good. So the answer for RQ 1A is that VSS is still a candidate in the list of our contributions (see Section 1.6).

5.4 EXPERIMENT TWO: SER USING THE VSS ALGORITHM

In this section, we present the experimental details of SER using the VSS algorithm. The set-up of Experiment two is given in Subsection 5.4.1. Then, Subsection 5.4.2 discusses the procedure and criteria of the evaluation. Finally, the results of the experiment are reported and analyzed in Subsection 5.4.3.

5.4.1 Set-up of Experiment Two

In Experiment two, the VSS algorithm accomplishes SER; in other words, the acoustic features are extracted from the voiced segments based on the VSS algorithm. For each voiced segment, acoustic features are extracted. Recently (2015), Interspeech, which is nowadays one of the major SER conferences, proposed a baseline acoustic feature set. It is nowadays the most well-recognized acoustic feature set and we use it as the state-of-the-art option for our experiment baseline (Schuller et al., 2015). There are seven types of features are used for acoustic feature set, which includes: Linear Predictive Cepstral Coefficients (LPCC), Mel-Frequency Cepstral Coefficients (MFCC), zero-crossing rate, speech rate, pitch, formants $(1-3)$, and magnitude. Beside the original acoustic features, four statistical descriptors are derived: mean, maximum, minimum, and standard deviation. The total recording times of MAS is 20,400.

Each fragment is represented by 198 features yielding a feature matrix of $198 \times 20,400$ elements. To reduce the dimensionality of this matrix, principal component analysis (PCA) is applied (see Section 3.2). Using the PCA script in the Matlab dimensionality reduction toolbox (Van der Maaten, Postma, & Van den Herik, 2009a), we reduce the number of features to the following numbers: either $10, 20, 30, 40, 50, 60, 70, 80$ or 90 components. For the classification, the SVM is used. For MAS, it maps the reduced set of features into the five emotional states. The SVM parameters c and g are optimized in the same manner as in Experiment One (see Table 5.3).

5.4.2 Evaluation Procedure

This subsection describes the procedure of evaluating SER performance using the VSS algorithm. Our evaluation compares (a) the performance of SER with

VSS pre-processing versus (b) the performance of SER without using the VSS algorithm.

The main evaluation is performed on the MAS corpus. The evaluation is carried out on each set of features using cross-validation procedures. Cross-validation combines (averaged) measures of fit (prediction error) to derive a more accurate estimate of the model prediction performance. To avoid overfitting due to the PCA and the SVM parameter optimization, the evaluation is performed using a nested 10×10 fold cross-validation, in which the optimization of the SVM parameters and the number of PCs was performed in the outer each-fold leave-out cycle and the evaluation in the inner cycle. All evaluations (cross-validation experiments) are repeated 10 times and the results are averaged.

5.4.3 Results of the SER Experiment

The SER classification is applied to the MAS corpus. Figure 5.3 visualizes the classification performances with and without VSS before feature extraction. Both sets of features are based on the state-of-the-art baseline. The comparative evaluation involved (1) employing the VSS algorithm and (2) not employing the VSS algorithm before the feature extraction.

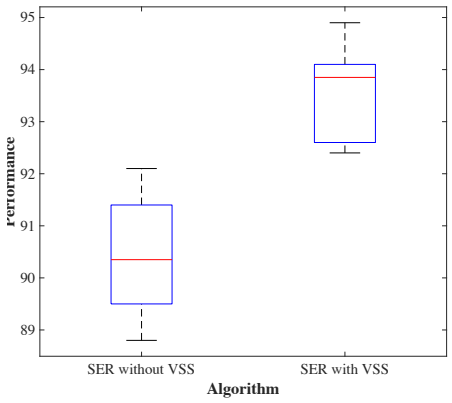


Figure 5.3: Comparison of SER performances obtained on the MAS database.

The SER after applying the VSS algorithm to the MAS corpus yields a 3.6% improvement in SER accuracy compared to those that do not involving the VSS algorithm. The experiment results demonstrate that using the VSS algorithm leads to a non-overlapping (in the statistical sense) improvement in SER performance. The results reveal that the VSS algorithm is effective at improving feature extraction in SER. The reason could be that most of the acoustic features are related to the voiced parts of speech, and more precise voiced parts could result in more precise acoustic feature extraction.

In the experiment, we performed PCA optimization in the outer loop of the cross-validation procedure. Figure 5.4 illustrates the optimal number of

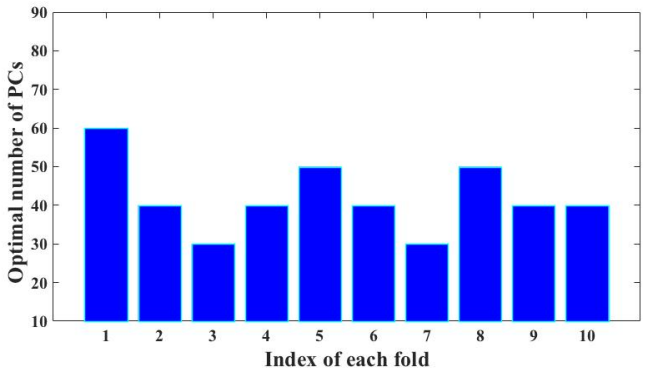


Figure 5.4: The optimal number of PCs for each fold in the cross validation.

principal component which we could obtain in each fold during the cross validation. As we can observe from the Figure 5.4, the x-axis indicates the index of each fold and the y-axis indicates the optimal number of principal components. According to the figure, we get a similar number of optimal PCs around 40 for each fold.

Table 5.5: Optimal numbers of Principal components for SER with and without VSS

Classification	Numbers of PCs
SER without VSS	50
SER with VSS	40

Table 5.5 lists the final optimal numbers of principal components for achieving the best SER performance with and without VSS in the cross-validation. As the table reveals, the highest performance of 93.7% for SER achieved with VSS is obtained while there are 40 principal components. For SER achieved without VSS, the best performance is obtained when 50 principal components are retained.

After examining the overall performance, we can gather insights into the individual emotional states. Table 5.6 and Table 5.7 show the confusion matrix of classification performance achieved with the optimal number of principal components (list in Table 5.5). Table 5.6 provides the confusion matrix of the classification performance for the five emotions without the VSS. As we can see from the table, the left-hand column indicates the names of the emotional state. From the top downward, they are *angry*, *happy*, *neutral*, *panic*, and *sad*, respectively. The diagonal demonstrates the best performance rate achieved for each emotional state. The remaining cells signify the error rate (i.e., recognizing the wrong emotion). As we can see, the classification performance ranges from 90.7% to 92.6%. *Angry* emotion is associated with the weakest performance at

90.7%, and *neutral* obtains the best performance at 92.6%. The values for *happy*, *panic*, and *sad* performance rates are 91.1%, 92.2% and 91.9%, respectively.

Table 5.6: Confusion matrix of the classification performance without VSS

	Performance (%)				
Emotion	Angry	Happy	Neutral	Panic	Sad
Angry	90.7%	3.8%	2.4%	2.0%	1.1%
Happy	4.2%	91.1%	1.5%	2.3%	0.9%
Neutral	2.3%	1.9%	92.6%	1.5%	1.7%
Panic	2.6%	1.9%	0.8%	92.2%	2.5%
Sad	1.5%	1.4%	3.2%	1.9%	91.9%

Table 5.7 contains the confusion matrix for the five emotions associated with the best-performing SVM classifier during VSS processing. *Neutral* and *sad* are most easily recognized, obtaining scores of 96.5% and 94.8%, respectively. Comparing (a) the classification without VSS (see Table 5.6), and (b) VSS is employed (see Table 5.7), the *neutral* and *sad* emotions achieve 3.9% and 2.9% increases. One possible explanation for this is that *neutral* and *sad* expressions are soft and smooth in speech. The pauses between words are more explicit than for the remaining emotions; therefore, the patterns that appear in the spectrogram are more distinguishable. In contrast, *panic* is the most difficult to recognize. When the VSS algorithm is applied, it has a 90.1% performance rate, weaker than the previous 92.2% performance. This declining trend distinguishes panic from all the other emotions. A possible explanation is that *panic* triggers a fast speaking rate and a lengthy speech duration. Hence, its pattern on the spectrogram is more complicated and less straightforward than the pattern seen for the other emotions.

Table 5.7: Confusion matrix of classification performance with VSS

	Performance(%)				
Emotion	Angry	Happy	Neutral	Panic	Sad
Angry	92.5%	2.7%	2.1%	2.4%	0.1%
Happy	3.0%	93.3%	1.0%	1.8%	0.7%
Neutral	1.5%	1.1%	96.5%	0.2%	0.4%
Panic	4.1%	2.3%	0.5%	91.8%	1.1%
Sad	0.4%	0.9%	2.7%	1.2%	94.8%

Finally, we formulate the answer to RQ 1B. Based on the results of Experiment two, the performance of SER is generally higher when the VSS algorithm

is applied before feature extraction, compared to the performance without VSS. The results indicate that using VSS for SER is a good answer for RQ 1B and a useful complement to current SER methods.

5.5 CHAPTER DISCUSSION

In this chapter, we proposed a new algorithm to more accurately detect voiced parts activity in speech. Moreover, we provided relevant details regarding the VSS algorithm. To investigate the effectiveness of the VSS algorithm, we evaluated our algorithm via two comparisons. First, the classification performance of VSS was compared with the Energy and ZCR algorithm, the statistical LR algorithm, and the DBN algorithm. Second, the SER performance was compared via involving VSS and not involving VSS.

So, the remaining question is how to explain why the VSS achieved better performance than the traditional algorithms. Based on the results, we may conclude that the VSS algorithm has five advantages. Below we discuss them one by one.

(1) It is important to note that the acoustic features extracted in previous studies and those extracted in our experiments are significantly different. Normally, previous VAD studies have typically extracted features entirely from a single dimension, either the frequency domain or the time domain. In contrast, our algorithm extracts features from a spectrogram that is a combination of time and frequency.

(2) The performances of the Energy and ZCR algorithm and the statistical LR algorithm have a higher standard deviation. It is possible that the background noise in speech has a significant impact on the performance of the two above-mentioned algorithms, because the noise can substantially influence the value of the acoustic features extracted from speech signals. However, it is likely that noise has less influence on spectrograms.

(3) The log-Gabor filter is a powerful tool for extracting features from a spectrogram. It allowed us flexibly to design the width and orientation of each filter. As demonstrated in Table 5.2, we can efficiently detect all the textural information of a spectrogram using only log-Gabor filters. It is difficult to miss any textural appearance of a spectrogram that is characterised by elongated oriented energy profiles.

(4) The language difference may also influence the performance. As described in Chapter 4, the major datasets were collected by human participants. Thus, participants' styles of expressing emotion in speech relies on their previous cultural and language habits. The Mandarin language is a tonal

language in which each word is a syllable, and so words are more separable in pronunciation than is the case in Western languages. Correspondingly, it may happy that, reflected on a spectrogram, each formant energy band may be more clearly displayed for Chinese language speech.

(5) Although the DBN algorithm obtained the best performance in the comparison, the VSS algorithm possesses higher computational efficiency. (5a) All in all, the results demonstrated that the VSS is an efficient algorithm and provides highly accurate voiced-part feature extraction. (5b) In summary, we may conclude that in experiment two, the results of our evaluation revealed that SER is more accurate when using VSS than when not using VSS.

5.6 ANSWER TO RESEARCH QUESTION ONE

In this chapter, we addressed the research question RQ1. Moreover, RQ1 has been partitioned into two sub-questions, RQ 1A and RQ 1B. We discuss them below.

RQ 1A: *What characteristics should a new algorithm possess for being more accurately in detecting a voiced part activity in speech?*

To answer RQ 1A, we proposed a novel VSS algorithm that uses a log-Gabor filter to extract the features from a spectrogram for subsequent voiced segment classification. To the best of our knowledge, our study is the first to analyze the relation between a spectrogram and VAD. The results of Experiment one show that the performance of VAD was improved using our VSS algorithm compared to the existing prevalent algorithms. Although the best performance of VAD in existing studies was achieved through a deep-learning algorithm, our algorithm had a much lower computational cost in achieving results close to the best performance. Here we may conclude that our results show that the VSS algorithm can be a useful and practical method to detect voice activity.

RQ 1B: *What is the gain in SER performance provided by the new voice detection algorithm as compared to the original algorithms?*

Since emotion expressions in speech are mainly contained in the voiced segments, it means that acoustic features are highly dependent on the voiced segments. Specifically, the results of Experiment two using SER show that when the VSS algorithm is used, the classification rate is higher in the Chinese corpus than when VSS is not used before feature extraction. The results convinced us that using VSS for SER is a good answer to RQ 1B and a useful complement to the current SER methods. In summary, our results indicate

that in the future, it is worth considering incorporating a VAD algorithm in SER software.

Taking the answer to RQ 1A and RQ 1B together, we are now able to provide the answer to RQ1.

RQ1: *Is it possible to design a new algorithm that improves the accuracy of detecting the voiced part activity in speech?*

Based on the answers to RQ 1A and RQ 1B, we may conclude that for RQ1 the following answer is adequate. The VSS algorithm is designed for accomplishing VAD via spectrograms. It yields an improved SER performance in terms of acoustic feature extraction. The results of experiment one and experiment two and our design of the VSS algorithm using spectograms and log-Gabor filters allow us to conclude that it is possible to design a new algorithm that is able to perform a more accurate detecting voice activity in speech.

In the following, our recommendation for VAD is that it can and should be further improved. Moreover, we must recognize and acknowledge that deep learning obtains the best performance in VAD. Deep learning has become the most popular and powerful method for machine learning and pattern recognition. For clarity and a proper understanding, we have anticipated on our results to be reported in Chapter 8. Deep learning has a strong ability in feature learning, and thus it would be the best choice for future studies, provided that (1) the cost for the relatively large computational time is reduced and (2) the large initiation cost for tuning the features is also reduced.

6 THE BASIS OF PRIMARY FEATURE

In this chapter[17], we address research question two (RQ2). We attempt to discover objective two-dimensional features on a spectrogram that would make classifying emotional states a simpler task.

RQ2: *How can we use two-dimensional features to analyze the spectrogram representation of speech?*

The remainder of this chapter is structured as follows. Section 6.1 outlines the inspiration for RQ2. The two-dimensional Gabor filter analysis of emotional speech and the different approaches used in our empirical study are presented in Section 6.2. In Section 6.3, the experiment set-up and results are detailed. In Section 6.4, the chapter discussion is presented. Finally, RQ2 is answered in Section 6.5.

6.1 INSPIRATION: PRIMARY FEATURE RESEARCH

Speech emotion recognition plays a significant role in human-machine interaction, which is becoming increasingly important given the ubiquity of computational devices in daily life (cf. Ringeval et al., 2015). Apart from verbal signals, human speech also contains non-verbal cues that provide (amongst other things) information on the affective state of the speaker, such as intensity, pitch and other spectral features. SER allows algorithms to detect non-verbal cues from users to obtain knowledge about, for instance, their level of frustration or stress, or other emotional affectation.

Current SER systems rely on machine learning. In these systems, features that are assumed to be relevant to the classification task under investigation are extracted from the speech signal. Classifiers are trained on the feature representations of the speech signal to estimate the appropriate classes for recognition. As illustrated in Section 3.1, this kind of construction of the appropriate features, so-called *feature construction*, is crucial to the recognition performance. Traditional speech processing and recognition methods often rely on temporal and spectral features that have proven to be relevant for speech-related tasks (cf. Chen, Mao, Xue, & Cheng, 2012). Well-known examples of such established features are linear predictive cepstral coefficients (LPCC) and mel frequency

17 This chapter is based on work by Y. Gu, E.O. Postma and H.X Lin. Vocal emotion recognition with log-Gabor filters. In Proceedings of the 5th International Workshop on Audio/Visual Emotion Challenge. ACM. 2015.

cepstral coefficients (MFCC), which consist of a wide variety of measurements of a speech signal. In contrast to the traditional methods, we attempted to use domain knowledge of the new features extracted from a spectrogram for feature construction.

The research in this chapter continues the work by Ezzat, Bouvrie, and Poggio by performing a spectro-temporal analysis of affective vocalizations through decomposing the associated spectrogram with two-dimensional Gabor filters (see Ezzat, Bouvrie, & Poggio, 2007). Based on previous studies on the emotional expression of voices and the resulting spectrogram display, we assume that each emotional state has a unique spectro-temporal signature in terms of the orientated energy bands detectable via properly tuned log-Gabor filters. We compare the SER performance of tuned log-Gabor filters with that achieved using standard acoustic features. From the experiments, we expect the results to demonstrate that using pairs of log-Gabor filters to extract features from a spectrogram yields a performance that at least matches the performance achieved via an approach based on traditional acoustic features. If true, their combined performance in terms of emotion recognition would be expected to outperform state-of-the-art SER algorithms. A successful combination of log-Gabor filters and traditional acoustic features would indicate that tuned log-Gabor filters definitively support the automatic recognition of emotions in Mandarin speech. The outcome would imply that log-Gabor filters may have advantages as regards to other speech-related tasks.

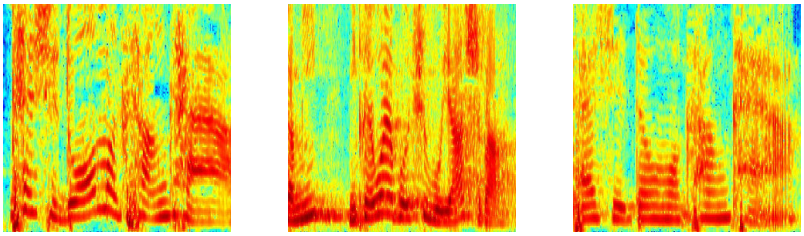
6.2 DETECTING PRIMARY FEATURES

A spectrogram is a visual representation of the spectrum of frequency for a sound signal and is thus a primary tool employed in this research (see Section 4.3). A spectrogram can represent various types of important information about speech signals (cf. Yin, Hohmann, & Nadeu, 2011). On a spectrogram, the horizontal axis represents time, and the vertical axis represents frequency. Energy is represented by the color intensity of each pixel at a particular frequency and particular time. In Subsection 6.2.1 we will illustrate how log-filters can be applied.

6.2.1 Application of log-Gabor Filters

To illustrate the application of log-Gabor filters to affective vocal expressions, Figure 6.1 contains five spectrograms for the utterance "He is a good person" in Chinese spoken with the following five emotional expressions: (a) *angry*, (b) *happy*, (c) *neutral*, (d) *panic* and (e) *sad*. Comparing the five spectrograms highlights, several differences are apparent. For the neutral emotion, the energy bands are primarily horizontal, while for the other four emotions differ-

ent orientational patterns are present. The differences in orientation reflect the spectro-temporal dynamics of vocal pitch, which can rise over time (upward orientation; e.g., angry, happy), remain stable (horizontal orientation; neutral), or fall over time (downward orientation, e.g., panic, sad)). As Figure 6.1 indicates, the shift from high values to low values corresponds to a shift from red to blue. The parallel horizontal energy bands (red bands) are generated by the vocal cord. For the neutral emotion, the horizontal energy bands reflect a constant frequency pattern. Correspondingly, energy bands with clear orientations provide the information cues regarding emotion. All of this can be considered as different types of spectro-temporal modulation. In Subsection 6.2.3 we provide detailed descriptions of the five orientations. We first would like to inform the reader on previous studies.



a angry **b** happy **c** neutral

d panic **e** sad

Figure 6.1: Five spectrograms of the utterance “He is a good person" spoken in Mandarin with five different emotions.

6.2.2 Previous Studies

Many previous studies have made important contributions that shed light on how different types of emotion are vocally expressed (cf. Kreiman & Sidtis, 2011; Bänziger & Scherer, 2005). For example, Hammerschmidt and Jürgens found the spectro-temporal energy bands are useful for describing acoustic characteristics of emotional vocal expressions (cf. Hammerschmidt & Jürgens, 2007). They observed that different vocal emotions were associated with different energy bands. Their study evaluated five types of vocal emotions: angry, happy, neutral, panic, and sad.

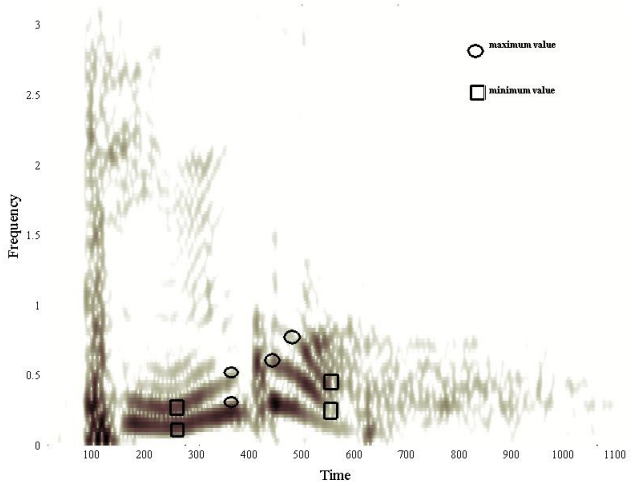


Figure 6.2: Spectrogram of the phrase "So bad" in Chinese expressed with an angry vocal emotion. The energy bands have an upward and sharp downward contour orientation. The minimum and maximum values of an energy band are indicated by a square and circle, respectively.

Figure 6.2 displays an example of an angry utterance "很坏" in Chinese (which means "So bad" in English), as collected by the MAS corpus. The spectro-temporal representation consists of two segments. The left segment consists of parallel energy bands that move upward. The right segment contains parallel energy bands that move downward. Hammerschmidt and Jürgens measured the minimum and maximum frequency values of the bands within each segment. In the figure, these extreme values are represented by squares (minimum values) and circles (maximum values). The slope of the line connecting the minimum and maximum values quantifies the orientation of the energy bands.

6.2.3 Descriptions of Five Orientations

We translate the quantitative orientation measurements by Hammerschmidt and Jürgens (2007) into five qualitative descriptions of the slope: horizontal, fast upward, slow upward, fast downward, and slow downward. Table 6.1 specifies the descriptions for each of the five types of emotion.

To detect the vocal emotions from their spectro-temporal signatures, we define two sets of tuned Gabor filters. As discussed above, the angle can be positive or negative. The orientation of the filter is set as follows: the horizontal slope is set to 0°, the fast upward slope equals 45°, and the slow upward slope is 30°. The downward slopes are defined by negative angles. We do not attempt to optimize the orientation through machine learning. We experiment with sin-

Table 6.1: Qualitative descriptions of the slopes of the first and second segment of five vocal emotions.

Emotion	First Segment	Second Segment
Angry	Fast upward	Slow downward
Happy	Fast upward	Fast downward
Neutral	Horizontal	Horizontal
Panic	Slow upward	Fast downward
Sad	Slow downward	Slow downward

gle filters (covering one segment) and double filters (covering two neighboring segments).

Table 6.2: Specification of the *single log-Gabor filters* tuned to the five emotions.

Emotion	Gabor filter	Orientation
Angry	G^1_{angry}	45°
Happy	G^1_{happy}	45°
Neutral	$G^1_{neutral}$	0°
Panic	G^1_{panic}	30°
Sad	G^1_{sad}	−30°

Table 6.3: Specification of the *log-Gabor filter pairs* tuned to the five emotions.

Emotion	Gabor filter	Left	Right
Angry	G^2_{angry}	45°	−30°
Happy	G^2_{happy}	45°	−45°
Neutral	$G^2_{neutral}$	0°	0°
Panic	G^2_{panic}	30°	−45°
Sad	G^2_{sad}	−30°	−30°

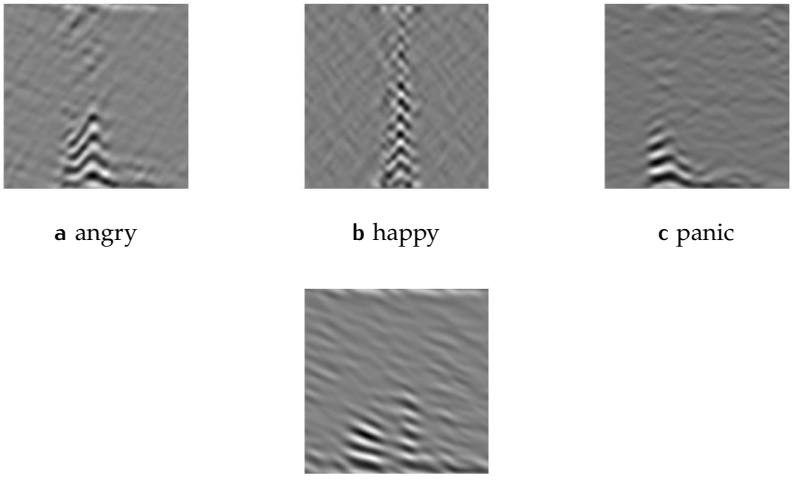
The first set of tuned Gabor filters consists of single filters tuned to the dominant orientation of the associated spectrogram. Table 6.2 lists the orientations of the single log-Gabor filters designed to detect the five emotions (including neutral).

The second set of tuned Gabor filters consist of horizontally contiguous pairs of filters that are tuned to the dominant combinations of orientations in the spectrograms. In addition, these combinations are estimated representa-

tions from a representative sample of emotional expressions. Table 6.3 lists the orientations of the log-Gabor filter pairs designed to detect the emotions. The orientations of the left and right filters of each contiguous pair are specified in the columns labeled left and right, respectively. *Neutral* and *sad* only have one dominant orientation direction. Meanwhile, *angry*, *happy*, and *panic* have two different orientations. Figure 6.3 displays the Gabor filter pair tuned to detect the characteristic orientations of the spectrogram associated with panic.

Figure 6.3: Illustration of G^2_{panic}.

Figure 6.4 illustrates the outcome after convolving the four emotional expressions happy, angry, panic, and sad with the filter pairs G^2_{angry}, G^2_{happy}, G^2_{panic} and G^2_{sad},. Comparing the four convolved images, the specific orientation patterns of the Gabor pairs are clearly visible. Provided that the tuned filters respond selectively to the emotion-specific orientations in the spectrograms, the filters support the automatic recognition of emotions from speech.



a angry **b** happy **c** panic

d sad

Figure 6.4: Convolution images obtained by convolving the spectrograms in Figure 6.1 with the associated Gabor filter pairs listed in Table 6.3.

Therefore, convolving the spectrogram with a Gabor filter of a given spatial frequency and orientation, the convolved spectrogram illustrates spectro-temporal patterns and the associated spatial frequency (width) and orientation, respectively. In this experiment, we only tune the orientation of the Gabor filters and average the range of spatial frequencies encompassing the widths of the energy bands of interest.

6.3 EXPERIMENT WITH LOG-GABOR FILTERS

To determine the effects of tuned Gabor filters in extracting useful features for the automatic recognition of affective speech, we perform a comparative evaluation focusing on (1) acoustic features, (2) untuned Gabor filters (all orientations), (3) tuned Gabor filters (single orientations and pairs of orientations), and (4) a combination of acoustic features and tuned Gabor filters. The remainder of this section covers three topics. Subsection 6.3.1 provides the details of the experiment set-up stage. Subsection 6.3.2 subsequently describes the experiment evaluation procedure. Finally, Subsection 6.3.3 discusses the results of the experiment.

6.3.1 Experiment Set-up

The procedure of the primary feature algorithm for SER contains three steps (A, B, and C; step B is subdivided into B1, B2, and B3), which are applied to each utterance in the corpus. The steps are (A) acoustic feature extraction; (B) Gabor-filter feature extraction: [B1] spectrogram calculation, [B2] convolution with Gabor filters, [B3] dimensionality reduction; and (C) classification. In the following, each of the three steps is described in detail.

A: Acoustic Features

All acoustic features used in our experiment correspond to the baseline features of the Interspeech challenge (cf. Schuller et al., 2015), which is the current state-of-the-art feature set in SER. The acoustic feature set for each utterance consists of the following seven features: Linear Predictive Cepstral Coefficients (LPCC), Mel-Frequency Cepstral Coefficients (MFCC), zero-crossing rate, speech rate, pitch, formants ($1-3$), and magnitude. Each feature is represented by four statistical descriptors: mean, maximum, minimum, and standard deviation. In the present day, acoustic features can be straightforwardly extracted by employing certain software packages. Different software products have different interfaces, some of which are particularly easy to use. For example, PRAAT is an acoustic analysis software used for phonetic speech and designed for the beginning researchers. It permits the user to analyze the majority of the voice cues relevant to speech emotion expression and is free to

download [18]. Voicebox is another speech processing toolbox for acoustic feature analysis and extraction. It primarily works on the Matlab software platform. In our experiment, the Voicebox toolbox is used as the tools for the acoustic feature extraction.

B: Log-Gabor Filters Feature Extraction

B1: Spectrogram Calculation

Each auditory signal (utterance) is transformed into a spectrogram using Matlab's spectral analysis function, which employs the short-time Fourier transform with a 20 ms Hamming window and an overlap of half the window length.

B2: Convolution with Gabor filters

Kovesi's log-Gabor functions [19] are used to perform the convolutions on the four quadrants of each spectrogram. The following parameters are employed. There are 12 orientations in equal steps covering 360 degrees for "untuned". For "tuned" condition, the 8 orientations and orientation pairs specified in Tables 6.2 and 6.3 are used. These filters are not tuned to the data, but to general principles extracted from the literature. The number of scales is 12; the minimum wavelength is 3 and sigmaOnf is 0.8. For each of the four spectrogram regions, each scale and orientation yields a single convolution image. The convolution values within each image are averaged yielding a total number of Gabor energy values equal to the number of orientations multiplied by the number of scales.

B3: Dimensionality Reduction

To reduce the redundancy of the Gabor energy values, Principal Component Analysis is applied. We utilize the PCA function that is incorporated within the dimensionality reduction toolbox (cf. Van der Maaten et al., 2009a) and optimize the number of retained components from 10 to 80 in steps of 10.

C: Classification

The dimensionality-reduced Gabor energy values are used for training an SVM classifier. We use the WEKA[20] implementation, with an RBF kernel. The parameter values Gamma g and Cost c are optimized by means of grid search. The range of value on a log2 scale ($g = G^2$, $c = C^2$) as depicted in Table 5.3. The WEKA is a suite of machine learning software, developed at the University of Waikato. It has high-functioning algorithms for data analysis and machine learning modeling, alongside graphical user interfaces for easy employment of these functions.

18 http://www.fon.hum.uva.nl/praat/

19 http://www.csse.uwa.edu.au/ pk/research/matlabfns/

20 http://www.cs.waikato.ac.nz/ml/weka/

In the comparative evaluation, acoustic features are treated similarly to the Gabor features (i.e., application of PCA and classification via an SVM). In the experiment involving acoustic features and Gabor features, all features are combined into a single feature vector and subjected to PCA and SVM.

