Modeling for Continuous Cued Speech Recognition in French using Advanced Machine Learning Methods

Li Liu

Communauté UNIVERSITÉ Grenoble Alpes

THÈSE

pour obtenir le grade de

DOCTEUR DE LA COMMUNAUTÉ UNIVERSITÉ GRENOBLE ALPES

Spécialité : SIGNAL IMAGE PAROLE TELECOMS

Arrêté ministériel : 25 mai 2016

Présentée par

Li LIU

Thèse dirigée par **Denis BEAUTEMPS**
et **Gang FENG**
préparée au sein du **Laboratoire Grenoble Images Parole Signal Automatique (Gipsa Lab)**
dans **l'École Doctorale Electronique, Electrotechnique, Automatique, Traitement du Signal (EEATS)**

Reconnaissance continue de la langue française parlée complétée à l'aide de méthodes avancées d'apprentissage automatique

Continuous Cued Speech Recognition in French using Advanced Machine Learning Methods

Thèse soutenue publiquement le **11 Septembre 2018**,
devant le jury composé de:

Monsieur Jean-Luc SCHWARTZ
Directeur de Recherche, CNRS, Président du jury
Madame Régine ANDRE-OBRECHT
Professeur des Universités, Université de Toulouse, Rapporteur
Madame Hélène MAYNARD
Maître de Conférences, Université d'Orsay, Habilitée à Diriger des Recherches, Rapporteur
Madame Jacqueline LEYBAERT
Professeur, Université Libre de Bruxelles, Examinateur
Monsieur Denis BEAUTEMPS
Chargé de Recherche, CNRS, Habilité à Diriger des Recherches, Directeur de thèse
Monsieur Gang FENG
Professeur des Universités, Grenoble INP, Co-directeur de thèse
Monsieur Thomas HUEBER
Chargé de Recherche, CNRS, Invité

We will be exploring on a long road,

with persistence and determination.

Nous explorerons sur une longue route,

avec persévérance et détermination.

--- Qu Yuan (c. 340–278 BC)

Acknowledgement

I am very happy to meet so many lovely people during my PhD and my past life. First of all, I would like to express my sincere gratitude to my supervisors Dr. Denis BEAUTEMPS and Prof. Gang FENG for their support, encouragement, brilliant guidance and reviewing this thesis. I am indeed fortunate to have them as my supervisors. My sincere gratitude also goes to Dr. Thomas HUEBER for his guidance and assistance in Convolutional Neural Networks and Hidden Markov Models. Thanks to them, I had the chance to make progress during the past three years in a stimulating scientific environment while enjoying a great freedom in my work. I thank them for their abilities to listen, their wise counsel and great availability.

Then I want to express my sincere gratitude to the members of the jury. I thank Régine ANDRE-OBRECHT and Hélène MAYNARD for their flexibility in accommodating the review of this thesis into their busy schedules, Jacqueline LEYBAERT for being the examiner and Jean-Luc SCHWARTZ for being the president of the jury. Meanwhile, I want to thank the volunteer Cued Speakers for their time spent in recording the dataset and Christophe SAVARIAUX for his help in the experimental setup. I also thank Henrietta IRELAND and Cate CALDER in the Cued Speech Association UK for their enthusiastic help in British English CS corpus recording.

Many thanks go to all the members of Speech and Cognition Department and GIPSA-lab who gave me a happy memory and enriched my life in France. This thesis would not be of the present form if without the valuable discussions with many colleagues and friends, especially Zuheng MING, Chengfang REN, Junshi XIA, Yang QI, Jingtao CHEN, Dacheng ZHANG, Canh NGUYEN and Omar MOHAMMED. Especially, I would like to thank Prof. Gang FENG again, for his kind and selfless help which made my life in France go well.

I also would like to thank Prof. Kainan XIANG and Prof. Vladas SIDORAVICIUS for their support and encouragement during my study in France, as well as Prof. Xiaoping ZHANG, for his assistance and suggestion for my future research plan.

Last but not the least, I would like to sincerely thank my mother Zhenggui LI and my father Sanling LIU for raising me up and teaching me to be an honest and positive person. I thank my sister Jiao LIU, and brother Can LIU, for their company and support in the process of pursuing my dream. Many thanks also go to my husband Jianze LI for his continued encouragement and support during my PhD study.

Abstract

This PhD thesis deals with the automatic continuous *Cued Speech* (CS) recognition in French based on the images of subjects without using any artificial landmark. In order to realize this objective, we extract high-level features of three information flows (lips, hand positions and shapes), and find an optimal approach to merge them for a robust CS recognition system. We first introduce a novel and powerful deep learning method based on the *Convolutional Neural Network* (CNN) for extracting the lips and hand shape features from raw images. The *adaptive background mixture models* (ABMMs) are applied to obtain the hand position features for the first time. Meanwhile, based on an advanced machine learning method *Constrained Local Neural Fields* (CLNF), we propose the *Modified CLNF* to extract the inner lips parameters (lips width A and lips height B), as well as another method named *adaptive ellipse model*. All these methods make significant contributions to the feature extraction in CS. Then, due to the asynchrony problem of three feature flows (i.e., lips, hand shapes and hand positions) in CS, the fusion of them is a challenging issue. In order to resolve it, firstly, a new resynchronization procedure is proposed to align hand and lips features based on the study of the temporal hand movement in CS. Then, we propose several approaches including feature-level and model-level fusion strategies combined with the context-dependent HMM. To achieve the CS recognition, we propose three tandem CNN-HMMs architectures. All these architectures are evaluated on the corpus without any artifice, and the CS recognition performance confirms the efficiency of our proposed methods. The result of 72.67% continuous phoneme recognition is comparable to the state of the art, which used the corpus with artifices and was for the isolated CS recognition case. In parallel, we investigate a specific study about the temporal organization of hand movements in CS, especially about its temporal segmentation, and the evaluations confirm the superior performance of our methods. In summary, this PhD thesis applies the advanced machine learning (especially the deep learning) methods to CS recognition work, which makes a significant step to the general automatic conversion problem of CS to audio speech.

Keywords: Cued Speech, Automatic Continuous Speech Recognition, Automatic Feature Extraction, Machine Learning, Deep Learning, Multi-modal Fusion, Context-dependent Modeling, CNN-HMMs.

Résumé

Cette thèse de doctorat traite de la reconnaissance automatique de la langue française Parlée Complétée (LPC), version française du Cued Speech (CS), à partir de l'image vidéo et sans marquage de l'information préalable à l'enregistrement vidéo. Afin de réaliser cet objectif, nous cherchons à extraire les caractéristiques de haut niveau de trois flux d'information (lèvres, positions de la main et formes), et fusionner ces trois modalités dans une approche optimale pour un système de reconnaissance de LPC robuste. Dans ce travail, nous avons introduit une méthode d'apprentissage profond avec les réseaux neurono convolutifs (CNN) pour extraire les formes de main et de lèvres à partir d'images brutes. Un modèle de mélange de fond adaptatif (ABMM) est proposé pour obtenir la position de la main. De plus, deux nouvelles méthodes nommées *Modified Constraint Local Neural Fields* (CLNF Modifié) et le modèle *Adaptive Ellipse Model* ont été proposés pour extraire les paramètres du contour interne des lèvres (étirement et ouverture aux lèvres). Le premier s'appuie sur une méthode avancée d'apprentissage automatique (CLNF) en vision par ordinateur. Toutes ces méthodes constituent des contributions significatives pour l'extraction des caractéristiques du LPC. En outre, en raison de l'asynchronie des trois flux caractéristiques du LPC, leur fusion est un enjeu important dans cette thèse. Afin de le résoudre, nous avons proposé plusieurs approches, y compris les stratégies de fusion au niveau données et modèle avec une modélisation HMM dépendant du contexte. Pour obtenir le décodage, nous avons proposé trois architectures CNN-HMMs. Toutes ces architectures sont évaluées sur un corpus de phrases codées en LPC en parole continue sans aucun artifice, et la performance de reconnaissance du LPC confirme l'efficacité de nos méthodes proposées. Le résultat (72.67%) est comparable à l'état de l'art qui utilisait des bases de données où l'information pertinente était préalablement repérée. En même temps, nous avons réalisé une étude spécifique concernant l'organisation temporelle des mouvements de la main, révélant une avance de la main en relation avec l'emplacement dans la phrase. En résumé, ce travail de doctorat propose les méthodes avancées d'apprentissage automatique issues du domaine de la vision par ordinateur et les méthodologies d'apprentissage profond dans le travail de reconnaissance du LPC, qui constituent un pas important vers le problème général de conversion automatique du LPC en parole audio.

Mot-clés: Reconnaissance automatique du LPC en continu, Extraction automatique de caractéristiques, Apprentissage automatique et Apprentissage profond, Fusion multimodale, Modèle dépendant du contexte, CNN-HMMs.

Contents

List of Figures

List of Tables

Acronyms

AAM Active Appearance Model. 67, 168

ABMM Adaptive Background Mixture Model. xviii, 3, 111–113, 115, 117, 132, 134, 145, 162, 168, 175

AC Aligned Concatenation. 142, 160, 169, 177

AI Artificial Intelligence. 47, 48

ALPC Association Nationale de la Langue Française Parlée Complétée. 7

ANN Artificial Neural Network. 4, 17, 169

ASM Active Shape Model. 67

ASR Audio Speech Recognition. 62, 94, 141, 142

AVSR Audio Visual Speech Recognition. 4, 19, 35, 58–60, 62, 67

Bi-LSTM Bi-directional Long Short Temporal Memory. 170

BP Back-Propagation. 49

BPTT Back-Propagation Through Time. 54, 135, 161

CCA Canonical Correlation Analysis. 62

CCS Chinese Cued Speech. xxii, 171, 172

CLM Constrained Local Model. xiii, 62, 67–70, 168

CLNF Constrained Local Neural Fields. 67, 69, 167

CNN Convolutional Neural Network. 3, 15, 47, 50, 136, 140, 168, 175

CS Cued Speech. 1, 7, 167, 173

CSAUK Cued Speech Association UK. 7, 27

CTC Connectionist Temporal Classification. 170

DCCA Deep Canonical Correlation Analysis. 62, 170

DCT Discrete Cosine Transform. xii, xv, 35, 36, 38, 82–84, 87

DI Direct Identification. 59–61, 142

DNN Deep Neural Network. 17, 50, 61

DoG Difference of Gaussian. 111

DR Dominant Recoding. 59, 60

DWT Discrete Wavelet Transform. 36

EM Expectation Maximization. 59, 146

FC Fully Connected. 52, 144–146

GMM Gaussian Mixture Model. 4, 14, 168, 175

GPU Graphics Processing Unit. 17, 135, 144, 161

HD-CTM Hybrid Dynamic Correlation Template Method. 68, 167

HMM Hidden Markov Model. 3, 4, 17, 58, 169, 177

HOG Histogram of Oriented Gradients. 18

KCCA Kernel Canonical Correlation Analysis. 62

KLT Kanade–Lucas–Tomasi. xii, 41, 43, 44, 168

LNF Local Neural Field. 69

LPC Langue française Parlée Complétée. xi, 7, 9, 11, 13, 25, 145, 173–175, 177

LSTM Long Short-Term Memory. 3, 19, 48, 55, 58, 62, 120, 131, 134, 138, 140, 161, 168, 171, 177

MLP Multi-Layer Perceptron. 4, 48, 49

Modified CLNF Modified Constrained Local Neural Fields. xxv, 3, 4, 68, 81, 87, 90, 106–108, 167, 175

MR Motor Recoding. 59, 60

MSE Mean Squared Error. 49

NCSA National Cued Speech Association. xi, 7, 8

PCA Principal Component Analysis. 35, 36, 168

PDM Point Distribution Model. 68, 69

ReLU Rectified Linear units. xiii, 50–52, 145, 146

RLMS Regularized Landmark Mean-Shift. xiii, 69, 70

RMSE Root Mean Square Error. 81, 84, 87, 167

RNN Recurrent Neural Network. 4, 19, 47, 53

ROI Region of Interest. 3, 169, 175

SI Separated Identification. 59, 60

SL Sign Language. 9, 11

STIP Space-Time Interest Point. 18

SVM Support Vector Machine. 17, 18, 69

tanh Hyperbolic Tangent function. xiii, 49–51

VLLT Validated Lower Lips Thickness. xiv, 76, 78, 81, 84, 109, 167

VSR Visual Speech Recognition. 19

WHO World Health Organization. 1, 173

WLAS Watch, Listen, Attend and Spell. 1, 19

General Introduction

Motivation

It was reported by World Health Organization (WHO)[1] that over 5% of the world's population (about 432 million adults and 34 million children) have a disabling hearing loss nowadays, and it is estimated that this number will rise over 900 million by 2050. Therefore, there will be a serious demand of automatic methods to help these people communicate easier and better. In fact, in the community of orally educated deaf people, *lip reading* [1]–[3] is still one of the main modalities of perceiving speech, and its benefits have been widely admitted in the world for a long time. However, the speech cannot be easily perceived thoroughly if the lip reader has no knowledge about the semantic context due to the ambiguity of the visual patterns in lip reading. In fact, many phonemes which look identical on lips (e.g., [p], [b] and [m]) cannot be well perceived only by the lips information. For example, for the moment, one of the best automatic lip readers Watch, Listen, Attend and Spell (WLAS) network [4] still obtains a 23.8% word error rate on the *Lip Reading in the Wild* (LRW) dataset.

To overcome the low performance of lip reading and improve the reading ability of deaf children, in 1967, Cornett [5] developed the Cued Speech (CS) system [5]–[9] which uses the hand gestures to complement the lips information to make all the phonemes of spoken languages clearly visible. Therefore, the similar lips shapes can be distinguished by adding the hand information, which allows the deaf people to completely understand spoken languages using the lips and hand information. This cuing system allows the people who are deaf, hard of hearing to access the basic, fundamental properties of spoken languages by the vision.

To improve the communications between the hearing impaired people and normal hearing people, it will be meaningful to realize an automatic conversion from visual modality to audio modality and inversely from audio modality to visual modality. This PhD thesis focuses on the automatic continuous recognition of CS in French in the conversion from visual modality to the text[2](phonemes). Its framework lies in the *multi-modal (audio-visual) speech processing*, and intersects with the *human machine communication*, *Artificial Intelligence* and *computer vision*.

Challenges

A CS recognition system requires an automatic recognition to decode not only the lips of the speakers but also the movements of their hand. In this thesis, the CS recognition contains three main procedures: *feature extraction*, *multi-stream (lips and hand) fusion* and *continuous*

[1] http://www.who.int/en/

[2] In fact, this conversion is an important step in the process of visual modality to audio.

CS recognition, which can be seen in Fig. 1.

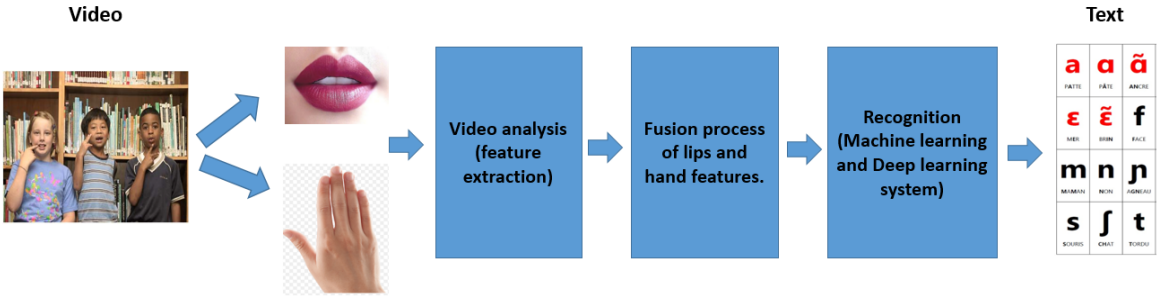


Figure 1: Framework of the automatic CS recognition from visual modality to phonemes.

In the process of realizing the automatic CS recognition in Fig. 1, there are three main challenges, which can be summarized as follows.

(1) The first challenge is the *feature extraction without using any artificial landmark*. In fact, in the state of the art [10], [11], the corpus was recorded with artifices for the convenience of the lips and hand feature extraction. To the best of our knowledge, no work has been dedicated to extracting the lips and hand features based on the raw images in CS. Therefore, to get rid of these artifices in the feature extraction step is our first challenge.

(2) The second one is the *temporal segmentation of the hand movements in CS*. Assuming that lips movement is synchronized with the audio sound [12]–[14], we can obtain the temporal segmentation of lips by temporally aligning the audio sound. However, since lips, hand position and hand shape are asynchronous in CS, hand can not share the same temporal segmentation with the lips movement. In the state of the art [10], [11], the temporal segmentation of the lips, hand position and shape streams was all directly realized by the audio signal. Thus, in this thesis, our second challenge is to take into account this asynchrony issue among these three streams, and propose methods to automatically segment the hand position and hand shape movement temporally.

(3) The third one is the *continuous CS recognition system which takes into account the fusion of the asynchronous feature streams*. In this thesis, the *continuous* CS recognition follows two conditions: the data is composed of continuous sentences[3]; and the temporal boundary is not given(i.e., every image frame is fed to the recognizer) in the test step. In the state of the art [10], [11], either the isolated CS recognition was studied in a continuous sentence corpus, or the continuous CS recognition was studied in an isolated word corpus, and the context information of CS was not modeled. In this thesis, we deal with the continuous CS recognition based on a continuous sentence corpus, and thus it is much more difficult. On the other hand, the multi-modalities in CS lead to another problem of how to merge the features from different streams. Therefore, how to realize the CS recognition taking into account the fusion of different modalities, as well as the context information of feature flows, is our third challenge.