6.3.2 Evaluation Procedure

This subsection describes the experiment performed to evaluate the performance of the tuned Gabor filter pairs. The aim of this subsection is to investigate how we can use two-dimensional features to analyze spectrogram representations of speech.

To determine the contribution of the visual features extracted by means of two-dimensional Gabor filters, we perform a comparative evaluation of the following five types of features for a recognition task: (1) acoustic features (e.g., MFCC and LPCC features), (2) untuned Gabor filters, (3) tuned Gabor filters, (4) tuned Gabor filter pairs, and (5) a combination of acoustic features and tuned Gabor filter pairs. The extracted features of all above types are trained and evaluated on the MAS corpus (see Section 4.2). Our evaluation focuses on the classification performance (see Definition 5.2) of these five feature types.

The SER performance evaluation for each set of features relies on the use of cross-validation procedures. In order to avoid overfitting due to PCA and SVM parameter optimization, we perform a 10×10 fold cross validation. The optimization was performed in the outer 10-fold CV cycle and the evaluation in the inner cycle. The advantage of 10 fold cross validation is that all feature sets are used for both training and validation, and each feature subset is used for validation exactly once. All evaluations (cross validation experiments) are repeated 10 times and the results are averaged.

6.3.3 Results of Experiment with log-Gabor Filters

In this subsection, we outline the results of the experiment in two areas: (A) the performance results for five algorithmic classifications and (B) the results of a confusion matrix of the respective emotional states.

A: Experiment Results of Classification Performance

Figure 6.5 contains box plots of the recognition performances obtained for the five sets of features. From left to right, these are: (a) acoustic features, (b) untuned single Gabor filters, (c) tuned single Gabor filters, (d) tuned Gabor filter pairs, and (e) acoustic + tuned Gabor filter pairs. All performances are averaged accuracies over all folds of cross-validation.

As seen in Table 6.4, untuned single Gabor filters have the lowest performance rate, 82.1%, with a high standard deviation. This is because untuned

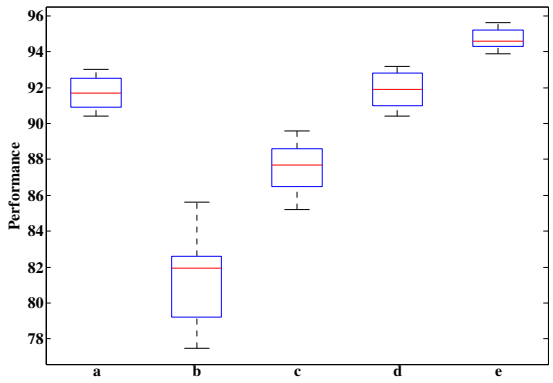


Figure 6.5: Recognition performances expressed in percentages obtained for the five sets of features.

Table 6.4: Confusion table of all feature performances

Performance Metric	Performance Rate (SD)
Acoustic features	91.7 (1.2)
Untuned single Gabor filters	82.1 (3.8)
Tuned single Gabor filters	87.9 (1.7)
Tuned Gabor filter pairs	92.2 (1.4)
Acoustic + Tuned Gabor filter pairs	94.1 (1.5)

Gabor filters are chosen from 8×12 different options for orientation filters, which can cause an excessive number of redundant features, and hence, underfitting or overfitting in terms of classification.

The tuned single Gabor filters result in an 87.9% accuracy rate. The improved performance of the tuned Gabor filters as compared to the untuned filters (which include the tuned orientations), indicates that a reduction of features is beneficial in terms of performance. However, the performance of the tuned single Gabor filters is still lower than that of the tuned Gabor filter pairs, which achieve a significantly higher recognition performance (92.2%). The reason is that a tuned single Gabor filter is designed to extract only one orientation for each type of emotion on a spectrogram, which may cause the omission of some useful features. In contrast, tuned Gabor filter pairs are designed with two orientations, especially for both slopes in the first and second segments of vocal emotions in a spectrogram. More details and fluctuations are therefore included in this filter design, effectively capturing all patterns of the five types of emotion.

Traditional acoustic features achieve a recognition rate of 91.8%, thereby outperforming the untuned and tuned (single) Gabor filters. In this sense, en-

coding oriented energy bands in the spectrogram does not necessarily lead to a better performance than the one obtained with traditional acoustic features. The tuned Gabor filter pairs perform comparably reasonable to the acoustic features. The combination of tuned Gabor filter pairs and acoustic features yields the best overall performance (94.6%), suggesting that this approach partly captures non-overlapping vocal characteristics.

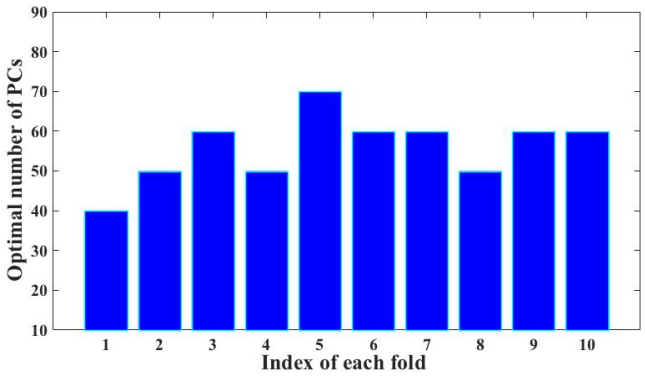


Figure 6.6: The optimal number of PCs for each fold.in 10 fold cross-validation based on the tune log-Gabor pairs.

The number of PCs was optimized in the outer loop of the cross-validation procedure as mentioned in Subsection 6.3.1. This evokes the question of how the optimal number of principal components varies in each fold during the cross-validation. Figure 6.6 illustrates the optimal number of principal components for the tuned Gabor filter pairs which is achieved in each fold through cross-validation. In Figure 6.6, the x-axis is the index of each fold and the y-axis is the optimal number of principal components. It shows that the optimal numbers are slightly different between folds but generally close to 60.

Table 6.5: Numbers of Principal components of all feature for the best performance

Classification	Numbers of PCs
Acoustic features	50
Untuned single Gabor filters	20
Tuned single Gabor filters	30
Tuned Gabor filter pairs	60
Acoustic + Tuned Gabor filter pairs	70

Table 6.5 illustrates the optimal numbers of principal components for the five feature sets associated with their best performances (reported in Table 6.4) in the cross-validation. As illustrated in the table, the optimal numbers of PCs differ for different feature sets. For the tuned Gabor filter pairs, the best performance (92.2%) is obtained when 60 principal components are retained.

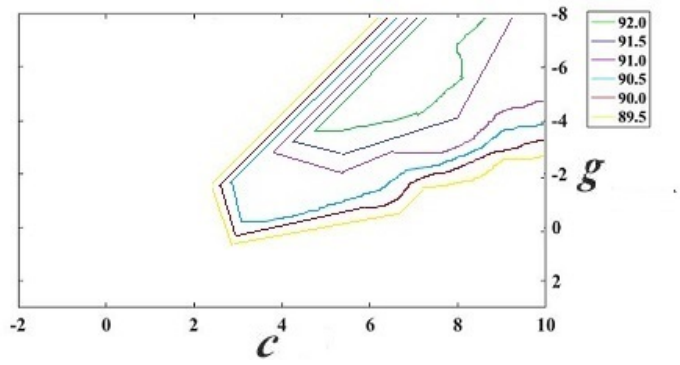


Figure 6.7: Contour plot illustrating the grid-search results for optimizing the SVM parameters c and g for the Gabor filter algorithm.

By using a RBF based SVM classifier for training, the best combination of c and g needs to be obtained. In this experiment, we use the grid search to obtain the best combination of SVM parameters c and g in the outer loop of the nested cross validation. As shown in Figure 6.7, it illustrates a contour plot of visualizing the results of the grid-search parameter optimization. We can observe that the x-axis is c and the y-axis is the g in figure. As shown in Figure 6.7, the performance varies dramatically with different values of parameters. The best performance of 92.2% accuracy is obtained according to the values $c = 8$ and $g = -6$ ($C = 256$ and $gamma = 0.015625$.). And variety color lines indicate different performance, which shows a contour plot of the performances obtained for the examined range of parameter values. The green line marks the performance over 92.0% through 10 fold cross validation.

B: Experiment Results for Individual Emotions

Tables 6.6 to 6.10 are confusion tables for five feature sets. These results are obtained based on the optimal number of principal components which is listed in Table 6.5. In these confusion tables, the predicted emotional states are represented by the rows, and the actual emotion states are shown by the columns. Hence, each confusion table illustrates how often each emotion state is correctly recognized (diagonal entries) and incorrectly recognized (off-diagonal entries). The percentages are calculated according to the times that emotion state is correctly recognized against the total times.

As Table 6.9 reveals, the neutral emotion achieves the highest recognition rate (93.2%). The reason for this is that the visual features of the neutral emotion in a spectrogram are distinct from those of the other emotions. As demonstrated in Figure 4.1, in most cases, the neutral emotion is mild, i.e., there are fewer emotional fluctuations than for other emotions (angry, happy, panic, sad). Correspondingly, the energy band of the neutral emotion on a spectrogram is often horizontal, with only a very slight slope. Thus, the neutral emotion is the most feasible to detect with a 0° log-Gabor filter.

Table 6.6: Confusion table of acoustic features.

	Performance (%)				
Emotion	Angry	Happy	Neutral	Panic	Sad
Angry	90.7%	3.8%	2.4%	2.0%	1.1%
Happy	4.2%	91.1%	1.5%	2.3%	0.9%
Neutral	2.3%	1.9%	92.6%	1.5%	1.7%
Panic	2.6%	1.9%	0.8%	92.2%	2.5%
Sad	1.5%	1.4%	3.2%	1.9%	91.9%

Table 6.7: Confusion table of untuned Gabor filters.

	Performance (%)				
Emotion	Angry	Happy	Neutral	Panic	Sad
Angry	81.6%	6.4%	3.5%	5.0%	3.5%
Happy	6.1%	81.1%	3.4%	5.2%	4.2%
Neutral	4.7%	3.9%	83.9%	3.5%	4.0%
Panic	4.8%	6.4%	2.9%	81.7%	4.2%
Sad	4.4%	5.1%	3.7%	4.6%	82.2%

From our findings, we may conclude that tuned log-Gabor filters (1) enhance the automatic recognition of emotions from speech and (2) may be beneficial as regards other speech-related tasks. The experiments are carried out on a database containing recordings of actors speaking. It would be valuable to test the presented methods on other speech databases in future work (cf. Ringeval, Sonderegger, Sauer, & Lalanne, 2013).

6.4 CHAPTER DISCUSSION

In this chapter, we investigated how the visual cues of speech emotion can be identified in a spectrogram and how two-dimensional features can be extracted from a spectrogram. As mentioned in Chapter 1, current SER methods are based on machine-learning technology, which relies strongly on feature relevance. Instead of relying on the traditional method using temporal and spectral features, which have proven their relevancy to speech-related tasks for SER, we designed two types of log-Gabor filters, single tuned Gabor filters and the tuned Gabor filter pairs, to construct our feature set. The constructed features were evaluated by comparing their SER performances with those of a

Table 6.8: Confusion table of tuned Gabor filters.

	Performance (%)				
Emotion	Angry	Happy	Neutral	Panic	Sad
Angry	88.1%	3.5%	2.7%	3.4%	2.3%
Happy	4.0%	86.8%	2.4%	3.7%	3.1%
Neutral	3.3%	2.9%	89.4%	2.5%	1.9%
Panic	3.5%	4.1%	2.8%	86.7%	2.9%
Sad	3.2%	3.0%	2.9%	2.4%	88.5%

Table 6.9: Confusion table of tuned Gabor filter pairs.

	Performance (%)				
Emotion	Angry	Happy	Neutral	Panic	Sad
Angry	91.6%	2.7%	2.2%	1.9%	1.6%
Happy	2.9%	92.1%	1.1%	2.1%	1.8%
Neutral	1.9%	2.3%	93.2%	1.4%	1.2%
Panic	1.8%	2.9%	1.4%	91.9%	1.9%
Sad	2.6%	2.2%	1.7%	1.8%	91.7%

state-of-the-art feature. Five types of emotional states were chosen for evaluation (angry, happy, neutral, panic and sad).

The results reveal that the tuned Gabor filter pairs that we designed achieved a higher performance rate than the state-of-the-art approach during the feature construction stage. This calls for the question why log-Gabor filters can detect features for classification more efficiently than the traditional state-of-the-art acoustic feature set. We believe that two reasons for the obtained improved classification performance exist.

The first reason is that we performed manual feature selection by inspecting several spectrograms of emotional speech. As a result, the slopes of the emotions in the spectrograms were qualitatively identified using the following five descriptions: horizontal, fast/slow upward, and fast/slow downward. In this way, the tuned Gabor filter pairs that we designed have the same orientations as the slope descriptors. This led to a limited set of features, which prevented the curse of dimensionality in capturing task-relevant information from the spectrogram.

The second reason for the improved classification performance is that Gabor filters are easy to redesign to fit any orientation of the energy band. Moreover each redesign is able to convolute the entire spectrogram. In sum, the

Table 6.10: Confusion table for the combination of acoustic features and tuned Gabor filter pairs.

	Performance (%)				
Emotion	Angry	Happy	Neutral	Panic	Sad
Angry	93.5%	2.4%	2.1%	1.7%	0.3%
Happy	3.2%	93.3%	1.0%	1.8%	0.7%
Neutral	1.7%	1.3%	95.5%	0.7%	0.8%
Panic	1.8%	1.3%	0.9%	93.9%	2.1%
Sad	0.6%	1.0%	2.8%	1.3%	94.3%

orientation tuning of Gabor filters is highly suitable for encoding these patterns.

In our finding we ignored the role of spatial frequency (the width of the energy bands) by including a wide range of spatial frequencies in our tuned Gabor filters. We consider it an open question whether there is a link between the width of the energy bands and the precision of the constructed features. It is possible that an improved tuning in terms of spatial frequency may further enhance the results. This question is left to future study.

At the end of the discussion, we establish that we thoroughly studied the use of the designed log-Gabor filters in extracting the two-dimensional features from a spectrogram. The two types of designed log-Gabor filters (single and pairs) were illustrated in Tables 6.2 and 6.3. The extracted log-Gabor features are highly representative of speech emotions.

6.5 ANSWER TO RESEARCH QUESTION TWO

The research question addressed in this chapter is RQ2, which reads as follows.

RQ2: *How can we use two-dimensional features to analyze the spectrogram representation of speech?*

In order to answer this research question, we first identified the features in spectrograms. By using domain knowledge, we found the relevant feature patterns of the spectrograms in relation to each emotional state. Secondly, we proposed using log-Gabor filters as tools for analyzing and extracting the features from spectrograms. This idea originated from Ezzat, Bouvrie, and Poggio (see Ezzat et al., 2007). We designed three types of Gabor filter (untuned single Gabor filter, tuned single Gabor filter, and tuned Gabor filter pairs) to extract features. The results of the experiment reveal that the untuned single Gabor filter and the tuned single Gabor filter did not achieve a satisfactory perfor-

mance rate. However, the tuned Gabor filter pairs performed a second-order analysis of the spectrogram, that yielded acceptable results. Based on these observations, combined with our findings, we may conclude that we can (1) use domain knowledge to identify the feature patterns in a spectrogram, and then (2) employ log-Gabor filters to extract features and analyze the spectrogram representation of speech. Moreover, the tuned log-Gabor filter pairs produced novel enhanced features and can boost the performance of SER.

7 LESS-INTENSIVE FEATURES IN A SPECTROGRAM

This chapter [21] addresses research question three (RQ3).

RQ3: Can we extract additional, likely less-intensive features via the composition of Gabor filters through a spectrogram?

In Chapter 6, we used tuned Gabor filter pairs to extract the primary feature patterns of five different emotions represented in spectrograms. In this chapter, we attempt to discover whether less-intensive emotional patterns can also be detected in a spectrogram. Thus far, we have focused on the most prominent emotional feature patterns of a spectrogram, those representing the most intensive expressions of emotions. However, emotions can be expressed in several parts of a sentence. Therefore, we further categorize emotional expressions in a sentence into primary and subsequent expressions according to their intensiveness. We refer to them as primary and subsequent patterns. Each emotion may show a unique feature pattern when manifested as a subsequent expression. We aim to explicitly reveal the feature patterns of the subsequent emotional expressions in a spectrogram. To detect them, we carry out experiments and assume that the detection of subsequent feature patterns is conducive to SER performance. Success in extracting subsequent emotional feature patterns would mean that they could be used to further improve the accuracy of SER.

The chapter is structured as follows. The term less-intensive features is introduced in Section 7.1. The experiment set-up and results are reported and analyzed in Section 7.2. A discussion is found in Section 7.3, while RQ3 is conclusively answered in Section 7.4.

7.1 MEANING OF LESS-INTENSIVE FEATURES

In Chapter 6, we identified the most apparent feature patterns and designed the corresponding tuned Gabor filter pairs. Hence, it comprises the foundation of this chapter. The feature patterns in Chapter 6 were extracted from the parts of the sentences that demonstrated the most intensive expressions of emotions in the spectrograms. We call these *the primary emotions* of a sentence. However, other parts of sentences may also contain less-intensive expressions of emotion.

21 This chapter is based on work by Y. Gu, E.O. Postma, H. Jaap van den Herik and H.X Lin; Speech Emotion Recognition with Log-Gabor Filters. In Proceedings of the 8th International Conference on Agents and Artificial Intelligence, ICAART, 2016.

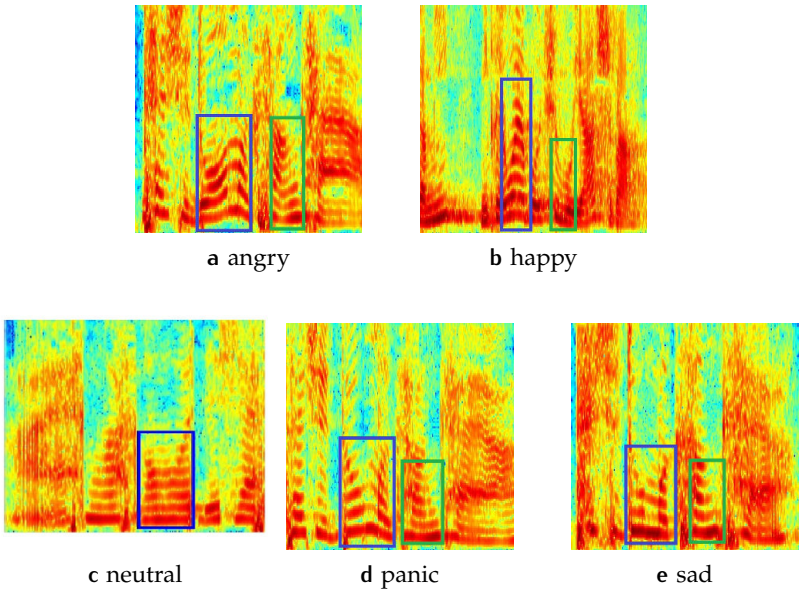


Figure 7.1: Five spectrograms of the utterance "He is a good person" with five different emotions marked by blue and green rectangular.

These less intensive expressions occur relatively frequently in these sentence elements.

Thus, the less-apparent parts of the spectrogram can also contain some emotional expressions. In other words, a spectrogram may also feature less-intensive feature patterns. Indeed, they may last for a shorter duration but should still deserve our attention. For the sake of convenience and clarity, we qualify the most apparent filter pairs and their feature patterns as *the primary pattern*, and the less-intensive filter pairs and their feature patterns as *the subsequent pattern*.

Figure 7.1 shows five spectrograms of the utterance "He is a good person" in Chinese. In the primary pattern of the feature extraction, the vocal expressions of the five emotions in the spectrograms are extracted: (a) *angry*, (b) *happy*, (c) *neutral*, (d) *panic*, and (e) *sad*. Each of the five spectrograms exhibits a distinct feature pattern in the form of an energy band. The primary features of the five emotions are outlined in blue rectangles, and each emotion demonstrates a unique feature pattern .

However, the emotion is not solely concentrated in the dominant parts of the sentence. The remaining parts may also contain a portion of the emotion. For example, a subsequent emotional expression may occur in the latter part of a sentence. We therefore attempt to identify the subsequent feature patterns in the spectrograms. In Figure 7.1, the green rectangles mark the less-intensive patterns.

7.1.1 Primary and Subsequent Patterns

For each emotion, the spectrogram demonstrates the primary and subsequent patterns. Their respective orientations are listed in Table 7.1 and Table 7.2. The log-Gabor filter pairs designed with specific orientations may help to extract feature patterns from the spectrograms. Therefore, to detect the spectro-temporal patterns, two groups of filters are set to the specific orientations. One group of filters, which we call *primary (tuned) Gabor filters*, correspond to the most prominent patterns, which we manually designed in Chapter 6. The other group, named *the subsequent filters*, correspond to the less-intensive patterns. We define the upward orientation as a positive angle and downward orientation as a negative angle. The orientation of the filter is set as follows: the horizontal orientation is 0°, the slow upward orientation is 30°, the fast upward orientation is 45°, and the downward orientation have symmetrical negative values to their upward counterparts.

7.1.2 The Neighboring Segment

Below, we discuss our experiment with primary and subsequent log-Gabor filters, which both of them have two neighboring segments. The segments for the primary Gabor filter are manually chosen based on the most expressive parts of the sentence. The primary filter group consists of the filter pairs specific to the orientation in the associated spectrogram. Table 7.1 lists the orientations of the primary log-Gabor filter pairs designed to detect the five emotions. Regarding the segments for the subsequent Gabor filters, we first predict the likelihood of patterns based on observations and experiences. Then, we shift the window over the entire spectrogram and apply feature selection methods to identify the most apparent patterns in the remaining parts of the spectrogram. Further details are provided in Subsection 7.2.1.

Table 7.1: Specification of the primary log-Gabor filter pairs for the five emotions

Speech emotion	Gabor filter	Left	Right
Angry	G^{d}_{angry}	45°	−30°
Happy	G^{d}_{happy}	45°	−45°
Neutral	$G^{d}_{neutral}$	0°	0°
Panic	G^{d}_{panic}	30°	−45°
Sad	G^{d}_{sad}	−30°	−30°

7.2 EXPERIMENT: LESS-INTENSIVE FEATURES WITH LOG-GABOR FILTER PAIRS

This section describes an experiment intended to extract the less-intensive features from a spectrogram using log-Gabor filters. The goal is to evaluate to what extent subsequent log-Gabor filter pairs are able to perform the task of revealing less-intensive features. The subsequent log-Gabor filters are an extension of the primary (tuned) log-Gabor filter pairs proposed in Chapter 6.

The remainder of this section outlines the three stages of the experiment. Subsection 7.2.1 provides the experiment set-up for the subsequent log-Gabor filters and the features extracted by them. In Subsection 7.2.2, the evaluation procedure is presented. Finally, the results of the experiment are discussed in Subsection 7.2.3.

7.2.1 Experiment Set-up

The extraction of subsequent Gabor filters is based on the following four steps which we follow to analyze each utterance in the corpus: (A) spectrogram calculation, (B) convolution with Gabor filters, (C) feature selection of subsequent features, and (D) classification. Each step is specified below.

A: Spectrogram Calculation

We transform each auditory signal (utterance) into a spectrogram, employing (i) Matlab's spectral analysis and (ii) a short-time Fourier transform with a Hamming window of 20 ms in length and an overlap length of 10 ms. So, the total number of spectrograms which we could obtain is 20,400 from the corpus. We divide them into two spectrogram subsets (spectrogram subset 1, and spectrogram subset 2) of an equal size of 10,200 spectrograms. We use one subset for feature selection and the other for classification in order to avoid overfitting.

B: Convolution with Gabor filters

Kovesi's log-Gabor functions[22] are used to perform the convolutions on the four quadrants of each spectrogram. The parameters used are as follows.

(1) Attempting to cover the entire 360 degrees, we used 8 and 12 in the experiment for the specific orientations illustrated in Table 7.1 and Table 7.2, which correspond to the primary and subsequent filter pairs, respectively.

(2) The Number of scales equals $= 12$; MiniWavelength is $= 3$, and SigmaOnf is $= 0.8$.

22 http://www.csse.uwa.edu.au/ pk/research/matlabfns/

(3) For each of the four spectrogram regions, each scale and orientation yield a single convolution image. The convolution values within each image are averaged, yielding a total number of Gabor energy values equal to the number of orientations multiplied by the number of scales.

C: Feature Selection of the Subsequent Feature

In Step (B), the subsequent features are extracted from the spectrogram set 1. We use the Fisher method (cf. Roffo, Melzi, & Cristani, 2015) from the Matlab Feature Selection Library toolbox[23] for selecting the subsequent features. The Fisher method computes the so-called F-score for a feature as the ratio of interclass separation and intraclass variance, in order to evaluate features independently. A larger F-score signifies a feature that is likely to contribute to the prediction task. The final feature list consists of the top-ranked features for each emotional state. The top-ranked features which are computed by the Fisher method can be thought as "automatically selected features". From the list of top-ranked features, we can find the corresponding log-Gabor filters' orientation. We further design "manually selected features" which is another type of subsequent feature. The motivation for us is that, different from the orientations in Table 7.1 which are most dominant in spectrogram, the less frequently appearing features in the top-ranking list which can be extracted by subsequent log-Gabor filter pairs are a good supplementary to the feature group. Therefore, we design the subsequent log-Gabor filter pairs' orientation based on the features that are listed in top-ranked "automatically selected features" but have not been used as primary features yet. Table 7.2 lists the orientations of the manually selected subsequent log-Gabor filter pairs.

Table 7.2: Specification of subsequent log-Gabor filter pairs for the five emotions

Vocal emotion	Gabor filter	Left	Right
Angry	G^s_{angry}	−45°	45°
Happy	G^s_{happy}	−30°	45°
Neutral	$G^s_{neutral}$	0°	0°
Panic	G^s_{panic}	−45°	30°
Sad	G^s_{sad}	−30°	30°

D: Classification

We use the corresponding subsequent log-Gabor filter to extract automatically selected features from the independent spectrogram subset 2 to train a new SVM Classification, which is dependent from the dataset

23 https://au.mathworks.com/matlabcentral/fileexchange/56937-feature-selection-library

used for the feature selection. Our experiment employs the WEKA[24] as the classification tool. A support vector machine (SVM) is used to classify the subsequent log-Gabor filter features. The SVM's c and g parameter values are optimized by means of a grid search. All Gabor energy values, including the primary and subsequent filter values, are used to train an SVM classifier. In Section 3.3, we provided a more detailed explanation of the SVM.

7.2.2 Evaluation Procedure

To evaluate the performance obtained via the subsequent log-Gabor filter features, we assess the accuracy of the five feature types: (a) acoustic features, (b) untuned features, (c) primary filter features, (d) subsequent filter features, and (e) primary filters with subsequent filter features. The first three types of features ((a) to (c)) are extracted via the procedure introduced in Chapter 6. The performance of the subsequent filter features is achieved as discussed in Subsection 7.2.1. The performance provides a criterion by which we evaluate whether the subsequent filter features are suitable and useful for improving the performance of SER.

The features discussed above are trained and assessed on the MAS corpus (see Section 4.2). In our assessment, we focus on the classification performance (see Definition 5.2) of the five feature types. For the subsequent features, we would like to determine if the selected subsequent log-Gabor filters are actually useful for the method. We start with using feature selection to find the top-ranked subsequent feature which we extract from the spectrogram subset 1. Then we use the corresponding subsequent log-Gabor filter to extract automatically selected features from the independent spectrogram subset 2 to train a new SVM Classification.

An evaluation of SER performance is conducted on subsequent features using cross-validation procedures. To avoid overfitting and to ensure that the data used for training and testing are selected randomly, the evaluation is performed using 10 times 10 fold cross validation on the subset 2. The optimization is performed in the outer each-fold cycle and the evaluation is performed in the inner cycle. The evaluations experiments are repeated 10 times and the final performance results are averaged.

7.2.3 Results of Experiment with Subsequent log-Gabor Filters

This subsection provides the results of the experiment in two parts: (A) the overall performance results of five classification and (B) the results of individual emotion states showed in a confusion matrix.

24 http://www.cs.waikato.ac.nz/ml/weka/

A: Experiment Results of Classification Performance

We use the Fisher feature selection method to obtain the top ten high F-ratio score features as explained in Subsection 7.2.1. Through those automatically selected features, we can find out the corresponding log-Gabor filters' orientation which were used to extract those features. And then, we can further design the subsequent Gabor filter pairs as listed in Table 7.2. The whole subsequent Gabor features are the combination of automatically selected feature and manually determined features.

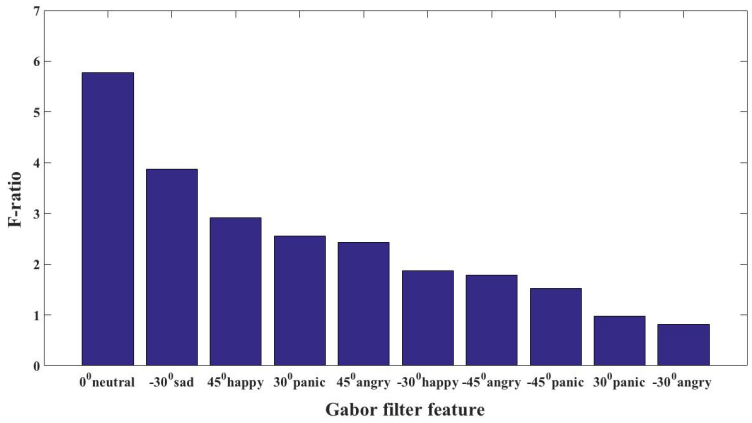


Figure 7.2: F-Ratio score for top ten Gabor filter features.

Figure 7.2 illustrates F-ratio scores of individual features which were extracted by the 144 filters (12 orientations $\times$12 scales) for each emotion state. As illustrated in the Figure 7.2, the x-axis indicates the Gabor filter features and the y-axis indicates the associated F-ratio scores. We can observe the top ten features which achieve the higher F-ratio scores. As shown in the figure, the horizontal (0° degree) feature for neutral emotion achieve the highest score of 5.78. And the following high F-ratio score features are $-30°$ feature for sad emotion (3.87), 45° feature for happy emotion (2.91), 30° feature for panic emotion (2.56) and 45° feature for angry emotion (2.43). Interestingly, these top five features correspond to the fine-tuned Gabor features in the previous Chapter. After the top five features, the rest of high F-ratio scores are $-30°$ features for happy emotion (1.87), $-45°$ feature for angry emotion (1.79), $-45°$ feature for panic emotion (1.52), $-30°$ feature for angry emotion, and 30° feature for sad emotion. The last five features of the top ten which have lower F-ratio scores are selected as subsequent features as they are unused in the previous Chapter. With these features at hand, it is possible to find out whether these features can help improve SER accuracy through subsequent log-Gabor filters.