[3] The continuous sentence is made up of continuous words (i.e., not isolated words), but there is a time gap between sentence and sentence.

Methodologies

To realize the automatic continuous recognition of French CS in the conversion from visual modality to phonemes, based on the corpora of five speakers, we develop several approaches and algorithms in this thesis. Corresponding to the above three challenges, these approaches and algorithms can be summarized as follows.

(1) For the first challenge, we propose two novel methods named Modified Constrained Local Neural Fields (Modified CLNF) [15], [16] and *adaptive ellipse model* [17] to extract the inner lips parameters. Meanwhile, Convolutional Neural Network (CNN) [18], [19] is applied on the raw lips Region of Interest (ROI) to extract the high-level pixel based lips features. For hand position features, we propose to use the Adaptive Background Mixture Model (ABMM) [20]–[22] to extract the hand position, which is taken as the center of the hand ROI. After the hand ROI is tracked, CNNs are then applied to extract hand shape features from ROI of the raw images.

(2) For the second challenge, we propose a *hand preceding model* to automatically predict the temporal segmentation of hand position movement by exploring the relationship between hand preceding time (i.e., the time that hand precedes lips movements) and the vowel positions in sentences. To evaluate the performance of the proposed method, hand position recognition is realized with the *multi-Gaussian* and Long Short-Term Memory (LSTM) [23] using different temporal segmentations of hand position. The results show that using the temporal segmentation predicted by the proposed method significantly improves the recognition performance compared with that using the audio based segmentation. For the hand shape stream, we propose the optimal temporal segmentation for hand shape realization based on the hand preceding model of the hand position.

(3) For the third challenge, the asynchronous multi-modal fusion and the continuous CS recognition are realized thanks to several new tandem architectures which combine the CNNs, Hidden Markov Model (HMM) [24], [25], different fusion strategies and *context-dependent modeling*. A new resynchronization procedure *aligned concatenation* (AC) is proposed to pre-process the multi-modal features in order to reduce the effect of the asynchrony and guarantee the quality of fusion. Without exploiting any dictionary or language model, the best proposed tandem CNN-HMM architecture can correctly identify about 72.7% of the continuous phonemes using the hand position given by ABMMs and 76.6% using the ground truth hand position. Notably, this result is comparable to the state of the art [10], which was for isolated CS recognition and used the corpus with artifices. In fact, this thesis is the first work to deal with the continuous CS recognition based on a sentence corpus without using any artifice.

Organization

This PhD thesis is organized as follows.

In Part I, we present the background of CS and the state of the art of the automatic CS recognition, as well as CS materials which will be used in this thesis. The methods of deep learning and multi-modal fusion are also introduced. In Chapter 1, we first introduce the history and development of CS, as well as the principles of its construction. Some other studies like CS perception, production and the automatic CS processing are also reported. Then, we report some previous work about the automatic lip reading, hand feature extraction, automatic CS recognition system and some other related topics. In Chapter 2, the experimental setup and the corpus are introduced, as well as the phonetic transcription, automatic alignment, the definition of the lips and hand parameters, and the classical pixel based feature extraction methods. In Chapter 3, we focus on the deep learning methods. A general framework of the artificial intelligence, machine learning and deep learning related with this thesis is presented, and the standard Multi-Layer Perceptron (MLP), CNN, Recurrent Neural Network (RNN) [26] and related deep learning methods are presented. In Chapter 4, we introduce the Hidden Markov Model (HMM) and the classical multi-modal fusion strategies as well as deep learning based fusion methods in Audio Visual Speech Recognition (AVSR). Then context-dependent modeling is presented. The ideas about the application of these methods in the CS recognition are also discussed.

In Part II, we first propose the methods for extracting the lips and hand features in CS. The temporal segmentation of the hand movements in CS is then studied, as well as the automatic continuous CS recognition. In Chapter 5, we present two studies for estimating the inner lips parameters. One is the Modified CLNF [15], [16] which effectively estimates the inner lips parameters even in different contexts, e.g., with hand occlusion or different lighting conditions. The other is the *adaptive ellipse model* [17] which is able to extract the inner lips parameters for any lips shape. In Chapter 6, We focus on the extraction of the hand position and hand ROI in CS. The hand position is tracked by the ABMMs [20], [21], [27], which model the background of the image by Gaussian Mixture Model (GMM), and the hand ROI is then determined based on the hand position. In Chapter 7, an automatic method is developed to segment the hand movements temporally. For the hand position movements, a hand preceding model which investigates the relationship between the vowel position in sentences and the hand preceding time is proposed. Based on the hand position result, the hand preceding model for hand shape is built. In Chapter 8, we propose several tandem CNN-HMM architectures for the CS recognition. The Artificial Neural Network (ANN) is applied for processing the hand position features, and CNNs are used for extracting the features of hand shapes and lips. A resynchronization procedure combined with the feature-level and model-level fusion methods [28] are used to merge the lips and hand streams within a context-dependent HMM-GMM decoder. Then the viseme, vowel, consonant and phoneme recognitions are conducted to evaluate these CNN-HMM architectures.

Finally, we summarize the main contributions and results of this thesis and give some suggestions for the future work.

Part I

Chapter 1

State of the art of Cued Speech Automatic Processing

Contents

1.1 Introduction to Cued Speech

Cued Speech (CS) is a system[1] using the hand codes as a complement to the natural lip reading, invented by Cornett [5] in 1967, to make the hearing impaired people access spoken language easier (see Fig. 1.1). In this system, as a combination of different hand shapes and positions near the face, the hand coding complements the lip reading to enhance the speech perception. More precisely, the hand shapes are used to code the consonants, while the hand positions on one side of the face or the neck are used to code the vowels.

CS is now becoming more and more attended by the world and has been adapted to over 60 languages for the moment. The National Cued Speech Association (NCSA)[2] for American English CS and Cued Speech Association UK (CSAUK)[3] for British English CS have been established to generalize this system. For the French CS, which is named Langue française Parlée Complétée (LPC) [7], the Association Nationale de la Langue Française Parlée Complétée (ALPC)[4] has been established as well. These associations have been trying their

[1] https://www.youtube.com/watch?v=jn4e9V3oigs
[2] http://www.cuedspeech.org/
[3] http://www.cuedspeech.co.uk/
[4] http://alpc.asso.fr/

best to improve the communications between the deaf or hearing impaired children and the normal hearing family members through the CS system.

In this chapter, we first present the CS system with a brief history of its background and construction. Then, other studies of CS such as the speech perception and automatic CS processing are reported. Finally, we explore the state of the art of CS recognition as well as the lips and hand feature extraction.

1.1.1 The motivation of Cued Speech

CS is invented to augment lip reading to improve the communication abilities of the hearing impaired people. Now we first briefly introduce the lip reading, also known as the speech reading, which is a way of perceiving speech by interpreting the movements of the mouth, lips and tongue visually when there is no available sound. A more common way of understanding the lip reading is to interpret the movements of only lips, which is adopted in this thesis. One of the most important applications of lip reading is to help the deaf or hearing impaired people access the spoken speech, which has undoubtedly improved the communication of these people a lot. However, there still exists a problem in this approach, which can only provide insufficient information sometimes. More precisely, it does not allow to recover some contrasts, for example [p] vs. [m], which is caused by the similarity of labial shapes. As a result, this problem makes it difficult for deaf or hearing impaired people to access speech only by the traditional oral education. Many methods have been proposed to overcome this problem up to now, and most of them use hand codings to provide additional information.

Figure 1.1: Examples of cueing American English CS (from the NCSA).

Now we present a more detailed description of how CS overcomes the limit of lip reading. As we know, spoken English has more than 40 phonemes, and spoken French has 36 phonemes. The normal hearing people can distinguish these different phonemes from auditory speech. However, some of them may look similar on the mouth, and thus it is difficult to distinguish them visually. As a result, a skilled lip reader cannot discern all the information during a conversational discourse without the help of the semantic and syntax context. The amount of information, which can be perceived, ranges from only about 20% to 60% [29]. For example, the phonemes [m, b], and [p] look very similar on the mouth. If one says the following two sentences (in French) without voice or in a high-noise environment, it would be almost impossible for the lip readers to distinguish them:

Il mange des frites;
Il marche très vite.

It is possible that the logic based on the context information may help an adult to figure out such pair of sentences. Take the following two sentences for example:

Il marche très vite;
Il marche des frites.

It is easy to see that the first one is the sentence we are listening to, while the second one is impossible. However, it is difficult for a child, who is two, three or four years old, to perceive higher-level context and logic thinking. Even for the adult people, if some of the details are missing or uncertain in the process of discourse, it will be difficult for them to keep on track. This is the reason why the CS system is introduced. A cue-reader in this system has access to all of the information through visual information. For the above problem (e.g., the confusion among [m], [b] and [p]), different hand shapes are used to distinguish these phonemes which look very similar on the lips shape.

The CS solves some concrete problems which are confronted by a large number of hearing impaired children. These problems[5] are summarized by Dr. Cornett as follows.

(1) "The limited communication in the early years, resulting in retarded personality development and delayed social maturation.

(2) The delayed and limited acquisition of verbal language, including its vocabulary, syntax, and common idioms. Rapid early verbal language rarely occurs in the very young child through only traditional methods, e.g., Sign Language (SL) and oral education.

(3) Failure to acquire an accurate and extensive model of the phonological details of the spoken language. Such a model is indispensable for accurate speech patterns and maximum development of speech reading ability. The needed base for reading is constituted by the needed linguistic competence in vocabulary, syntax, and idioms, together with the phonological model. The six years old child needs this base to learn to read easily and enjoyably.

(4) The lack of a convenient method of clear communication in the classroom, at home, and elsewhere, for use in interaction, for instruction, for clearing up the disagreement, for making clear pronunciation, and for increased awareness of and involvement in whatever is going on."

1.1.2 The construction of Cued Speech

In this subsection, we will introduce the criteria of the CS construction, especially about the American English CS and LPC.

[5] http://www.cuedspeech.org/cued-speech-what-and-why.php

1.1.2.1 The criteria of Cued Speech

The first CS system was built for the American English by Cornett [5] based on four hand positions and eight hand shapes (see Fig. 1.2), in which two major criteria were set: the minimum effort for encoding, and the maximum visual contrast. The number of hand shapes and hand positions is limited in order to save the energy for encoding CS. In CS, the basic unit is CV (consonant-vowel) syllable. Hand shape codes consonants and position codes vowels of syllables. To optimize the CS system, consonants with similar lips shapes should be grouped into different hand shape groups. The same principle is used for the vowel case. The other point is to maximize the visual contrast for each group. In American English CS system, Dr. Cornett used the frequency tables in [30] to group all the phonemes. The aim is to make the coding spend minimum energy and facilitate the hand movement in CS coding. As special cases, the isolated consonants and vowels are coded by the corresponding shape and the neutral shape (No. 5), respectively, at the side position (see Fig. 1.2).

As a result of above criteria, the most commonly used consonants such as [t], [m] and [f] are encoded by the shape No. 5 which is easier to remember and perform. For the vowels, Dr. Cornett defined three lips configurations: open, flattened-relaxed and round. According to the vowel viseme, hand positions are distributed to distinguish the vowels in a viseme. The diphthongs are encoded by shifting the hand between two vowel positions.

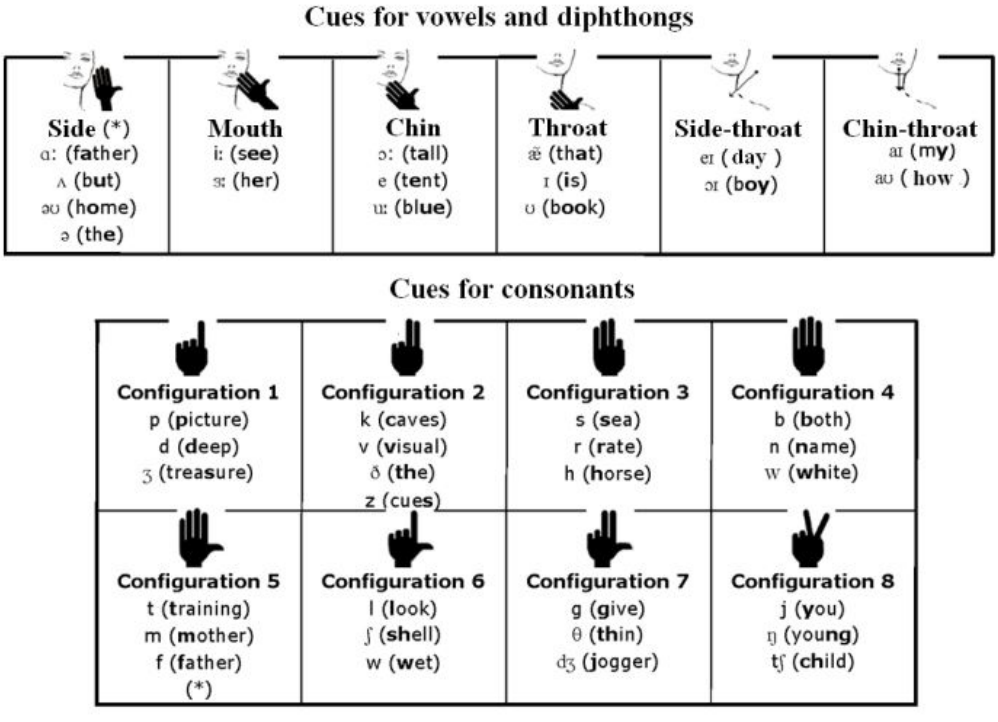


Figure 1.2: Manual cues of American English CS (from [31]).

1.1.2.2 LPC: The French version of Cued Speech

The French CS was called *Langage Codé Cornett* when it was first introduced in 1977. Then, it evolved into *Langage Parlé Complété*, and was finally changed to Langue française Parlée Complétée (LPC) [7] (see Fig. 1.1) to show that it is completely based on the French language. LPC possess the two criteria as American English CS, as well as the functioning principle and the coding unit (see Fig. 1.3). However, there are still some difference between LPC and American English CS. LPC uses five hand positions to code French vowels instead of four. Besides, French language does not have diphthongs. Therefore, there is not rules for coding diphthongs by sliding the hand position movements. In order to make an easier adaptation of LPC, the hand shape codings are similar to the American English CS.

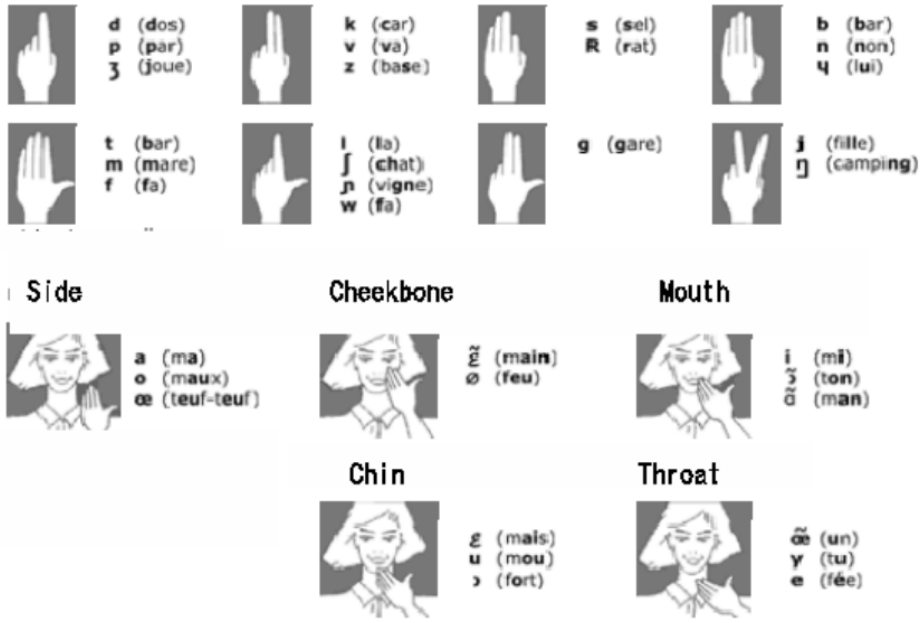


Figure 1.3: Manual cues of LPC (from [10]).

1.1.3 Other studies of Cued Speech

Concerning other studies of CS, one frequent question is what is the difference between CS and Sign Language (SL). In this subsection, we will first introduce the SL and then show the main difference between CS and SL. Moreover, the studies of CS in speech perception and automatic CS processing will be introduced.

1.1.3.1 Comparisons between Cued Speech and Sign Language

We first briefly introduce Sign Language (SL), which is a widely used communication method for deaf people. SL is a language that uses manual communication to convey a speaker's ideas.

The manual communication may include the hand gestures, movement, the orientation of the fingers, arms or body, and facial expressions simultaneously. An example signing "Hello" in SL can be seen in Fig. 1.4. SL should not be confused with the "body language", since SL is a full language as other spoken languages and has its specific grammar rule. However, body language is not a language like SL, and it is a non-verbal communication system which expresses or conveys the information by physical behaviors, including the facial expression, body posture, gesture, eye movement and touch.

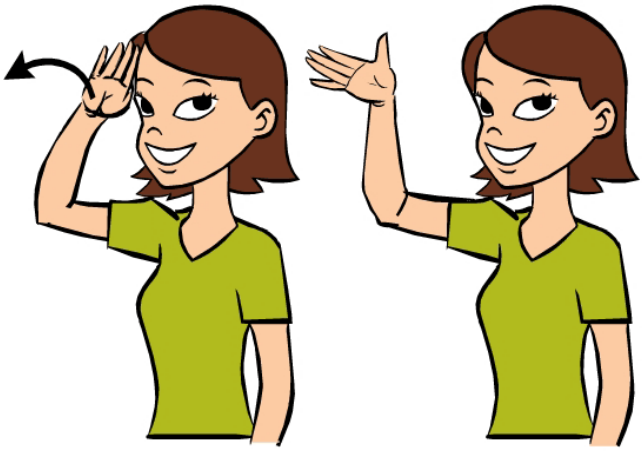


Figure 1.4: One example of SL, signing "hello"[6].

Now we show the differences between these two approaches. Firstly, SL was developed in the early 18th century while CS was invented in 1967. This may be the main reason why CS is not so popular as SL. Secondly, SL is a language with its own grammar and syntax, while CS is a visual representation of the spoken languages. SL is developed to show a language by signs with a word as the unit, instead of a phonetic unit in the CS system. For example, the signs for "dog" do not show the phonetic properties of the word as [dog]. The signs for "cat" do not indicate the phonemes [cat]. However, CS codes in phonetic level. Thirdly, to become fluent, any new language including SL often needs to take several years, while CS can be learned in a fast 20 hours training. It was reported in [32] that parents worry about that if there is no available SL model, their child may have some delay of the first-language.

Dr. Cornett pointed out that SL would always be a part of the deaf community and the CS is not going to replace the SL. However, the deaf children are able to master and learn native language easily by using CS.

1.1.3.2 The studies of Cued Speech in speech perception

Improving the speech perception efficiently is an important point of CS [6], [10], [33]–[35]. In practice, CS has shown good performance for normal hearing parents communicating with

[6] https://www.babysignlanguage.com/dictionary/first-signs/?v=11aedd0e4327.

their hearing impaired children, as well as the communication between teachers and hearing impaired children in the class.

Now we briefly introduce two well-known studies on CS perception. Nicholls et al. [6] carried an experiment to test if it is workable for hearing impaired children to perceive English words only based on visual information without any audio sound. The result gave a positive answer and confirmed the effectiveness of CS to improve the speech perception for hearing impaired children. In [35], thirty-one French deaf children were asked to perceive the French words and pseudo-words. The result confirmed that LPC can reduce the ambiguities of lip reading, and deaf children can understand the words and pseudo-words easier and faster using LPC (with very similar performance compared with the normal hearing people). A more detailed review of the effectiveness of LPC for the development of French language perception can be referred to [36].

1.1.3.3 The studies of Cued Speech production

The CS production mainly focuses on the temporal organization of lips and hand movements. In 1967, Cornett [5] emphasized that audio speech should leave some time for hand realization. In 1988, Cornett [31] proposed the *Autocuer* system, in which the LED lighting was derived from a phoneme recognition of the audio speech signal. The result showed that hand precedes the sound production about 150 to 200 ms. The first systematic study on the CS production was [37], where Attina et al. found that the hand reaches its target roughly between 171 to 256 ms before the phoneme being visible at lips. Their corpus is constituted by nonsense French syllabic sequences decomposed as *C1V1C1V1C2V2C1V1* (such as "mamabuma" logatome). Besides, in [38], Aboutabit et al. found that the hand precedes lips movement about 144.17 $\pm$ 80.68ms based on the syllables extracted from continuous sentences, which is coherent with [37].

1.1.3.4 Temporal segmentation of hand movement in Cued Speech

As we know, lips and hand are asynchronous in CS. Therefore, the temporal segmentations of them should be different. There are some automatic methods to obtain the audio based segmentation [12], [13], [39], which can be used for lips. For the hand position movement, the prior work [38], [40], [41] proposed an automatic method to temporally segment the hand movement based on the corpus with color marks. As hand positions can be extracted by tracking the blue color on subject's hand, the temporal segmentation of hand movement can be obtained based on Gaussian modeling of the hand positions and a minimum of the velocity. More precisely, as shown in Fig. 1.5, by plotting the smoothed x and y coordinates of the hand position, some local plateau defined by a set of successive identical position numbers will be obtained.

Firstly, five trained Gaussian classifiers corresponding to five positions are used to classify the hand positions in each frame. Sequential frames with a sequence number (1-5) are ob-

tained. Then a threshold is applied to the velocity to specify the boundaries of the target. The position where velocity is higher than the threshold is considered as a transition. Otherwise, it is the target hand position. By the above two criteria, the beginning and the end of each hand position interval can be determined.

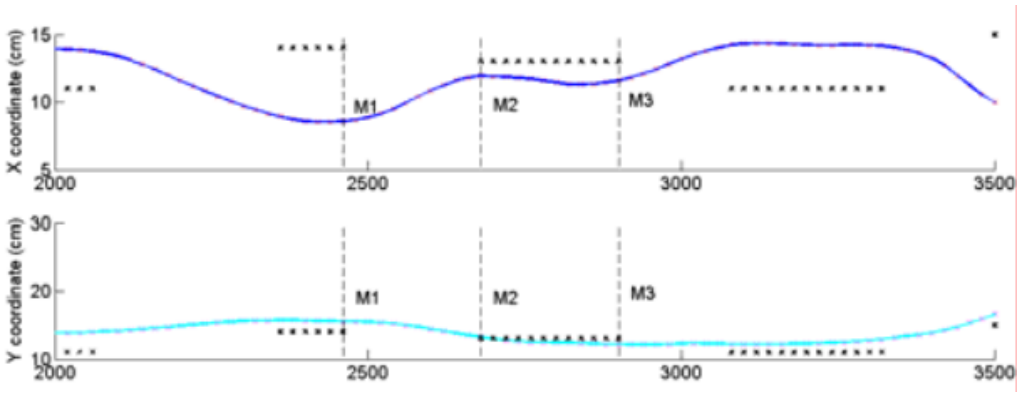


Figure 1.5: The automatic method to temporally segment the hand movement based on artificial blue marks. Two curves are the smoothed x and y coordinates of hand positions with plateaus, respectively. The black points are the classified hand positions obtained by Gaussian classifier (from [38]).

However, this method needs both spatial positions of the hand back and target finger, and thus it is not possible to directly apply this method to the database without painting any artificial mark.

1.1.3.5 The studies of automatic Cued Speech processing

The automatic processing of CS is located in the domains of synthesis and recognition. For the synthesis domain, it started with the AutoCuer system [42] in the field of assistive technology. Then, Duchnowski et al. [43] continued the idea of using audio recognition for American English, and developed a video system that displays a synthesis hand in superimposition to the image of the speaker's face. To improve their system for better use of CS deaf lip readers, the authors empirically advanced the hand with 100ms before the temporal segmentation was given by the recognition system. In 2004, at the *Institut de la Communication Parlée* (ICP) in Grenoble, Attina et al. [37] completed the pioneer work on CS production that pointed out an advance of the hand cues to the lips (between 171 to 256ms). They presented the first text to visual speech synthesizer including rules for the hand advance. In 2005, Gibert et al. [44] realized an articulatory 3D synthesizer from the analysis of a VICON recording[7]. Finally, Ming. et al. [45] built a mapping of CS hand positions from audio speech spectral parameters using the Gaussian Mixture Model (GMM).

The work on automatic processing of CS was extended to the recognition field with the first result on vowel recognition from image videos in [40], which was conducted in the context of French ANR TELMA project [46]. Then, the isolated CS phoneme recognition was realized

[7] This recording was conducted in the context of French ANR ARTUS project.

in [10]. After, the continuous CS speech based on an isolated word corpus was conducted in [11]. In all the previous work on CS recognition, the video images were recorded with artifices applied to the CS speaker before the recording (blue sticks on the lips, blue marks on the hand and forehead) in order to mark the pertinent information and make their further extraction easier.

1.2 The state of the art of Cued Speech Recognition

CS recognition aims to develop an automatic system to enable or enhance the communication between the deaf and hearing people. This automatic system has to recognize lips shapes and hand gestures (hand positions and hand shapes) to recognize CS. As in Section 1.1.3.5, researchers have been trying to explore the CS recognition which consists of lips, hand feature extraction and recognition. In this section, we present the state of the art of them.

1.2.1 The state of the art of lips feature extraction

Lips feature extraction is one of the most active and challenging research areas in speech processing and computer vision, and it is certainly an important task for CS recognition as well. We now introduce the lips feature extraction in CS case, and the general methods in some other fields.

In the literature, a common way [10], [11] to extract lips feature was realized by tracking the color landmarks on speaker's lips (see Fig. 1.6). A threshold was applied to this gray level image to segment the blue lips. In [10], they obtained 68% accuracy for vowel recognition and 52.1% accuracy for consonant recognition using only lips features. However, the disadvantage of this approach is that the threshold for detecting the blue color is difficult to satisfy in some cases such as when the hand overlaps lips area or the lighting condition changes slightly.

The machine learning methods have also been used for lips feature extraction in CS. Stillitano et al. [47] used active contours combined with parametric models to extract the lips contour in CS. Their algorithm could get an accurate segmentation of the lips contour. However, this algorithm sets many thresholds for the lips parameter model, and is not robust for unstable experimental conditions.

The lips feature detection is also an essential problem in some other areas. Given the lips ROI, several algorithms such as snakes model [48], templates based methods [49], and active shape and appearance models [50]–[52] can be used to obtain the lips contour. In order to extract the nonlinear properties of the lips image, several deep learning approaches for visual speech recognition have been presented recently. Hlavac [53] applied the Convolutional Neural Network (CNN) [26] to the lips landmark detection. The error is 0.97 pixels per point on the test data. Most errors come from the chin area, since no robust feature could lock the exact positions of the landmarks. Noda et al. [54] extracted the visual features using CNN for each frame from raw images. Using only the visual features, the accuracy of their method is about

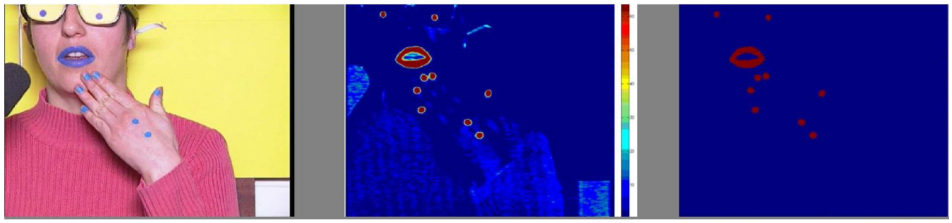


Figure 1.6: Three steps for lips shape and hand gesture detection based on blue marks in CS. From the left to right, it shows the original image, the feature map extracted from the blue component and the effect after applying the thresholds for the blue marks (from [10]).

20% for recognition of Japanese phonemes. Using only the lips features, Li et al. [55] proposed a lip reading approach to the isolated word recognition based on a dynamic feature and CNN, and 71.76% accuracy was obtained.

1.2.2 Hand feature extraction

Let's us recall that hand coding is used to complement the lip reading to make the phonemes visible in CS, and thus it carries a lot of significant information. Moreover, it is also an essential topic in other research fields like motion capture system, SL recognition. Now we present the hand tracking in CS and some other fields.

In CS, hand feature includes the hand position and shape. The correct hand shape feature is required for consonant recognition, and the correct hand position is required for vowel recognition. In the literature, the classical methods to extract the hand features were based on the artifices on subject's hand. Heracleous et al. [10], [11] proposed a method based on the x and y coordinates of the color landmarks placed on the fingers (see Fig. 1.7). The coordinates of the color landmarks on the finger are used as features for the hand shape and hand position modeling. Except for the above method which uses colored marks on the speaker's hand, in [56], speaker wore a one-colored glove in order to make the hand segmentation (see Fig. 1.8).

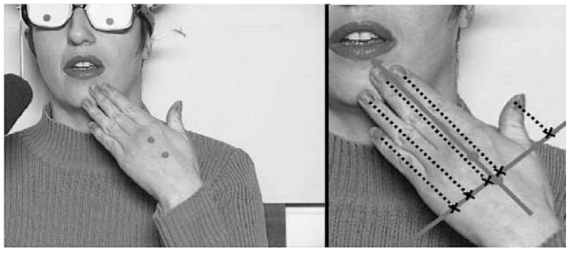


Figure 1.7: Hand shape feature extraction based on the projection of the blue marks. The left image shows the CS speaker, and the right image shows the projection method to obtain the hand shape features (from [10]).

Meanwhile, hand feature extraction is also very active in some other fields, such as SL and gesture recognition. Lots of studies have also been dedicated to the hand tracking of some other fields in the literature. For the hand shape feature extraction in SL, Gonzalez et

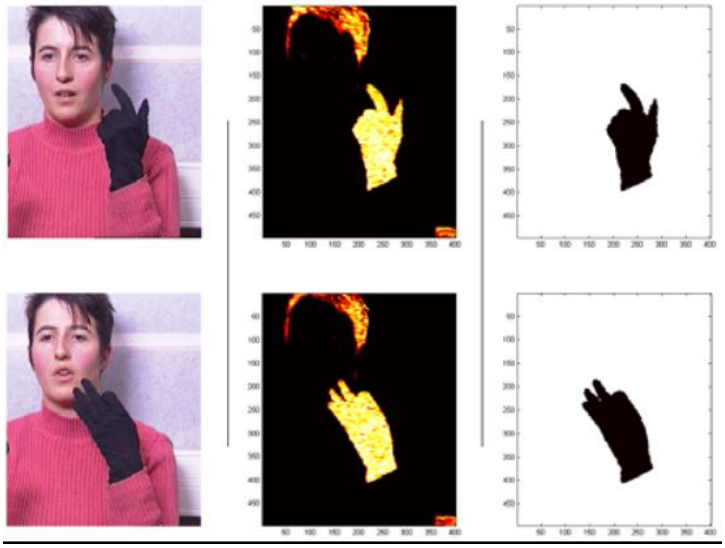


Figure 1.8: Hand segmentation based on the detecting glove. The original images, similarity color maps and segmented hand shapes are shown from left to right (from [56]).

al. [57] presented a method based on the combination of pixel color and edge orientation to segment the hand over the face accurately. Experiments using the Dicta-Sign corpus[8] indicate that this method can extract the hand effectively based on their own SL corpus. In [58], an American Language recognition system was presented with a vocabulary of 30 words. An appearance based representation was constructed and a hand classification was realized by a Hidden Markov Model (HMM) [24], [25]. An error rate of 10.91% was achieved on the RWTH-BOSTON-50 database. The approach in [59] used the *Microsoft Kinect* to extract the appearance based hand features and track the 2D and 3D positions. The classification results were obtained by comparing an HMM approach with the *sequential pattern boosting* (SP-boosting). This resulted in an accuracy of 85.1% with 40 gestures. An Artificial Neural Network (ANN) [60] was used for the classification. This method is able to recognize 20 Italian gestures with a cross-validation accuracy of 91.7%. On the other hand, in recent years, some related work on SL recognition (hand shape recognition) based on deep learning has been explored [59], [61]–[69]. Due to the availablity of large datasets like RWTHPHOENIX-Weather-2014 [67], researchers paid more attention to the continuous hand shape recognition using Deep Neural Network (DNN) in SL. CNNs were widely used in image processing and with good performance. Pigou et al. [68] considered a hand recognition system using the Microsoft Kinect (see Fig. 1.9), CNNs and Graphics Processing Unit (GPU) acceleration. The DNN was combined with HMM-based temporal modeling in [66]. In 2017, Camgoz et al. [62] proposed the deep learning architecture *SubUNets* for SL recognition. The result showed this architecture achieved more than 30% improvements compared with the state of the art.

In the field of people's affective and cognitive mental state recognition, it is also important to analyze facial expressions and hand gestures. Mahmoud et al. [70] presented an automatic detection for the hand-over-face gestures based on multi-modal Support Vector Machine

[8] http://www.sign-lang.uni-hamburg.de/dicta-sign/portal/

(SVM) with the feature of Histogram of Oriented Gradients (HOG) and Space-Time Interest Point (STIP). Experiments showed that the method recognizes the hand shape (fingers, open hand, closed hand, two hands) with an accuracy of 36%, and the hand action (static and dynamic) with an accuracy of 76%. Hand gesture recognition is also important for designing touch-less interfaces in cars. Such interfaces are very friendly and allow drivers to concentrate on driving while intercommunicating with other things, e.g., audio and air conditioning, and thus improve drivers' comfort and safety.

(a) RGB (b) Depth map (c) User index (d) Skeleton

Figure 1.9: Hand recognition system using the Microsoft Kinect for SL (from [68]).

Figure 1.10: The refined coding scheme for hand-over-face gesture descriptors (from [70]).

More importantly, it is worthy to mention the recent popular gesture pose tracking system *open-pose* [71]–[73], which is very robust even with the occlusion (see Fig. 1.11). This approach learns the body guesteures by a non-parametric representation which called Part Affinity Field (PAF). Motivated by its attractive advantages, we test it in our CS case for hand tracking, and the experimental results will be presented in Section 2.3.4.

Figure 1.11: A visualization of open-pose system for body gesture detection (from [73]).

1.2.3 Cued Speech recognition

Audio Visual Speech Recognition (AVSR) is a way which uses the information of lips to help audio speech recognition, while Visual Speech Recognition (VSR) only considers the information of video image such as lips and tongue. CS is located in VSR and can be considered as a visual supplement component to augment lip reading. Automatic CS recognition shares some issues with some other fields of multi-modal speech processing, such as AVSR[9] [28], VSR (i.e., automatic lip reading) [74], silent speech interfaces [75], [76], and gesture recognition, which includes the SL recognition introduced in Section 1.2.2. Now we present the state of the art of the automatic lip reading and the CS recognition, and the fusion methodologies will be introduced in Chapter 4.