The boxplot in Figure 7.3 shows the performances of the different feature sets, consisting of (a) acoustic features, (b) untuned Gabor filters, (c) primary (tuned) Gabor filter pairs, (d) subsequent Gabor filter pairs, and (e) primary Gabor filter pairs + subsequent Gabor filter pairs.

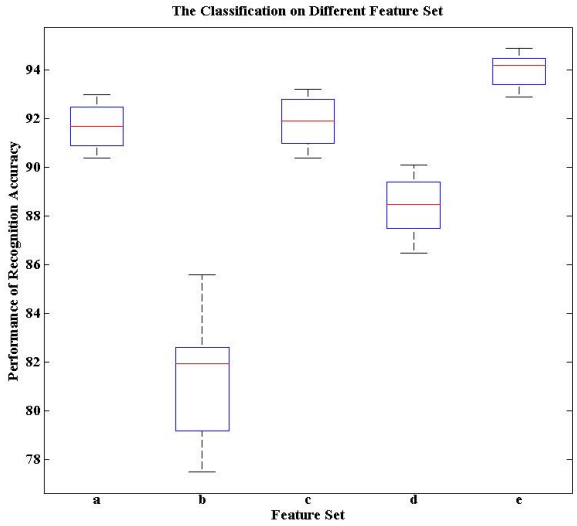


Figure 7.3: Recognition performances obtained for the five sets of features.

Table 7.3: Table of subsequent feature performance

Performance Metric	Performance Rate (SD)
Acoustic features	91.7 (1.2)
Untuned single Gabor filters	82.1 (3.8)
Primary (tuned) Gabor filter pairs	92.2 (1.4)
Subsequent Gabor filter pairs	88.6 (1.7)
Primary + Subsequent Gabor filter pairs	93.8 (0.9)

Table 7.3 lists the performance rates and corresponding standard deviation values of these features. As we can see from the performance rates in Figure 7.3 and Table 7.3, the subsequent Gabor filters are significantly less accurate than the primary Gabor filters. It is apparent that the primary pattern is the prominent pattern in the spectrogram and appears more often in the speech signal. The subsequent Gabor filters perform better than the untuned Gabor filters, indicating that the orientation feature is crucial due to the Gabor filter feature. The combination of primary and subsequent feature sets achieves the highest performance, which indicates that the combination is more effective than the primary Gabor filters alone.

B: The Results of Individual Emotion States

Tables 7.4 to 7.7 illustrate the confusion matrix of the five emotional states: *angry*, *happy*, *neutral*, *panic*, and *sad*. Table 7.7 demonstrates that the recognition rate is always higher for the *neutral* emotion (95.8%) than for the other emo-

Table 7.4: Confusion table of acoustic features

	Performance (%)				
Emotion	Angry	Happy	Neutral	Panic	Sad
Angry	90.7%	3.8%	2.4%	2.0%	1.1%
Happy	4.2%	91.1%	1.5%	2.3%	0.9%
Neutral	2.3%	1.9%	92.6%	1.5%	1.7%
Panic	2.6%	1.9%	0.8%	92.2%	2.5%
Sad	1.5%	1.4%	3.2%	1.9%	91.9%

Table 7.5: Confusion table of primary Gabor filter pairs

	Performance (%)				
Emotion	Angry	Happy	Neutral	Panic	Sad
Angry	91.6%	2.7%	2.2%	1.9%	1.6%
Happy	2.9%	92.1%	1.1%	2.1%	1.8%
Neutral	1.9%	2.3%	93.2%	1.4%	1.2%
Panic	1.8%	2.9%	1.4%	91.9%	1.9%
Sad	2.6%	2.2%	1.7%	1.8%	91.7%

tions. This is because the *neutral* emotion has a unique horizontal pattern that is significantly different from that displayed by the other emotions. Moreover, the performance of *sad* is higher than the remaining three emotions due to its distinct slow downward orientation pattern. The results correlate to the characteristics of the *sad* emotion in speech: a slow speaking rate and no dramatic change. *Panic* only achieves 87.1% accuracy on the subsequent Gabor filters, which means that its less emotion-intensive pattern is less visible in the spectrogram and thus less useful. However, for *panic*, the combination of primary and subsequent filters results in an improved recognition rate of 93.9%. Hence, the subsequent features cannot work independently, but can still be a valuable additional feature.

Therefore, based on the above experiment results, we may conclude that the subsequent Gabor filters are an effective complement to primary filters in the spectrogram. Moreover, a combination of primary and subsequent filters yields a better performance than the acoustic features, suggesting that the combination captures partly non-overlapping vocal characteristics that are different from acoustic features. This means that the subsequent Gabor filter features improve the SER rate.

Table 7.6: Confusion table of subsequent Gabor filters

	Performance (%)				
Emotion	Angry	Happy	Neutral	Panic	Sad
Angry	88.2%	3.7%	2.6%	3.3%	2.2%
Happy	3.7%	87.3%	2.7%	3.4%	2.9%
Neutral	3.2%	2.4%	90.5%	2.2%	1.7%
Panic	3.4%	3.8%	2.6%	87.1%	3.1%
Sad	2.9%	2.7%	2.2%	2.3%	89.9%

Table 7.7: Confusion table of the combination of primary and subsequent Gabor filter pairs

	Performance (%)				
Emotion	Angry	Happy	Neutral	Panic	Sad
Angry	92.8%	2.3%	2.0%	1.8%	1.1%
Happy	2.9%	93.1%	1.6%	1.8%	1.2%
Neutral	1.2%	1.1%	95.4%	1.0%	0.9%
Panic	1.8%	1.3%	0.9%	93.9%	2.1%
Sad	0.9%	1.3%	1.3%	2.2%	93.8%

7.3 CHAPTER DISCUSSION

The aim of the research discussed in this chapter was to extract less intensive features from a spectrogram. We considered how the log-Gabor algorithm could be further improved (see Section 7.1) and how the algorithm relates to the primary (tuned) Gabor filter pairs outlined in Chapter 6. To meet our research aim, we proposed using subsequent filter pairs to extract the subsequent feature patterns. To investigate the effectiveness of the subsequent features extracted using subsequent log-Gabor filter pairs, we compared five kinds of feature sets. The results of the experiment demonstrate that the subsequent log-Gabor filter pairs are a robust and meaningful feature extraction algorithm for SER.

To the best of our knowledge, our study is the first to use a combination of log-Gabor filters and spectrograms for SER. Overall, in our study we identified four types of feature, namely the untuned single Gabor filter, tuned single Gabor filter, primary (tuned) Gabor filter pairs, and subsequent Gabor filter pairs. Each fulfills a different SER function in terms of identifying different feature pattern.

In general, our research regarding the log-Gabor filter algorithm revealed that both the primary Gabor pairs and subsequent Gabor pairs used in our experiments respond directly to the emotional phenomena in a spectrogram. We see that (1) the primary Gabor filter pairs detect the more expressive emotion in speech and achieve a higher accuracy, and (2) the subsequent filter pairs focus on the less intensive expression and yield a weaker performance. However, we also note that (3) the combination of both primary and subsequent filter pairs has the highest accuracy rate. From this third observation, we may conclude that the subsequent pairs successfully detect less-intensive emotional expressions, and constitute a suitable complement to the primary feature set.

7.4 ANSWER TO RESEARCH QUESTION THREE

To expand our study (see Chapter 6), we explored the subsequent log-Gabor filter pairs. The corresponding research question addressed in this chapter was RQ3.

RQ3: *Can we extract additional, likely less-intensive features via the composition of Gabor filters through a spectrogram?*

To answer this question, we used the subsequent Gabor filter pairs to extract the less-intensive features from the speech spectrogram. The subsequent feature itself generally cannot achieve a sufficient result in a classification performance competition. However, we obtain the strongest performance by combining the primary and subsequent Gabor filter pairs. This means that less-intensive features can be successfully extracted by the subsequent Gabor filter pairs too, which represent a useful complement to the existing Gabor feature group capable of accomplishing a more accurate SER. Hence, our answer to RQ 3 is that the subsequent feature is an adequate complement to the feature group for SER. Yet, this conclusion does not mark the end of our research.

The next step in this study is to improve and expand the log-Gabor algorithm in SER by investigating the use of a deep-learning algorithm as a learning tool for extracting and measuring features from a spectrogram. The effectiveness of feature extraction accomplishing via deep-learning instead of a manual design is a worthy topic of exploration. Thus, the next step is to connect the spectrogram with a deep-learning algorithm such as a convolutional neural network.

8 DEEP LEARNING FOR SPEECH EMOTION RECOGNITION

This chapter is addressing research question four (RQ4).

RQ4: *Can we apply the deep-learning method to the spectrogram outcomes to extract "visual" features to increase the accuracy of SER?*

In our studies discussed in Chapter 6 and 7, we introduced approaches to extracting novel features from a spectrogram using log-Gabor filters. Our experiments treated the speech signal as a visual image. In this chapter, we shift the focus to a deep-learning (DL) algorithm: a convolutional neural network (CNN), which suggests a more complex model and an improved ability to detect new features. The new features may be completely new for human beings, or even incomprehensible for the human brain. However, in Go, chess, Shogi, poker and Dota 2 [25], the DL algorithm has proven successful. Employing a deep-learning algorithm should therefore constitute a strategy capable of enhancing our findings presented in the previous chapters.

The course of this chapter is as follows. Section 8.1 provides a very brief overview of deep learning. Section 8.2 then presents the background of CNNs. The experimental set-up for using CNN to automatically learn the features from a spectrogram is outlined in Section 8.3 along with the results of the experiment. In Section 8.4, we present a discussion regarding this method. In Section 8.5, we conclude with our answer to RQ4.

8.1 DEEP LEARNING

DL is a set of techniques that allows a system to automatically discover the needed feature representations from raw data. These networks are typically multiple layers deep, which is why they are called *deep learning*. The well-known progress of DL can be traced through Google Deep Mind's performance in Go, chess, and Shogi competitions over the last two years.

October to December 2015 was a memorable period for humankind as AlphaGo, which was developed by Google DeepMind, defeated the European Go champion Fan Hui in a five-game match without losing one game (5-0) (cf. Silver et al., 2016). It was a milestone for DL and marked the first time that a computer program could beat a human professional player in a Go game.

25 Dota 2 is a free online game which allows multiplayers attend in one battle. The game is the sequel of the Defense of the Ancients (DotA), which was originally from the Blizzard Entertainment's Warcraft III: The Frozen Throne.

This achievement had been anticipated by many researchers to take at least two decades, with the notable exception of Jaap van den Herik who predicted 2020 (see website Rehm, former Go Champion of the Netherlands). Only two months after the match, in March 2016, AlphaGo defeated Lee Sedol in a five-game Go match (4-1) (cf. Silver et al., 2016; C. Lee et al., 2016). That a DL computer program could beat the world's third ranked Go player marked a historic milestone. The game of Go was invented in ancient China nearly 2,000 years ago. It is currently believed to be the world's oldest and most complicated board game. Until Sedol's match, computer programs found it nearly impossible to beat a high-ranking player in a Go game, because of the inability of the computer to adequately evaluate an arbitrary position (Fullerton, 2014), an essential difference between Go and chess.

In December 2016, the mysterious chess player "Master" started to play in online Go games from a platform that was based in China. "Master" defeated top-ranked Chinese and foreign players in 60 successive games playing each day (three days in total) and achieved an unprecedented 60 games winning streak. Several famous professional Go players such as Gu Li and Chang Hao, were defeated. On January 4, 2017, the AlphaGo research team announced that "Master" is actually an upgraded version of AlphaGo (Silver et al., 2016).

Only several months after its initial win against the professional Go players, in May 2017, AlphaGo outmatched Ke Jie, the human world champion, in a three-game match (cf. Silver, Schrittwieser, et al., 2017). This was a milestone that evidenced DL's overwhelming capacity to play the game of Go.

However, the power of DL is not limited to Go. The use of artificial intelligence to challenge another artificial intelligence already occurred in 2017. Just several months after AlphaGo sensationally bested the top-ranked Go players, the successor AI program AlphaGo Zero destroyed the AlphaGo. In brief, we add that AlphaGo was trained by a training set of approximately 40 million play by grandmaster games (24 million human play and 16 million self play games). AlphaGo Zero did not have any prior knowledge except for the playing rules. It developed its own strategies and it was trained solely via self-play. (see Silver, Schrittwieser, et al., 2017). Ultimately AlphaGo Zero won the 100-game match against AlphaGo with 100 strikes (cf. Silver, Hubert, et al., 2017a).

After Alpha-Go Zero, the DeepMind team developed an Alpha Zero program for chess and Shogi (see Silver, Hubert, et al., 2017b). It achieved a superhuman level of playing chess and Shogi games within 24 hours of learning. In chess, Alpha Zero won against Stockfish, which was the world-champion among professionals (TCEC), with 28 wins, 72 draws, and zero losses. And in Shogi, the leading program Elmo was defeated by 92-8.

To sum up, over the last two decades, the DL approach has gained increasing attention from researchers due to the ability to process a massive amount of raw data. However, since DL was limited by the underlying hardware's computational ability, only in recent years has DL has the ability to become a mainstream method of developing an effective machine-learning algorithm.

The DL consists of a variety of architectures, such as deep neural networks (DNN), and recurrent neural networks (RNN). Furthermore, a variety of application fields have succeeded in utilizing the DL technology, and these include image recognition (cf. He, Zhang, Ren, & Sun, 2016), speech recognition (cf. Chan, Jaitly, Le, & Vinyals, 2016; Xiong et al., 2017), and automatic emotion recognition (cf. Neumann & Vu, 2017).

The initial concept of a DNN was inspired by studies on the biological functioning of the human brain. In terms of architecture, a DNN attempts to simulate the human brain's biological neural units, which are connected through axons (Preparata & Shamos, 2012). Each neural unit is connected to other units, and the links between them can be dynamically changed by the activated function. A DNN can structurally consist of many hidden layers between the input and output layers. Each hidden layer receives input from the previous layer and allows the composition of a feature to serve as the output to the next layer. In this way, a DNN shows its capacity to learn the high-level representative features from the raw data. Therefore, it can effectively classify data. With sufficient training on the data and appropriate training strategies, DNNs can achieve an excellent performance on certain recognition tasks (Ioffe & Szegedy, 2015).

8.2 CNN: CONVOLUTIONAL NEURAL NETWORKS

A special kind of multilayer perceptron, called a CNN, has led to breakthroughs on many recognition and prediction tasks. Convolutional neural networks were initially inspired by the visual system's structure, Hubel and Wiesel's early work found that a cat's visual system contains a complicated arrangement of cells (Hubel & Wiesel, 1968). These cells are sensitive to small subregions of the visual field, called a receptive field. The first model design was proposed by Fukushima based on the local connectivities of the transformation of an image (Fukushima, 1980). Fukushima discovered that a well-organized invariant output can be obtained by applying the same parameter setting of the neural units to the patch of the previous layer at any location. Researchers have been turning renewed attention to this early neural network due to two recent developments: (1) the large volume of training data that is currently available and (2) the increasing computational ability of hardware. In recent studies, CNNs used for pattern recognition systems have clearly demonstrated their advantages as regards handwritten character recognition (Poznanski & Wolf, 2016), a task that has served as a machine-learning benchmark for many years.

Speech emotion recognition (SER) is a processing method of predicting an emotional state based on the features of the speech signal cues produced by a speaker. Emotional expression is neither always explicit, nor is it always based on individual acoustic cues, and this problem further influences feature extraction. As a result, a high-performance algorithm generally requires a high-

quality feature extraction and selection from the feature group of the speech signal.

In the previous two chapters (Chapter 6 and 7), we have described how we manually devised features (log-Gabor filters) to capture the visual features of the spectrograms associated with speech emotions (see Gu, Postma, Lin, & Van den Herik, 2016b). Convolutional neural networks are known to be capable of inferring a hierarchical feature representation. Therefore, in this chapter, we employ a CNN to automatically detect and extract features from spectrograms for SER.

8.3 EXPERIMENT: FEATURES LEARNED FROM A CNN

This section describes an experiment in which we use CNN to extract features from spectrograms for SER. The aim of the experiment is to extract visual features for two purposes: (1) to determine whether a DL algorithm can extract the features from spectrogram to increase the performance of SER, and (2) to verify the findings outlined in Chapter 6 and 7.

The reminder of the section is organized as follows. Subsection 8.3.1 presents details concerning the set-up of the CNN experiment. Subsequently, Subsection 8.3.2 describes the evaluation procedure for this experiment. Finally, the results of the experiment are reported and analyzed in Subsection 8.3.3.

8.3.1 Experiment Set-up

Below, we present a CNN algorithm to investigate feature learning from a spectrogram for SER. The design experiments involve the CNN learning throughout the various procedures. The MatConvNet[26] toolbox is used to run the CNN. The CNN experiment set-up consists of three steps: (A) spectrogram generation, (B) transfer learning for Pre-trained CNN, and (C) CNN fine-tuning on MAS dataset. The details of each of these steps are outlined below.

A: Spectrogram generation

Each auditory signal (utterance) is transformed in a spectrogram using Matlab's spectral analysis function. It has a short-time Fourier transform with a 20 ms Hamming window and an overlap of half a window length.

B: Transfer learning for Pre-trained CNN

The use of pre-trained CNNs on a specific task, so-called transfer learning (Pan, Yang, et al., 2010), alleviates the use of large datasets. As we commonly know, a well-trained CNN requires a considerable number of datasets and resources. It will be difficult to train a whole CNN from

26 http://www.vlfeat.org/matconvnet/

the scratch because researchers are seldom able to obtain a significant amount of datasets. Therefore, to be practical, using transfer learning for object recognition can prevent the demand of large dataset and is highly recommended by researchers. The main condition for using pre-trained CNNs for transfer learning is that novel tasks resemble the task the CNN was pre-trained on (Sharif Razavian, Azizpour, Sullivan, & Carlsson, 2014).

The ImageNet-VGG is chosen as the pre-trained CNN model to be used in our experiment (Chatfield, Simonyan, Vedaldi, & Zisserman, 2014). ImageNet is one of the favorite datasets which consist of 1.2 million images associated with 1000 different categories. ImageNet-VGG is well-trained CNN for classifying natural images. However, our task is completely different from what ImageNet-VGG is usually used for. Apart from the natural images, we have spectrograms for different training objectives. The question that is evoked is why we would expect to use transfer learning through ImageNet-VGG to work for our task?

The motivation for us to do this is that, in trained CNNs the features extracted by the later layers become progressively more specific to the classes which greatly depend on the original dataset and task. But features which are learned from the first (earlier) layer contains more generic features, which we call *generic features*, could be useful for a variety of recognition tasks. Therefore, by finding features extracted by these first layers, it is possible for us to retrain the classifier on a new dataset (i.e., MAS) by fine-tuning the higher layers in the network and "freeze" the learning in the earlier layers. The pre-trained CNN also shows an interesting phenomenon: the features which are learned by first layer resemble Gabor filters. Interestingly, in our previous Chapter 6 and 7, we applied the log-Gabor filters as the feature extractor.

The architecture of the CNN model which we use in this experiment is a multi-layer CNN model. There are two types of layers at the heart of the model: convolutional layers and fully connected layers (Zheng, Yu, & Zou, 2015). Each layer has a topographic structure, and each neuron is associated with a fixed two dimensional position that corresponds to a location in the input image. The architecture of CNN consists of eight layers, in which five layers are convolutional layers and the last three layers are fully connected layers. The convolutional layers are used to detect the feature pattern from the spectrogram. In each convolution layer, the filters make the convolution over the input space and are well-suited for exploiting the strong spatially local correlation present in spectrogram. The architecture comprised of several convolutional layers ensures that the learned “filters” produce the strongest response to a spatially local input pattern. Another crucial CNN element is the pooling layers, which are a form of down-sampling. The earlier fully connected layers

are regularized using dropout. The last fully connected layer is used as a classifier.

The hyperparameters for CNN training are used as follows. Table 8.1 lists the setting of relevant CNN parameter values that are used in the experiment. In this model, stochastic gradient descent (SGD) is used as the learning algorithm. The convolution filters are 5×5 for the first convolutional layers and 3×3 for the following four convolution layers. The value of stride and the padding which we choose to use is 2 and 1, respectively. The weight decay is 0.0005. It is better to use a smaller initial learning rate due to control the over-fitting. Thus, the learning rate is initialized at 0.01 for the layers that are being fine-tuned. And a step decay schedule for the learning rate schedule has been chosen to use in our experiment. The learning rate is decreased by every 20 epochs (from 0.01, 0.001,... to 0.000001). A drop-out layer is inserted after the first two fully-connected layers and the drop out ratio is 50%.

Table 8.1: Overview of the parameter values of our CNN

Name of parameter	Value of parameter
Momentum	0.9
Initial Learning Rate	0.01
NumEpochs	100
WeightDecay	0.0005

C: CNN fine-tuning on MAS dataset

For the fine-tuning of the pre-trained CNN, MAS dataset is prepared and separated into a training set, a test set, and a validation set. Thus, we set up our dataset with a training, a validation and a test directory in this manner [27]. The last layers are retrained in the fine-tuning. The output dimensionality of the last layer equals to the number of classes, i.e., 5. The weights of a zero mean and 0.01 variance are initialized by a Gaussian distribution function. We used softmax at the output layer and used the cross-entropy loss function. No data augmentation is applied in this experiment.

8.3.2 Evaluation Procedure

To determine a CNN's SER performance, we evaluate the classification performance (see Definition 5.3) by comparing the following algorithms based on:

27 In Matlab, we need to give each state of emotion spectrogram for their corresponding index labels from 1 to 5, and the created dataset will be saved as 'imdb' file for further use.

(a) acoustic features, (b) primary log-Gabor filter pairs, (c) primary and subsequent log-Gabor filter pairs, and (d) the CNN. In this experiment, we train and evaluate all four algorithms on the MAS corpus (see Section 4.2).

A: Acoustic features

The acoustic features are extracted via the same procedures introduced in Chapter 6. All the acoustic features used in our experiment correspond to the baseline features of the Interspeech challenge (Schuller et al., 2013). This feature-set is the current state-of-the-art in SER, and facilitates a standard comparison.

B: Primary log-Gabor filter pairs

The primary log-Gabor filter features are obtained as described in Chapter 6.

C: Primary and subsequent log-Gabor filter pairs

The procedure used for the primary and subsequent log-Gabor filter features is as discussed in Chapter 7. This approach's performance provides a criterion by which to evaluate whether the CNN has utility in terms of improving the SER performance.

D: The pre-trained CNN

For the CNN, the dataset is separated into a training set, a test set, and a validation set to ensure that the models are not overtrained to fit the affective styles of a particular speaker. We also use cross-validation as in all previous chapters. The training set is split manually into 10 fold, and a Imdb file is created for each fold. The average classification performance of the CNN learned feature is evaluated by comparing with the 'state-of-the-art' acoustic feature algorithm (Schuller et al., 2015), and the primary log-Gabor filter pairs and subsequent log-Gabor filter pairs (see Chapters 6 and 7).

Table 8.2: The training, validation and text set in number of recording

	Number of recordings
Training set	10,200
Validation set	5,100
Test set	5,100

8.3.3 Results of the CNN Experiment

Table 8.3 provides the classification performances using the feature sets: (a) acoustic features, (b) primary log-Gabor filter pairs, (c) primary and subse-

quent log-Gabor filter pairs and (d) CNN learned features. The numbers in the table represent the recognition accuracy rates for the MAS corpus. As shown in Table 8.3, the left column is the algorithm's name, the right column is the performance rate obtained. The algorithms' performance rates will be discussed from the weakest to the strongest as follows.

Table 8.3: CNN classification performance on the MAS database

Performance Metric (MAS)	Performance Rate (SD)
Acoustic features	91.7 (1.2)
Primary Gabor filter pairs	92.2 (1.4)
Primary and Subsequent Gabor filter pairs	93.8 (0.9)
CNN learned features	94.6 (0.9)

The performance (91.8%) of the acoustic feature algorithm proposed by Schuller is currently considered as the state-of-the-art (Schuller et al., 2015). The acoustic feature algorithm results in the weakest performance (91.8%). The possible reasons are (1) even though there are a huge number of the types of acoustic features, they still cannot fully cover all the useful information about emotion, as well as (2) overlapping of acoustic features may cause redundant information and hence affect the performance. The performance of log-Gabor filter pairs is based on the previous work in Chapter 6 on using the primary Gabor filter pairs. The algorithm using primary log-Gabor filter pairs alone achieves a slightly higher accuracy of 92.2%. The combination of primary and subsequent filter algorithms as proposed in Chapter 7 obtains the second highest performance rate of 93.8%. The highest performance rate of 94.6% is obtained by the CNN.

As demonstrated above, both within the combination and alone primary log-Gabor filter pairs are better at improving accuracy than acoustic features but still not as good as the CNN. Log-Gabor filter pairs perform better than acoustic features in that they are manually designed and can effectively extract the emotional expression patterns from a spectrogram. Yet, log-Gabor filter pairs may "overlook" some emotional expressions in spectrogram. The CNN offers the best performance due to the following two reasons.

(1) The CNN has a high ability to learn high-level features which are very useful for classification. The CNN performs feature learning by building a local connectivity pattern between neurons of adjacent layers. The input layer can extract the low-level features from spectrogram, such as edges and curves of emotional expression patterns. With the addition of deeper layers, the low-level features can be combined into high-level features. For example, the first layer can work as filter to extract the simplest features from spectrogram. After that, the subsets of units in the

first layer can be combined into an input of the hidden units in the second layer which represent more complicated features. These second-layer units can further generate the hidden units in the third layer.

(2) Compared with acoustic features or log-Gabor filter pairs, the CNN does not need to do plenty of feature construction work, such as the preliminary design of the characteristics of features, the labeling of features and decision on the number of features. In the case of the CNN, the entire dataset can be directly imported for feature learning.

The CNN algorithm achieves a better performance than those yielded by other algorithms, despite offering only a slight advantage in terms of accuracy. If we could explore our CNN algorithm by increasing the amount of training data or deep neural network layers, we would likely obtain more promising results. That said, the current achievement is still quite beneficial in terms of demonstrating DL's contribution to SER. And also this confirms our previous findings in Chapter 6 that the primary and less-intensive patterns in spectrograms are useful features for SER.

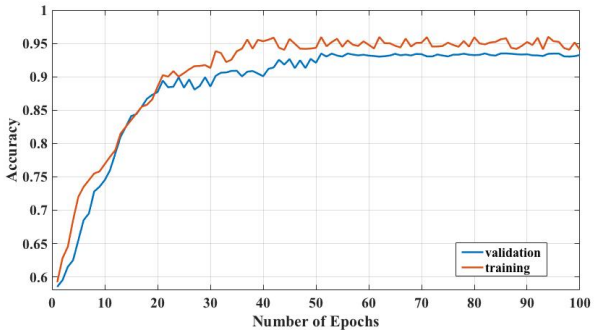


Figure 8.1: The performance of CNN training process on the MAS dataset.

In the CNN experiment, we separate the MAS dataset into a training set, a test set, and a validation set to ensure that the models are not overtrained. Half of original dataset is used as training data, the rest of dataset is set equal as validation and test data. The performance of CNN training is shown in Figure 8.1. As illustrated in Figure, the orange line indicates the performance on the training data and the blue line shows the performance on the validation data. After 100 epochs, the accuracy can be achieved at 95%, and performance on the validation data shows it is bit over-fitting.

Table 8.4 offers details on the recognition performances for the five types of emotions. Compared with Table 7.7, we can find there are some improvements. The CNN can obtain 94.0% and 93.8% on *angry* and *happy*, 1.2% and 0.9% higher than log-Gabor filter, respectively. The emotional expression patterns for *angry* and *happy* are both complex and can be easily mistaken for each other. However, the CNN has the ability to learn the feature differences between the two emotions to better recognize them.

Table 8.4: Confusion table of the CNN spectrogram features

	Performance (%)				
Emotion	Angry	Happy	Neutral	Panic	Sad
Angry	94.9%	1.6%	1.2%	1.4%	0.9%
Happy	2.2%	93.8%	1.6%	1.8%	1.2%
Neutral	1.3%	1.1%	95.9%	1.0%	0.9%
Panic	1.6%	1.3%	1.1%	94.0%	2.0%
Sad	1.1%	1.1%	1.2%	2.1%	94.4%

In regard to CNN's performances for the five emotions, the accuracy is higher for the *neutral* (95.9%) than for the other emotions, which is consistent with our previous finding through the log-Gabor filter. As the emotional expression pattern for the neutral emotion is unique and highly recognizable, the CNN can efficiently extract this pattern through the feature learning process.

8.4 CHAPTER DISCUSSION

In this chapter, we investigated the use of deep learning in extracting features from a spectrogram. To evaluate the CNN algorithm, we designed an experiment for the CNN, and compared the CNN with (1) the acoustic feature algorithm, (2) primary log-Gabor filter pairs algorithm and (3) primary and subsequent log-Gabor filter pairs algorithm. The results of this experiment reveal that the CNN offers improvements over all three alternative algorithms. CNN demonstrates its value also in terms of boosting the recognition performance. Thus, the CNN improves on existing algorithms for learning features from spectrograms. This finding also underlines our previous findings, which we manually achieved after designing log-Gabor filter pairs to extract feature patterns from spectrograms.

8.5 ANSWER TO RESEARCH QUESTION FOUR

This chapter targeted RQ4, which reads as follows.

RQ4: *Can we apply the deep-learning method to the spectrogram outcomes to extract "visual" features to increase the accuracy of SER?*

To answer this question, we used the CNN algorithm for SER in this chapter. To investigate the effectiveness of the CNN-learned features, we examined

and evaluated the CNN-learned features, we likewise employed the acoustic feature algorithm and two log-Gabor filter algorithms that we proposed in Chapters 6 and 7.

Their performances were evaluated for the MAS database. The results reveal that the CNN-learned features perform the best among all the algorithms. Based on these findings, we may conclude that the CNN is able to learn important features from spectrograms. Moreover, it demonstrates its contribution by increasing the accuracy of SER. Therefore, the answer to RQ4 is as follows: The CNN can successfully extract features from a spectrogram and improve on the conventional SER performance.

The current disadvantage of the CNN is that they require a very lengthy computing period to train data from very large datasets. This is an obstacle that could be overcome in future work by using a more efficient parallel computing algorithm on large Graphics processing unit (GPU) platforms to meet the capacity requirements of computing systems.