Automatic lip reading

Automatic lip reading plays an important part in CS recognition. In the literature, many studies [77]–[83] have been dedicated to this field. An extensive review of the automatic lip reading can be referred to [84]. For the traditional methods, two steps are included generally. The first is the visual feature extraction and the second is the classification. Besides, deep learning is becoming very popular in automatic lip reading. The general framework includes a CNN to extract the lips features and a Recurrent Neural Network (RNN) to construct the model temporally. For example, the *LipNet* was proposed in [85] as the first end-to-end sentence-level lip reading model. It achieved 95.2% accuracy on the GRID corpus [86], outperforming the state of the art [87]. Chung et al. [88] developed the Watch, Listen, Attend and Spell (WLAS) model based on Long Short-Term Memory (LSTM). This model used an

[9] Indeed, in a large scope, we usually think that CS recognition is located in the AVSR area since lip reading can be seen as an audio articulatory activity.

attention mechanism to operate the visual and audio data, and the evaluation on the BBC television database confirmed the good performance.

Automatic Cued Speech recognition

For the automatic phoneme CS recognition, the fusion of three modalities (i.e., lips, hand position and hand shape) is required. For vowel recognition, lips and hand position features are merged. For consonant recognition, lips and hand shape features are merged. Now we present the state of the art of automatic CS recognition including the isolated and continuous cases, while a summary can be seen in Table 1.1.

Table 1.1: A summary of the previous studies on CS recognition. [11] is for the continuous recognition case with corpus of French words, while others are for the isolated recognition case with corpus of sentences. The audio-based temporal segmentations are used for the features of three streams, and the recognition classifier is HMM-GMM except that the simple Gaussian classifier is used in [41].

Vowel recognition (lips + hand positions)			
	Features	Fusion strategy	Results
Aboutabit et al. [41]	Corpus with artifices	decision-level	77.6%
Heracleous et al. [89]	Corpus with artifices	feature-level	85.1%
Heracleous et al. [10]	Corpus with artifices	model-level	87.6%
Heracleous et al. [11]	Corpus with artifices	feature-level	88.9%
Consonant recognition (lips + hand shapes)			
Heracleous et al. [10], [11], [90]	Corpus with artifices	feature-level	78.9%
Aboutabit et al. [91]	Corpus with artifices	model-level	79.6%
Phoneme recognition (lips + hand shapes + hand positions)			
Heracleous et al. [10]	Corpus with artifices	feature-level	61.5%
Heracleous et al. [10]	Corpus with artifices	decision-level (Two pass schemes + two GMMs)	70.9%
Heracleous et al. [10]	Corpus with artifices	decision-level (Two pass schemes + eight GMMs)	74.4%
Heracleous et al. [11]	Corpus with artifices	feature-level	82.9%

For the **isolated CS recognition**, Aboutabit et al. [41] focused on the identification of vowels by merging CS hand positions and lips information of SC corpus. Hand position was conducted using the Gaussian classifier which took the 2D hand positions as input. The

vowel recognition used the merged features of the lips and hand position, and obtained 77.6% identification correctness.

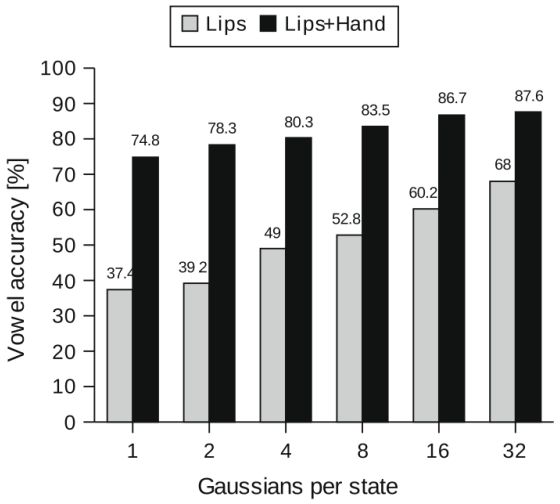
In [10], Heracleous et al. used the context-independent HMM-GMMs to decode a set of isolated phonemes extracted from CS sentences, i.e., the temporal boundaries of each phoneme to be recognized in the video was given at the test stage. In fact, the audio based temporal segmentation was used for the temporal alignments of the lips, hand position and shape. The corpus was derived from a video recording of the CS speaker with blue colors on lips and hand pronouncing and coding a set of 262 French sentences. The experiments concerning the vowel, consonant and phoneme recognitions were presented.

For vowel and consonant recognitions, lips and hand features are merged using one stream HMM and multi-stream HMMs. The results are shown in Fig. 1.12, we see that using GMMs (with 32 components) and the model-level fusion, vowel recognition obtains 87.6% accuracy (see Fig. 1.12(a)), and consonant recognition 78.9% (see Fig. 1.12(b)). We can observe that the vowel and consonant recognition accuracy increased with the the increasing number of Gaussian mixture components. Besides, the case combining lips and hand information clearly outperformed the case using lips information only.

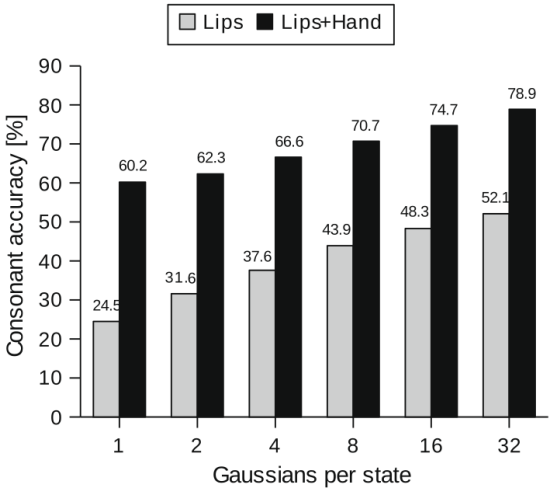
For the phoneme recognition, Heracleous et al. [10] proposed three approaches for the feature fusion. The first method is simply concatenating lips, hand position and shape features, while the second and third one are the one pass scheme and two-pass scheme, respectively. More precisely, for the first pass, the input data are fed to GMMs. The input is considered as a vowel or a consonant based on the likelihood of GMMs. When the likelihood of the vowel-GMM is higher than that of the consonant-GMM, the decision is made for a vowel. For the second pass, vowel and consonant recognitions are realized by HMMs. The second and third methods have different numbers of GMMs and training units. Two GMMs are used for training vowels and consonants in the second method, while eight GMMs are used for training 3 vowel visemes and 5 consonant visemes in the third method. It is shown that the third method gives the best phoneme recognition (74.4%). Besides, isolated word recognition experiments based on the word corpus introduced in Section 2.2.1 were conducted [10]. The normal hearing subject obtained a higher accuracy (94.9%) than the hearing impaired subject (89%).

For the **continuous CS recognition**, the prior work [11] explored it based on context-independent HMM-GMMs. However, the dataset in [11] is composed only of isolated words repeated several times (not continuous sentences). Fig. 1.13(a) shows the phoneme correctness using the information of lips only, hand shape only and merged feature in case of the normal hearing speaker. We can observe that the correctness increase significantly when merging the hand features and lips information. More precisely, it achieves correctness: vowel 88.9%, consonant 86%, and phoneme 82.9%. Compared with the consonant recognition, vowel recognition achieved a higher recognition performance. It may be due to the limited visual information of lips (e.g., [k], [g]), which causes more confusions between different consonants. Indeed, the lips contrast for vowel may be more evident than the consonant case.

The CS recognition results for deaf speakers are shown in Fig. 1.13(b). We can also see

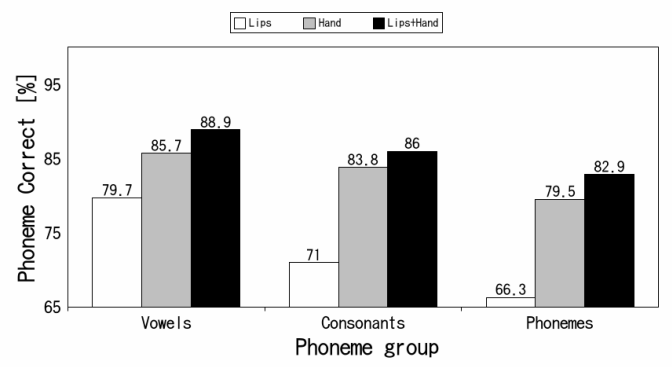


(a)

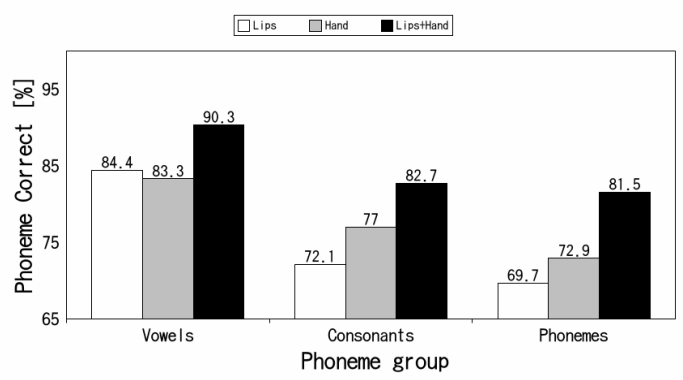


(b)

Figure 1.12: CS (a) vowel and (b) consonant recognition results based on the multi-stream HMM model-level fusion in [10].

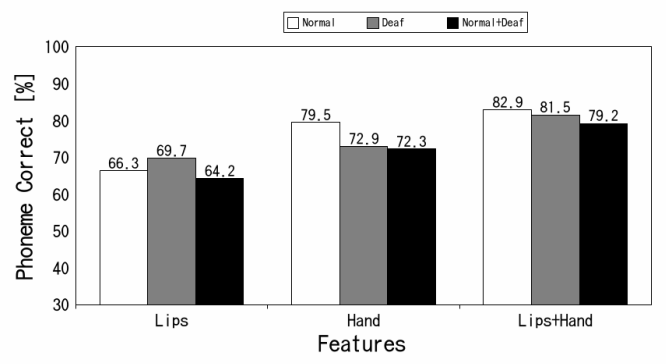


(a)



(b)

Figure 1.13: CS recognition results of only lips, only hand and merged feature of them for (a) normal hearing and (b) deaf speakers (from [11]).



(c)

Figure 1.14: Comparison of the phoneme recognition correctness for normal hearing and deaf speakers (from [11]).

that the recognition performance increases significantly when merging the lips and hand shape information. The results of normal hearing and deaf speakers can be seen in Fig. 1.14, where very slight difference is shown. Notably, we observe that deaf speaker shows better ability in lip reading than the normal hearing subject. It is reasonable because they are used to the lip reading in daily communication and have nice perception and production of lip reading. Finally, the results of the intra-speakers experiment confirm that the normal hearing and deaf speakers have very similar performance in CS recognition.

Chapter 2

Cued Speech Materials

Contents

2.1 Introduction

In current **big data** times, the database is very critical to any statistical analysis. In this thesis, since there is no existing CS database without using any artificial mark, we employ CS teachers and speakers to utter French words and sentences used in the LPC to record our own database. In this chapter, we will introduce these databases. An organization of this chapter is shown in Fig. 2.1. Firstly, the data preparation and recording setup will be presented in Section 2.2. Then, in Section 2.3, we will focus on the data processing, which includes the processing of texts (sentences or words), audio speech signals and CS videos.

2.2 Database preparation and recording

The data recording contains the speaker's upper body (face and neck) and the audio sound. It is carried out in a sound-proof room in Gipsa-lab, France. The CS speakers are professional French CS translators. The speaker is seated in front of a computer screen which shows the French words or sentences to be coded. A microphone and a camera are set up for the recording. The RGB images are saved in BMP or PNG format with size 720×576 at the rate of 50 Hz. The recorded audio sound signal is digitalized at 22,050 kHz, which is to help the temporal segmentation of words and sentences.

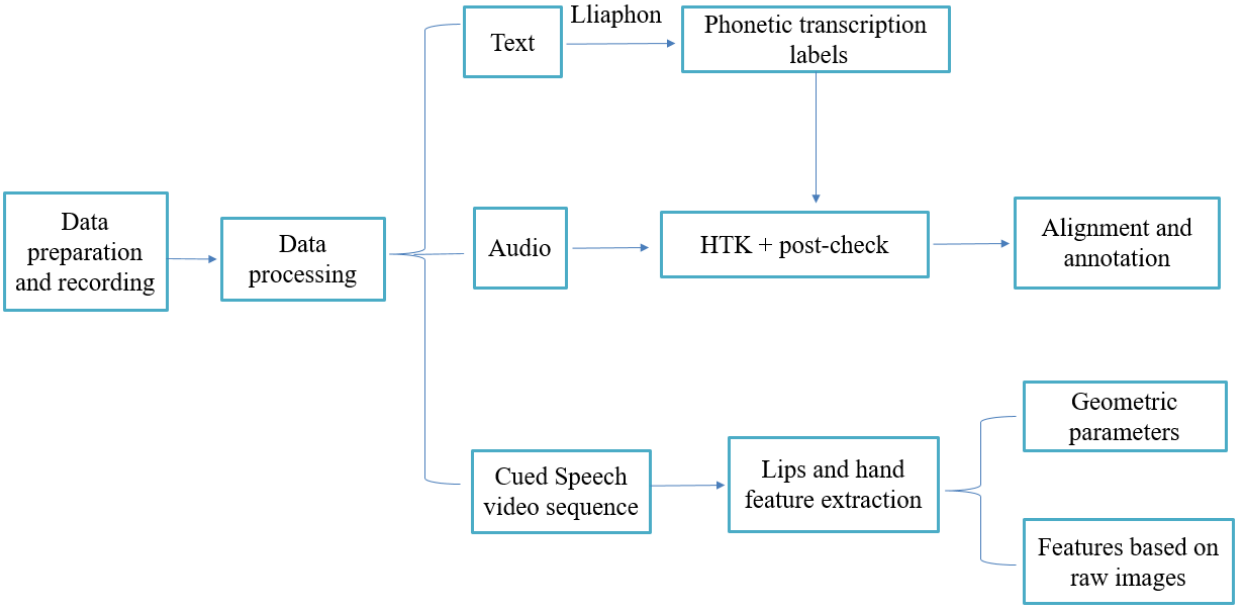


Figure 2.1: The organization of Chapter 2 including text, audio and video processing.

In this thesis, we work on five subjects named as *MD*, *DB*, *ChS*, *SC* and *LM* (see Fig. 2.3). With audio speech sound, three of them (*MD*, *DB* and *ChS*) utter French words and others (*SC* and *LM*) utter French sentences. The audio sound is synchronized with the CS video to help us align the articulatory data. A summary of these five corpora is shown in Table 2.2.

2.2.1 Five corpora in this thesis

The word corpora contain a CS interpreter and two normal speakers. The corpus *MD* was previously recorded, and was exploited by Ming et al. [45] for audio to lips and hand ges-

Table 2.1: A summary of the five corpora in this thesis.

Speaker	Speech unit and amount	Speech type	Color landmarks
MD	50x10 words	CS	no landmark on lips
DB	50x10 words	normal speaker	no landmark on lips
ChS	50x10 words	normal speaker	no landmark on lips
SC	267 sentences	CS	blue landmarks on lips and hand
LM	238x2 sentences	CS	no landmark

tures mapping in CS. The speaker *MD* is a female native speaker of French. In this corpus, she pronounced and coded 50 isolated French words (see Appendix A.2). These words are constituted by the digits from 0 to 31, twelve months, and six ordinary French words such as *Bonjour*, and they were repeated ten times. For this speaker, one blue landmark was glued on the forehead, and five blue landmarks were glued on the extremity of the right hand's finger. A pre-prepared grid paper was used in the recording in order to transfer the pixel unit to centimeter (see Fig. 2.2). The two normal speakers (*DB* and *ChS*) are male French, and they uttered the same French words as *MD*. Other recording setups are also the same as *MD*. The main difference is that they speak normally instead of CS.

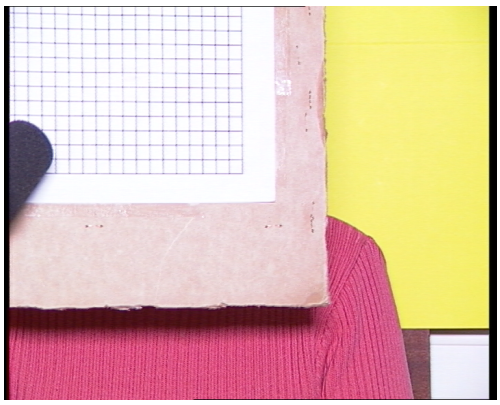


Figure 2.2: Experimental setup of the transformation from pixel to millimeter.

The sentence corpora without using any artifice are collected from one female normal hearing CS speaker named *LM*. The experimental setup is the same as *MD*. The speaker *LM* pronounced and coded a set of 238 French sentences derived from a corpus in [44], [92]. Each sentence was repeated twice by the speaker resulting in a set of 476 sentences (totally 11772 phonemes). One example is *Ma chemise est roussie*. The sentence corpus named *SC* with color marks was recorded for the previous study [10], [40]. The speaker's lips and hand were painted by blue color landmarks. She coded 262 French sentences which are also derived from the corpus in [44], [92]. The sentences uttered by these two subjects are not the same but with a large part of sentences in common.

To validate our proposed algorithms for other languages, we also recorded the first British English CS corpus. It was coded by a CS expert from the CSAUK. The videos with 720x1280 RGB images (50 fps) were recorded. Until now, we have recorded 100 English sentences, and the recording processing is still ongoing.

In order to see the balance status of phonemes in each corpus, we calculate the number of vowels, consonants, and phonemes in the corpora *MD*, *SC* and *LM* (see Table 2.2). The word corpora *DB* and *ChS* have the same amount of vowels, consonants and phonemes as *MD*. It can be seen from Fig. 2.4 that the phonemes are not well balanced for all these three corpora. Notably, the word corpus has a different distribution with the sentence corpus concerning the phoneme amount. Note that there are some unbalance of the corpus *MD* since the selected 50 isolated French words are often made by some common phonemes such as *vingt-et-un* and *trente-et-un*.

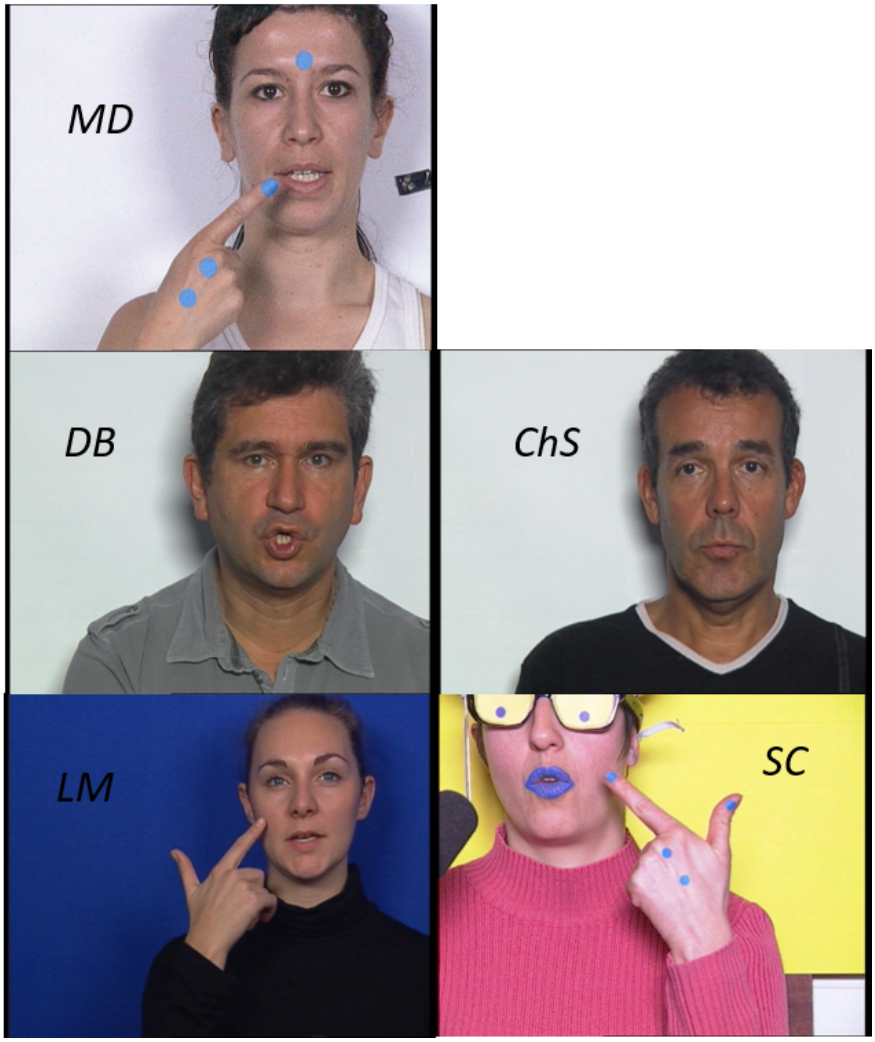
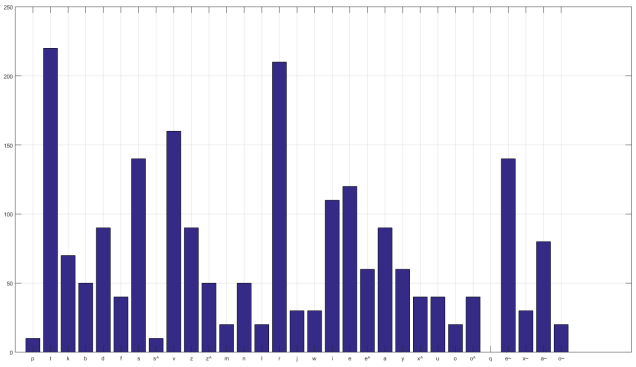


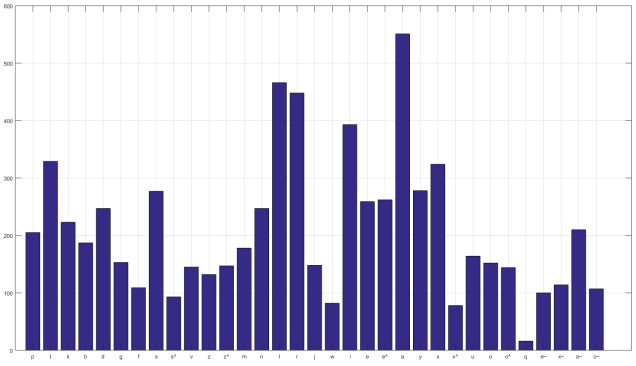
Figure 2.3: Images of five subjects in this thesis. *MD, DB* and *ChS* utter French words, while *LM* and *SC* utter French sentences.

Table 2.2: A summary of the vowels, consonants and phonemes for the corpora of this thesis.

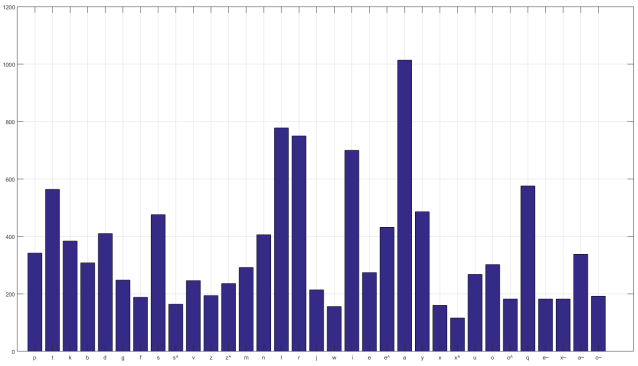
Speakers	Vowels	Consonants	Phonemes
MD	850	1290	2140
DB	850	1290	2140
ChS	850	1290	2140
SC	3152	3816	6968
LM	5404	6356	11760



(a)



(b)



(c)

Figure 2.4: The statistical analysis of the amount of phonemes for corpora: (a) *MD*, (b) *SC* and (c) *LM*.

Table 2.3: Phoneme to viseme mapping in the French language (from [93]), including five consonant visemes and three vowel visemes.

Consonants		Vowels	
Viseme	Phonemes	Viseme	Phonemes
C1	/p/, /b/, /m/	V3	/ɔ̃/, /y/, /o/ /ø/, /u/
C2	/f/, /v/	V1	/a/, /ɛ̃/, /i/ /œ̃/, /e/, / ɛ/
C3	/t/, /d/, /s/ /z/, /n/, /ɲ/	V2	/ɑ̃/, /ɔ/, /œ/
C4	/ʃ/, /ʒ/		
C5	/k/ , /g/ /ʀ/, /l/		

Table 2.4: French phonetic reference table. Phonemes with blue color denote the labels with changes.

Simplified label	p	t	k	b	d	g	f	s	s^
Reference label	p	t	k	b	d	g	f	s	ʃ
Simplified label	v	z	z^	m	n	l	r	j	w
Reference label	v	z	ʒ	n	n	l	r	j	w
Simplified label	i	e	e^	a	y	x	x^	u	o
Reference label	i	e	ɛ	a	y	ø	œ	u	o
Simplified label	o^	e~	x~	a~	o~	q			
Reference label	ɔ	ɛ̃	œ̃	ã	ɔ̃	ə			

2.2.2 Definition of the lips viseme

We now introduce some prerequisites in this thesis. Firstly, a term named *viseme* will be introduced. As we know, in French, there are 36 phonemes including 17 consonants, 16 vowels, and 3 semi-vowels. Some consonants and vowels have very similar lips shapes. We call these consonants and vowels with very similar lips shapes as a viseme. The phoneme to viseme mapping [93] in the French language is shown in Table 2.3, where we can see that there are five consonant visemes and three vowel visemes.

Secondly, in this thesis, we use some simple labels to mark some French phonemes (see Table 2.4). For example, [ʃ] for 's^' and [ʒ] for 'z^'. The reason is that, in the programming of *Matlab* and *Python*, the simple labels can be directly used as the "char" variable, while the standard French phonetic type needs extra controls.

2.3 Data processing

As shown in Fig. 2.1, after the recording and preprocessing steps, we go to the data processing which contains the processing of texts, audio speech signals and videos. The text processing concerns the acquisition of the phonetic transcription. The audio speech signal is used to obtain the alignment of each phoneme. In the video processing step, we will present the definitions of lips and hand features, as well as some basic methods for the feature extraction. It should be noted that our proposed methods for lips and hand feature extractions will be introduced in Chapter 5 and Chapter 6, respectively.

2.3.1 Text data processing: phonetic transcription

The phonetic transcription is to translate the raw text characters (see Appendix A.1 and Appendix A.2) into phonetic sequences without using any audio signal. More precisely, the raw French sentences or words of the audio signals is first done manually in naturally written text, and then translated into phone sequences using *Lliaphon* phonetizer[1] [94], which is a text-to-speech application and translates the texts into phonetic descriptions. Lliaphon is designed for French texts and it is free to use for non-commercial and non-military applications. However, when directly applying it to our case, some errors appear. In fact, the errors in the phonetic decoding are because of that the real audio sound or hand coding may not always correspond to the phonetic pronunciation. In summary, in the phonetic transcription, there are two types of errors. Now we illustrate these two aspects by two typical examples.

The first error is the phonetic transcription error when Lliaphon processes the raw text. The following three main problems are shown in Fig. 2.5. (1) There should be a 'z', while no '_' (silence) is inserted. (2) Lliaphon usually makes some confusions between 'q' and 'x'. (3) A '_' should be added between two successive vowels ('o^' and 'o~' in this sentence). Besides, in most cases, when the syllable is ended by a consonant, one [ə] should be added in the acoustic speech, while it will not be added in the raw transcription. Given these errors, some problems will appear in the automatic alignment.

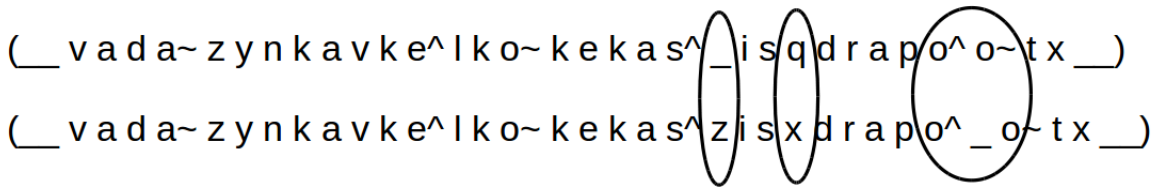


Figure 2.5: One example phonetic error from Lliaphon. The first row is the transcription by LLiaphon, while the second row is the ground truth.

The second error is mainly due to the various pronunciations or codings made by speakers. For example, in the audio speech, [ø] and [ə] are often confused, as well as [o] and [ɔ],

[1] https://gna.org/projects/lliaphon

since their pronunciations are very similar. At the same time, in the hand coding, [o] and [ɔ] are often mistakenly coded. For the vowel [o], the ground truth hand position is "side", but it often mistakenly points to the "chin" which indicates the vowel [ɔ].

The above errors will influence the performance not only on the temporal alignment but also on the final automatic CS recognition. As a consequence, it is necessary to have a post check for phonetic transcriptions.

2.3.2 Acoustic data alignment

In Section 2.3.1, we have converted the text of sentences into a sequence of phonemes including the silence. Note that the speech signal is recorded synchronously with the CS data. Phoneme sequences are then automatically aligned on audio signals using a standard speech recognition system (based on a set of tied-state context-dependent phonetic HMM trained using the *HTK* toolkit[2] and a *forced alignment* procedure). The forced alignment is a technique to take an orthographic transcription of an audio file and generate a time-aligned version using a pronunciation dictionary to look up phones for words. Also, depending on prior knowledge, we define an initial transition probability between different phonemes. The feature will be fed to a pre-trained acoustic model to obtain the likelihood score of the phoneme recognition using the *Viterbi* algorithm. The phoneme with the maximum likelihood score will be the right candidate of the current temporal boundary. We remark that different acoustic models may produce slightly different forced alignment results.

However, even with a manual temporal segmentation, it is sometimes hard to determine a correct temporal boundary of each phoneme. Thus, it is natural that the forced alignment cannot obtain a perfect alignment. Now we give some examples showing the errors of automatic alignment and the results of the post check alignment (see Fig. 2.6 and Fig. 2.7).

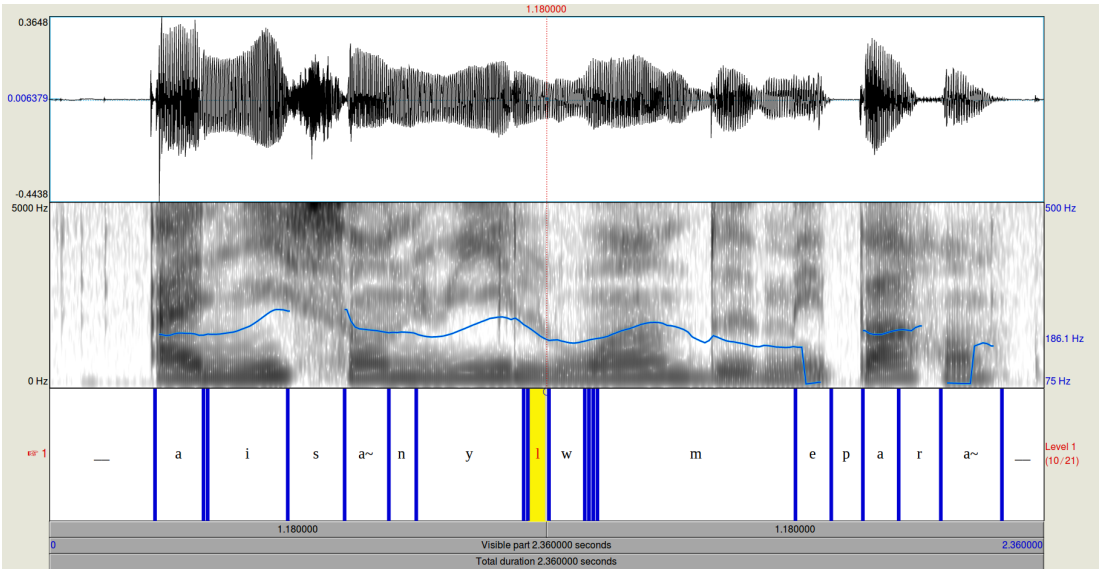
As in Fig. 2.6(a), the automatic alignment for the long sentence *Annie s'ennuie loin de mes parents* is bad (especially in the middle of the sentence). The corrected alignment is shown in Fig. 2.6(b). We observe that errors appear more evidently when there is a sequence of voiced phonemes.

It is shown in Fig. 2.7 that if there is no silence between two successive vowels in the phonetic transcription, the automatic alignment will perform badly as in the example 'o˜' and 'oˆ'.

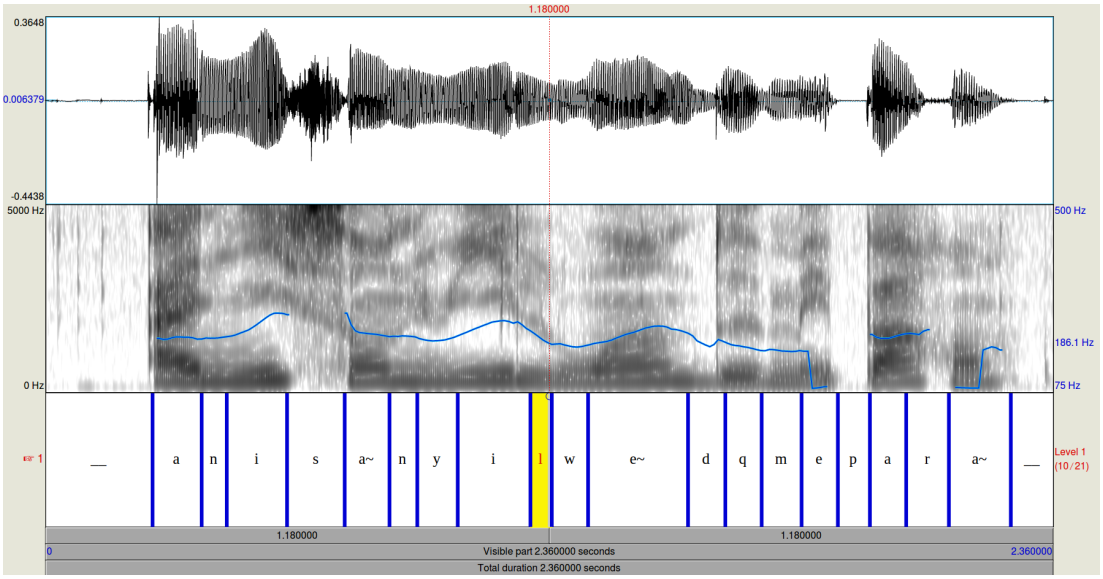
2.3.3 Video data processing

In CS recognition, one essential step is the video image processing which contains the lips, hand position and hand shape feature extraction. Now we briefly introduce the definitions of parametric feature and some common basic pixel based feature extraction methods.