9 CONCLUSIONS AND FUTURE WORK

In this chapter, our conclusions are presented in three parts. First, in Section 9.1, we summarize the results of this thesis by referring to the research questions (RQs) that were formulated in Chapter 1. We offer our answers to each question. Secondly, in Section 9.2, we provide overall conclusions based on the answers to the four RQs. Section 9.3 addresses the problem statement. Finally, in Section 9.4, we will offer six recommendations for future research.

9.1 ANSWERS TO THE RESEARCH QUESTIONS

The main topic of this study has been SER. Cutting across the speech emotions, we have explored four algorithms for using SER to distinguish emotional states. The four algorithms focused on spectrograms containing visual information for feature construction and feature learning, which enabled us to detect and distinguish an emotional state from a speech signal. The research was guided by four RQs. Below we summarize the answers to each of them.

RQ1: *Is it possible to design a new algorithm that improves the accuracy of detecting the voiced part activity in speech?*

RQ1 is divided into two sub-questions, RQ 1A and RQ 1B. They read as follows.

RQ 1A: *What characteristics should a new algorithm possess for being more accurately in detecting voiced part activity in speech?*

RQ 1B: *What is the gain in SER performance provided by the new voice detection algorithm as compared to the original algorithms?*

The RQ 1 was investigated in Chapter 5. RQ 1A and RQ 1B are also addressed in Chapter 5. The brief view is as follows. To answer RQ 1A we proposed a new algorithm, which we called voiced segment selection (VSS). We reviewed various existing studies on VAD, and on the basis of our literature review, we identified the state-of-the-art algorithm for use in comparisons. The VSS algorithm uses log-Gabor filters to extract the features from a spectrogram. Our assessment of the use of Gabor filters to extract visual spectro-temporal features from the speech spectrogram revealed the feasibility of an idea originally proposed by Ezzat, Bouvrie, and Poggio (2007). We ran computational experiments on the VSS algorithm using the MAS database. The results of

the experiment demonstrate that VAD performance improved relative to existing prevalent algorithms. The results obtained using the VSS algorithm are similar to those achieved by the deep-learning algorithm, but the former are associated with a (much) lower computational cost. Regarding RQ 1B, we determined that when the VSS algorithm is used, the classification rate is higher in the Chinese corpus than when VSS is not used before feature extraction. The results indicate that using VSS for SER is a good answer for RQ 1B and a useful complement to current SER methods. Thus, in summary, the answer to RQ1 is that the VSS algorithm can be a useful and practical algorithm for VAD.

RQ2: *How can we use two-dimensional features to analyze the spectrogram representation of speech?*

The second RQ is addressed in Chapter 6. Inspired by Ezzat et al. (2007), we concentrated on the parts of a sentence that demonstrate intensive expressions of emotions in a spectrogram. We used two-dimensional Gabor filter pairs to detect and extract the feature patterns of the emotional information from a spectrogram. The feature patterns were new features that complemented the feature group. Employing these tuned Gabor filter pairs to perform a second-order analysis of the spectrogram produced satisfactory results. We believe that there are two reasons for this strong performance. First, we performed manual feature selection by inspecting numerous spectrograms of emotional speech. This led to a limited set of features, which prevented dimensionality by capturing task-relevant information from the spectrogram. Second, the Gabor filters detected spectrogram characteristics that were relevant for the task at hand. The orientation patterns in the spectrogram reflect the time-varying frequency compositions of the utterances associated with each emotion. The orientation tuning of Gabor filters is highly suitable for encoding these patterns.

RQ3: *Can we extract additional, likely less-intensive features via the composition of Gabor filters through a spectrogram?*

Chapter 7 investigated the RQ3. To answer RQ3, we further categorized emotional expressions in a sentence into primary and less-intensive feature patterns according to their intensiveness. Moreover, we revealed the feature patterns of the subsequent emotion expressions in a spectrogram. In Chapter 6, we proposed primary (tuned) log-Gabor filter pairs for the primary feature patterns. Chapter 7 subsequently served as an extension of Chapter 6, by further developing subsequent log-Gabor filters that were specific to the less-intensive feature patterns in the spectrogram. We evaluated the effectiveness of those subsequent log-Gabor filter features for SER. For this evaluation, we compared the performances of five types of feature-based algorithms. A combination of

primary and subsequent feature patterns proved to be the most effective algorithm. This combination outperformed the state-of-the-art algorithms. We can therefore conclude that these subsequent log-Gabor filters are a useful complement to primary features for SER. The research further revealed that the combination of the two approaches is a successful follow up.

RQ4: *Can we apply the deep-learning method to the spectrogram outcomes to extract "visual" features to increase the accuracy of SER?*

The answer to the RQ4 is derived from the research discussed in Chapter 8. In that chapter, we described an experiment that used a deep-learning algorithm to extract features from a spectrogram. In the experiment, a convoluted neural network (CNN) was able to automatically learn the features from the spectrogram. In total, 16 layers were designed for the CNN's structure (13 convolutional layers, and 3 pooling layers). The evaluation consisted of comparing four algorithms: (1) the state-of-the-art acoustic feature algorithm, (2) the primary log-Gabor filter algorithm proposed in Chapter 6, (3) the combination of primary and subsequent log-Gabor filters algorithm, and (4) the CNN algorithm. The CNN algorithm yielded the best performance. Hence, the answer to RQ4 is as follows: (1) A CNN can extract features from a spectrogram and improve the SER performance, and (2) the CNN's performance surpassed that achieved by our previous algorithm of manually using a log-Gabor filter for feature construction from a spectrogram. Thus, the CNN supported our previous findings, and demonstrated that automatic feature learning should be the main algorithm used in future work.

9.2 CONCLUSION BASED ON THE RESEARCH QUESTIONS

From the answers to the four RQs we may conclude that using a spectrogram for VAD and SER yields a meaningful improvement in the SER performance. Moreover, we identified three key achievements. Below, we specify the achievements of this study in three pillars.

(1) A combination of spectrograms and log-Gabor filters can improve the performance of VAD. Moreover, the accuracy of SER can be enhanced by using the VSS algorithm as a pre-processing algorithm.

(2) The primary and subsequent log-Gabor filter pairs can be assessed by the primary and less-intensive feature patterns in a spectrogram. Both primary and subsequent features in a spectrogram are crucial features that can contribute to SER.

(3) The CNN is highly capable of learning features from a spectrogram. These features learned by CNN from a spectrogram are valuable for enhancing SER performance. They also supported our earlier findings regarding feature construction using spectrograms and log-Gabor filters.

9.3 RESPONDING TO THE PROBLEM STATEMENT

In this section, we respond to the problem statement. Our answer is based on the overall results reported in the detailed answers to our research questions.

In Chapter 1, we outlined the importance of SER, which can establish an interaction between humans and computer software (applications). We stated that speech emotion recognition may have important applications in daily life, and that SER applications therefore required a strong ability to recognize the emotions expressed by a person. However, due to issues with limited accuracy, the performance of SER required further improvements, which brought us to our problem statement.

Problem Statement: To what extent can we further improve SER accuracy using spectrogram information?

Looking back to our problem statement, the findings presented in this dissertation can serve to prove that a combination of spectrogram and log-Gabor filter is effective in improving the outcome of feature extraction.

Overall, the insights of this dissertation show a new direction for feature extraction that has become a crucial part of SER. Thus, in the future, SER might be as important as other things in daily life. We may further conclude that using spectrograms and log-Gabor filters in this thesis have enhanced the performance of SER and provided related advantages assumed to be used in the future.

9.4 FUTURE RESEARCH

The ultimate goal of this research is to enhance the performance of emotion recognition in human speech. The results of our experiments offer convincing support for our proposed algorithms. However, there are still additional topics that need attention in future research on SER. In this section, we provide six recommendations for future research.

1. In this experiment, we did not consider the width of the energy bands. In addition, we used a range of values for the spatial frequency in the log-Gabor filters. Choosing a precise value for the width of the energy bands, and potentially improving the tuning in terms of the spatial frequency, may lead to even more accurate results. This hypothesis is recommended to be considered in future research.

2. As explained in Section 4.2, laboratory-collected emotion databases are most suitable for use in research and represent the majority of existing databases. The experiments in this study were conducted using a database of voice-acted data. It would therefore be valuable to test our four algorithms with other neutral environment speech databases in future research (Ringeval et al., 2013).

3. In Chapter 1, we introduced two types of information transmitted in a speech conversation: verbal and non-verbal information. We only focused on non-verbal transmission in this study. However, verbal transmission is typically discourse-related and contains the context and situation of a dialogue. Therefore, verbal information is indispensable for improving our understanding of human emotion. Future researchers will need to search for a single method capable of processing both verbal and non-verbal segments of speech for SER.

4. Emotion expression is conveyed by not only speech expression but also non-verbal means, such as facial expressions, speech recognition, and body language, as reviewed in Chapter 1. Emotion recognition based on facial expression has been extensively studied in previous computer science studies. The emotion recognition process in this study used speech as its only modality function. In future research, more attention should be paid to the less-researched combination of audio and video signals to achieve a more accurate recognition of emotional states.

5. In Chapter 8, we used the Convolutional Neural Network algorithm as the deep-learning mechanism for our research. However, the CNN requires a rather long computing time to train data from very large datasets. Therefore, for future research, a more efficient parallel computing algorithm on large GPU platforms should be used to meet the capacity requirements of computing systems.

6. This research was limited by the MAS corpus, as only five types of emotional states were collected and available for use in our experiments. It would be useful to create a database with a broader range of emotional states, as certain applications may require a more nuanced emotional classification than that offered by the five fundamental states. Therefore, future research should expand this scope and create a database with more emotional states for use in experiments.

REFERENCES

Anagnostopoulos, C.-N., Iliou, T., & Giannoukos, I. (2015). Features and classifiers for emotion recognition from speech: a survey from 2000 to 2011. *Artificial Intelligence Review*, *43*(2), 155–177.

Apple, W., Streeter, L. A., & Krauss, R. M. (1979). Effects of pitch and speech rate on personal attributions. *Journal of Personality and Social Psychology*, *37*(5), 715.

Bach, J.-H., Kollmeier, B., & Anemuller, J. (2010). Modulation-based detection of speech in real background noise: Generalization to novel background classes. In *Acoustics speech and signal processing (icassp), 2010 IEEE international conference on* (pp. 41–44).

Bachu, R., Kopparthi, S., Adapa, B., & Barkana, B. (2008). Voiced/unvoiced decision for speech signals based on zero-crossing rate and energy. In K. M. Elleithy (Ed.), *Scss (2)* (p. 279-282). Springer.

Bänziger, T., & Scherer, K. R. (2005). The role of intonation in emotional expressions. *Speech Communication*, *46*(3), 252–267.

Belin, P., Zatorre, R. J., & Ahad, P. (2002). Human temporal-lobe response to vocal sounds. *Cognitive Brain Research*, *13*(1), 17–26.

Benyassine, A., Shlomot, E., Su, H.-y., & Yuen, E. (1997). A robust low complexity voice activity detection algorithm for speech communication systems. In *Speech coding for telecommunications proceedings, 1997, 1997 IEEE workshop on* (pp. 97–98).

Bermingham, M. L., Pong-Wong, R., Spiliopoulou, A., Hayward, C., Rudan, I., Campbell, H., ... others (2015). Application of high-dimensional feature selection: evaluation for genomic prediction in man. *Scientific Reports*, *5*, 10312.

Bouvrie, J., Ezzat, T., & Poggio, T. (2008). Localized spectro-temporal cepstral analysis of speech. In *Acoustics, speech and signal processing, 2008. icassp 2008. IEEE international conference on* (pp. 4733–4736).

Breitenstein, C., Lancker, D. V., & Daum, I. (2001). The contribution of speech rate and pitch variation to the perception of vocal emotions in a german and an american sample. *Cognition & Emotion*, *15*(1), 57–79.

Buck, R. W. (2000). The epistemology of reason and affect. *The neuropsychology of emotion*, 31–55.

Buisman, H., & Postma, E. O. (2012). The log-gabor method: speech classification using spectrogram image analysis. In *Interspeech* (pp. 518–521).

Burnett, G. C. (2007, July 17). *Detecting voiced and unvoiced speech using both acoustic and nonacoustic sensors.* Google Patents. (US Patent 7,246,058)

Busso, C., Lee, S., & Narayanan, S. (2009). Analysis of emotionally salient aspects of fundamental frequency for emotion detection. *Audio, Speech, and Language Processing, IEEE Transactions on*, *17*(4), 582–596.

Cabanac, M. (2002). What is emotion? *Behavioural Processes*, *60*(2), 69–83.

Cen, C.-Q., Liang, Y.-Y., Chen, Q.-R., Chen, K.-Y., Deng, H.-Z., Chen, B.-Y., & Zou, X.-B. (2017). Investigating the validation of the chinese mandarin version of the social responsiveness scale in a mainland china child population. *BMC psychiatry*, *17*(1), 51.

Chan, W., Jaitly, N., Le, Q., & Vinyals, O. (2016). Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. In *Acoustics, speech and signal processing (icassp), 2016 IEEE international conference on* (pp. 4960–4964).

Chang, C.-C., & Lin, C.-J. (2011). Libsvm: a library for support vector machines. *ACM transactions on intelligent systems and technology (TIST)*, *2*(3), 27.

Chastagnol, C., & Devillers, L. (2012). Personality traits detection using a parallelized modified sffs algorithm. *Computing*, *15*, 16.

Chatfield, K., Simonyan, K., Vedaldi, A., & Zisserman, A. (2014). Return of the devil in the details: Delving deep into convolutional nets. In *British machine vision conference.*

Chaudhari, S., & Kagalkar, R. (2014). A review of automatic speaker recognization and identifying speaker emotion using voice signal. *International Journal of Science and Research (IJSR)*, *3*(11).

Chen, Mao, X., Xue, Y., & Cheng, L. (2012). Speech emotion recognition: Features and classification models. *Digital Signal Processing*, 1154-1160.

Chen, L., Mao, X., Xue, Y., & Cheng, L. (2012). Speech emotion recognition: Features and classification models. *Digital Signal Processing*, *22*(6), 1154-1160.

Chen, S.-H., Chang, Y., & Truong, T.-K. (2007). An improved voice activity detection algorithm for gsm adaptive multi-rate speech codec based on wavelet

and support vector machine. In H. G. Okuno & M. Ali (Eds.), *Iea/aie* (Vol. 4570, p. 915-924). Springer.

Collobert, R., Puhrsch, C., & Synnaeve, G. (2016). Wav2letter: an end-to-end convnet-based speech recognition system. *arXiv preprint arXiv:1609.03193*.

Dahake, P. P., Shaw, K., & Malathi, P. (2016). Speaker dependent speech emotion recognition using mfcc and support vector machine. In *Automatic control and dynamic optimization techniques (icacdot), international conference on* (pp. 1080–1084).

Darwin, C. (1872). The expression of the emotions in man and animals, New York. *Appleton and Company*.

Davis, A., Nordholm, S., & Togneri, R. (2006). Statistical voice activity detection using low-variance spectrum estimation and an adaptive threshold. *Audio, Speech, and Language Processing, IEEE Transactions on*, *14*(2), 412–424.

Davitz, J. R. (1964). Auditory correlates of vocal expression of emotional feeling. *The communication of emotional meaning*, 101–112.

Dechter, R. (1986). *Learning while searching in constraint-satisfaction problems*. University of California, Computer Science Department, Cognitive Systems Laboratory.

Ekman, P., Sorenson, E. R., & Friesen, W. V. (1969). Pan-cultural elements in facial displays of emotion. *Science*, *164*(3875), 86–88.

El Ayadi, M., Kamel, M. S., & Karray, F. (2011). Survey on speech emotion recognition: Features, classification schemes, and databases. *Pattern Recognition*, *44*(3), 572–587.

Eyben, F., Weninger, F., Squartini, S., & Schuller, B. (2013). Real-life voice activity detection with lstm recurrent neural networks and an application to hollywood movies. In *Acoustics, speech and signal processing (icassp), 2013 IEEE international conference on* (pp. 483–487).

Ezzat, T., Bouvrie, J. V., & Poggio, T. (2007). Spectro-temporal analysis of speech using 2-d gabor filters. In *Interspeech* (pp. 506–509).

Fayek, H. M., Lech, M., & Cavedon, L. (2016). Modeling subjectiveness in emotion recognition with deep neural networks: Ensembles vs soft labels. In *Neural networks (IJCNN), 2016 international joint conference on* (pp. 566–570).

Fayek, H. M., Lech, M., & Cavedon, L. (2017). Evaluating deep learning architectures for speech emotion recognition. *Neural Networks*, *92*, 60–68.

Forsell, M. (2007). *Acoustic correlates of perceived emotions in speech*. Numerisk analys och datalogi, Kungliga Tekniska högskolan.

Frijda, N. H., Kuipers, P., & Ter Schure, E. (1989). Relations among emotion, appraisal, and emotional action readiness. *Journal of personality and social psychology*, *57*(2), 212.

Fukushima, K. (1980). Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, *36*(4), 193–202.

Fullerton, T. (2014). *Game design workshop: a playcentric approach to creating innovative games*. CRC press.

Gabor, D. (1946). Theory of communication. part 1: The analysis of information. *Journal of the Institution of Electrical Engineers-Part III: Radio and Communication Engineering*, *93*(26), 429–441.

Gendron, M., & Barrett, L. F. (2009). Reconstructing the past: A century of ideas about emotion in psychology. *Emotion Review*, *1*(4), 316–339.

Germain, F. G., Sun, D. L., & Mysore, G. J. (2013). Speaker and noise independent voice activity detection. In *Interspeech* (pp. 732–736).

Gong, X., & Cortese, C. (2017). A socialist market economy with chinese characteristics: The accounting annual report of china mobile. In *Accounting forum*.

Goudbeek, M., & Scherer, K. (2010). Beyond arousal: Valence and potency/-control cues in the vocal expression of emotion. *The Journal of the Acoustical Society of America*, *128*(3), 1322–1336.

Gu, Y., Postma, E., & Lin, H.-X. (2015). Vocal emotion recognition with log-gabor filters. In *Proceedings of the 5th international workshop on audio/visual emotion challenge* (pp. 25–31).

Gu, Y., Postma, E. O., Lin, H.-X., & Van den Herik, H. J. (2016a). Speech emotion recognition using voiced segment selection algorithm. In *Proceedings of 22nd European Conference on Artificial Intelligence ECAI 2016* (pp. 1682–1683).

Gu, Y., Postma, E. O., Lin, H.-X., & Van den Herik, H. J. (2016b). Speech emotion recognition with log-gabor filters. In *Proceedings of International Conference on Agents and Artificial Intelligence, ICAART (2016)* (pp. 446–452).

Gupta, O., Raviv, D., & Raskar, R. (2018). Illumination invariants in deep video expression recognition. *Pattern Recognition*, *76*, 25–35.

Hammerschmidt, K., & Jürgens, U. (2007). Acoustical correlates of affective prosody. *Journal of Voice*, *21*(5), 531–540.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770–778).

Heilman, K. M., & Gilmore, R. L. (1998). Cortical influences in emotion. *Journal of Clinical Neurophysiology*, *15*(5), 409–423.

Hinton, G., Deng, L., Yu, D., Dahl, G. E., Mohamed, A.-r., Jaitly, N., ... others (2012). Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal processing magazine*, *29*(6), 82–97.

Hubel, D. H., & Wiesel, T. N. (1968). Receptive fields and functional architecture of monkey striate cortex. *The Journal of physiology*, *195*(1), 215–243.

Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International conference on machine learning* (pp. 448–456).

Izard, C. E. (1992). Basic emotions, relations among emotions, and emotion-cognition relations. *Psychological Review*, *99*(3), 561–565.

Izard, C. E. (2013). *Human emotions*. Springer Science & Business Media.

James, W. (1884). What is an emotion? *Mind*, *9*(34), 188–205.

Jin, Q., Li, C., Chen, S., & Wu, H. (2015). Speech emotion recognition with acoustic and lexical features. In *Acoustics, speech and signal processing (icassp), 2015 IEEE international conference on* (pp. 4749–4753).

Joseph, S. M., & Babu, A. P. (2016). Wavelet energy based voice activity detection and adaptive thresholding for efficient speech coding. *International Journal of Speech Technology*, *19*(3), 537–550.

Jürgens, R., Hammerschmidt, K., & Fischer, J. (2011). Authentic and play-acted vocal emotion expressions reveal acoustic differences. *Frontiers in psychology*, *2*, 180–182.

Ketai, R. (1975). Affect, mood, emotion, and feeling: semantic considerations. *The American Journal of Psychiatry*, *132*, 1215-1217.

Kiktova-Vozarikova, E., Juhar, J., & Cizmar, A. (2015). Feature selection for acoustic events detection. *Multimedia Tools and Applications*, *74*(12), 4213–4233.

Koolagudi, S. G., & Rao, K. S. (2012). Emotion recognition from speech: a review. *International journal of speech technology*, *15*(2), 99–117.

Kotani, K., Yoshimi, T., Nanjo, H., & Isahara, H. (2016). A corpus of writing, pronunciation, reading, and listening by learners of english as a foreign language. *English Language Teaching*, *9*(9), 139.

Kramer, E. (1964). Elimination of verbal cues in judgments of emotion from voice. *The Journal of Abnormal and Social Psychology*, *68*(4), 390.

Kreiman, J., & Sidtis, D. (2011). *Foundations of voice studies: An interdisciplinary approach to voice production and perception*. John Wiley & Sons.

Le, D., & Provost, E. M. (2013). Emotion recognition from spontaneous speech using hidden markov models with deep belief networks. In *Automatic speech recognition and understanding (asru), 2013 IEEE workshop on* (pp. 216–221).

Lee, C., Mower, E., Busso, C., Lee, S., & Narayanan, S. (2011). Emotion recognition using a hierarchical binary decision tree approach. *Speech Communication*, *53*(9-10), 1162–1171.

Lee, C., Wang, M.-H., Yen, S.-J., Wei, T.-H., Wu, I.-C., Chou, P.-C., ... Yan, T.-H. (2016). Human vs. computer go: Review and prospect [discussion forum]. *IEEE Computational Intelligence Magazine*, *11*(3), 67–72.

Lee, J., & Tashev, I. (2015). High-level feature representation using recurrent neural network for speech emotion recognition. In *Interspeech* (pp. 1537–1540).

Li, X., & Akagi, M. (2016). Automatic speech emotion recognition in chinese using a three-layered model in dimensional approach. In *2016 risp international workshop on nonlinear circuits, communications and signal processing (ncsp'16).*

Liu, H., & Motoda, H. (2012). *Feature selection for knowledge discovery and data mining* (Vol. 454). Springer Science & Business Media.

Longé, M. R., Eyraud, R., & Hullfish, K. C. (2017, October 10). *Multimodal disambiguation of speech recognition.* Google Patents. (US Patent 9,786,273)

Low, L.-S. A., Maddage, N. C., Lech, M., & Allen, N. (2009). Mel frequency cepstral feature and gaussian mixtures for modeling clinical depression in adolescents. In *Cognitive informatics, 2009. icci'09. 8th IEEE international conference on* (pp. 346–350).

Luneski, A., Konstantinidis, E., & Bamidis, P. (2010). Affective medicine: a review of affective computing efforts in medical informatics. *Methods of information in medicine*, *49*(3), 207–218.

Mao, Q., Dong, M., Huang, Z., & Zhan, Y. (2014). Learning salient features for speech emotion recognition using convolutional neural networks. *IEEE Transactions on Multimedia*, *16*(8), 2203–2213.

Mencattini, A., Martinelli, E., Costantini, G., Todisco, M., Basile, B., Bozzali, M., & Di Natale, C. (2014). Speech emotion recognition using amplitude modulation parameters and a combined feature selection procedure. *Knowledge-Based Systems*, *63*, 68–81.

Meyer, B. T., & Kollmeier, B. (2011). Robustness of spectro-temporal features against intrinsic and extrinsic variations in automatic speech recognition. *Speech Communication*, *53*(5), 753–767.

Mirsamadi, S., Barsoum, E., & Zhang, C. (n.d.). Automatic speech emotion recognition using recurrent neural networks with local attention. In *Acoustics, speech and signal processing (icassp), 2017 IEEE International Conference on, pages=2227–2231, year=2017, organization=IEEE.*

Myers, D. G. (2004). Theories of emotion. *Psychology: Seventh Edition, New York, NY: Worth Publishers*, *500*.

Nan, S., Sun, L., Chen, B., Lin, Z., & Toh, K.-A. (2017). Density-dependent quantized least squares support vector machine for large data sets. *IEEE transactions on neural networks and learning systems*, *28*(1), 94–106.

Neumann, M., & Vu, N. T. (2017). Attentive convolutional neural network based speech emotion recognition: A study on the impact of input features, signal length, and acted speech. *arXiv preprint arXiv:1706.00612*.

Nygaard, L. C., & Queen, J. S. (2008). Communicating emotion: Linking affective prosody and word meaning. *Journal of Experimental Psychology: Human Perception and Performance*, *34*(4), 1017.

Ortony, A., Clore, G. L., & Collins, A. (1990). *The cognitive structure of emotions*. Cambridge university press.

Pan, S. J., Yang, Q., et al. (2010). A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, *22*(10), 1345–1359.

Pao, T.-L., Chien, C. S., Chen, Y.-T., Yeh, J.-H., Cheng, Y.-M., & Liao, W.-Y. (2007). Combination of multiple classifiers for improving emotion recognition in mandarin speech. In *Intelligent information hiding and multimedia signal processing, 2007. iihmsp 2007. third international conference on* (Vol. 1, pp. 35–38).

Pao, T.-L., Liao, W.-Y., Chen, Y.-T., Yeh, J.-H., Cheng, Y.-M., & Chien, C. S.

(2007). Comparison of several classifiers for emotion recognition from noisy mandarin speech. In *Intelligent information hiding and multimedia signal processing, 2007. iihmsp 2007. third international conference on* (Vol. 1, pp. 23–26).

Piana, S., Stagliano, A., Odone, F., Verri, A., & Camurri, A. (2014). Real-time automatic emotion recognition from body gestures. *arXiv preprint arXiv:1402.5047*.

Picard, R. W., & Picard, R. (1997). *Affective computing* (Vol. 252). MIT press Cambridge.

Pittam, J., & Scherer, K. R. (1993). *Vocal expression and communication of emotion.* Guilford Press.

Plutchik, R., & Kellerman, H. (2013a). *Biological foundations of emotion* (Vol. 3). Academic press.

Plutchik, R., & Kellerman, H. (2013b). *Theories of emotion* (Vol. 1). Academic Press.

Postma-Nilsenová, M., Postma, E., & Gu, Y. (2014). No effect of language experience on spectral/fundamental listener type distribution: A comparison of chinese and dutch. In *Fourth international symposium on tonal aspects of languages.*

Postma-Nilsenová, M., Postma, E., Tsoumani, O., & Gu, Y. (n.d.). Biases in auditory perception: Listener-specific preference.

Poznanski, A., & Wolf, L. (2016). Cnn-n-gram for handwriting word recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 2305–2314).

Preparata, F. P., & Shamos, M. (2012). *Computational geometry: an introduction*. Springer Science & Business Media.

Qi, Z., Tian, Y., & Shi, Y. (2013). Robust twin support vector machine for pattern classification. *Pattern Recognition*, *46*(1), 305–316.

Ramakrishnan, S., & El Emary, I. M. (2013). Speech emotion recognition approaches in human computer interaction. *Telecommunication Systems*, 1–12.

Ringeval, F., Schuller, B., Valstar, M., Jaiswal, S., Marchi, E., Lalanne, D., ... Pantic, M. (2015, October). AV+EC 2015 – The First Affect Recognition Challenge Bridging Across Audio, Video, and Physiological Data. In *Proceedings of the 5th International Workshop on Audio/Visual Emotion Challenge (AVEC), ACM MM.* Brisbane, Australia.

Ringeval, F., Sonderegger, A., Sauer, J., & Lalanne, D. (2013, April). Introducing the RECOLA Multimodal Corpus of Remote Collaborative and Affective Interactions. In *Proceedings of Face & Gestures 2013, 2nd IEEE International Workshop on Emotion Representation, Analysis and Synthesis in Continuous Time and Space (EmoSPACE).* Shanghai, China.

Roffo, G., Melzi, S., & Cristani, M. (2015, Dec). Infinite feature selection. In *2015 IEEE international conference on computer vision (iccv)* (p. 4202-4210). doi: 10.1109/ICCV.2015.478

Russell, J. A. (1980). A circumplex model of affect. *Journal of personality and social psychology*, *39*(6), 1161.

Sadjadi, S. O., & Hansen, J. H. (2013). Unsupervised speech activity detection using voicing measures and perceptual spectral flux. *IEEE Signal Processing Letters*, *20*(3), 197–200.

Scherer, K. R. (1972). Acoustic concomitants of emotional dimensions: Judging affect from synthesized tone sequences. *Eastern Psychological Association*, 1–8.

Scherer, K. R. (1995). Expression of emotion in voice and music. *Journal of voice*, *9*(3), 235–248.

Scherer, K. R., & Ekman, P. (1982). *Handbook of methods in nonverbal behavior research* (Vol. 2). Cambridge University Press Cambridge.

Scherer, K. R., & Zei, B. (1988). Vocal indicators of affective disorders. *Psychotherapy and psychosomatics*, *49*(3-4), 179–186.

Schirmer, A., & Simpson, E. (2007). Brain correlates of vocal emotional processing in men and women. *Voice and Emotion*, 75–86.

Schuller, B., Steidl, S., Batliner, A., Hantke, S., Hónig, F., Orozco-Arroyave, J. R., ... Weninger, F. (2015). The interspeech 2015 computational paralinguistics challenge: Nativeness, parkinson's & eating condition. In *Sixteenth annual conference of the international speech communication association.*

Schuller, B., Steidl, S., Batliner, A., Vinciarelli, A., Scherer, K., Ringeval, F., ... others (2013). The interspeech 2013 computational paralinguistics challenge: social signals, conflict, emotion, autism.

Schuller, B., Vlasenko, B., Eyben, F., Rigoll, G., & Wendemuth, A. (2009). Acoustic emotion recognition: A benchmark comparison of performances. In *Automatic speech recognition & understanding, 2009. asru 2009. IEEE workshop on* (pp. 552–557).

Schuller, B., Vlasenko, B., Eyben, F., Wollmer, M., Stuhlsatz, A., Wendemuth, A., & Rigoll, G. (2010). Cross-corpus acoustic emotion recognition: Variances and strategies. *IEEE Transactions on Affective Computing*, *1*(2), 119–131.

Sebe, N., Cohen, I., Gevers, T., & Huang, T. S. (2006). Emotion recognition based on joint visual and audio cues. In *Pattern recognition, 2006. icpr 2006. 18th international conference on* (Vol. 1, pp. 1136–1139).

Sharif Razavian, A., Azizpour, H., Sullivan, J., & Carlsson, S. (2014). Cnn features off-the-shelf: an astounding baseline for recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops* (pp. 806–813).

Sharma, P., & Rajpoot, A. K. (2013). Automatic identification of silence, unvoiced and voiced chunks in speech. *Journal of Computer Science and Information Technology*, *3*(5), 87–96.

Sidtis, J. J., & Van Lancker Sidtis, D. (2003). A neurobehavioral approach to dysprosody. In *Seminars in speech and language* (Vol. 24, pp. 93–106).

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., ... others (2016). Mastering the game of go with deep neural networks and tree search. *Nature*, *529*(7587), 484–489.

Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., ... others (2017a). Mastering chess and shogi by self-play with a general reinforcement learning algorithm. *arXiv preprint arXiv:1712.01815*.

Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., ... others (2017b). Mastering chess and shogi by self-play with a general reinforcement learning algorithm. *arXiv preprint arXiv:1712.01815*.

Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., ... others (2017). Mastering the game of go without human knowledge. *Nature*, *550*(7676), 354-357.

Skinner, E. R. (1935). A calibrated recording and analysis of the pitch, force and quality of vocal tones expressing happiness and sadness; and a determination of the pitch and force of the subjective concepts of ordinary, soft, and loud tones. *Communications Monographs*, *2*(1), 81–137.

Sohn, J., Kim, N. S., & Sung, W. (1999). A statistical model-based voice activity detection. *IEEE signal processing letters*, *6*(1), 1–3.

Souli, S., & Lachiri, Z. (2011). Environmental sounds classification based on visual features. In *Progress in pattern recognition, image analysis, computer vision,*

and applications (pp. 459–466). Springer.

Steunebrink, B. R. (2010). *The logical structure of emotions.* Ph.D. thesis. Utrecht University, the Netherlands.

Stuhlsatz, A., Meyer, C., Eyben, F., Zielke, T., Meier, G., & Schuller, B. (2011). Deep neural networks for acoustic emotion recognition: raising the benchmarks. In *Acoustics, speech and signal processing (icassp), 2011 IEEE international conference on* (pp. 5688–5691).

Suh, Y., & Kim, H. (2012). Multiple acoustic model-based discriminative likelihood ratio weighting for voice activity detection. *IEEE Signal Processing Letters*, *19*(8), 507–510.

Suzuki, A., Watanabe, S., Taheno, Y., Kosugi, T., Kasuya, T., & Senter, D. (2002). Possibility of detecting deception by voice analysis. *Polygraph*, *31*(2), 129–134.

Tao, J., & Tan, T. (2005). Affective computing: A review. In *Affective computing and intelligent interaction* (pp. 981–995). Springer.

Trigeorgis, G., Ringeval, F., Brueckner, R., Marchi, E., Nicolaou, M. A., Schuller, B., & Zafeiriou, S. (2016). Adieu features? end-to-end speech emotion recognition using a deep convolutional recurrent network. In *Acoustics, speech and signal processing (icassp), 2016 IEEE international conference on* (pp. 5200–5204).

Tziolas, A. C., Morrison, N., & Armstrong, E. M. (2017). The interstellar mission. In *Star ark* (pp. 210–254). Springer.

Uhrin, D., Chmelikova, Z., Tovarek, J., Partila, P., & Voznak, M. (2016). One approach to design of speech emotion database. In *Spie defense+ security* (pp. 98500B–98500B).

Van der Maaten, L. (2009). *Feature extraction from visual data*. Ph.D. thesis. Tilburg University, the Netherlands.

Van der Maaten, L., & Hinton, G. (2008). Visualizing data using t-sne. *Journal of Machine Learning Research*, *9*(Nov), 2579–2605.

Van der Maaten, L., Postma, E. O., & Van den Herik, H. J. (2009a). Dimensionality reduction: a comparative. *Journal of Machine Learning Research*, *10*, 66–71.

Van der Maaten, L., Postma, E. O., & Van den Herik, H. J. (2009b). *Dimensionality reduction: A comparative review* (Tech. Rep. No. TiCC-TR 2009-005). Tilburg center for Cognition and Communication, Tilburg University.

Van Lancker, D. (1991). Personal relevance and the human right hemisphere. *Brain and Cognition*, *17*(1), 64–92.

Wang, K., An, N., Li, B. N., Zhang, Y., & Li, L. (2015). Speech emotion recognition using fourier parameters. *IEEE Transactions on Affective Computing*, *6*(1), 69–75.

Wang, K.-C. (2015). Time-frequency feature representation using multi-resolution texture analysis and acoustic activity detector for real-life speech emotion recognition. *Sensors*, *15*(1), 1458–1478.

Watzlawick, P., Bavelas, J. B., Jackson, D. D., & O'Hanlon, B. (2011). *Pragmatics of human communication: A study of interactional patterns, pathologies and paradoxes*. WW Norton & Company.

Williams, C. E., & Stevens, K. N. (1972). Emotions and speech: Some acoustical correlates. *The Journal of the Acoustical Society of America*, *52*(4B), 1238–1250.

Witten, I. H., & Frank, E. (2005). *Data mining: Practical machine learning tools and techniques*. Morgan Kaufmann.

Wu, Parsons, T. D., & Narayanan, S. S. (2010). Acoustic feature analysis in speech emotion primitives estimation. In *Interspeech* (pp. 785–788).

Wu, S., Falk, T. H., & Chan, W.-Y. (2009). Automatic recognition of speech emotion using long-term spectro-temporal features. In *Digital signal processing, 2009 16th international conference on* (pp. 1–6).

Wu, T., Yang, Y., Wu, Z., & Li, D. (2006). Masc: a speech corpus in mandarin for emotion analysis and affective speaker recognition. In *Speaker and language recognition workshop* (pp. 1–5).

Xiong, W., Droppo, J., Huang, X., Seide, F., Seltzer, M., Stolcke, A., ... Zweig, G. (2017). The microsoft 2016 conversational speech recognition system. In *Acoustics, speech and signal processing (icassp), 2017 IEEE international conference on* (pp. 5255–5259).

Yin, H., Hohmann, V., & Nadeu, C. (2011). Acoustic features for speech recognition based on gammatone filterbank and instantaneous frequency. *Speech Communication*, *53*(5), 707–715.

Yoo, I.-C., Lim, H., & Yook, D. (2015). Formant-based robust voice activity detection. *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, *23*(12), 2238–2245.

Yu, T., & Hansen, J. H. (2010). Discriminative training for multiple observation likelihood ratio based voice activity detection. *Signal Processing Letters, IEEE*,

17(11), 897–900.

Zanettin, F., Bernardini, S., & Stewart, D. (2014). *Corpora in translator education*. Routledge.

Zhang, X.-L., & Wang, D. (2014). Boosted deep neural networks and multi-resolution cochleagram features for voice activity detection. In *Fifteenth annual conference of the international speech communication association.*

Zhang, X.-L., & Wu, J. (2013). Deep belief networks based voice activity detection. *Audio, Speech, and Language Processing, IEEE Transactions on*, *21*(4), 697–710.

Zheng, W., Yu, J., & Zou, Y. (2015). An experimental study of speech emotion recognition based on deep convolutional neural networks. In *Affective computing and intelligent interaction (acii), 2015 international conference on* (pp. 827–831).

A APPENDICES

In the following we provide additional contents to the text in the form of two appendices. Appendix A.1 is the Utterances of Mandarin Affective Speech, and appendix A.2 is titled the URLs of the relevant tools.

A.1 THE UTTERANCES OF MANDARIN AFFECTIVE SPEECH

The Appendix A.1 provides the utterances of Mandarin Affective Speech. It contains the utterance in Chinese and also its English translation.

Table A.1: Mandarin Affective Speech Corpus part 1

Number	Utterance in Chinese	English Translation
1	你是个好人。	He is a good person.
2	我们那边有网球运动场、餐馆、酒吧和一个面包店。	There are a tennis court, restaurant, bar and bakery close to where we live.
3	你今天去医院看过病了吗?	Have you seen a doctor at the hospital today?
4	这个湖是人工的，还是自然形成的?	Is this lake artificial or natural?
5	你去把空调打开。	Please turn the air-conditioner on.
6	他是多么慷慨啊。	He is so generous.
7	为什么你不给他看那本小说呢?	Why don't you show him that novel?
8	我应该在信里写些什么呢?	What should I write in the letter?
9	我们那天去欧洲的温莎城堡玩?	Which day shall we visit Windsor Castle?
10	老翁挖了大约五平米的池塘养鱼。	The old man dug a pond of about 5 square meters for fish-farming.

Table A.2: Mandarin Affective Speech Corpus part 2

Number	Utterance in Chinese	English Translation
11	小明，你陪外婆补牙齿吧。	Xiaoming, could you accompany your grandmother to have dental fillings?
12	今天晚上会下雨。	It will rain tonight.
13	港口来了好多警察。	So many policemen have come to the port.
14	考试结束时间快到了。	The examination is about to finish.
15	我们室友总是把寝室弄的很脏。	My roommate always makes the dorm very messy.
16	你抄我的物理作业。	You copied my physics assignment.
17	我最要好的朋友要移民去欧洲了。	My best friend is going to emigrate to Europe.
18	他们家小狗死掉了。	Their family's dog is dead.
19	明天要去富春漂流了。	We are going rafting tomorrow in Fuchun.
20	金桥门水果摊要开门了	The fruit shop that is located in Jinqiaomen will open soon.

A.2 URLS OF THE RELEVANT TOOLS

Appendix A.2 provides the URLs of the relevant tools in this thesis.

Table A.3: The URL of the relevant tools

No	Software	URLs
1	Skype Translator。	https://www.skype.com/en/features/skype-translator
2	Google Translator	https://support.google.com/translate
3	Baidu Voice	http://http://yuyin.baidu.com/
4	Mandarin Affective Speech	http://catalog.ldc.upenn.edu/LDC2007S09/
5	Spectrogram Generation Function	Phttps://au.mathworks.com/help/signal/ref/spectrogram.html
6	Log-Gabor Function	http://www.csse.uwa.edu.au/pk/research/matlabfns/
7	WIKA	http://www.cs.waikato.ac.nz/ml/weka/
8	Praat	http://www.fon.hum.uva.nl/praat/
9	MatConvNet	http://www.vlfeat.org/matconvnet/

A.3 MATLAB CODE FOR VSS ALGORITHM

Appendix A.3 provides two ways of setting the VSS algorithm.

Code 1: manually setting the threshold for the voiced part selection

```
[y, fs]=audioread('D:\emotion recognition\wav\test\sa(10).wav');
wlen=320;inc=160;
win=hamming(wlen);
nfft=1024;
filename =spectrogram(y,win,inc,nfft,fs);
surf(T,F,10*log10(P),'edgecolor','none'); axis tight;
view(0,90);
xlabel('Time (Seconds)'); ylabel('Hz');

I= filename;

EO= gaborconvolve(I, 12, 12, 3, 1.35, 0.8, 1.5,  0);    %%using the log-
    Gabor filter to process the image
% y=abs(EO);
BO=cell(12,12);
for i=1:12
    for j=1:12
        BO{i,j}=abs(EO{i,j});
    end
end
meaning=zeros(size(EO{1,1}));            %%Get the vector of each (Ns, No)
    ,
                                         %%The convolution values within
                                             each image is averaged by the
                                              time sequence.
                                          %%The row of the matrix EO{i,j}
                                              is frequency, while the
                                              column of the matrix is time
                                              .
for u=1:12
    for v=1:12
        meaning(((u-1)*12+v),:)=mean(BO{u,v});
    end
end
% song=mean(meaning);
dataset=song;
[dataset_scale,ps]=mapminmax(dataset,0,1);    %%normalize the vector
n=size(song,2);                               %%To get the point-in-time
    of spectrogram
q=size(y,1);
t=q/fs;                                       %%to get the time span of
    the voice
ts=t/(q-1);                                   %%to get the time span
    between the sample of the voice
```

```
Ts=t/(n-1);                                    %%to get the time span
    between the point-in-time of spectrogram
p=round(Ts/ts);                                %%to get the number of
    sample between the point-in-time of spectrogram
% start=0;
wout=zeros(160000,1);
for k=1:n
  if dataset_scale(1,k)>0.229                  %%set the threshhold of the
      voiced
                                               %%and unvoiced segment
      wout((1+(k-1)*p):k*p,1)=y((1+(k-1)*p):k*p,1);
  else
      continue
  end
end
out=wout(find(wout~=0));                      %%get the voiced segment
% wavname=['D:\newtraing\an' int2str(i) '.wav'];
audiowrite('sa10.wav',out,fs);
```

Code 2: only the voiced feature extraction

```

clear, close all

nscale = 12; % number of scales used 10
norient = 12; % number of orientations (0-pi) used = 12
minWaveLength = 3; % number of pixels minimum wave length, must be at
    least 2 for nyquist
mult = 1.35; % factor increase in wavelength per scale - used 1.5
sigmaOnf = 0.8; % sd of gaussian

[y, fs]=audioread('D:\emotion recognition\wav\test\sa(10).wav');
wlen=320;inc=160;
win=hamming(wlen);
nfft=1024;
filename =spectrogram(y,win,inc,nfft,fs);
surf(T,F,10*log10(P),'edgecolor','none'); axis tight;
view(0,90);
xlabel('Time (Seconds)'); ylabel('Hz');

I= filename;

EO= gaborconvolve(I, 1, norient, waveLength, mult, sigmaOnf, Lnorm, 0);
       %%using the log-Gabor filter to process the image
% y=abs(EO);
BO=cell(12,12);
for i=1:12
    for j=1:12
        BO{i,j}=abs(EO{i,j});
    end
```

```
end
meaning=zeros(size(EO{1,1}));           %%Get the vector of each (Ns, No)
    ,
                                        %%The convolution values within
                                            each image is averaged by the
                                             time sequence.
                                         %%The row of the matrix EO{i,j}
                                             is frequency, while the
                                             column of the matrix is time
                                             .
for u=1:12
    for v=1:12
        meaning(((u-1)*12+v),:)=mean(BO{u,v});
    end
end
% song=mean(meaning);
dataset=song;
[dataset_scale,ps]=mapminmax(dataset,0,1);    %%normalize the vector
```

A.4 MATLAB CODE FOR LOG-GABOR FILTERS

Appendix A.4 provides the code for log-Gabor filters.

```
clear, close all
tic

%% data params
File = dir(strcat('C:\Users\TiCC\Documents\MATLAB\spectrogramalgorithm\
    data\happiness\','*.wav'));

outputFileName = 'happy_features.csv';
outputPath = './';

nscale = 12; % number of scales used 10
norient = 6; % number of orientations (0-pi) used = 12
minWaveLength = 3; % number of pixels minimum wave length, must be at
    least 2 for nyquist
mult = 1.35; % factor increase in wavelength per scale - used 1.5
sigmaOnf = 0.8; % sd of gaussian
Lnorm = 0; % norm of the filter

features = []; % features and class label

happyfeatures =[];

for f = 1:200 % run script on each of the files (TIF)
    fileName = strcat('C:\Users\TiCC\Documents\MATLAB\spectrogramalgorithm
        \data\happiness\',File(f).name);
    fprintf('Processing %i of %i files\n',f,length(File))

    if ~isempty(strfind(fileName,'.jpg')) % only process if it is a tif
        file

        disp(fileName)
        im = spectrogram(fileName);

        %% extract a patch from the image
        patch_size = 2^floor(log2(min(size(im,1),size(im,2)))); % smallest
             2^n size square patch
        h = size(im,1);
        w = size(im,2);
        im = im(h/2-floor(patch_size/2)+1:h/2+ceil(patch_size/2),w/2-floor
            (patch_size/2)+1:w/2+ceil(patch_size/2),:);

        %% convert to gray scale if RGB
        nr_vectors = size(im);

        if length(nr_vectors)>2
```

```
    im = rgb2gray(im);
end

total_energy = zeros(size(nscale,norient));

%% apply gaborconvolve per scale

magnitudeSumFeatures = zeros(16,nscale); % placeholder for
    features for this image
magnitudeVarFeatures = zeros(16,nscale); % placeholder for
    features for this image



for j = 1:nscale
    waveLength = minWaveLength * mult^(j-1);
    [Es,Bs] = gaborconvolve(im, 1, norient, waveLength, mult,
        sigmaOnf, Lnorm, 0);

    magnitudeOrient = zeros([size(Es{1}),norient]);

    for es = 1:norient
        magnitudeOrient(:,:,es) = abs(Es{es});
    end

    for h = 1:4
        for w = 1:4

            magnitudeOrientPart = magnitudeOrient(1+patch_size/4*(
                h-1):patch_size/4*h,1+patch_size/4*(w-1):patch_
                size/4*w,:);

            magnitudeSumFeatures(w+(h-1)*4,j) = sum(sum(sum(
                magnitudeOrientPart))/(patch_size/4)^2); % mean
                energy per pixel for this scale
            magnitudeVarFeatures(w+(h-1)*4,j) = std(sum(sum(
                magnitudeOrientPart))/mean(sum(sum(
                magnitudeOrientPart)))); % measure for 'ellipsity'
                 of texture

        end
    end

end

magnitudeSumFeatures = bsxfun(@rdivide,magnitudeSumFeatures,sum(
    magnitudeSumFeatures,2)); % scale by sum of energy to cancel
    out light intensity
scaleFeatures = [magnitudeSumFeatures,magnitudeVarFeatures];
```

```
        happyfeatures = cat(1,happyfeatures,scaleFeatures);

    end

end

% output features to file
disp('Writing angerfeatures and labels to file')
outputName = strcat(outputPath,outputFileName);
dlmwrite(outputName,happyfeatures);
```

A.5 MATLAB CODE FOR CNN ALGORITHM

Appendix A.5 provides the CNN code based on both the designed and the pre-trained model.

A.5.1 Matlab Code for CNN Code 1

Code 1: The CNN model based on the 7 layers small model

```
function [net, info] = cnn_spectrogram_fiveemomiddle(varargin)

run(fullfile(fileparts(mfilename('fullpath')),...
  '..', '..', 'matlab', 'vl_setupnn.m')) ;

opts.batchNormalization = false ;
opts.network = [] ;
opts.networkType = 'simplenn' ;
[opts, varargin] = vl_argparse(opts, varargin) ;

sfx = opts.networkType ;
if opts.batchNormalization, sfx = [sfx '-bnorm'] ; end
opts.expDir = fullfile(vl_rootnn, 'data', ['emotionfivemiddle-' sfx]) ;
[opts, varargin] = vl_argparse(opts, varargin) ;

opts.dataDir = fullfile(vl_rootnn, 'data', 'emotion') ;
opts.imdbPath = fullfile(opts.expDir, 'imdb.mat');
opts.train = struct() ;
opts = vl_argparse(opts, varargin) ;
if ~isfield(opts.train, 'gpus'), opts.train.gpus = []; end;

% -------------------------------------------------------------------
%                                                          Prepare data
% -------------------------------------------------------------------

if isempty(opts.network)
  net = cnn_spectrogram_fiveemomiddle_init('batchNormalization', opts.
      batchNormalization, ...
    'networkType', opts.networkType) ;
else
  net = opts.network ;
  opts.network = [] ;
end

if exist(opts.imdbPath, 'file')
  imdb = load(opts.imdbPath) ;
else
  imdb = getemoImdb(opts) ;
  mkdir(opts.expDir) ;
  save(opts.imdbPath, '-struct', 'imdb') ;
```

```
end

net.meta.classes.name = arrayfun(@(x)sprintf('%d',x),1:5,'UniformOutput',
    false) ;

% --------------------------------------------------------------------
%                                                                Train
% --------------------------------------------------------------------

switch opts.networkType
  case 'simplenn', trainfn = @cnn_train ;
  case 'dagnn', trainfn = @cnn_train_dag ;
end

[net, info] = trainfn(net, imdb, getBatch(opts), ...
  'expDir', opts.expDir, ...
  net.meta.trainOpts, ...
  opts.train, ...
  'val', find(imdb.images.set == 3)) ;

% --------------------------------------------------------------------
function fn = getBatch(opts)
% --------------------------------------------------------------------
switch lower(opts.networkType)
  case 'simplenn'
    fn = @(x,y) getSimpleNNBatch(x,y) ;
  case 'dagnn'
    bopts = struct('numGpus', numel(opts.train.gpus)) ;
    fn = @(x,y) getDagNNBatch(bopts,x,y) ;
end

% --------------------------------------------------------------------
function [images, labels] = getSimpleNNBatch(imdb, batch)
% --------------------------------------------------------------------
images = imdb.images.data(:,:,:,batch) ;
labels = imdb.images.labels(1,batch) ;

% --------------------------------------------------------------------
function inputs = getDagNNBatch(opts, imdb, batch)
% --------------------------------------------------------------------
images = imdb.images.data(:,:,:,batch) ;
labels = imdb.images.labels(1,batch) ;
if opts.numGpus > 0
  images = gpuArray(images) ;
end
inputs = {'input', images, 'label', labels} ;

% --------------------------------------------------------------------
function imdb = getemoImdb(opts)
```

```
% -------------------------------------------------------------------
% Preapre the imdb structure, returns image data with mean image
    subtracted
load('test.mat');
load('train.mat');
x1 = Train1;
x2 = Test1;

data = single(cat(4, x1, x2));

y1=cat(2,ones(1,1000),ones(1,1000)+1,ones(1,1000)+2,ones(1,1000)+3,ones
    (1,1000)+4);
y2=cat(2,ones(1,400),ones(1,400)+1,ones(1,400)+2,ones(1,400)+3,ones(1,400)
    +4);

set = [ones(1,numel(y1)) 3*ones(1,numel(y2))];

data = single(cat(4, x1, x2));
dataMean = mean(data(:,:,:,set == 1), 4);


imdb.meta.classes = {'angry', 'happy','neutral','panic','sad'} ;
imdb.images.labels = cat(2, y1, y2) ;
imdb.images.set = set ;
imdb.meta.sets = {'train', 'val', 'test'} ;
imdb.images.data = bsxfun(@minus, data, dataMean);
imdb.images.data_mean = single(dataMean);
```

```
function net = cnn_spectrogram_fiveemomiddle_init(varargin)
opts.batchNormalization = true ;
opts.networkType = 'simplenn' ;
opts = vl_argparse(opts, varargin) ;

rng('default');
rng(0) ;

f=1/100 ;
net.layers = {} ;
net.layers{end+1} = struct('type', 'conv', ...
                           'weights', {{f*randn(5,5,3,20, 'single'), zeros
                               (1, 20, 'single')}}, ...
                           'stride', 1, ...
                           'pad', 0) ;
net.layers{end+1} = struct('type', 'pool', ...
                           'method', 'max', ...
                           'pool', [2 2], ...
                           'stride', 2, ...
                           'pad', 0) ;
```

```
net.layers{end+1} = struct('type', 'conv', ...
                           'weights', {{f*randn(5,5,20,50, 'single'),zeros
                              (1,50,'single')}}, ...
                           'stride', 1, ...
                           'pad', 0) ;
net.layers{end+1} = struct('type', 'pool', ...
                           'method', 'max', ...
                           'pool', [2 2], ...
                           'stride', 2, ...
                           'pad', 0) ;
net.layers{end+1} = struct('type', 'conv', ...
                           'weights', {{f*randn(4,4,50,500, 'single'),
                              zeros(1,500,'single')}}, ...
                           'stride', 1, ...
                           'pad', 0) ;
net.layers{end+1} = struct('type', 'relu') ;
net.layers{end+1} = struct('type', 'conv', ...
                           'weights', {{f*randn(1,1,500,10, 'single'),
                              zeros(1,10,'single')}}, ...
                           'stride', 1, ...
                           'pad', 0) ;
net.layers{end+1} = struct('type', 'softmaxloss') ;

% optionally switch to batch normalization
if opts.batchNormalization
  net = insertBnorm(net, 1) ;
  net = insertBnorm(net, 4) ;
  net = insertBnorm(net, 7) ;
end

% Meta parameters
net.meta.inputSize = [28 28 1] ;
net.meta.trainOpts.learningRate = 0.0005 ;
net.meta.trainOpts.numEpochs = 200 ;
net.meta.trainOpts.batchSize = 100 ;

% Fill in defaul values
net = vl_simplenn_tidy(net) ;

% Switch to DagNN if requested
switch lower(opts.networkType)
  case 'simplenn'
    % done
  case 'dagnn'
    net = dagnn.DagNN.fromSimpleNN(net, 'canonicalNames', true) ;
    net.addLayer('top1err', dagnn.Loss('loss', 'classerror'), ...
      {'prediction', 'label'}, 'error') ;
    net.addLayer('top5err', dagnn.Loss('loss', 'topkerror', ...
      'opts', {'topk', 5}), {'prediction', 'label'}, 'top5err') ;
```

```
  otherwise
    assert(false) ;
end

% --------------------------------------------------------------------
function net = insertBnorm(net, l)
% --------------------------------------------------------------------
assert(isfield(net.layers{l}, 'weights'));
ndim = size(net.layers{l}.weights{1}, 4);
layer = struct('type', 'bnorm', ...
               'weights', {{ones(ndim, 1, 'single'), zeros(ndim, 1, '
                   single')}}, ...
               'learningRate', [1 1 0.05], ...
               'weightDecay', [0 0]) ;
net.layers{l}.biases = [] ;
net.layers = horzcat(net.layers(1:l), layer, net.layers(l+1:end)) ;
```

A.5.2 Matlab Code for CNN code 2

```
function [net, info] = cnn_emotion (varargin)


run(fullfile(fileparts(mfilename('fullpath')), ...
  '..', '..', 'matlab', 'vl_setupnn.m')) ;

opts.modelType = 'lenet' ;
[opts, varargin] = vl_argparse(opts, varargin) ;

opts.expDir = fullfile(vl_rootnn, 'data', ...
  sprintf('emotion256-%s', opts.modelType)) ;
[opts, varargin] = vl_argparse(opts, varargin) ;

opts.dataDir = fullfile(vl_rootnn, 'data','emotion256') ;
opts.imdbPath = fullfile(opts.expDir, 'imdb.mat');
opts.whitenData = true ;
opts.contrastNormalization = true ;
opts.networkType = 'simplenn' ;
opts.train = struct() ;
opts = vl_argparse(opts, varargin) ;
if ~isfield(opts.train, 'gpus'), opts.train.gpus = []; end;

%
    ------------------------------------------------------------------------

%                                                    Prepare model and
    data
```

```
%
    ------------------------------------------------------------------------


switch opts.modelType
  case 'lenet'
    net = cnn_emotion_init('networkType', opts.networkType) ;
  case 'nin'
    net = cnn_emotion_init_nin('networkType', opts.networkType) ;
  otherwise
    error('Unknown model type ''%s''.', opts.modelType) ;
end

if exist(opts.imdbPath, 'file')
  imdb = load(opts.imdbPath) ;
else
  imdb = getemoImdb(opts) ;
  mkdir(opts.expDir) ;
  save(opts.imdbPath, '-struct', 'imdb') ;
end

net.meta.classes.name = imdb.meta.classes(:)' ;

%
    ------------------------------------------------------------------------

%
    Train
%
    ------------------------------------------------------------------------


switch opts.networkType
  case 'simplenn', trainfn = @cnn_train ;
  case 'dagnn', trainfn = @cnn_train_dag ;
end

[net, info] = trainfn(net, imdb, getBatch(opts), ...
  'expDir', opts.expDir, ...
  net.meta.trainOpts, ...
  opts.train, ...
  'val', find(imdb.images.set == 3)) ;

%
    ------------------------------------------------------------------------

function fn = getBatch(opts)
```

```
%
    -------------------------------------------------------------------------

switch lower(opts.networkType)
  case 'simplenn'
    fn = @(x,y) getSimpleNNBatch(x,y) ;
  case 'dagnn'
    bopts = struct('numGpus', numel(opts.train.gpus)) ;
    fn = @(x,y) getDagNNBatch(bopts,x,y) ;
end

%
    -------------------------------------------------------------------------

function [images, labels] = getSimpleNNBatch(imdb, batch)
%
    -------------------------------------------------------------------------

images = imdb.images.data(:,:,:,batch) ;
labels = imdb.images.labels(1,batch) ;
if rand > 0.5, images=fliplr(images) ; end

%
    -------------------------------------------------------------------------

function inputs = getDagNNBatch(opts, imdb, batch)
%
    -------------------------------------------------------------------------

images = imdb.images.data(:,:,:,batch) ;
labels = imdb.images.labels(1,batch) ;
if rand > 0.5, images=fliplr(images) ; end
if opts.numGpus > 0
  images = gpuArray(images) ;
end
inputs = {'input', images, 'label', labels} ;

%
    -------------------------------------------------------------------------

function imdb = getemoImdb(opts)
%
    -------------------------------------------------------------------------

% Preapre the imdb structure, returns image data with mean image
    subtracted
traintdir = 'Train';
trainFiles = dir(fullfile(traintdir, '*.jpg') );
train1=zeros(64,64,3,length(trainFiles));
```

```
nFiles = length(trainFiles)
      for i = 1:nFiles
          currentFile = fullfile(traintdir, trainFiles(i).name);
          currentImage1 = imread(currentFile);
          currentImage2 = imresize(currentImage1,[64,64]);
          train1(:,:,:,i) = currentImage2;
      end

testdir = 'Test';
testFiles = dir(fullfile(testdir, '*.jpg') );
test1=zeros(64,64,3,length(testFiles));
mFiles = length(testFiles)
      for i = 1:mFiles
          currenttestFile = fullfile(testdir, testFiles(i).name);
          currenttest2 = imread(currenttestFile);
          currenttest3 = imresize(currenttest2,[64,64]);
          test1(:,:,:,i) = currenttest3;
      end


data = single(cat(4, train1, test1));


y1=cat(2,ones(1,2000),ones(1,2000)+1,ones(1,2000)+2,ones(1,2000)+3,ones
    (1,2000)+4);
y2=cat(2,ones(1,1000),ones(1,1000)+1,ones(1,1000)+2,ones(1,1000)+3,ones
    (1,1000)+4);

set = [ones(1,numel(y1)) 3*ones(1,numel(y2))];

dataMean = mean(data(:,:,:,set == 1), 4);

imdb.meta.classes = {'angry';'happy';'neutral';'panic';'sad'} ;
imdb.meta.sets = {'train', 'val', 'test'} ;
imdb.images.labels = cat(2, y1, y2) ;
imdb.images.set = set;
imdb.images.data = bsxfun(@minus, data, dataMean);
```

```
function net = cnn_emotion_init(varargin)
opts.networkType = 'simplenn' ;
opts = vl_argparse(opts, varargin) ;

lr = [.1 2] ;

net.layers = {} ;

% Block 1
net.layers{end+1} = struct('type', 'conv', ...
```

```
                           'weights', {{0.01*randn(5,5,3,32, 'single'),
                              zeros(1, 32, 'single')}}, ...
                           'learningRate', lr, ...
                           'stride', 1, ...
                           'pad', 2) ;
net.layers{end+1} = struct('type', 'pool', ...
                           'method', 'max', ...
                           'pool', [3 3], ...
                           'stride', 2, ...
                           'pad', [0 1 0 1]) ;
net.layers{end+1} = struct('type', 'relu') ;

% Block 2
net.layers{end+1} = struct('type', 'conv', ...
                           'weights', {{0.05*randn(5,5,32,32, 'single'),
                              zeros(1,32,'single')}}, ...
                           'learningRate', lr, ...
                           'stride', 1, ...
                           'pad', 2) ;
net.layers{end+1} = struct('type', 'relu') ;
net.layers{end+1} = struct('type', 'pool', ...
                           'method', 'avg', ...
                           'pool', [3 3], ...
                           'stride', 2, ...
                           'pad', [0 1 0 1]) ; % Emulate caffe

% Block 3
net.layers{end+1} = struct('type', 'conv', ...
                           'weights', {{0.05*randn(5,5,32,64, 'single'),
                              zeros(1,64,'single')}}, ...
                           'learningRate', lr, ...
                           'stride', 1, ...
                           'pad', 2) ;
net.layers{end+1} = struct('type', 'relu') ;
net.layers{end+1} = struct('type', 'pool', ...
                           'method', 'avg', ...
                           'pool', [3 3], ...
                           'stride', 2, ...
                           'pad', [0 1 0 1]) ; % Emulate caffe

% Block 4
net.layers{end+1} = struct('type', 'conv', ...
                           'weights', {{0.05*randn(4,4,64,64, 'single'),
                              zeros(1,64,'single')}}, ...
                           'learningRate', lr, ...
                           'stride', 1, ...
                           'pad', 0) ;
net.layers{end+1} = struct('type', 'relu') ;
```

```
% Block 5
net.layers{end+1} = struct('type', 'conv', ...
                           'weights', {{0.05*randn(1,1,64,10, 'single'),
                              zeros(1,10,'single')}}, ...
                           'learningRate', .1*lr, ...
                           'stride', 1, ...
                           'pad', 0) ;

% Loss layer
net.layers{end+1} = struct('type', 'softmaxloss') ;

% Meta parameters
net.meta.inputSize = [64 64 3] ;
net.meta.trainOpts.learningRate = [0.05*ones(1,30) 0.005*ones(1,10)
    0.0005*ones(1,5)] ;
net.meta.trainOpts.weightDecay = 0.0001 ;
net.meta.trainOpts.batchSize = 100 ;
net.meta.trainOpts.numEpochs = numel(net.meta.trainOpts.learningRate) ;

% Fill in default values
net = vl_simplenn_tidy(net) ;

% Switch to DagNN if requested
switch lower(opts.networkType)
  case 'simplenn'
    % done
  case 'dagnn'
    net = dagnn.DagNN.fromSimpleNN(net, 'canonicalNames', true) ;
    net.addLayer('error', dagnn.Loss('loss', 'classerror'), ...
             {'prediction','label'}, 'error') ;
  otherwise
    assert(false) ;
end
```