[2] http://htk.eng.cam.ac.uk/

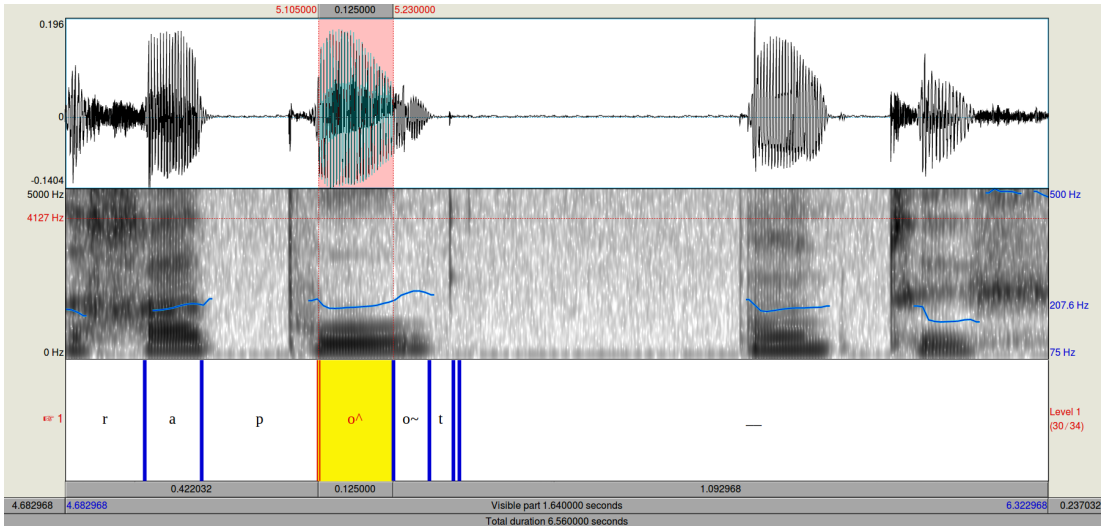


(a)

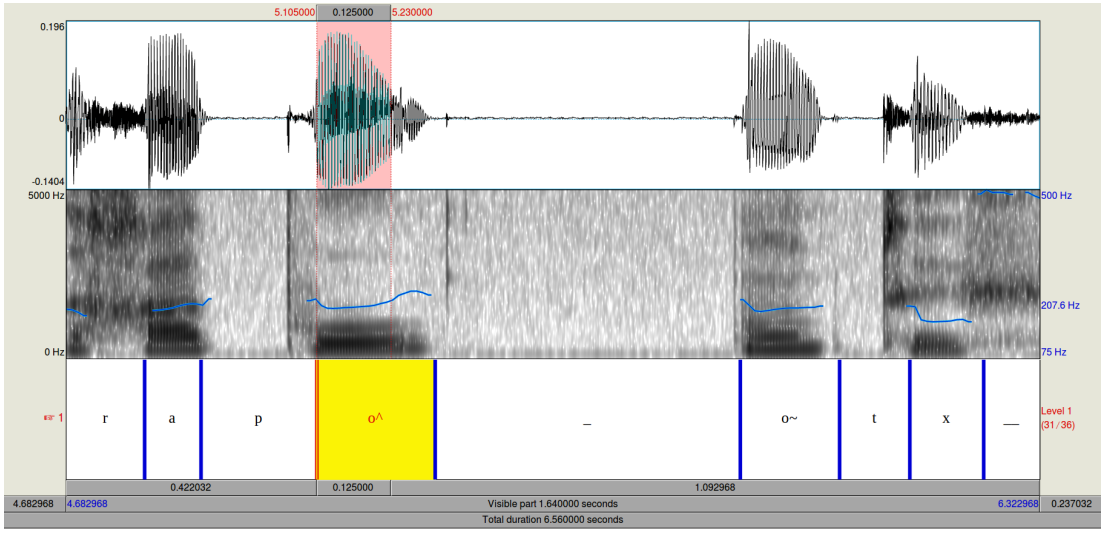


(b)

Figure 2.6: The alignment of the sentence *Annie s'ennuie loin de mes parents.* (a) is the alignment obtained by the automatic alignment algorithm. (b) is the ground truth alignment.



(a)



(b)

Figure 2.7: The alignment of the sentence *Va dans une cave quelconque et caches-y ce drapeau honteux*. (a) is the alignment obtained by the HMM automatic alignment algorithm. (b) is the ground truth alignment.

2.3.3.1 Lips features

The lips feature extraction is indispensable in CS recognition. In general, there are three levels of lips feature: geometric parameter level, contour level and pixel level (based on image) feature. In this thesis, we explore the lips feature in geometric parameter and pixel levels. In terms of the geometric parameter, the lips opening height, width [90], [95]–[97] are considered as lips features. Now we introduce the measures of geometric parameters of lips, as well as the common pixel based methods, which includes Principal Component Analysis (PCA) [98], [99] and Discrete Cosine Transform (DCT) [100]–[103].

Geometric parameters of lips

It was reported in [104] that the lips parameters such as the horizontal width, vertical height and the area of inner and outer lips are used to describe the lips shape. One example of the application is the AVSR task [105], [106]. In this thesis, we use the inner lips width A and inner lips height B as the lips features. We observe that the inner lips area has a correlation with the value $A \times B$, and the calculations of A and B can be referred to [3].

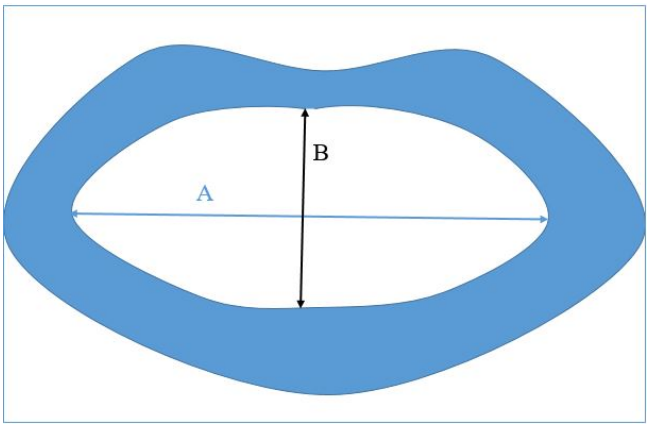


Figure 2.8: Lips parameters A and B.

As introduced in Section 1.2.1, the traditional methods used the artificial landmarks to extract the lips contour to calculate these parameters. In Chapter 5, we will propose two methods to calculate them without using any artificial landmark.

Pixel based lips features

Now we introduce two pixel based feature extraction methods (PCA and DCT), which are both applied to the raw images and able to reduce the high dimension of features.

(1) PCA based lips features

Principal Component Analysis (PCA) is widely used for the dimension reduction and feature orthogonalization [98], [100], [107]–[116]. It is an unsupervised and linear technique which aims at finding a decomposition basis that best explains the variation of pixel intensity in a set of training frames. In this thesis, the PCA is performed on a set of N training frames (normalized by its mean value in the pixel intensity domain at the training stage). The resulting basis vectors are often called *eigenlips* [108] when applying this technique to the lips images. At the feature extraction stage, each new frame is projected onto the set of these basis vectors. Visual features are defined as the D first coordinates in that decomposition basis. When applying PCA to the lips ROI in our case, in order to keep the eigenvectors that carry 85% of the variance, we set $D = 22$ (see Fig. 2.9). All the parameters of PCA are calculated based on 1000 images which are randomly extracted from the whole database (for example, corpus *LM*). The reconstruction of the lips based on different numbers of PCA components is shown in Fig. 2.10. We can see that the image with 40 components (see Fig. 2.10(d)) is better than the one with 22 components (see Fig. 2.10(c)). In this thesis, PCA will be used several times as a baseline to be compared with our proposed methods.

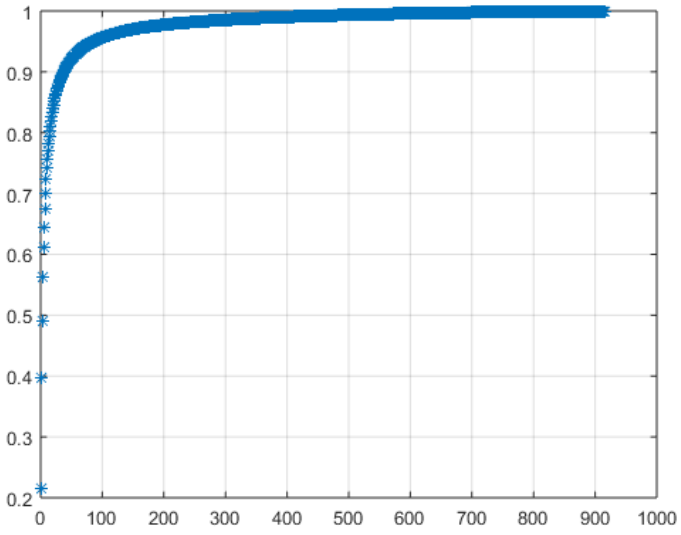


Figure 2.9: Variance ration of PCA components on lips ROI. The abscissa is the number of PCA components and y axis is the variance ratio.

(2) DCT based lips feature

Apart from PCA, there are still some other popular linear image transforms [117] to obtain the lip reading features, such as DCT [100]–[102], [109], [118], [119], Discrete Wavelet Transform (DWT) [120], and the *Hadamard and Haar transforms* [119].

The energy located in the low frequency domain is packed by DCT. Therefore, we can

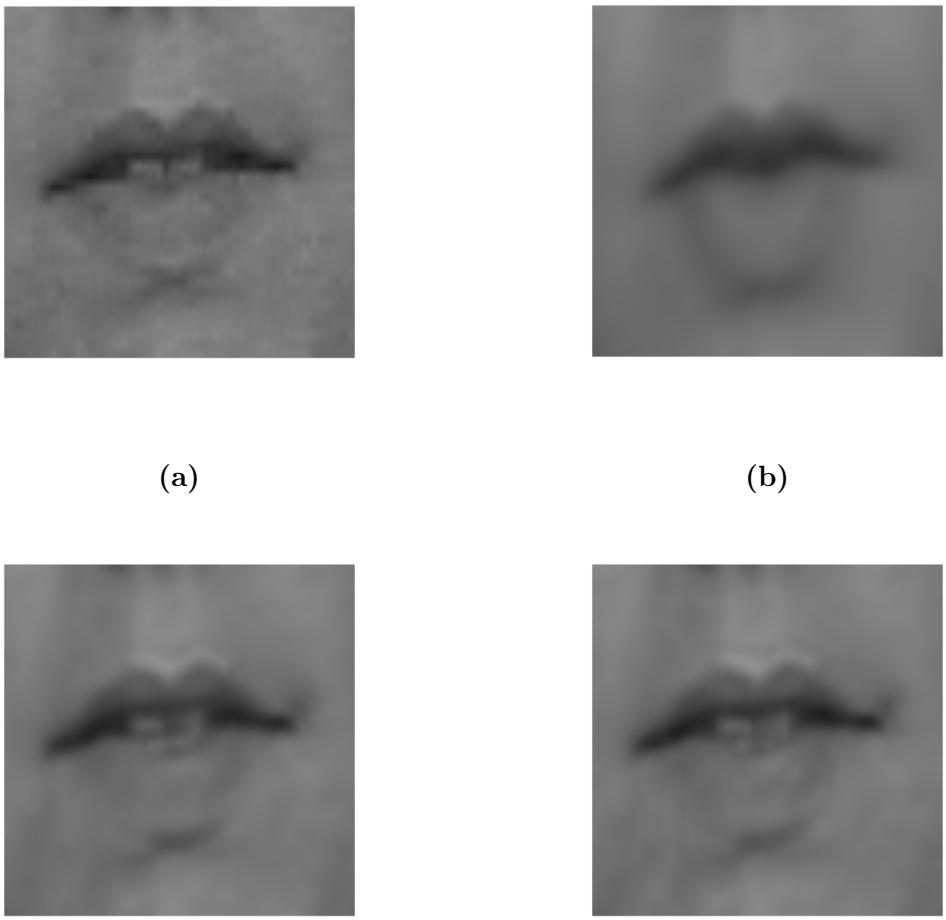


Figure 2.10: Examples of the reconstruction of lips ROI based on the PCA coefficients. (a) is the original lips ROI. (b) is the mean image of the lips ROI. (c) is the one with 22 components which can explain 85% variance. (d) is the one with 40 components which can explain 90.5% variance.

ignore the high frequency energy in order to reduce the dimension of the DCT feature.

In [45], a mask is used to select the most significant coefficients located in the top left of the DCT matrix in order to reduce the dimension. We can see that the reconstructed images (see Fig. 2.11(e) and Fig. 2.11(f)) can well represent the original gray-scale images (see Fig. 2.11(c) and Fig. 2.11(d)).

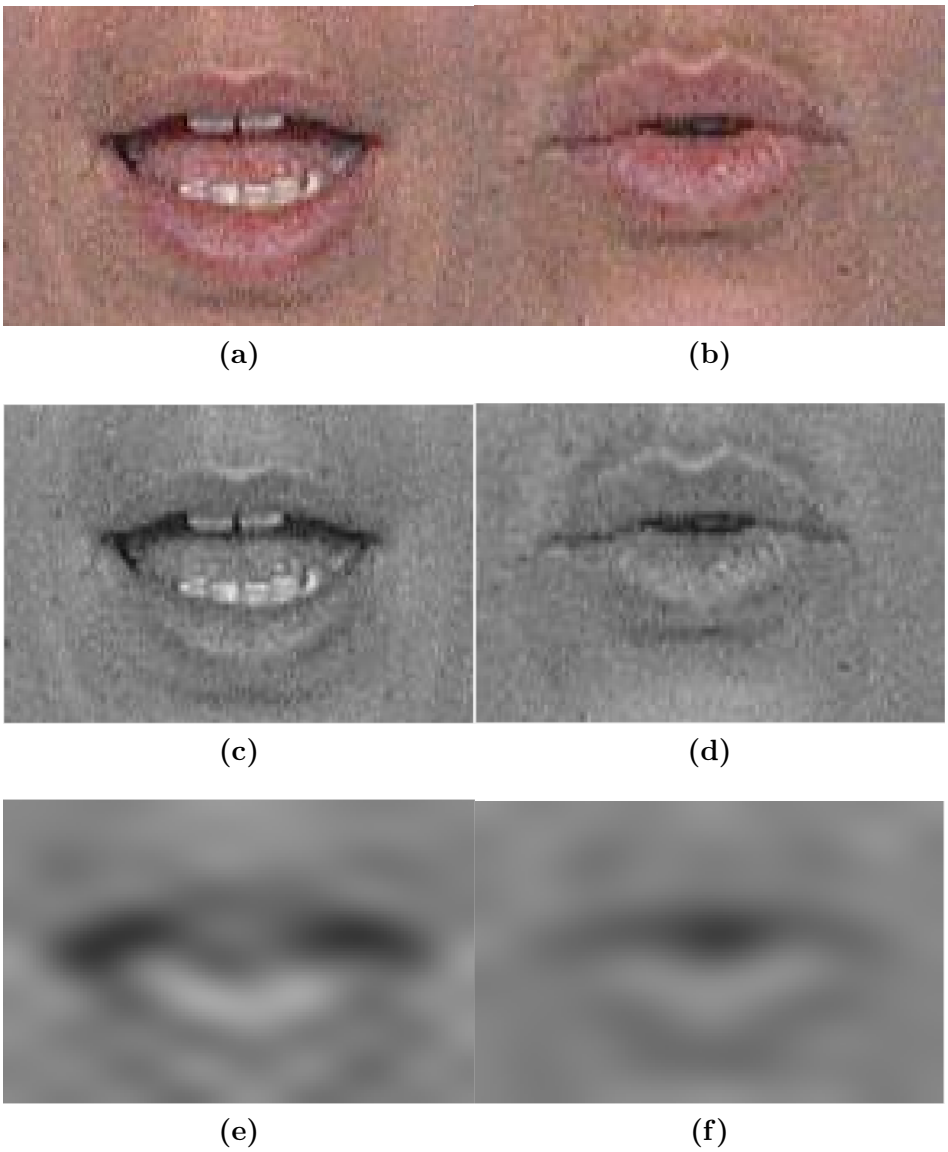


Figure 2.11: Examples of lips reconstruction based on DCT coefficients. (a) and (b) are original RGB lips ROI. (c) and (d) are the corresponding original gray lips ROI. (e) and (f) are the corresponding inverse DCT reconstruction lips ROI.

In our experiment, we found that the DCT based feature extraction method is sensitive to the environment noise such as rotations and lighting conditions, thus we do not use DCT lips features for CS recognition. However, it will be used for building a closed and round lips

detector in Section 5.2.2.6.

2.3.3.2 Hand position and hand shape features

Recall that the hand shape codes the consonant, and the hand position codes the vowel in CS. The extracted features of hand shape and position are very essential in CS recognition. Now we introduce the characteristics of the hand shape and position features.