A.5.3 Matlab Code for CNN Pre-trained model

Appendix A.5.3 provides the CNN feature based on the Imagenet pre-trained model.

```
function imdb = cnn_image_setup_data(varargin)

opts.dataDir = fullfile('data','image') ;
opts.lite = false ;
opts = vl_argparse(opts, varargin) ;

% --------------------------------------------------------------------
% Load categories metadata
%
    --------------------------------------------------------------------------


metaPath = fullfile(opts.dataDir, 'classInd.txt') ;

fprintf('using metadata %s\n', metaPath) ;
tmp = importdata(metaPath);
nCls = numel(tmp);
% check the label
if nCls ~= 5
  error('Wrong meta file %s',metaPath);
end
% read the label name
cats = cell(1,nCls);
for i=1:numel(tmp)
  t = strsplit(tmp{i});
  cats{i} = t{2};
end
% select data file
imdb.classes.name = cats ;
imdb.imageDir.train = fullfile(opts.dataDir, 'train') ;
imdb.imageDir.test = fullfile(opts.dataDir, 'test') ;

%% -------------------------------------------------------------------
%                                               load image names and labels
%
    --------------------------------------------------------------------------


name = {};
labels = {} ;
imdb.images.sets = [] ;
%%
fprintf('searching training images ...\n') ;
train_label_path = fullfile(opts.dataDir, 'train_label.txt') ;
```

```
train_label_temp = importdata(train_label_path);
temp_l = train_label_temp.data;
for i=1:numel(temp_l)
    train_label{i} = temp_l(i);
end
if length(train_label) ~= length(dir('C:\Users\17253260\Documents\MATLAB\
    Newtrainingmodel\data\image\train\*.jpg'))
    error('training data is not equal to its label!!!');
end

i = 1;
for d = dir('C:\Users\17253260\Documents\MATLAB\Newtrainingmodel\data\
    image\train\*.jpg')'
    name{end+1} = d.name;
    labels{end+1} = train_label{i} ;
    if mod(numel(name), 10) == 0, fprintf('.') ; end
    if mod(numel(name), 500) == 0, fprintf('\n') ; end
    imdb.images.sets(end+1) = 1;%train
    i = i+1;
end
%%
fprintf('searching testing images ...\n') ;
test_label_path = fullfile(opts.dataDir, 'test_label.txt') ;
test_label_temp = importdata(test_label_path);
temp_l = test_label_temp.data;
for i=1:numel(temp_l)
    test_label{i} = temp_l(i);
end
if length(test_label) ~= length(dir('C:\Users\17253260\Documents\MATLAB\
    Newtrainingmodel\data\image\test\*.jpg'))
    error('testing data is not equal to its label!!!');
end
i = 1;
for d = dir('C:\Users\17253260\Documents\MATLAB\Newtrainingmodel\data\
    image\test\*.jpg')'
    name{end+1} = d.name;
    labels{end+1} = test_label{i} ;
    if mod(numel(name), 10) == 0, fprintf('.') ; end
    if mod(numel(name), 500) == 0, fprintf('\n') ; end
    imdb.images.sets(end+1) = 3;%test
    i = i+1;
end
%%
labels = horzcat(labels{:}) ;
imdb.images.id = 1:numel(name) ;
imdb.images.name = name ;
imdb.images.label = labels ;
```

```
function [net, info] = cnn_newnn(varargin)
% Demonstrates fine-tuning a pre-trained CNN based on imagenet dataset

run(fullfile(fileparts(mfilename('fullpath')), ...
  '..', 'matconvnet-1.0-beta24', 'matlab', 'vl_setupnn.m')) ;
opts.dataDir = fullfile('data','image') ;
opts.expDir  = fullfile('exp', 'image') ;
opts.modelPath = fullfile('models','imagenet-vgg-f.mat');
[opts, varargin] = vl_argparse(opts, varargin) ;

opts.numFetchThreads = 12 ;

opts.lite = false ;
opts.imdbPath = fullfile(opts.expDir, 'imdb.mat');

opts.train = struct() ;
opts.train.gpus = [];
opts.train.batchSize = 8 ;
opts.train.numSubBatches = 4 ;
opts.train.learningRate = 1e-4 * [ones(1,10), 0.1*ones(1,5)];

opts = vl_argparse(opts, varargin) ;
if ~isfield(opts.train, 'gpus'), opts.train.gpus = []; end;

%
    -------------------------------------------------------------------------

%                                                              Prepare
    model
%
    -------------------------------------------------------------------------

net = load(opts.modelPath);
net = prepareDINet(net,opts);
%
    -------------------------------------------------------------------------

%                                                               Prepare
    data
%
    -------------------------------------------------------------------------

if exist(opts.imdbPath,'file')
  imdb = load(opts.imdbPath) ;
else
  imdb = cnn_image_setup_data('dataDir', opts.dataDir, 'lite', opts.lite)
      ;
  mkdir(opts.expDir) ;
  save(opts.imdbPath, '-struct', 'imdb') ;
```

```
end

imdb.images.set = imdb.images.sets;

% Set the class names in the network
net.meta.classes.name = imdb.classes.name ;
net.meta.classes.description = imdb.classes.name ;

imageStatsPath = fullfile(opts.expDir, 'imageStats.mat') ;
if exist(imageStatsPath)
  load(imageStatsPath, 'averageImage') ;
else
    averageImage = getImageStats(opts, net.meta, imdb) ;
    save(imageStatsPath, 'averageImage') ;
end
% % ?????????
net.meta.normalization.averageImage = averageImage;
%
    -------------------------------------------------------------------------

%
    Learn
%
    -------------------------------------------------------------------------

% train==1  val==3
opts.train.train = find(imdb.images.set==1) ;
opts.train.val = find(imdb.images.set==3) ;
% ??
[net, info] = cnn_train_dag(net, imdb, getBatchFn(opts, net.meta), ...
                      'expDir', opts.expDir, ...
                      opts.train) ;

%
    -------------------------------------------------------------------------

%
    Deploy
%
    -------------------------------------------------------------------------

net = cnn_imagenet_deploy(net) ;
modelPath = fullfile(opts.expDir, 'net-deployed.mat');

net_ = net.saveobj() ;
save(modelPath, '-struct', 'net_') ;
clear net_ ;
```

```
%
    ------------------------------------------------------------------------

function fn = getBatchFn(opts, meta)
%
    ------------------------------------------------------------------------

useGpu = numel(opts.train.gpus) > 0 ;

bopts.numThreads = opts.numFetchThreads ;
bopts.imageSize = meta.normalization.imageSize ;
bopts.border = meta.normalization.border ;
% bopts.averageImage = [];
bopts.averageImage = meta.normalization.averageImage ;
% bopts.rgbVariance = meta.augmentation.rgbVariance ;
% bopts.transformation = meta.augmentation.transformation ;

fn = @(x,y) getDagNNBatch(bopts,useGpu,x,y) ;


%
    ------------------------------------------------------------------------

function inputs = getDagNNBatch(opts, useGpu, imdb, batch)
%
    ------------------------------------------------------------------------

if imdb.images.set(1) == 1
       load('train.mat');
       images = Train1;
else
        load('test.mat');
        images = Test1;
end

isVal = ~isempty(batch) && imdb.images.set(batch(1)) ~= 1 ;

if ~isVal
  % training
  im = cnn_imagenet_get_batch(images, opts, ...
                              'prefetch', nargout == 0) ;
else
  % validation: disable data augmentation
  im = cnn_imagenet_get_batch(images, opts, ...
                              'prefetch', nargout == 0, ...
                              'transformation', 'none') ;
end

if nargout > 0
```

```
  if useGpu
    im = gpuArray(im) ;
  end
  labels = imdb.images.label(batch) ;
  inputs = {'input', im, 'label', labels} ;
end

%
    ---------------------------------------------------------------------

function averageImage = getImageStats(opts, meta, imdb)
%
    ---------------------------------------------------------------------

train = find(imdb.images.set == 1) ;
batch = 1:length(train);
fn = getBatchFn(opts, meta) ;
train = train(1: 100: end);
avg = {};
for i = 1:length(train)
    temp = fn(imdb, batch(train(i):train(i)+99)) ;
    temp = temp{2};
    avg{end+1} = mean(temp, 4) ;
end

%averageImage = mean(cat(4,avg{:}),4) ;

%averageImage = gather(averageImage);
```

```
function net = prepareDINet(net,opts)

fc8l = cellfun(@(a) strcmp(a.name, 'fc8'), net.layers)==1;

nCls = 5;
sizeW = size(net.layers{fc8l}.weights{1});
if sizeW(4)~=nCls
  net.layers{fc8l}.weights = {zeros(sizeW(1),sizeW(2),sizeW(3),nCls,'
      single'), ...
    zeros(1, nCls, 'single')};
end

net.layers{end} = struct('name','loss', 'type','softmaxloss') ;

net = dagnn.DagNN.fromSimpleNN(net, 'canonicalNames', true) ;

net.addLayer('top1err', dagnn.Loss('loss', 'classerror'), ...
    {'prediction','label'}, 'top1err') ;
net.addLayer('top5err', dagnn.Loss('loss', 'topkerror', ...
```

```
    'opts', {'topK',5}), ...
{'prediction','label'}, 'top5err') ;
```

SUMMARY

Whether a person is speaking privately with family members or giving a presentation at a conference, emotion is an inevitable element of speech. One of the most important functions of emotion is to support interpersonal communication. The appropriate use of emotional expression helps to achieve better communication, enhance friendship and mutual respect, and improve relationships. Because of the significant impact of emotion on the human exchange of information, the recognition and understanding of emotions in communication behavior has become a prominent multidisciplinary research topic. The earliest modern scientific studies on emotion can be traced back to the work by Darwin (see Darwin, 1872). Following Darwin, emotion studies were dominated by behavioral psychologists for more than one hundred years. In this field, William James established the contemporary research theory of emotion (cf. James, 1884). Since that point, the topic has expanded to a variety of disciplines (see Tao & Tan, 2005). Given the wide range of emotional information that a listener receives from speech, it is not surprising that researchers from a range of disciplines are interested in studying speech emotion.

In this research, we aimed to create a novel method for a computer to recognize emotion through non-verbal speech cues in the Mandarin language. The objective was to enable the computer to detect a Mandarin speaker's different emotional states. Our goal was to identify an alternative to current methods that accurately characterize non-verbal speech emotion in several languages. In this study, we disregarded the verbal aspects of speech and primarily focused on the non-verbal aspects.

Speech emotion recognition could have important applications in people's daily lives. However, due to SER's current limited accuracy, SER approaches need improvement, which was the basis for our PS. For our Problem Statement (PS), which reads as follows,

PS: To what extent can we improve SER accuracy using spectrogram information?

From the PS we formulated four RQs.

RQ1: Is it possible to design a new algorithm that improves the accuracy of detecting the voiced part activity in speech?

RQ2: How can we use two-dimensional features to analyze the spectrogram representation of speech?

RQ3: Can we extract additional, and likely less-intensive features via the composition of Gabor filters through a spectrogram?

RQ4: Can we apply the deep-learning method to the spectrogram outcomes to extract "visual" features to increase the accuracy of SER?

The answers to the research questions enable us to formulate our conclusion to the problem statement.

Chapter 1 describes the research topic. It provides an overview of SER and a description of the implementation of the algorithms. The PS and four RQs are outlined. Our research methodology is subsequently described, and the major contributions are listed.

Chapter 2 reviews the definition of emotion and the emotional states that are used in our experiments. It further describes how emotion affects expression in speech.

Chapter 3 presents an overview of the three stages of SER. First, we review feature extraction and the most commonly used acoustic features in SER. We subsequently provide details regarding the feature extraction method used in our research. Second, feature selection is reviewed. Third, the classification algorithms are analyzed.

Chapter 4 provides a brief explanation of the tools and techniques used in this study. We first describe the databases chosen for our experiments. The spectrogram and log-Gabor filters are then introduced as the key tools in our research. They are used extensively in the experiments.

Chapter 5 answers RQ1. We propose a new algorithm, the VSS algorithm, which produces a more accurate segmentation of speech signals by dealing with the voiced signal segments as image processing. In Section 5.1, we briefly review the crucial emotional information contained in the voiced and unvoiced aspects of speech. We observe that the most important features of speech are the voiced aspects. However, there are no clear boundaries between voiced and unvoiced aspects of speech. Current methods primarily rely on the exploitation of the intensity of speech signals for SER and produce unsatisfactory results. Researchers are calling for more precision regarding the boundary between the voiced and unvoiced aspects of speech. Thus, if we would be able to improve the performance of VAD, we could consequently influence and enhance feature extraction for SER. To this end, our study proposes a new algorithm, the VSS algorithm, which can produce an accurate segmentation of speech signals by using log-Gabor filters to detect voiced aspects of speech on a spectrogram. The VSS algorithm is evaluated by (1) a comparison with the current most successful VAD algorithm and (2) a comparison of SER performance with and without the VSS algorithm. The first comparison showed that the VSS algorithm is a more accurate algorithm for voiced segment detection than the one currently in use. Our research results indicate that the VSS algorithm improves SER accuracy.

Chapter 6 answers RQ2. We describe the primary log-Gabor filter algorithm, which uses log-Gabor filters to extract the spectro-temporal features of the emotional information from a spectrogram. A spectrogram can visually dis-

play the combined time, frequency, and energy information in an image of a speech signal. Specifically, in each spectrogram image, vertical and horizontal axes represent the time and frequency, while colors represent the energy of the speech signal. As a spectrogram simultaneously illustrates multiple indicators of the speech signal, we believe that it has the potential to produce new feature groups that we have not encountered previously. Thus, the second RQ sought to identify a new kind of feature containing efficient emotional information that does not overlap with the existing feature group. The unique pattern that we designed for each type of emotion in the spectrogram was illustrated. The recognition performance demonstrated that the new features are efficient for the feature group.

Chapter 7 answers RQ3. This chapter proposes a further study seeking to categorize emotional expressions in a sentence into primary and less intensive emotional expressions according to their intensiveness. The chapter reveals the feature patterns of the subsequent emotional expressions in a spectrogram. We use log-Gabor filters to identify and extract the subsequent feature patterns for different emotions on a spectrogram. Whereas Chapter 6 concentrated on the parts of a sentence that demonstrate intensive expression of emotion, in this chapter, we further conduct feature extraction using additional Gabor filters on the feature patterns of less intensive emotional expressions.

Chapter 8 answers RQ4. It describes the CNN algorithm and the reasons that it is necessary for our work. First, the architecture of the neural network is illustrated. We then explain the details of feature learning from a spectrogram using the CNN. The classification of data is subsequently demonstrated for each type of emotion. Finally, the comparison of the algorithms' performance is analyzed.

Chapter 9 provides an answer to the PS. The chapter begins by summarizing the answers to RQ1, RQ2, RQ3, and RQ4. We then review the PS, formulate conclusions, and offer six recommendations for future investigation aimed at further improving SER.

SAMENVATTING

Of iemand nu privé met familieleden spreekt of een presentatie geeft tijdens een conferentie, emotie is een onvermijdelijk element van spreken. Een van de belangrijkste functies van emotie is het ondersteunen van interpersoonlijke communicatie. Het juiste gebruik van emotionele expressie helpt om betere communicatie tot stand te brengen, vriendschap en wederzijds respect te verbeteren en relaties te versterken. Vanwege de significante invloed van emotie op de menselijke uitwisseling van informatie, is de herkenning en het begrip van emoties in communicatiegedrag een prominent multidisciplinair onderzoeksonderwerp geworden. De vroegste moderne wetenschappelijke studies over emotie zijn terug te voeren op het werk van Darwin (see Darwin, 1872). Na Darwin werden emotiestudies gedomineerd door gedragspsychologen gedurende meer dan honderd jaar. Op dit gebied heeft William James de hedendaagse onderzoekstheorie van de emotie gevestigd (cf. James, 1884). Sindsdien is het onderwerp uitgebreid tot een verscheidenheid aan disciplines (see Tao & Tan, 2005). Gezien het brede scala aan emotionele informatie die een luisteraar van spraak ontvangt, is het niet verrassend dat onderzoekers uit verschillende disciplines geïnteresseerd zijn in het bestuderen van spraakemotie.

In dit onderzoek hebben we geprobeerd een nieuwe methode te ontwikkelen voor een computer om emoties te herkennen aan de hand van non-verbale spraakelementen in het Mandarijn. Het doel was om de computer in staat te stellen de verschillende emotionele toestanden van een Mandarijnspreker te detecteren. Ons doel was ook om een alternatief te vinden voor de huidige methoden die de non-verbale spraakemotie in verschillende talen accuraat karakteriseren. In deze studie negeerden we de verbale aspecten van spraak en richten ons in eerste instantie op de non-verbale aspecten.

Spraak-emotieherkenning kan belangrijke toepassingen hebben in het dagelijks leven van mensen. De huidige nauwkeurigheid van de SER is beperk, daarom behoeft de SER-aanpak verbetering. Dit was de basis voor onze Probleemstelling (PS). Voor onze Probleemstelling (PS), die zei,

PS: In hoeverre kunnen we de nauwkeurigheid van de SER verbeteren met behulp van spectrogram-informatie?

Van de PS hebben we vier Onderzoeksvragen (OVs) afgeleid.

OV1: Is het mogelijk om een nieuw algoritme te ontwerpen dat de nauwkeurigheid van het detecteren van de stemhebbende onderdeelactiviteit in spraak verbetert?

OV2: Hoe kunnen we tweedimensionale functies gebruiken om de spectrogramrepresentatie van spraak te analyseren?

OV3: Kunnen we additionele en waarschijnlijk minder intensieve functies extraheren via de samenstelling van Gabor-filters met behulp van een spectrogram?

OV4: Kunnen we de deep-learning methode toepassen op de spectogram-uitkomsten om "visuele" functies te extraheren om de nauwkeurigheid van SER te vergroten?

De antwoorden op de onderzoeksvragen stellen ons in staat onze conclusie te formuleren voor de probleemstelling.

Hoofdstuk 1 beschrijft het onderzoeksthema. Het geeft een overzicht van de SER en een beschrijving van de implementatie van de algoritmen. De PS en de vier OVs worden geformuleerd. Onze onderzoeksmethodologie wordt vervolgens beschreven. Voorts worden onze belangrijkste vondsten vermeld.

Hoofdstuk 2 beschrijft de definitie van emotie en de emotionele toestanden die in onze experimenten worden gebruikt. Het beschrijft verder hoe emotie de expressie in de spraak beinvloedt.

Hoofdstuk 3 geeft een overzicht van de drie fasen van de SER. Eerst bespreken we feature-extractie en de meest gebruikte akoestische functies in SER. Daarenboven geven we details over de functie-extractiemethode die in ons onderzoek is gebruikt. Ten tweede wordt functieselectie beoordeeld. Ten derde worden de classificatie-algoritmen geanalyseerd.

Hoofdstuk 4 geeft een korte toelichting op de hulpmiddelen en technieken die in dit onderzoek zijn gebruikt. We beschrijven eerst de databases die voor onze experimenten zijn gekozen. De spectrogram- en log-Gabor-filters worden vervolgens geïntroduceerd als de belangrijkste instrumenten in ons onderzoek. Ze worden veel gebruikt in de experimenten.

Hoofdstuk 5 antwoordt OV1. We stellen een nieuw algoritme voor, het VSS-algoritme, dat een nauwkeurige segmentatie van spraaksignalen produceert door de stemhebbende signaalsegmenten aan te pakken als beeldverwerking. In Sectie 5.1 bespreken we kort de cruciale emotionele informatie die is vervat in de stemhebbende en stemloze aspecten van spraak. We merken op dat de belangrijkste kenmerken van spraak de stemhebbende aspecten zijn. Er zijn echter geen duidelijke grenzen tussen stemhebbende en niet-stemhebbende aspecten van spraak. De huidige methoden zijn hoofdzakelijk gebaseerd op de benutting van de intensiteit van spraaksignalen voor de SER en produceren onbevredigende resultaten. Onderzoekers vragen om meer precisie met betrekking tot de grens tussen stemhebbende en stemloze aspecten. Als we aldus de prestaties van VAD zouden kunnen verbeteren, zouden we bijgevolg de feature-extractie voor SER kunnen beïnvloeden en verbeteren. Hiertoe stelt onze studie een nieuw algoritme voor, het VSS-algoritme, dat een nauwkeurige segmentatie van spraaksignalen kan produceren met behulp van log-Gabor filters om stemhebbende aspecten van spraak op een spectrogram te detecteren. Het VSS-algoritme wordt geëvalueerd door (1) een vergelijking met het meest succesvolle VAD-algoritme en (2) een vergelijking van SER-prestaties met en

zonder het VSS-algoritme. De eerste vergelijking toont aan dat het VSS-algoritme een meer accuraat algoritme is voor stemdetectiesectie dan het huidige dat wordt gebruikt. Onze onderzoeksresultaten geven aan dat het VSS-algoritme de SER-nauwkeurigheid verbetert.

Hoofdstuk 6 antwoordt OV2. We beschrijven het primaire log-Gabor filteralgoritme, dat log-Gabor-filters gebruikt om de spectro-temporele kenmerken van de emotionele informatie uit een spectrogram te extraheren. Een spectrogram kan de gecombineerde tijd-, frequentie- en energie-informatie visueel weergeven in een beeld van een spraaksignaal. Specifiek stellen in elk spectrogrambeeld de verticale en horizontale assen de tijd en frequentie voor, terwijl kleuren de energie van het spraaksignaal vertegenwoordigen. Omdat een spectrogram tegelijkertijd meerdere indicatoren van het spraaksignaal illustreert, geloven we dat het de potentie heeft nieuwe functiegroepen te produceren die we nog niet eerder zijn tegengekomen. OV2 probeert dus een nieuw soort functie te identificeren met efficiënte emotionele informatie die niet overlapt met de bestaande functiegroep. Het unieke patroon dat we hebben ontworpen voor elk type emotie in het spectrogram, wordt hier geïllustreerd. De prestaties die voor de herkenning geboekt zijn, hebben aangetoond dat de nieuwe functies efficiënt zijn voor de functiegroep.

Hoofdstuk 7 antwoordt OV3. In dit hoofdstuk wordt een verdere studie voorgesteld om emotionele expressies in een zin te categoriseren in primaire en minder intensieve emotionele expressies op basis van hun intensiviteit. Het hoofdstuk onthult de kenmerkpatronen van de daaropvolgende emotionele expressies in een spectrogram. We gebruiken log-Gabor filters om de volgende kenmerkpatronen voor verschillende emoties op een spectrogram te identificeren en extraheren. Terwijl Hoofdstuk 6 zich concentreerde op de delen van een zin die intense uitdrukking van emotie tonen, voeren we in dit hoofdstuk eerst verder feature-extractie uit met extra Gabor filters op de feature-patronen van minder intensieve emotionele expressies.

Hoofdstuk 8 antwoordt OV4. Het beschrijft het CNN-algoritme en de redenen waarom dit noodzakelijk is voor ons werk. Eerst wordt de architectuur van het neurale netwerk geïllustreerd. Vervolgens leggen we de details van het leren van functies uit vanuit een spectrogram met behulp van de CNN. De classificatie van gegevens wordt vervolgens gedemonstreerd voor elk type emotie. Ten slotte wordt de vergelijking van de prestaties van de algoritmen geanalyseerd.

Hoofdstuk 9 geeft een antwoord op de PS. Het hoofdstuk begint met een samenvatting van de antwoorden op OV1, OV2, OV3 en OV4. Vervolgens beschouwen we de PS, formuleren we conclusies en geven we aanbevelingen voor toekomstig onderzoek dat gericht is op het verder verbeteren van de SER.

CURRICULUM VITAE

Yu Gu was born in Xi'An, China on 7th, January 1987. He received a bachelor's degree in computer science from Xi Dian University and a master's degree in computer software and theory from Guang Xi University in China.

In 2012, Gu was granted a scholarship by the Chinese Scholarship Council (CSC) to pursue a Ph.D. program in the Netherlands. He subsequently enrolled at the Tilburg Centre for Cognition and Communication (TiCC) in the Faculty of Humanities at Tilburg University in the Netherlands. The results of his Ph.D. project titled "Automatic Emotion Recognition From Mandarin Speech" is reported in this thesis.

PUBLICATIONS

CONFERENCE PROCEEDINGS

Yu Gu, Eric Postma, Hai-Xiang Lin and Jaap van den Herik. Speech Emotion Recognition Using Voiced Segment Selection Algorithm. *In Proceedings of 22nd European Conference on Artificial Intelligence ECAI2016*, The Hague, the Netherlands, 2016, pp. 1682-1683.

Yu Gu, Eric Postma, Hai-Xiang Lin and Jaap van den Herik. Speech Emotion Recognition with Log-Gabor Filters. *8th International Conference on Agents and Artificial Intelligence ICAART 2016*, Rome, Italy, 2016, pp. 446-452.

Yu Gu, Eric Postma, Hai-Xiang Lin. Vocal Emotion Recognition with Log-Gabor Filters. *In Proceedings of the 5th International Workshop on Audio/Visual Emotion Challenge (AVEC '15)*. Brisbane, Australia, 2015, pp. 26-31.

Marie Postma-Nilsenova, Eric Postma and Yu Gu. No Effect of Language Experience on spectral/fundamental Listener Type Distribution: A comparison of Chinese and Dutch. *Fourth International Symposium on Tonal Aspects of Languages*. Nijmegen, the Netherlands, 2014.

Marie Postma-Nilsenova, Eric Postma, Olga Tsoumani, Yu Gu. *Biases in Auditory Perception: Listener-Specific Preference*, 2014.