Hand position features

Coordinates in the spatial space normally represent the hand position. However, there is no unique point which permits to define the hand position. We present three possible points to describe the hand position.

(1) The first point is on the hand back (see the blue point in Fig. 2.12), and we can track this point without any interruption since it is always visible. This point could give a good description of the whole hand movement. However, it is still difficult to automatically extract this point without using any artifice. Therefore, the extraction of this point is carried out manually, and this point is assumed to be a reference in this thesis.

(2) The second one is the target finger (see the green point in Fig. 2.12) which points to the vowel position allowing CS readers to understand what the CS coders want to express. Evidently, this is the best choice for the hand position recognition. However, the target position is not always realized by the same finger since the hand shape is variable during the coding process. The automatic CS hand finger tracking in 2D is still an unsolved problem. Therefore, we manually extract the target finger in the following way: the 2D position of the index finger is used if no middle finger appears.

(3) The third one is the gravity center of the hand, which will be introduced in Chapter 6.

The classical method in [10], [11], [40], [41] is to mark the information with the landmarks placed on the subject's hand back and finger's extremity. Then a color detection algorithm is applied to track the hand position. In Chapter 6, an automatic method will be presented to track the gravity of the hand in case of no artificial color landmark.

Hand shape features

For the hand shape, there are two levels of descriptions: parameter and pixel. In the state of the art [10], [11], different colors were painted on the finger's extremity to define the hand shape feature. To get rid of the artifices, we think about other approaches. One possible way is to extract the hand contour to describe the hand shape, which is difficult due to the dynamical changes of hand on both position and shape. The other way is to extract the

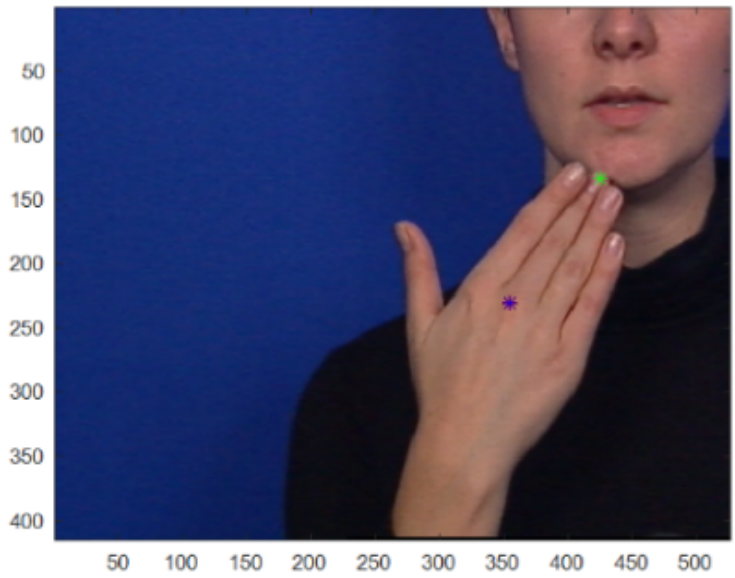


Figure 2.12: Two classical points used to track the hand position movements. One is on the hand back (blue point), and the other is the target finger point (green point).

pixel based hand features. In order to reach this goal, we need to determine a hand ROI (see Fig. 2.13), which will be introduced in Chapter 6. Then, we apply PCA on the hand ROI to extract the hand shape features. Other techniques for hand shape feature extraction will be presented in Chapter 8.

Figure 2.13: Visualization of hand ROI (corpus *LM*).

Applying PCA to the hand shape ROI, we can obtain the PCA components and variance of the hand shape (see Fig. 2.14). In order to keep the eigenvectors that carry 85% of the variance, we set $D = 34$. The reconstruction of hand shapes based on different numbers of PCA components is shown in Fig. 2.15. It can be seen that the image with 30 components (see Fig. 2.15(c)) is sufficiently clear to describe the hand shape.

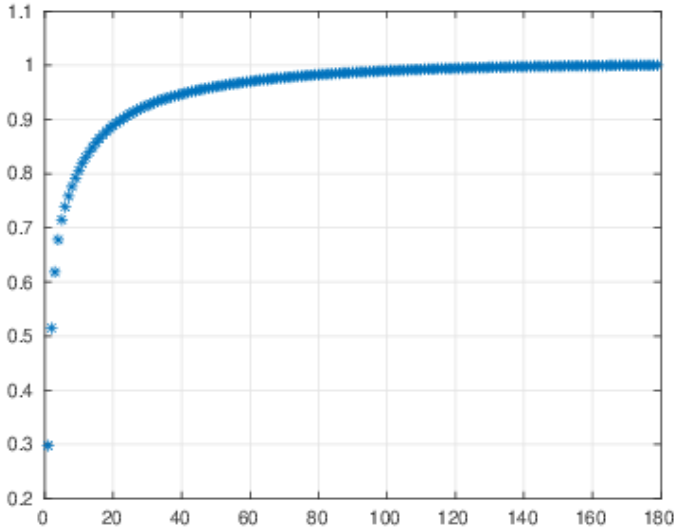


Figure 2.14: PCA performance on the hand shape ROI.

2.3.4 Other methods in Cued Speech feature extraction

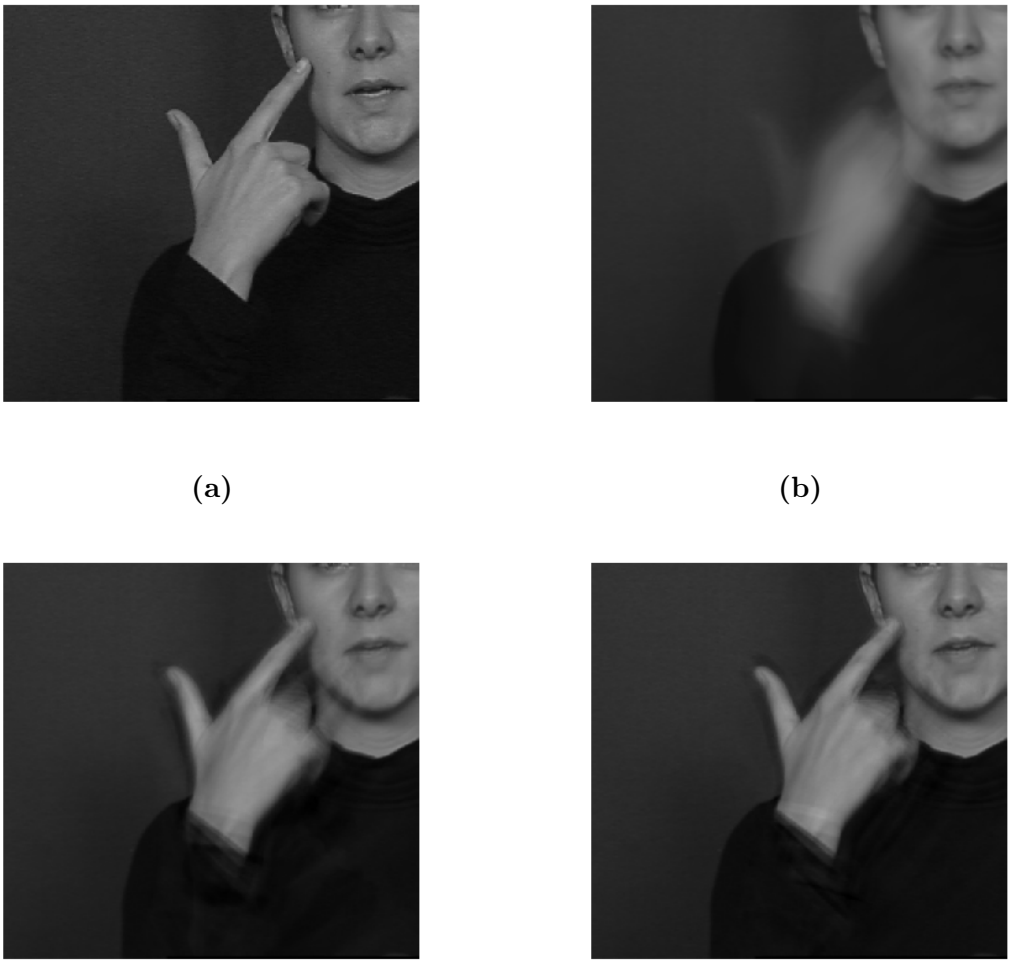
Except for the above classical statistical methods, some other approaches in computer vision fields were also tried for the CS feature extraction. Now, we briefly introduce the Kanade–Lucas–Tomasi (KLT) feature tracker [121], [122] and *open-pose* system [71]–[73].

KLT feature tracker

In computer vision, Kanade–Lucas–Tomasi (KLT) [121], [122] feature tracker is an approach to feature extraction. It is very effective for tracking the objects without changing the shape. Note that the hand moves with shape changes in CS. KLT is not suitable for the hand shape detection. However, we can use it for lips tracking since lips have less changes. The applications of KLT to hand and lips ROI are shown in Fig. 2.16. The green points are the "good" target feature points detected by KLT. In Fig. 2.17, we present different cases when applying KLT to the real-time tracking of lips and hand in corpus *LM*. We found that most of the time, the performance of lips tracking is better than the hand. In Fig. 2.17(c), the good feature of hand even disappears. It may be due to the large variabilities of the hand shape and less changes of the lips.

Open-pose system for human pose estimation

The open pose system is robust for the human pose estimation [71], [73], even with the occlusion (see Fig. 1.11). It can be seen from Fig. 2.18(a) and Fig. 2.18(b) that the open-pose system works perfectly for the gesture (facial feature, hand and body) estimation. However,



(a) **(b)**

(c) **(d)**

Figure 2.15: (a) is the original hand. (b) is the mean image of the hand. (c) is the one with 30 components which can explain 80% of the variance. (d) is the one with 50 components which can explain 85% of the variance.

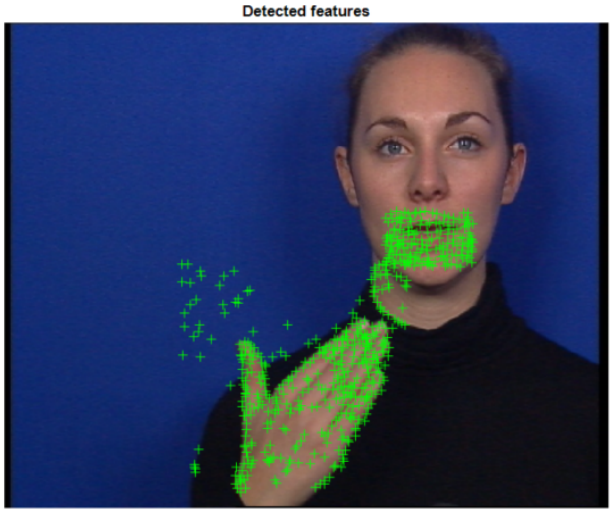
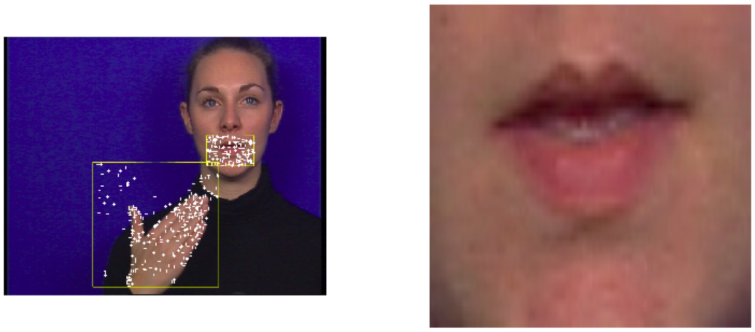
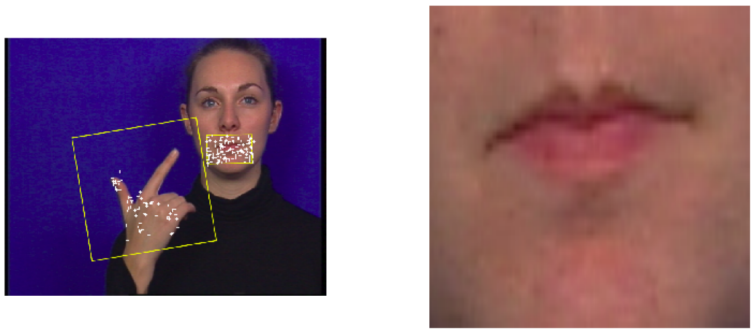


Figure 2.16: Visualization of KLT for lips and hand tracking.

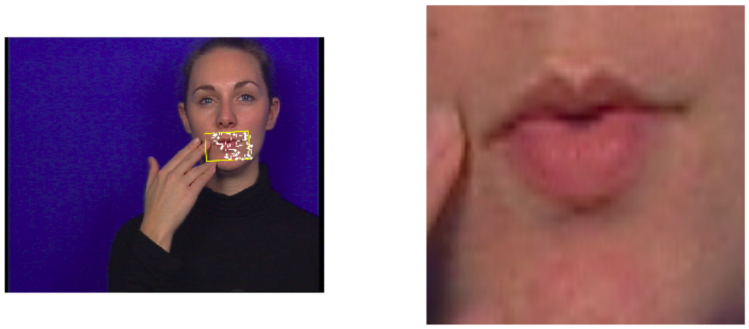
from Fig. 2.18(c) and Fig. 2.18(d), the open-pose system works well for the facial features but not for the hand and body gestures estimation. The reason is that our corpus does not record the whole upper body of the speaker in Fig. 2.18(c) and Fig. 2.18(d), and thus it does not have enough good initial features. In order to use the open-pose system directly, speakers need to expose their whole body in the recording, instead of only the body above neck in our case. Therefore, it is not enough to validate this method. We may suggest to record the whole body of CS speaker so that this method can be used in the future work.



(a)



(b)



(c)

Figure 2.17: Real-time tracking of lips and hand by KLT. The white points are the good features captured by KLT, and the yellow rectangle is the resulted ROI of hand and lips.

(a) (b)

(c) (d)

Figure 2.18: Visualization of open-pose for lips and hand tracking. (a) and (b) show the result for the open-pose applied on the database which records the whole upper body of the subjects. (c) and (d) show the result on our corpus *LM*, which only records part of the upper body.

Chapter 3

Deep Learning Methods

Contents

3.1 Introduction

Artificial Intelligence (AI) is a powerful tool that can make a machine act like a human. As a significant step for AI, *machine learning* provides the machine with the necessary data to **learn** how to do something without being explicitly programmed. Algorithms such as *decision tree learning, inductive logic programming, clustering* or *reinforcement learning* help them make sense of the input data. Furthermore, *deep learning* is a form of advanced machine learning that enables computers to learn from experience and understand the world concerning a hierarchy of concepts. If machine learning is a subset of the AI, then deep learning can be called a subset of the machine learning (see Fig. 3.1).

In 1986, Rina Dechter firstly introduced the term deep learning to the machine learning field. It aims at automatically learning the data representations, which can be supervised, partially supervised or unsupervised [26], [123], [124]. Deep learning architectures [26], e.g., Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN), have been widely applied to many fields including *computer vision* [18], *speech recognition* [125], *natural language processing* [126] and *audio recognition* [127], where the results are very nice [128] and even sometimes outperform human experts [18]. As the two most popular ones, CNNs are located at the heart of the image processing, while RNNs are very powerful in the continuous sequence modeling.

[1] http://bisintek.com/science/2017/12/27/knowing-basic-artificial-intelligence/

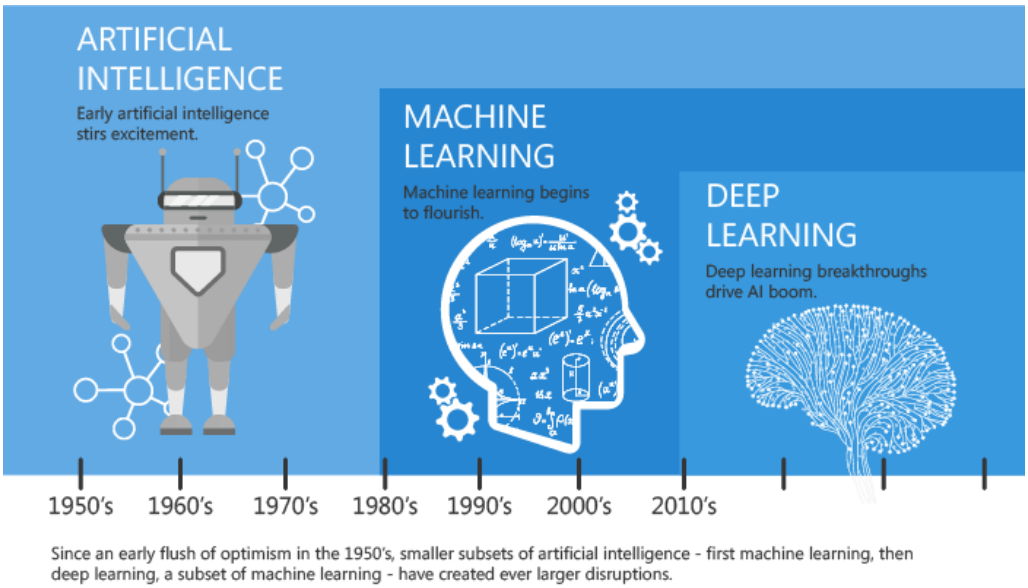


Figure 3.1: The relationship and the development of AI, machine learning and deep learning[1].