SIKS DISSERTATION SERIES

1998

1 Johan van den Akker (CWI) *DEGAS: An Active, Temporal Database of Autonomous Objects*

2 Floris Wiesman (UM) *Information Retrieval by Graphically Browsing Meta-Information*

3 Ans Steuten (TUD) *A Contribution to the Linguistic Analysis of Business Conversations*

4 Dennis Breuker (UM) *Memory versus Search in Games*

5 E. W. Oskamp (RUL) *Computerondersteuning bij Straftoemeting*

1999

1 Mark Sloof (VUA) *Physiology of Quality Change Modelling: Automated modelling of*

2 Rob Potharst (EUR) *Classification using decision trees and neural nets*

3 Don Beal (UM) *The Nature of Minimax Search*

4 Jacques Penders (UM) *The practical Art of Moving Physical Objects*

5 Aldo de Moor (KUB) *Empowering Communities: A Method for the Legitimate User-Driven*

6 Niek J. E. Wijngaards (VUA) *Re-design of compositional systems*

7 David Spelt (UT) *Verification support for object database design*

8 Jacques H. J. Lenting (UM) *Informed Gambling: Conception and Analysis of a Multi-Agent Mechanism*

2000

1 Frank Niessink (VUA) *Perspectives on Improving Software Maintenance*

2 Koen Holtman (TUe) *Prototyping of CMS Storage Management*

3 Carolien M. T. Metselaar (UvA) *Sociaal-organisatorische gevolgen van kennistechnologie*

4 Geert de Haan (VUA) *ETAG, A Formal Model of Competence Knowledge for User Interface*

5 Ruud van der Pol (UM) *Knowledge-based Query Formulation in Information Retrieval*

6 Rogier van Eijk (UU) *Programming Languages for Agent Communication*

7 Niels Peek (UU) *Decision-theoretic Planning of Clinical Patient Management*

8 Veerle Coupé (EUR) *Sensitivity Analyis of Decision-Theoretic Networks*

9 Florian Waas (CWI) *Principles of Probabilistic Query Optimization*

10 Niels Nes (CWI) *Image Database Management System Design Considerations, Algorithms and Architecture*

11 Jonas Karlsson (CWI) *Scalable Distributed Data Structures for Database Management*

2001

1 Silja Renooij (UU) *Qualitative Approaches to Quantifying Probabilistic Networks*

2 Koen Hindriks (UU) *Agent Programming Languages: Programming with Mental Models*

3 Maarten van Someren (UvA) *Learning as problem solving*

4 Evgueni Smirnov (UM) *Conjunctive and Disjunctive Version Spaces with Instance-Based Boundary Sets*

5 Jacco van Ossenbruggen (VUA) *Processing Structured Hypermedia: A Matter of Style*

6 Martijn van Welie (VUA) *Task-based User Interface Design*

7 Bastiaan Schonhage (VUA) *Diva: Architectural Perspectives on Information Visualization*

8 Pascal van Eck (VUA) *A Compositional Semantic Structure for Multi-Agent Systems Dynamics*

9 Pieter Jan 't Hoen (RUL) *Towards Distributed Development of Large Object-Oriented Models*

10 Maarten Sierhuis (UvA) *Modeling and Simulating Work Practice*

11 Tom M. van Engers (VUA) *Knowledge Management*

2002

1 Nico Lassing (VUA) *Architecture-Level Modifiability Analysis*
2 Roelof van Zwol (UT) *Modelling and searching web-based document collections*
3 Henk Ernst Blok (UT) *Database Optimization Aspects for Information Retrieval*
4 Juan Roberto Castelo Valdueza (UU) *The Discrete Acyclic Digraph Markov Model in Data Mining*
5 Radu Serban (VUA) *The Private Cyberspace Modeling Electronic*
6 Laurens Mommers (UL) *Applied legal epistemology: Building a knowledge-based ontology of*
7 Peter Boncz (CWI) *Monet: A Next-Generation DBMS Kernel For Query-Intensive*
8 Jaap Gordijn (VUA) *Value Based Requirements Engineering: Exploring Innovative*
9 Willem-Jan van den Heuvel (KUB) *Integrating Modern Business Applications with Objectified Legacy*
10 Brian Sheppard (UM) *Towards Perfect Play of Scrabble*
11 Wouter C. A. Wijngaards (VUA) *Agent Based Modelling of Dynamics: Biological and Organisational Applications*
12 Albrecht Schmidt (UvA) *Processing XML in Database Systems*
13 Hongjing Wu (TUe) *A Reference Architecture for Adaptive Hypermedia Applications*
14 Wieke de Vries (UU) *Agent Interaction: Abstract Approaches to Modelling, Programming and Verifying Multi-Agent Systems*
15 Rik Eshuis (UT) *Semantics and Verification of UML Activity Diagrams for Workflow Modelling*
16 Pieter van Langen (VUA) *The Anatomy of Design: Foundations, Models and Applications*
17 Stefan Manegold (UvA) *Understanding, Modeling, and Improving Main-Memory Database Performance*

2003

1 Heiner Stuckenschmidt (VUA) *Ontology-Based Information Sharing in Weakly Structured Environments*
2 Jan Broersen (VUA) *Modal Action Logics for Reasoning About Reactive Systems*
3 Martijn Schuemie (TUD) *Human-Computer Interaction and Presence in Virtual Reality Exposure Therapy*
4 Milan Petkovic (UT) *Content-Based Video Retrieval Supported by Database Technology*
5 Jos Lehmann (UvA) *Causation in Artificial Intelligence and Law: A modelling approach*
6 Boris van Schooten (UT) *Development and specification of virtual environments*
7 Machiel Jansen (UvA) *Formal Explorations of Knowledge Intensive Tasks*
8 Yongping Ran (UM) *Repair Based Scheduling*
9 Rens Kortmann (UM) *The resolution of visually guided behaviour*
10 Andreas Lincke (UvT) *Electronic Business Negotiation: Some experimental studies on the interaction between medium, innovation context and culture*
11 Simon Keizer (UT) *Reasoning under Uncertainty in Natural Language Dialogue using Bayesian Networks*
12 Roeland Ordelman (UT) *Dutch speech recognition in multimedia information retrieval*
13 Jeroen Donkers (UM) *Nosce Hostem: Searching with Opponent Models*
14 Stijn Hoppenbrouwers (KUN) *Freezing Language: Conceptualisation Processes across ICT-Supported Organisations*
15 Mathijs de Weerdt (TUD) *Plan Merging in Multi-Agent Systems*
16 Menzo Windhouwer (CWI) *Feature Grammar Systems: Incremental Maintenance of Indexes to Digital Media Warehouses*
17 David Jansen (UT) *Extensions of Statecharts with Probability, Time, and Stochastic Timing*
18 Levente Kocsis (UM) *Learning Search Decisions*

2004

1 Virginia Dignum (UU) *A Model for Organizational Interaction: Based on Agents, Founded in Logic*
2 Lai Xu (UvT) *Monitoring Multi-party Contracts for E-business*
3 Perry Groot (VUA) *A Theoretical and Empirical Analysis of Approximation in Symbolic Problem Solving*
4 Chris van Aart (UvA) *Organizational Principles for Multi-Agent Architectures*
5 Viara Popova (EUR) *Knowledge discovery and monotonicity*
6 Bart-Jan Hommes (TUD) *The Evaluation of Business Process Modeling Techniques*
7 Elise Boltjes (UM) *Voorbeeldig onderwijs: voorbeeldgestuurd onderwijs, een opstap naar abstract denken, vooral voor meisjes*

8 Joop Verbeek (UM) *Politie en de Nieuwe Internationale Informatiemarkt, Grensregionale politiële gegevensuitwisseling en digitale expertise*
9 Martin Caminada (VUA) *For the Sake of the Argument: explorations into argument-based reasoning*
10 Suzanne Kabel (UvA) *Knowledge-rich indexing of learning-objects*
11 Michel Klein (VUA) *Change Management for Distributed Ontologies*
12 The Duy Bui (UT) *Creating emotions and facial expressions for embodied agents*
13 Wojciech Jamroga (UT) *Using Multiple Models of Reality: On Agents who Know how to Play*
14 Paul Harrenstein (UU) *Logic in Conflict. Logical Explorations in Strategic Equilibrium*
15 Arno Knobbe (UU) *Multi-Relational Data Mining*
16 Federico Divina (VUA) *Hybrid Genetic Relational Search for Inductive Learning*
17 Mark Winands (UM) *Informed Search in Complex Games*
18 Vania Bessa Machado (UvA) *Supporting the Construction of Qualitative Knowledge Models*
19 Thijs Westerveld (UT) *Using generative probabilistic models for multimedia retrieval*
20 Madelon Evers (Nyenrode) *Learning from Design: facilitating multidisciplinary design teams*

2005

1 Floor Verdenius (UvA) *Methodological Aspects of Designing Induction-Based Applications*
2 Erik van der Werf (UM) *AI techniques for the game of Go*
3 Franc Grootjen (RUN) *A Pragmatic Approach to the Conceptualisation of Language*
4 Nirvana Meratnia (UT) *Towards Database Support for Moving Object data*
5 Gabriel Infante-Lopez (UvA) *Two-Level Probabilistic Grammars for Natural Language Parsing*
6 Pieter Spronck (UM) *Adaptive Game AI*
7 Flavius Frasincar (TUe) *Hypermedia Presentation Generation for Semantic Web Information Systems*
8 Richard Vdovjak (TUe) *A Model-driven Approach for Building Distributed Ontology-based Web Applications*
9 Jeen Broekstra (VUA) *Storage, Querying and Inferencing for Semantic Web Languages*
10 Anders Bouwer (UvA) *Explaining Behaviour: Using Qualitative Simulation in Interactive Learning Environments*
11 Elth Ogston (VUA) *Agent Based Matchmaking and Clustering: A Decentralized Approach to Search*
12 Csaba Boer (EUR) *Distributed Simulation in Industry*
13 Fred Hamburg (UL) *Een Computermodel voor het Ondersteunen van Euthanasiebeslissingen*
14 Borys Omelayenko (VUA) *Web-Service configuration on the Semantic Web: Exploring how semantics meets pragmatics*
15 Tibor Bosse (VUA) *Analysis of the Dynamics of Cognitive Processes*
16 Joris Graaumans (UU) *Usability of XML Query Languages*
17 Boris Shishkov (TUD) *Software Specification Based on Re-usable Business Components*
18 Danielle Sent (UU) *Test-selection strategies for probabilistic networks*
19 Michel van Dartel (UM) *Situated Representation*
20 Cristina Coteanu (UL) *Cyber Consumer Law, State of the Art and Perspectives*
21 Wijnand Derks (UT) *Improving Concurrency and Recovery in Database Systems by Exploiting Application Semantics*

2006

1 Samuil Angelov (TUe) *Foundations of B2B Electronic Contracting*
2 Cristina Chisalita (VUA) *Contextual issues in the design and use of information technology in organizations*
3 Noor Christoph (UvA) *The role of metacognitive skills in learning to solve problems*
4 Marta Sabou (VUA) *Building Web Service Ontologies*
5 Cees Pierik (UU) *Validation Techniques for Object-Oriented Proof Outlines*
6 Ziv Baida (VUA) *Software-aided Service Bundling: Intelligent Methods & Tools for Graphical Service Modeling*
7 Marko Smiljanic (UT) *XML schema matching: balancing efficiency and effectiveness by means of clustering*
8 Eelco Herder (UT) *Forward, Back and Home Again: Analyzing User Behavior on the Web*

9 Mohamed Wahdan (UM) *Automatic Formulation of the Auditor's Opinion*
10 Ronny Siebes (VUA) *Semantic Routing in Peer-to-Peer Systems*
11 Joeri van Ruth (UT) *Flattening Queries over Nested Data Types*
12 Bert Bongers (VUA) *Interactivation: Towards an e-cology of people, our technological environment, and the arts*
13 Henk-Jan Lebbink (UU) *Dialogue and Decision Games for Information Exchanging Agents*
14 Johan Hoorn (VUA) *Software Requirements: Update, Upgrade, Redesign - towards a Theory of Requirements Change*
15 Rainer Malik (UU) *CONAN: Text Mining in the Biomedical Domain*
16 Carsten Riggelsen (UU) *Approximation Methods for Efficient Learning of Bayesian Networks*
17 Stacey Nagata (UU) *User Assistance for Multitasking with Interruptions on a Mobile Device*
18 Valentin Zhizhkun (UvA) *Graph transformation for Natural Language Processing*
19 Birna van Riemsdijk (UU) *Cognitive Agent Programming: A Semantic Approach*
20 Marina Velikova (UvT) *Monotone models for prediction in data mining*
21 Bas van Gils (RUN) *Aptness on the Web*
22 Paul de Vrieze (RUN) *Fundaments of Adaptive Personalisation*
23 Ion Juvina (UU) *Development of Cognitive Model for Navigating on the Web*
24 Laura Hollink (VUA) *Semantic Annotation for Retrieval of Visual Resources*
25 Madalina Drugan (UU) *Conditional log-likelihood MDL and Evolutionary MCMC*
26 Vojkan Mihajlovic (UT) *Score Region Algebra: A Flexible Framework for Structured Information Retrieval*
27 Stefano Bocconi (CWI) *Vox Populi: generating video documentaries from semantically annotated media repositories*
28 Borkur Sigurbjornsson (UvA) *Focused Information Access using XML Element Retrieval*

2007

1 Kees Leune (UvT) *Access Control and Service-Oriented Architectures*
2 Wouter Teepe (RUG) *Reconciling Information Exchange and Confidentiality: A Formal Approach*
3 Peter Mika (VUA) *Social Networks and the Semantic Web*
4 Jurriaan van Diggelen (UU) *Achieving Semantic Interoperability in Multi-agent Systems: a dialogue-based approach*
5 Bart Schermer (UL) *Software Agents, Surveillance, and the Right to Privacy: a Legislative Framework for Agent-enabled Surveillance*
6 Gilad Mishne (UvA) *Applied Text Analytics for Blogs*
7 Natasa Jovanovic' (UT) *To Whom It May Concern: Addressee Identification in Face-to-Face Meetings*
8 Mark Hoogendoorn (VUA) *Modeling of Change in Multi-Agent Organizations*
9 David Mobach (VUA) *Agent-Based Mediated Service Negotiation*
10 Huib Aldewereld (UU) *Autonomy vs. Conformity: an Institutional Perspective on Norms and Protocols*
11 Natalia Stash (TUe) *Incorporating Cognitive/Learning Styles in a General-Purpose Adaptive Hypermedia System*
12 Marcel van Gerven (RUN) *Bayesian Networks for Clinical Decision Support: A Rational Approach to Dynamic Decision-Making under Uncertainty*
13 Rutger Rienks (UT) *Meetings in Smart Environments: Implications of Progressing Technology*
14 Niek Bergboer (UM) *Context-Based Image Analysis*
15 Joyca Lacroix (UM) *NIM: a Situated Computational Memory Model*
16 Davide Grossi (UU) *Designing Invisible Handcuffs. Formal investigations in Institutions and Organizations for Multi-agent Systems*
17 Theodore Charitos (UU) *Reasoning with Dynamic Networks in Practice*
18 Bart Orriens (UvT) *On the development an management of adaptive business collaborations*
19 David Levy (UM) *Intimate relationships with artificial partners*
20 Slinger Jansen (UU) *Customer Configuration Updating in a Software Supply Network*

21 Karianne Vermaas (UU) *Fast diffusion and broadening use: A research on residential adoption and usage of broadband internet in the Netherlands between 2001 and 2005*

22 Zlatko Zlatev (UT) *Goal-oriented design of value and process models from patterns*

23 Peter Barna (TUe) *Specification of Application Logic in Web Information Systems*

24 Georgina Ramírez Camps (CWI) *Structural Features in XML Retrieval*

25 Joost Schalken (VUA) *Empirical Investigations in Software Process Improvement*

2008

1 Katalin Boer-Sorbán (EUR) *Agent-Based Simulation of Financial Markets: A modular, continuous-time approach*

2 Alexei Sharpanskykh (VUA) *On Computer-Aided Methods for Modeling and Analysis of Organizations*

3 Vera Hollink (UvA) *Optimizing hierarchical menus: a usage-based approach*

4 Ander de Keijzer (UT) *Management of Uncertain Data: towards unattended integration*

5 Bela Mutschler (UT) *Modeling and simulating causal dependencies on process-aware information systems from a cost perspective*

6 Arjen Hommersom (RUN) *On the Application of Formal Methods to Clinical Guidelines, an Artificial Intelligence Perspective*

7 Peter van Rosmalen (OU) *Supporting the tutor in the design and support of adaptive e-learning*

8 Janneke Bolt (UU) *Bayesian Networks: Aspects of Approximate Inference*

9 Christof van Nimwegen (UU) *The paradox of the guided user: assistance can be counter-effective*

10 Wauter Bosma (UT) *Discourse oriented summarization*

11 Vera Kartseva (VUA) *Designing Controls for Network Organizations: A Value-Based Approach*

12 Jozsef Farkas (RUN) *A Semiotically Oriented Cognitive Model of Knowledge Representation*

13 Caterina Carraciolo (UvA) *Topic Driven Access to Scientific Handbooks*

14 Arthur van Bunningen (UT) *Context-Aware Querying: Better Answers with Less Effort*

15 Martijn van Otterlo (UT) *The Logic of Adaptive Behavior: Knowledge Representation and Algorithms for the Markov Decision Process Framework in First-Order Domains*

16 Henriette van Vugt (VUA) *Embodied agents from a user's perspective*

17 Martin Op 't Land (TUD) *Applying Architecture and Ontology to the Splitting and Allying of Enterprises*

18 Guido de Croon (UM) *Adaptive Active Vision*

19 Henning Rode (UT) *From Document to Entity Retrieval: Improving Precision and Performance of Focused Text Search*

20 Rex Arendsen (UvA) *Geen bericht, goed bericht. Een onderzoek naar de effecten van de introductie van elektronisch berichtenverkeer met de overheid op de administratieve lasten van bedrijven*

21 Krisztian Balog (UvA) *People Search in the Enterprise*

22 Henk Koning (UU) *Communication of IT-Architecture*

23 Stefan Visscher (UU) *Bayesian network models for the management of ventilator-associated pneumonia*

24 Zharko Aleksovski (VUA) *Using background knowledge in ontology matching*

25 Geert Jonker (UU) *Efficient and Equitable Exchange in Air Traffic Management Plan Repair using Spender-signed Currency*

26 Marijn Huijbregts (UT) *Segmentation, Diarization and Speech Transcription: Surprise Data Unraveled*

27 Hubert Vogten (OU) *Design and Implementation Strategies for IMS Learning Design*

28 Ildiko Flesch (RUN) *On the Use of Independence Relations in Bayesian Networks*

29 Dennis Reidsma (UT) *Annotations and Subjective Machines: Of Annotators, Embodied Agents, Users, and Other Humans*

30 Wouter van Atteveldt (VUA) *Semantic Network Analysis: Techniques for Extracting, Representing and Querying Media Content*

31 Loes Braun (UM) *Pro-Active Medical Information Retrieval*

32 Trung H. Bui (UT) *Toward Affective Dialogue Management using Partially Observable Markov Decision Processes*

33 Frank Terpstra (UvA) *Scientific Workflow Design: theoretical and practical issues*

34 Jeroen de Knijf (UU) *Studies in Frequent Tree Mining*

35 Ben Torben Nielsen (UvT) *Dendritic morphologies: function shapes structure*

2009

1 Rasa Jurgelenaite (RUN) *Symmetric Causal Independence Models*

2 Willem Robert van Hage (VUA) *Evaluating Ontology-Alignment Techniques*

3 Hans Stol (UvT) *A Framework for Evidence-based Policy Making Using IT*

4 Josephine Nabukenya (RUN) *Improving the Quality of Organisational Policy Making using Collaboration Engineering*

5 Sietse Overbeek (RUN) *Bridging Supply and Demand for Knowledge Intensive Tasks: Based on Knowledge, Cognition, and Quality*

6 Muhammad Subianto (UU) *Understanding Classification*

7 Ronald Poppe (UT) *Discriminative Vision-Based Recovery and Recognition of Human Motion*

8 Volker Nannen (VUA) *Evolutionary Agent-Based Policy Analysis in Dynamic Environments*

9 Benjamin Kanagwa (RUN) *Design, Discovery and Construction of Service-oriented Systems*

10 Jan Wielemaker (UvA) *Logic programming for knowledge-intensive interactive applications*

11 Alexander Boer (UvA) *Legal Theory, Sources of Law & the Semantic Web*

12 Peter Massuthe (TUE, Humboldt-Universitaet zu Berlin) *Operating Guidelines for Services*

13 Steven de Jong (UM) *Fairness in Multi-Agent Systems*

14 Maksym Korotkiy (VUA) *From ontology-enabled services to service-enabled ontologies (making ontologies work in e-science with ONTO-SOA)*

15 Rinke Hoekstra (UvA) *Ontology Representation: Design Patterns and Ontologies that Make Sense*

16 Fritz Reul (UvT) *New Architectures in Computer Chess*

17 Laurens van der Maaten (UvT) *Feature Extraction from Visual Data*

18 Fabian Groffen (CWI) *Armada, An Evolving Database System*

19 Valentin Robu (CWI) *Modeling Preferences, Strategic Reasoning and Collaboration in Agent-Mediated Electronic Markets*

20 Bob van der Vecht (UU) *Adjustable Autonomy: Controling Influences on Decision Making*

21 Stijn Vanderlooy (UM) *Ranking and Reliable Classification*

22 Pavel Serdyukov (UT) *Search For Expertise: Going beyond direct evidence*

23 Peter Hofgesang (VUA) *Modelling Web Usage in a Changing Environment*

24 Annerieke Heuvelink (VUA) *Cognitive Models for Training Simulations*

25 Alex van Ballegooij (CWI) *RAM: Array Database Management through Relational Mapping*

26 Fernando Koch (UU) *An Agent-Based Model for the Development of Intelligent Mobile Services*

27 Christian Glahn (OU) *Contextual Support of social Engagement and Reflection on the Web*

28 Sander Evers (UT) *Sensor Data Management with Probabilistic Models*

29 Stanislav Pokraev (UT) *Model-Driven Semantic Integration of Service-Oriented Applications*

30 Marcin Zukowski (CWI) *Balancing vectorized query execution with bandwidth-optimized storage*

31 Sofiya Katrenko (UvA) *A Closer Look at Learning Relations from Text*

32 Rik Farenhorst (VUA) *Architectural Knowledge Management: Supporting Architects and Auditors*

33 Khiet Truong (UT) *How Does Real Affect Affect Affect Recognition In Speech?*

34 Inge van de Weerd (UU) *Advancing in Software Product Management: An Incremental Method Engineering Approach*

35 Wouter Koelewijn (UL) *Privacy en Politiegegevens: Over geautomatiseerde normatieve informatie-uitwisseling*

36 Marco Kalz (OUN) *Placement Support for Learners in Learning Networks*

37 Hendrik Drachsler (OUN) *Navigation Support for Learners in Informal Learning Networks*

38 Riina Vuorikari (OU) *Tags and self-organisation: a metadata ecology for learning resources in a multilingual context*

39 Christian Stahl (TUE, Humboldt-Universitaet zu Berlin) *Service Substitution: A Behavioral Approach Based on Petri Nets*

40 Stephan Raaijmakers (UvT) *Multinomial Language Learning: Investigations into the Geometry of Language*

41 Igor Berezhnyy (UvT) *Digital Analysis of Paintings*

42 Toine Bogers (UvT) *Recommender Systems for Social Bookmarking*

43 Virginia Nunes Leal Franqueira (UT) *Finding Multi-step Attacks in Computer Networks using Heuristic Search and Mobile Ambients*

44 Roberto Santana Tapia (UT) *Assessing Business-IT Alignment in Networked Organizations*

45 Jilles Vreeken (UU) *Making Pattern Mining Useful*

46 Loredana Afanasiev (UvA) *Querying XML: Benchmarks and Recursion*

2010

1 Matthijs van Leeuwen (UU) *Patterns that Matter*

2 Ingo Wassink (UT) *Work flows in Life Science*

3 Joost Geurts (CWI) *A Document Engineering Model and Processing Framework for Multimedia documents*

4 Olga Kulyk (UT) *Do You Know What I Know? Situational Awareness of Co-located Teams in Multidisplay Environments*

5 Claudia Hauff (UT) *Predicting the Effectiveness of Queries and Retrieval Systems*

6 Sander Bakkes (UvT) *Rapid Adaptation of Video Game AI*

7 Wim Fikkert (UT) *Gesture interaction at a Distance*

8 Krzysztof Siewicz (UL) *Towards an Improved Regulatory Framework of Free Software. Protecting user freedoms in a world of software communities and eGovernments*

9 Hugo Kielman (UL) *A Politiele gegevensverwerking en Privacy, Naar een effectieve waarborging*

10 Rebecca Ong (UL) *Mobile Communication and Protection of Children*

11 Adriaan Ter Mors (TUD) *The world according to MARP: Multi-Agent Route Planning*

12 Susan van den Braak (UU) *Sensemaking software for crime analysis*

13 Gianluigi Folino (RUN) *High Performance Data Mining using Bio-inspired techniques*

14 Sander van Splunter (VUA) *Automated Web Service Reconfiguration*

15 Lianne Bodenstaff (UT) *Managing Dependency Relations in Inter-Organizational Models*

16 Sicco Verwer (TUD) *Efficient Identification of Timed Automata, theory and practice*

17 Spyros Kotoulas (VUA) *Scalable Discovery of Networked Resources: Algorithms, Infrastructure, Applications*

18 Charlotte Gerritsen (VUA) *Caught in the Act: Investigating Crime by Agent-Based Simulation*

19 Henriette Cramer (UvA) *People's Responses to Autonomous and Adaptive Systems*

20 Ivo Swartjes (UT) *Whose Story Is It Anyway? How Improv Informs Agency and Authorship of Emergent Narrative*

21 Harold van Heerde (UT) *Privacy-aware data management by means of data degradation*

22 Michiel Hildebrand (CWI) *End-user Support for Access to Heterogeneous Linked Data*

23 Bas Steunebrink (UU) *The Logical Structure of Emotions*

24 Zulfiqar Ali Memon (VUA) *Modelling Human-Awareness for Ambient Agents: A Human Mindreading Perspective*

25 Ying Zhang (CWI) *XRPC: Efficient Distributed Query Processing on Heterogeneous XQuery Engines*

26 Marten Voulon (UL) *Automatisch contracteren*

27 Arne Koopman (UU) *Characteristic Relational Patterns*

28 Stratos Idreos (CWI) *Database Cracking: Towards Auto-tuning Database Kernels*

29 Marieke van Erp (UvT) *Accessing Natural History: Discoveries in data cleaning, structuring, and retrieval*

30 Victor de Boer (UvA) *Ontology Enrichment from Heterogeneous Sources on the Web*

31 Marcel Hiel (UvT) *An Adaptive Service Oriented Architecture: Automatically solving Interoperability Problems*

32 Robin Aly (UT) *Modeling Representation Uncertainty in Concept-Based Multimedia Retrieval*

33 Teduh Dirgahayu (UT) *Interaction Design in Service Compositions*

34 Dolf Trieschnigg (UT) *Proof of Concept: Concept-based Biomedical Information Retrieval*

35 Jose Janssen (OU) *Paving the Way for Lifelong Learning: Facilitating competence development through a learning path specification*
36 Niels Lohmann (TUe) *Correctness of services and their composition*
37 Dirk Fahland (TUe) *From Scenarios to components*
38 Ghazanfar Farooq Siddiqui (VUA) *Integrative modeling of emotions in virtual agents*
39 Mark van Assem (VUA) *Converting and Integrating Vocabularies for the Semantic Web*
40 Guillaume Chaslot (UM) *Monte-Carlo Tree Search*
41 Sybren de Kinderen (VUA) *Needs-driven service bundling in a multi-supplier setting: the computational e3-service approach*
42 Peter van Kranenburg (UU) *A Computational Approach to Content-Based Retrieval of Folk Song Melodies*
43 Pieter Bellekens (TUe) *An Approach towards Context-sensitive and User-adapted Access to Heterogeneous Data Sources, Illustrated in the Television Domain*
44 Vasilios Andrikopoulos (UvT) *A theory and model for the evolution of software services*
45 Vincent Pijpers (VUA) *e3alignment: Exploring Inter-Organizational Business-ICT Alignment*
46 Chen Li (UT) *Mining Process Model Variants: Challenges, Techniques, Examples*
47 Jahn-Takeshi Saito (UM) *Solving difficult game positions*
48 Bouke Huurnink (UvA) *Search in Audiovisual Broadcast Archives*
49 Alia Khairia Amin (CWI) *Understanding and supporting information seeking tasks in multiple sources*
50 Peter-Paul van Maanen (VUA) *Adaptive Support for Human-Computer Teams: Exploring the Use of Cognitive Models of Trust and Attention*
51 Edgar Meij (UvA) *Combining Concepts and Language Models for Information Access*

2011

1 Botond Cseke (RUN) *Variational Algorithms for Bayesian Inference in Latent Gaussian Models*
2 Nick Tinnemeier (UU) *Organizing Agent Organizations. Syntax and Operational Semantics of an Organization-Oriented Programming Language*
3 Jan Martijn van der Werf (TUe) *Compositional Design and Verification of Component-Based Information Systems*
4 Hado van Hasselt (UU) *Insights in Reinforcement Learning: Formal analysis and empirical evaluation of temporal-difference*
5 Base van der Raadt (VUA) *Enterprise Architecture Coming of Age: Increasing the Performance of an Emerging Discipline*
6 Yiwen Wang (TUe) *Semantically-Enhanced Recommendations in Cultural Heritage*
7 Yujia Cao (UT) *Multimodal Information Presentation for High Load Human Computer Interaction*
8 Nieske Vergunst (UU) *BDI-based Generation of Robust Task-Oriented Dialogues*
9 Tim de Jong (OU) *Contextualised Mobile Media for Learning*
10 Bart Bogaert (UvT) *Cloud Content Contention*
11 Dhaval Vyas (UT) *Designing for Awareness: An Experience-focused HCI Perspective*
12 Carmen Bratosin (TUe) *Grid Architecture for Distributed Process Mining*
13 Xiaoyu Mao (UvT) *Airport under Control. Multiagent Scheduling for Airport Ground Handling*
14 Milan Lovric (EUR) *Behavioral Finance and Agent-Based Artificial Markets*
15 Marijn Koolen (UvA) *The Meaning of Structure: the Value of Link Evidence for Information Retrieval*
16 Maarten Schadd (UM) *Selective Search in Games of Different Complexity*
17 Jiyin He (UvA) *Exploring Topic Structure: Coherence, Diversity and Relatedness*
18 Mark Ponsen (UM) *Strategic Decision-Making in complex games*
19 Ellen Rusman (OU) *The Mind ' s Eye on Personal Profiles*
20 Qing Gu (VUA) *Guiding service-oriented software engineering: A view-based approach*
21 Linda Terlouw (TUD) *Modularization and Specification of Service-Oriented Systems*
22 Junte Zhang (UvA) *System Evaluation of Archival Description and Access*
23 Wouter Weerkamp (UvA) *Finding People and their Utterances in Social Media*
24 Herwin van Welbergen (UT) *Behavior Generation for Interpersonal Coordination with Virtual Humans On Specifying, Scheduling and Realizing Multimodal Virtual Human Behavior*

25 Syed Waqar ul Qounain Jaffry (VUA) *Analysis and Validation of Models for Trust Dynamics*

26 Matthijs Aart Pontier (VUA) *Virtual Agents for Human Communication: Emotion Regulation and Involvement-Distance Trade-Offs in Embodied Conversational Agents and Robots*

27 Aniel Bhulai (VUA) *Dynamic website optimization through autonomous management of design patterns*

28 Rianne Kaptein (UvA) *Effective Focused Retrieval by Exploiting Query Context and Document Structure*

29 Faisal Kamiran (TUe) *Discrimination-aware Classification*

30 Egon van den Broek (UT) *Affective Signal Processing (ASP): Unraveling the mystery of emotions*

31 Ludo Waltman (EUR) *Computational and Game-Theoretic Approaches for Modeling Bounded Rationality*

32 Nees-Jan van Eck (EUR) *Methodological Advances in Bibliometric Mapping of Science*

33 Tom van der Weide (UU) *Arguing to Motivate Decisions*

34 Paolo Turrini (UU) *Strategic Reasoning in Interdependence: Logical and Game-theoretical Investigations*

35 Maaike Harbers (UU) *Explaining Agent Behavior in Virtual Training*

36 Erik van der Spek (UU) *Experiments in serious game design: a cognitive approach*

37 Adriana Burlutiu (RUN) *Machine Learning for Pairwise Data, Applications for Preference Learning and Supervised Network Inference*

38 Nyree Lemmens (UM) *Bee-inspired Distributed Optimization*

39 Joost Westra (UU) *Organizing Adaptation using Agents in Serious Games*

40 Viktor Clerc (VUA) *Architectural Knowledge Management in Global Software Development*

41 Luan Ibraimi (UT) *Cryptographically Enforced Distributed Data Access Control*

42 Michal Sindlar (UU) *Explaining Behavior through Mental State Attribution*

43 Henk van der Schuur (UU) *Process Improvement through Software Operation Knowledge*

44 Boris Reuderink (UT) *Robust Brain-Computer Interfaces*

45 Herman Stehouwer (UvT) *Statistical Language Models for Alternative Sequence Selection*

46 Beibei Hu (TUD) *Towards Contextualized Information Delivery: A Rule-based Architecture for the Domain of Mobile Police Work*

47 Azizi Bin Ab Aziz (VUA) *Exploring Computational Models for Intelligent Support of Persons with Depression*

48 Mark Ter Maat (UT) *Response Selection and Turn-taking for a Sensitive Artificial Listening Agent*

49 Andreea Niculescu (UT) *Conversational interfaces for task-oriented spoken dialogues: design aspects influencing interaction quality*

2012

1 Terry Kakeeto (UvT) *Relationship Marketing for SMEs in Uganda*

2 Muhammad Umair (VUA) *Adaptivity, emotion, and Rationality in Human and Ambient Agent Models*

3 Adam Vanya (VUA) *Supporting Architecture Evolution by Mining Software Repositories*

4 Jurriaan Souer (UU) *Development of Content Management System-based Web Applications*

5 Marijn Plomp (UU) *Maturing Interorganisational Information Systems*

6 Wolfgang Reinhardt (OU) *Awareness Support for Knowledge Workers in Research Networks*

7 Rianne van Lambalgen (VUA) *When the Going Gets Tough: Exploring Agent-based Models of Human Performance under Demanding Conditions*

8 Gerben de Vries (UvA) *Kernel Methods for Vessel Trajectories*

9 Ricardo Neisse (UT) *Trust and Privacy Management Support for Context-Aware Service Platforms*

10 David Smits (TUe) *Towards a Generic Distributed Adaptive Hypermedia Environment*

11 J. C. B. Rantham Prabhakara (TUe) *Process Mining in the Large: Preprocessing, Discovery, and Diagnostics*

12 Kees van der Sluijs (TUe) *Model Driven Design and Data Integration in Semantic Web Information Systems*

13 Suleman Shahid (UvT) *Fun and Face: Exploring non-verbal expressions of emotion during playful interactions*

14 Evgeny Knutov (TUe) *Generic Adaptation Framework for Unifying Adaptive Web-based Systems*

15 Natalie van der Wal (VUA) *Social Agents. Agent-Based Modelling of Integrated Internal and Social Dynamics of Cognitive and Affective Processes*

16 Fiemke Both (VUA) *Helping people by understanding them: Ambient Agents supporting task execution and depression treatment*

17 Amal Elgammal (UvT) *Towards a Comprehensive Framework for Business Process Compliance*

18 Eltjo Poort (VUA) *Improving Solution Architecting Practices*

19 Helen Schonenberg (TUe) *What's Next? Operational Support for Business Process Execution*

20 Ali Bahramisharif (RUN) *Covert Visual Spatial Attention, a Robust Paradigm for Brain-Computer Interfacing*

21 Roberto Cornacchia (TUD) *Querying Sparse Matrices for Information Retrieval*

22 Thijs Vis (UvT) *Intelligence, politie en veiligheidsdienst: verenigbare grootheden?*

23 Christian Muehl (UT) *Toward Affective Brain-Computer Interfaces: Exploring the Neurophysiology of Affect during Human Media Interaction*

24 Laurens van der Werff (UT) *Evaluation of Noisy Transcripts for Spoken Document Retrieval*

25 Silja Eckartz (UT) *Managing the Business Case Development in Inter-Organizational IT Projects: A Methodology and its Application*

26 Emile de Maat (UvA) *Making Sense of Legal Text*

27 Hayrettin Gurkok (UT) *Mind the Sheep! User Experience Evaluation & Brain-Computer Interface Games*

28 Nancy Pascall (UvT) *Engendering Technology Empowering Women*

29 Almer Tigelaar (UT) *Peer-to-Peer Information Retrieval*

30 Alina Pommeranz (TUD) *Designing Human-Centered Systems for Reflective Decision Making*

31 Emily Bagarukayo (RUN) *A Learning by Construction Approach for Higher Order Cognitive Skills Improvement, Building Capacity and Infrastructure*

32 Wietske Visser (TUD) *Qualitative multi-criteria preference representation and reasoning*

33 Rory Sie (OUN) *Coalitions in Cooperation Networks (COCOON)*

34 Pavol Jancura (RUN) *Evolutionary analysis in PPI networks and applications*

35 Evert Haasdijk (VUA) *Never Too Old To Learn: On-line Evolution of Controllers in Swarm- and Modular Robotics*

36 Denis Ssebugwawo (RUN) *Analysis and Evaluation of Collaborative Modeling Processes*

37 Agnes Nakakawa (RUN) *A Collaboration Process for Enterprise Architecture Creation*

38 Selmar Smit (VUA) *Parameter Tuning and Scientific Testing in Evolutionary Algorithms*

39 Hassan Fatemi (UT) *Risk-aware design of value and coordination networks*

40 Agus Gunawan (UvT) *Information Access for SMEs in Indonesia*

41 Sebastian Kelle (OU) *Game Design Patterns for Learning*

42 Dominique Verpoorten (OU) *Reflection Amplifiers in self-regulated Learning*

43 Anna Tordai (VUA) *On Combining Alignment Techniques*

44 Benedikt Kratz (UvT) *A Model and Language for Business-aware Transactions*

45 Simon Carter (UvA) *Exploration and Exploitation of Multilingual Data for Statistical Machine Translation*

46 Manos Tsagkias (UvA) *Mining Social Media: Tracking Content and Predicting Behavior*

47 Jorn Bakker (TUe) *Handling Abrupt Changes in Evolving Time-series Data*

48 Michael Kaisers (UM) *Learning against Learning: Evolutionary dynamics of reinforcement learning algorithms in strategic interactions*

49 Steven van Kervel (TUD) *Ontologogy driven Enterprise Information Systems Engineering*

50 Jeroen de Jong (TUD) *Heuristics in Dynamic Sceduling: a practical framework with a case study in elevator dispatching*

2013

1 Viorel Milea (EUR) *News Analytics for Financial Decision Support*

2 Erietta Liarou (CWI) *MonetDB/DataCell: Leveraging the Column-store Database Technology for Efficient and Scalable Stream Processing*

3 Szymon Klarman (VUA) *Reasoning with Contexts in Description Logics*

4 Chetan Yadati (TUD) *Coordinating autonomous planning and scheduling*

5 Dulce Pumareja (UT) *Groupware Requirements Evolutions Patterns*

6 Romulo Goncalves (CWI) *The Data Cyclotron: Juggling Data and Queries for a Data Warehouse Audience*

7 Giel van Lankveld (UvT) *Quantifying Individual Player Differences*

8 Robbert-Jan Merk (VUA) *Making enemies: cognitive modeling for opponent agents in fighter pilot simulators*

9 Fabio Gori (RUN) *Metagenomic Data Analysis: Computational Methods and Applications*

10 Jeewanie Jayasinghe Arachchige (UvT) *A Unified Modeling Framework for Service Design*

11 Evangelos Pournaras (TUD) *Multi-level Reconfigurable Self-organization in Overlay Services*

12 Marian Razavian (VUA) *Knowledge-driven Migration to Services*

13 Mohammad Safiri (UT) *Service Tailoring: User-centric creation of integrated IT-based homecare services to support independent living of elderly*

14 Jafar Tanha (UvA) *Ensemble Approaches to Semi-Supervised Learning Learning*

15 Daniel Hennes (UM) *Multiagent Learning: Dynamic Games and Applications*

16 Eric Kok (UU) *Exploring the practical benefits of argumentation in multi-agent deliberation*

17 Koen Kok (VUA) *The PowerMatcher: Smart Coordination for the Smart Electricity Grid*

18 Jeroen Janssens (UvT) *Outlier Selection and One-Class Classification*

19 Renze Steenhuizen (TUD) *Coordinated Multi-Agent Planning and Scheduling*

20 Katja Hofmann (UvA) *Fast and Reliable Online Learning to Rank for Information Retrieval*

21 Sander Wubben (UvT) *Text-to-text generation by monolingual machine translation*

22 Tom Claassen (RUN) *Causal Discovery and Logic*

23 Patricio de Alencar Silva (UvT) *Value Activity Monitoring*

24 Haitham Bou Ammar (UM) *Automated Transfer in Reinforcement Learning*

25 Agnieszka Anna Latoszek-Berendsen (UM) *Intention-based Decision Support. A new way of representing and implementing clinical guidelines in a Decision Support System*

26 Alireza Zarghami (UT) *Architectural Support for Dynamic Homecare Service Provisioning*

27 Mohammad Huq (UT) *Inference-based Framework Managing Data Provenance*

28 Frans van der Sluis (UT) *When Complexity becomes Interesting: An Inquiry into the Information eXperience*

29 Iwan de Kok (UT) *Listening Heads*

30 Joyce Nakatumba (TUe) *Resource-Aware Business Process Management: Analysis and Support*

31 Dinh Khoa Nguyen (UvT) *Blueprint Model and Language for Engineering Cloud Applications*

32 Kamakshi Rajagopal (OUN) *Networking For Learning: The role of Networking in a Lifelong Learner's Professional Development*

33 Qi Gao (TUD) *User Modeling and Personalization in the Microblogging Sphere*

34 Kien Tjin-Kam-Jet (UT) *Distributed Deep Web Search*

35 Abdallah El Ali (UvA) *Minimal Mobile Human Computer Interaction*

36 Than Lam Hoang (TUe) *Pattern Mining in Data Streams*

37 Dirk Börner (OUN) *Ambient Learning Displays*

38 Eelco den Heijer (VUA) *Autonomous Evolutionary Art*

39 Joop de Jong (TUD) *A Method for Enterprise Ontology based Design of Enterprise Information Systems*

40 Pim Nijssen (UM) *Monte-Carlo Tree Search for Multi-Player Games*

41 Jochem Liem (UvA) *Supporting the Conceptual Modelling of Dynamic Systems: A Knowledge Engineering Perspective on Qualitative Reasoning*

42 Léon Planken (TUD) *Algorithms for Simple Temporal Reasoning*

43 Marc Bron (UvA) *Exploration and Contextualization through Interaction and Concepts*

2014

1 Nicola Barile (UU) *Studies in Learning Monotone Models from Data*

2 Fiona Tuliyano (RUN) *Combining System Dynamics with a Domain Modeling Method*

3 Sergio Raul Duarte Torres (UT) *Information Retrieval for Children: Search Behavior and Solutions*

4 Hanna Jochmann-Mannak (UT) *Websites for children: search strategies and interface design - Three studies on children's search performance and evaluation*

5 Jurriaan van Reijsen (UU) *Knowledge Perspectives on Advancing Dynamic Capability*

6 Damian Tamburri (VUA) *Supporting Networked Software Development*

7 Arya Adriansyah (TUe) *Aligning Observed and Modeled Behavior*

8 Samur Araujo (TUD) *Data Integration over Distributed and Heterogeneous Data Endpoints*

9 Philip Jackson (UvT) *Toward Human-Level Artificial Intelligence: Representation and Computation of Meaning in Natural Language*

10 Ivan Salvador Razo Zapata (VUA) *Service Value Networks*

11 Janneke van der Zwaan (TUD) *An Empathic Virtual Buddy for Social Support*

12 Willem van Willigen (VUA) *Look Ma, No Hands: Aspects of Autonomous Vehicle Control*

13 Arlette van Wissen (VUA) *Agent-Based Support for Behavior Change: Models and Applications in Health and Safety Domains*

14 Yangyang Shi (TUD) *Language Models With Meta-information*

15 Natalya Mogles (VUA) *Agent-Based Analysis and Support of Human Functioning in Complex Socio-Technical Systems: Applications in Safety and Healthcare*

16 Krystyna Milian (VUA) *Supporting trial recruitment and design by automatically interpreting eligibility criteria*

17 Kathrin Dentler (VUA) *Computing healthcare quality indicators automatically: Secondary Use of Patient Data and Semantic Interoperability*

18 Mattijs Ghijsen (UvA) *Methods and Models for the Design and Study of Dynamic Agent Organizations*

19 Vinicius Ramos (TUe) *Adaptive Hypermedia Courses: Qualitative and Quantitative Evaluation and Tool Support*

20 Mena Habib (UT) *Named Entity Extraction and Disambiguation for Informal Text: The Missing Link*

21 Kassidy Clark (TUD) *Negotiation and Monitoring in Open Environments*

22 Marieke Peeters (UU) *Personalized Educational Games: Developing agent-supported scenario-based training*

23 Eleftherios Sidirourgos (UvA/CWI) *Space Efficient Indexes for the Big Data Era*

24 Davide Ceolin (VUA) *Trusting Semi-structured Web Data*

25 Martijn Lappenschaar (RUN) *New network models for the analysis of disease interaction*

26 Tim Baarslag (TUD) *What to Bid and When to Stop*

27 Rui Jorge Almeida (EUR) *Conditional Density Models Integrating Fuzzy and Probabilistic Representations of Uncertainty*

28 Anna Chmielowiec (VUA) *Decentralized k-Clique Matching*

29 Jaap Kabbedijk (UU) *Variability in Multi-Tenant Enterprise Software*

30 Peter de Cock (UvT) *Anticipating Criminal Behaviour*

31 Leo van Moergestel (UU) *Agent Technology in Agile Multiparallel Manufacturing and Product Support*

32 Naser Ayat (UvA) *On Entity Resolution in Probabilistic Data*

33 Tesfa Tegegne (RUN) *Service Discovery in eHealth*

34 Christina Manteli (VUA) *The Effect of Governance in Global Software Development: Analyzing Transactive Memory Systems*

35 Joost van Ooijen (UU) *Cognitive Agents in Virtual Worlds: A Middleware Design Approach*

36 Joos Buijs (TUe) *Flexible Evolutionary Algorithms for Mining Structured Process Models*

37 Maral Dadvar (UT) *Experts and Machines United Against Cyberbullying*

38 Danny Plass-Oude Bos (UT) *Making brain-computer interfaces better: improving usability through post-processing*

39 Jasmina Maric (UvT) *Web Communities, Immigration, and Social Capital*

40 Walter Omona (RUN) *A Framework for Knowledge Management Using ICT in Higher Education*

41 Frederic Hogenboom (EUR) *Automated Detection of Financial Events in News Text*

42 Carsten Eijckhof (CWI/TUD) *Contextual Multidimensional Relevance Models*

43 Kevin Vlaanderen (UU) *Supporting Process Improvement using Method Increments*

44 Paulien Meesters (UvT) *Intelligent Blauw: Intelligence-gestuurde politiezorg in gebiedsgebonden eenheden*

45 Birgit Schmitz (OUN) *Mobile Games for Learning: A Pattern-Based Approach*

46 Ke Tao (TUD) *Social Web Data Analytics: Relevance, Redundancy, Diversity*

47 Shangsong Liang (UvA) *Fusion and Diversification in Information Retrieval*

2015

1 Niels Netten (UvA) *Machine Learning for Relevance of Information in Crisis Response*
2 Faiza Bukhsh (UvT) *Smart auditing: Innovative Compliance Checking in Customs Controls*
3 Twan van Laarhoven (RUN) *Machine learning for network data*
4 Howard Spoelstra (OUN) *Collaborations in Open Learning Environments*
5 Christoph Bösch (UT) *Cryptographically Enforced Search Pattern Hiding*
6 Farideh Heidari (TUD) *Business Process Quality Computation: Computing Non-Functional Requirements to Improve Business Processes*
7 Maria-Hendrike Peetz (UvA) *Time-Aware Online Reputation Analysis*
8 Jie Jiang (TUD) *Organizational Compliance: An agent-based model for designing and evaluating organizational interactions*
9 Randy Klaassen (UT) *HCI Perspectives on Behavior Change Support Systems*
10 Henry Hermans (OUN) *OpenU: design of an integrated system to support lifelong learning*
11 Yongming Luo (TUe) *Designing algorithms for big graph datasets: A study of computing bisimulation and joins*
12 Julie M. Birkholz (VUA) *Modi Operandi of Social Network Dynamics: The Effect of Context on Scientific Collaboration Networks*
13 Giuseppe Procaccianti (VUA) *Energy-Efficient Software*
14 Bart van Straalen (UT) *A cognitive approach to modeling bad news conversations*
15 Klaas Andries de Graaf (VUA) *Ontology-based Software Architecture Documentation*
16 Changyun Wei (UT) *Cognitive Coordination for Cooperative Multi-Robot Teamwork*
17 André van Cleeff (UT) *Physical and Digital Security Mechanisms: Properties, Combinations and Trade-offs*
18 Holger Pirk (CWI) *Waste Not, Want Not!: Managing Relational Data in Asymmetric Memories*
19 Bernardo Tabuenca (OUN) *Ubiquitous Technology for Lifelong Learners*
20 Loïs Vanhée (UU) *Using Culture and Values to Support Flexible Coordination*
21 Sibren Fetter (OUN) *Using Peer-Support to Expand and Stabilize Online Learning*
22 Zhemin Zhu (UT) *Co-occurrence Rate Networks*
23 Luit Gazendam (VUA) *Cataloguer Support in Cultural Heritage*
24 Richard Berendsen (UvA) *Finding People, Papers, and Posts: Vertical Search Algorithms and Evaluation*
25 Alexander Hogenboom (EUR) *Sentiment Analysis of Text Guided by Semantics and Structure*
26 Sándor Héman (CWI) *Updating compressed colomn stores*
27 Janet Bagorogoza (TiU) *Knowledge Management and High Performance; The Uganda Financial Institutions Model for HPO*
28 Hendrik Baier (UM) *Monte-Carlo Tree Search Enhancements for One-Player and Two-Player Domains*
29 Kiavash Bahreini (OU) *Real-time Multimodal Emotion Recognition in E-Learning*
30 Yakup Koç (TUD) *On the robustness of Power Grids*
31 Jerome Gard (UL) *Corporate Venture Management in SMEs*
32 Frederik Schadd (TUD) *Ontology Mapping with Auxiliary Resources*
33 Victor de Graaf (UT) *Gesocial Recommender Systems*
34 Jungxao Xu (TUD) *Affective Body Language of Humanoid Robots: Perception and Effects in Human Robot Interaction*

2016

1 Syed Saiden Abbas (RUN) *Recognition of Shapes by Humans and Machines*
2 Michiel Christiaan Meulendijk (UU) *Optimizing medication reviews through decision support: prescribing a better pill to swallow*
3 Maya Sappelli (RUN) *Knowledge Work in Context: User Centered Knowledge Worker Support*
4 Laurens Rietveld (VU) *Publishing and Consuming Linked Data*
5 Evgeny Sherkhonov (UVA) *Expanded Acyclic Queries: Containment and an Application in Explaining Missing Answers*
6 Michel Wilson (TUD) *Robust scheduling in an uncertain environment*
7 Jeroen de Man (VU) *Measuring and modeling negative emotions for virtual training*
8 Matje van de Camp (TiU) *A Link to the Past: Constructing Historical Social Networks from Unstructured Data*

9 Archana Nottamkandath (VU) *Trusting Crowdsourced Information on Cultural Artefacts*

10 George Karafotias (VUA) *Parameter Control for Evolutionary Algorithms*

11 Anne Schuth (UVA) *Search Engines that Learn from Their Users*

12 Max Knobbout (UU) *Logics for Modelling and Verifying Normative Multi-Agent Systems*

13 Nana Baah Gyan (VU) *The Web, Speech Technologies and Rural Development in West Africa - An ICT4D Approach*

14 Ravi Khadka (UU) *Revisiting Legacy Software System Modernization*

15 Steffen Michels (RUN) *Hybrid Probabilistic Logics - Theoretical Aspects, Algorithms and Experiments*

16 Guangliang Li (UVA) *Socially Intelligent Autonomous Agents that Learn from Human Reward*

17 Berend Weel (VU) *Towards Embodied Evolution of Robot Organisms*

18 Albert Meroño Peñuela (VU) *Refining Statistical Data on the Web*

19 Julia Efremova (Tu/e) *Mining Social Structures from Genealogical Data*

20 Daan Odijk (UVA) *Context & Semantics in News & Web Search*

21 Alejandro Moreno Célleri (UT) *From Traditional to Interactive Playspaces: Automatic Analysis of Player Behavior in the Interactive Tag Playground*

22 Grace Lewis (VU) *Software Architecture Strategies for Cyber-Foraging Systems*

23 Fei Cai (UVA) *Query Auto Completion in Information Retrieval*

24 Brend Wanders (UT) *Repurposing and Probabilistic Integration of Data; An Iterative and data model independent approach*

25 Julia Kiseleva (TU/e) *Using Contextual Information to Understand Searching and Browsing Behavior*

26 Dilhan Thilakarathne (VU) *In or Out of Control: Exploring Computational Models to Study the Role of Human Awareness and Control in Behavioural Choices, with Applications in Aviation and Energy Management Domains*

27 Wen Li (TUD) *Understanding Geo-spatial Information on Social Media*

28 Mingxin Zhang (TUD) *Large-scale Agent-based Social Simulation - A study on epidemic prediction and control*

29 Nicolas Höning (TUD) *Peak reduction in decentralised electricity systems - Markets and prices for flexible planning*

30 Ruud Mattheij (UvT) *The Eyes Have It*

31 Mohammad Khelghati (UT) *Deep web content monitoring*

32 Eelco Vriezekolk (UT) *Assessing Telecommunication Service Availability Risks for Crisis Organisations*

33 Peter Bloem (UVA) *Single Sample Statistics, exercises in learning from just one example*

34 Dennis Schunselaar (TUE) *Configurable Process Trees: Elicitation, Analysis, and Enactment*

35 Zhaochun Ren (UVA) *Monitoring Social Media: Summarization, Classification and Recommendation*

36 Daphne Karreman (UT) *Beyond R2D2: The design of nonverbal interaction behavior optimized for robot-specific morphologies*

37 Giovanni Sileno (UvA) *Aligning Law and Action - a conceptual and computational inquiry*

38 Andrea Minuto (UT) *Materials that Matter - Smart Materials meet Art & Interaction Design*

39 Merijn Bruijnes (UT) *Believable Suspect Agents; Response and Interpersonal Style Selection for an Artificial Suspect*

40 Christian Detweiler (TUD) *Accounting for Values in Design*

41 Thomas King (TUD) *Governing Governance: A Formal Framework for Analysing Institutional Design and Enactment Governance*

42 Spyros Martzoukos (UVA) *Combinatorial and Compositional Aspects of Bilingual Aligned Corpora*

43 Saskia Koldijk (RUN) *Context-Aware Support for Stress Self-Management: From Theory to Practice*

44 Thibault Sellam (UVA) *Automatic Assistants for Database Exploration*

45 Bram van de Laar (UT) *Experiencing Brain-Computer Interface Control*

46 Jorge Gallego Perez (UT) *Robots to Make you Happy*

47 Christina Weber (UL) *Real-time foresight - Preparedness for dynamic innovation networks*

48 Tanja Buttler (TUD) *Collecting Lessons Learned*

49 Gleb Polevoy (TUD) *Participation and Interaction in Projects. A Game-Theoretic Analysis*

50 Yan Wang (UVT) *The Bridge of Dreams: Towards a Method for Operational Performance Alignment in IT-enabled Service Supply Chains*

2017

1 Jan-Jaap Oerlemans (UL) *Investigating Cybercrime*

2 Sjoerd Timmer (UU) *Designing and Understanding Forensic Bayesian Networks using Argumentation*

3 Daniël Harold Telgen (UU) *Grid Manufacturing; A Cyber-Physical Approach with Autonomous Products and Reconfigurable Manufacturing Machines*

4 Mrunal Gawade (CWI) *Multi-core Parallelism in a Column-store*

5 Mahdieh Shadi (UVA) *Collaboration Behavior*

6 Damir Vandic (EUR) *Intelligent Information Systems for Web Product Search*

7 Roel Bertens (UU) *Insight in Information: from Abstract to Anomaly*

8 Rob Konijn (VU) *Detecting Interesting Differences:Data Mining in Health Insurance Data using Outlier Detection and Subgroup Discovery*

9 Dong Nguyen (UT) *Text as Social and Cultural Data: A Computational Perspective on Variation in Text*

10 Robby van Delden (UT) *(Steering) Interactive Play Behavior*

11 Florian Kunneman (RUN) *Modelling patterns of time and emotion in Twitter #anticipointment*

12 Sander Leemans (TUE) *Robust Process Mining with Guarantees*

13 Gijs Huisman (UT) *Social Touch Technology - Extending the reach of social touch through haptic technology*

14 Shoshannah Tekofsky (UvT) *You Are Who You Play You Are: Modelling Player Traits from Video Game Behavior*

15 Peter Berck (RUN) *Memory-Based Text Correction*

16 Aleksandr Chuklin (UVA) *Understanding and Modeling Users of Modern Search Engines*

17 Daniel Dimov (UL) *Crowdsourced Online Dispute Resolution*

18 Ridho Reinanda (UVA) *Entity Associations for Search*

19 Jeroen Vuurens (UT) *Proximity of Terms, Texts and Semantic Vectors in Information Retrieval*

20 Mohammadbashir Sedighi (TUD) *Fostering Engagement in Knowledge Sharing: The Role of Perceived Benefits, Costs and Visibility*

21 Jeroen Linssen (UT) *Meta Matters in Interactive Storytelling and Serious Gaming (A Play on Worlds)*

22 Sara Magliacane (VU) *Logics for causal inference under uncertainty*

23 David Graus (UVA) *Entities of Interest — Discovery in Digital Traces*

24 Chang Wang (TUD) *Use of Affordances for Efficient Robot Learning*

25 Veruska Zamborlini (VU) *Knowledge Representation for Clinical Guidelines, with applications to Multimorbidity Analysis and Literature Search*

26 Merel Jung (UT) *Socially intelligent robots that understand and respond to human touch*

27 Michiel Joosse (UT) *Investigating Positioning and Gaze Behaviors of Social Robots: People's Preferences, Perceptions and Behaviors*

28 John Klein (VU) *Architecture Practices for Complex Contexts*

29 Adel Alhuraibi (UvT) *From IT-BusinessStrategic Alignment to Performance: A Moderated Mediation Model of Social Innovation, and Enterprise Governance of IT*

30 Wilma Latuny (UvT) *The Power of Facial Expressions*

31 Ben Ruijl (UL) *Advances in computational methods for QFT calculations*

32 Thaer Samar (RUN) *Access to and Retrievability of Content in Web Archives*

33 Brigit van Loggem (OU) *Towards a Design Rationale for Software Documentation: A Model of Computer-Mediated Activity*

34 Maren Scheffel (OU) *The Evaluation Framework for Learning Analytics*

35 Martine de Vos (VU) *Interpreting natural science spreadsheets*

36 Yuanhao Guo (UL) *Shape Analysis for Phenotype Characterisation from High-throughput Imaging*

37 Alejandro Montes Garcia (TUE) *WiBAF: A Within Browser Adaptation Framework that Enables Control over Privacy*

38 Alex Kayal (TUD) *Normative Social Applications*

39 Sara Ahmadi (RUN) *Exploiting properties of the human auditory system and compressive sensing methods to increase noise robustness in ASR*

40 Altaf Hussain Abro (VUA) *Steer your Mind: Computational Exploration of Human Control in Relation to Emotions, Desires and Social Support For applications in human-aware support systems*

41 Adnan Manzoor (VUA) *Minding a Healthy Lifestyle: An Exploration of Mental Processes and a Smart Environment to Provide Support for a Healthy Lifestyle*

42 Elena Sokolova (RUN) *Causal discovery from mixed and missing data with applications on ADHD datasets*

43 Maaike de Boer (RUN) *Semantic Mapping in Video Retrieval*

44 Garm Lucassen (UU) *Understanding User Stories - Computational Linguistics in Agile Requirements Engineering*

45 Bas Testerink (UU) *Decentralized Runtime Norm Enforcement*

46 Jan Schneider (OU) *Sensor-based Learning Support*

47 Jie Yang (TUD) *Crowd Knowledge Creation Acceleration*

48 Angel Suarez (OU) *Collaborative inquiry-based learning*

2018

1 Han van der Aa (VUA) *Comparing and Aligning Process Representations*

2 Felix Mannhardt (TUE) *Multi-perspective Process Mining*

3 Steven Bosems (UT) *Causal Models For Well-Being: Knowledge Modeling, Model-Driven Development of Context-Aware Applications, and Behavior Prediction*

4 Jordan Janeiro (TUD) *Flexible Coordination Support for Diagnosis Teams in Data-Centric Engineering Tasks*

5 Hugo Huurdeman (UVA) *Supporting the Complex Dynamics of the Information Seeking Process*

6 Dan Ionita (UT) *Model-Driven Information Security Risk Assessment of Socio-Technical Systems*

7 Jieting Luo (UU) *A formal account of opportunism in multi-agent systems*

8 Rick Smetsers (RUN) *Advances in Model Learning for Software Systems*

9 Xu Xie (TUD) *Data Assimilation in Discrete Event Simulations*

10 Julienka Mollee (VUA) *Moving forward: supporting physical activity behavior change through intelligent technology*

11 Mahdi Sargolzaei (UVA) *Enabling Framework for Service-oriented Collaborative Networks*

12 Xixi Lu (TUE) *Using behavioral context in process mining*

13 Seyed Amin Tabatabaei (VUA) *Computing a Sustainable Future*

14 Bart Joosten (UVT) *Detecting Social Signals with Spatiotemporal Gabor Filters*

15 Naser Davarzani (UM) *Biomarker discovery in heart failure*

16 Jaebok Kim (UT) *Automatic recognition of engagement and emotion in a group of children*

17 Jianpeng Zhang (TUE) *On Graph Sample Clustering*

18 Henriette Nakad (UL) *De Notaris en Private Rechtspraak*

19 Minh Duc Pham (VUA) *Emergent relational schemas for RDF*

20 Manxia Liu (RUN) *Time and Bayesian Networks*