Because of the powerful capability of CNN for image feature extraction, in this thesis, we propose to use CNNs for the lips and hand feature extraction without using any artificial mark. On the other hand, it was shown in [129], [130] that the RNN based speech recognition outperforms the state of the art. Note that CS is a kind of visual speech in which the temporal context information is essential. We use the RNN based method for temporal information acquisition in the CS recognition. Since the CS data is limited for phoneme recognition, we will adapt the Long Short-Term Memory (LSTM) for CS hand position recognition in Chapter 6.

In Section 3.2, we will first introduce the Multi-Layer Perceptron (MLP), as well as some basic conceptions and properties in deep learning methods. Then, in Section 3.3 and Section 3.4, we will introduce the CNNs and RNNs, respectively.

3.2 Multi-Layer Perceptron

A Multi-Layer Perceptron (MLP) [131] is a basic feed-forward ANN, which is composed of at least three layers of nodes (see Fig. 3.2). Except in the input layer, each node is a neuron with a nonlinear activation function, which distinguishes the MLP from a linear perceptron. Formally, a MLP with one hidden layer is a function $f : \mathbb{R}^D \to \mathbb{R}^L$ defined as

$$y = f(x) = G(b^{(2)} + W^{(2)}h(x)), \tag{3.1}$$

where D is the size of input vector x, L is the size of the output vector $f(x)$ and

$$h(x) = s(b^{(1)} + W^{(1)}x) \tag{3.2}$$

constitutes the hidden layer. In (3.1) and (3.2), $b^{(1)}, b^{(2)}$ are the bias vectors, $W^{(1)}, W^{(2)}$ are the weight matrices, and G and s are the activation functions. To learn the set of parameters

$$\theta = \{W^{(2)}, b^{(2)}, W^{(1)}, b^{(1)}\}$$

in the training process, the MLP needs a supervised learning technique called the Back-Propagation (BP) [26], which will be introduced in Section 3.2.1. Typical choices for the activation function are the *sigmoid* and Hyperbolic Tangent function (tanh), which will be introduced in Section 3.2.2.

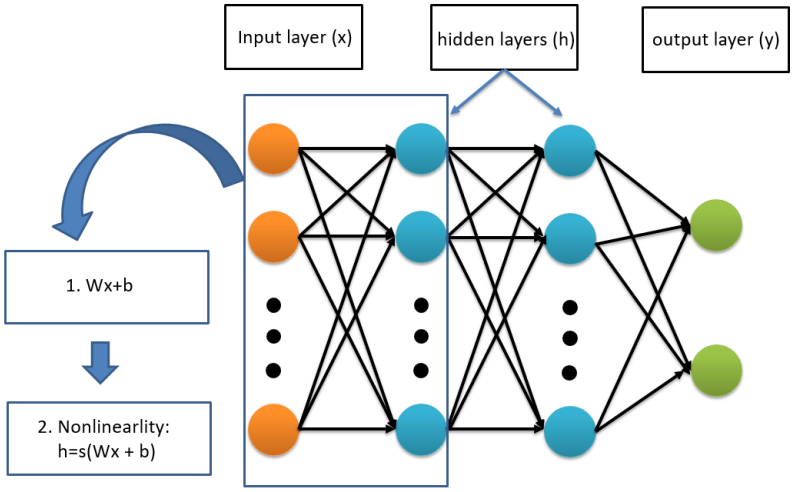


Figure 3.2: A MLP with two hidden layers.

3.2.1 Back-Propagation algorithm

For learning the parameters of an ANN, the Back-Propagation (BP) algorithm is the workhorse of the gradient descent optimization algorithm[2] [132], which aims to calculate the gradient of the loss function and then adjust the weights and biases.

Now we first introduce the error function $E(y, y')$, which is used to measure the compatibility between the ground truth y' and the output y. The standard choice is the Mean Squared Error (MSE), i.e., the square of the Euclidean distance. The error function over n training examples can be written as a sum:

$$E = \frac{1}{2} \sum_{x} \| y(x) - y'(x) \|^2 . \tag{3.3}$$

To train an ANN, an essential step is to optimize the parameters via minimizing (3.3), in which a complete iteration includes the feed-forward step, BP step and updating the parameters.

[2] This is a classical iterative algorithm to minimize a differentiable function in optimization theory.

3.2.2 Nonlinear activation function

Nonlinearity is one of the most important properties in ANNs, which are considered as universal function approximators. In other words, they can compute and learn any functions, and any process can be represented as a functional computation. Hence, we need to apply a nonlinear activation function to make the network more powerful with the ability to represent nonlinear complex arbitrary functions between inputs and outputs. The other essential feature of the activation function is the differentiability, which is needed to perform the BP optimization strategy.

Three commonly used activation functions are (1) the *sigmoid* function in the form of $\sigma(x) = 1/(1 + e^{-x})$, which is a s-shaped curve (see Fig. 3.3) between 0 and 1; (2) Hyperbolic Tangent function (tanh) in the form of $\tanh(x) = (1 - e^{-2x})/(1 + e^{-2x})$, whose output is zero centered (see Fig. 3.4), and (3) Rectified Linear units (ReLU) function with the form $\text{ReLU}(x) = \max(0, x)$ (see Fig. 3.5), which is proposed to deal with and rectify the *gradient vanishing problem* [133], and has become very popular in the past few years.

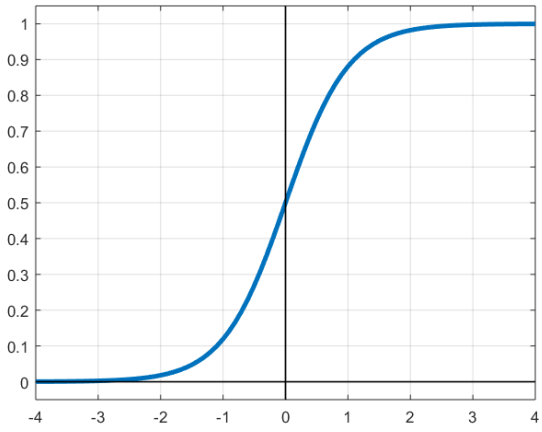


Figure 3.3: Sigmoid activation function.

3.3 Convolutional Neural Networks

Convolutional Neural Network (CNN) [26] is a special kind of feed-forward Deep Neural Network (DNN), and has been doing very nice work in image recognition tasks. It includes an input layer, output layer and multiple hidden layers (see Fig. 3.6), while a hidden layer typically includes convolutional layers and pooling layers. In this section, we give a brief description of the CNN, while more details can be referred to [26].

We take the CNN architecture in Fig. 3.6 as an example. The input of this example is a RGB image of a bird. Firstly, several filters will be convoluted for this image with a nonlinear activation function (normally it is ReLU). The principle of 2D *convolution* calculation is

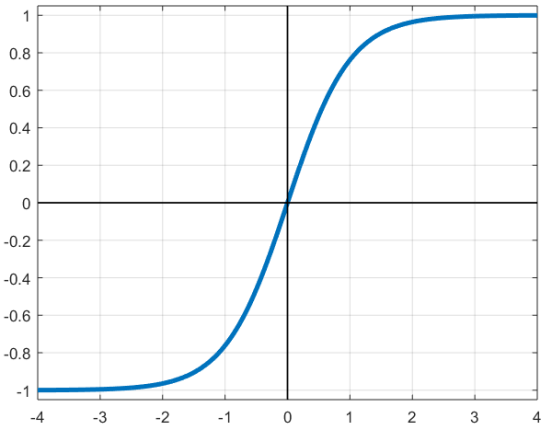


Figure 3.4: tanh activation function.

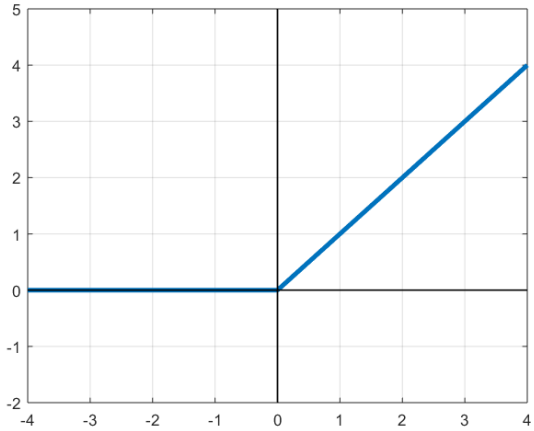


Figure 3.5: ReLU activation function.

shown in Fig. 3.7 and the *feature map*[3] is generated by applying ReLU to the output of the convolutional layer. Then, one pooling process will be conducted on the current feature map, and two Fully Connected (FC) layers will be used to combine all the elements of the feature vector. After that, a softmax layer will be applied to compute the posterior probabilities of all the classes. In this example, the posterior probabilities of being a bird, cat, sunset, dog, flower and so on will be calculated by the softmax layer, and the class label corresponding to the highest posterior probability will be the result of this classification problem.

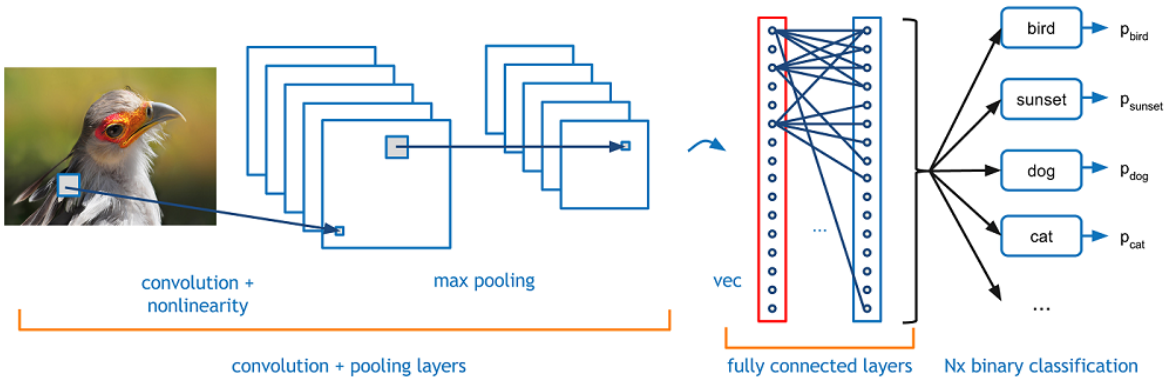


Figure 3.6: An example of CNN[4].

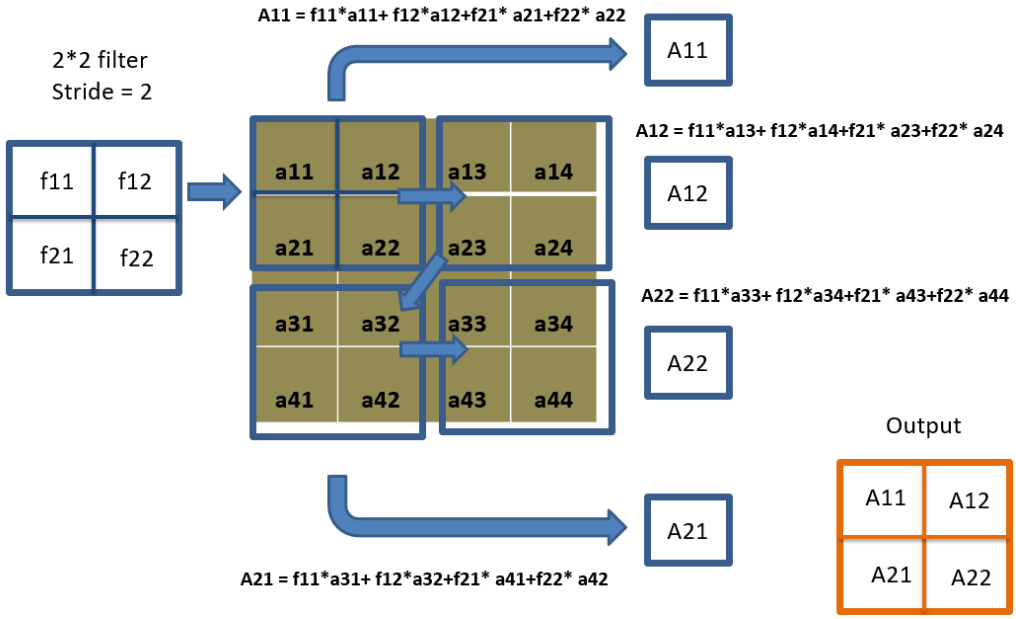


Figure 3.7: An example of 2D convolution calculation. We draw boxes with arrows to indicate how the 2×2 filter moves (with stride 2). The output is formed by applying the kernel to the corresponding region of the input image.

[3] A feature map is the output activation for a given filter.

[4] https://adeshpande3.github.io/A-Beginner%27s-Guide-To-Understanding-Convolutional-Neural-Networks/

The pooling process is used to reduce the dimensionality of each feature map while keeping the most important information. This process can be of various types, such as *Average, Max and Sum*. In the average pooling, a spatial window with a certain size (for example, 2×2) is defined, and then the average value of all the elements from the rectified feature map within that window will be calculated and used. Instead of the average element, we could also take the maximum (see Fig. 3.8) or the sum of all the elements in that window.

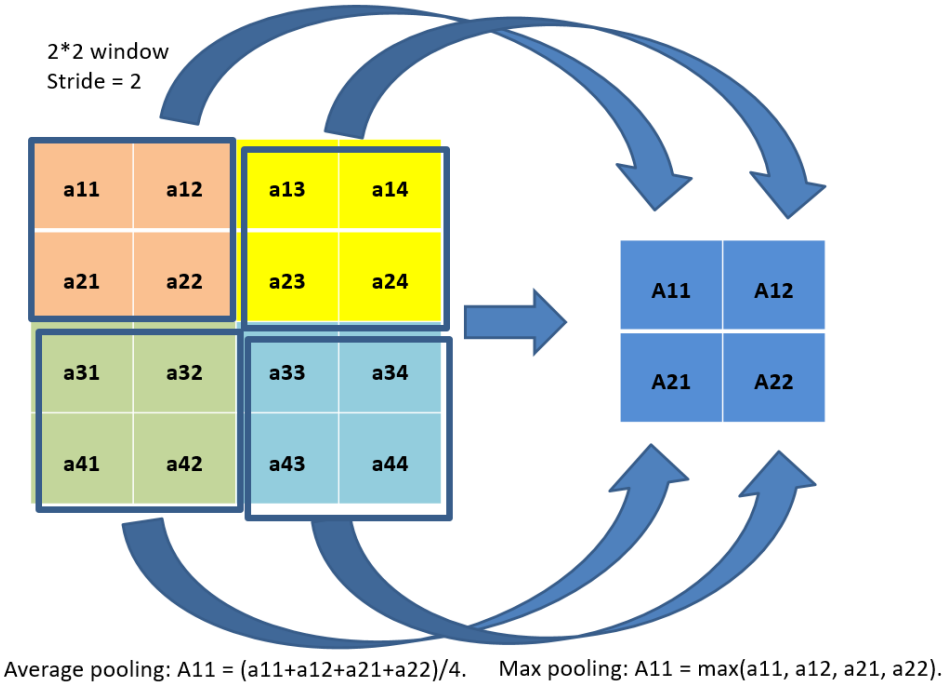


Figure 3.8: The principle of Average and Max pooling.

For the training of CNNs, the BP algorithm is used based on the *mini-batch gradient descent* [134], which is a variation of the gradient descent algorithm and splits the training dataset into small batches for calculating the errors and updating the coefficients. Moreover, the *dropout* technique [135] is often used to reduce the *over-fitting* [26] in the training process.

3.4 Recurrent Neural Networks

Unlike to the standard DNNs focusing on the static data, the Recurrent Neural Network (RNN) [136] was proposed to process the dynamic sequential data. It possess the connections between time steps (see Fig. 3.9), which allows to exhibit dynamical temporal behaviors. Some parameters are trained to save the internal memory of the temporal information. In particular, when coupled with a **large dataset**, RNN is very powerful. Recently, it has wide applications such as *unsegmented, connected handwriting recognition* [137], and *speech recognition* [138], [139]. In this section, we will give a brief description of the RNN. More details can be referred to [136], [140].

3.4.1 Vanilla Recurrent Neural Networks

The Vanilla RNN was introduced in [141], and can be described as:

$$s_t = f(Ws_{t-1} + Ux_t),$$
$$o_t = \text{softmax}(Vs_t),$$

at time t, where x_t is the input, s_t is the hidden state, o_t is the output and U, V and W are the weight matrices (see Fig. 3.9). In this model, s_t is the "memory" of the network, and can be calculated based on s_{t-1} and x_t. The usual choice of activation function f is tanh or ReLU.

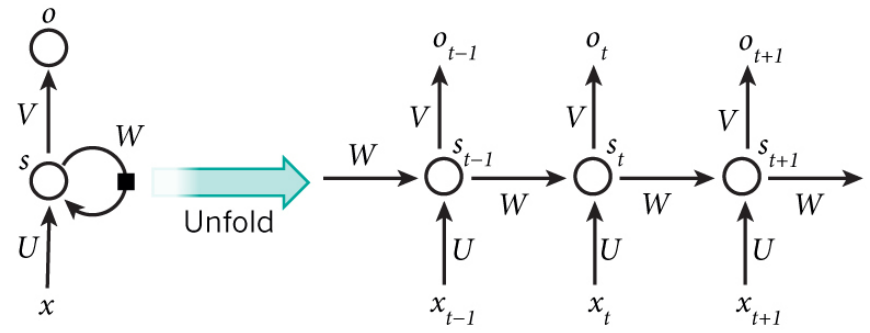


Figure 3.9: Left is the folded RNN, while the right one is the unfolded RNN with time series[5].

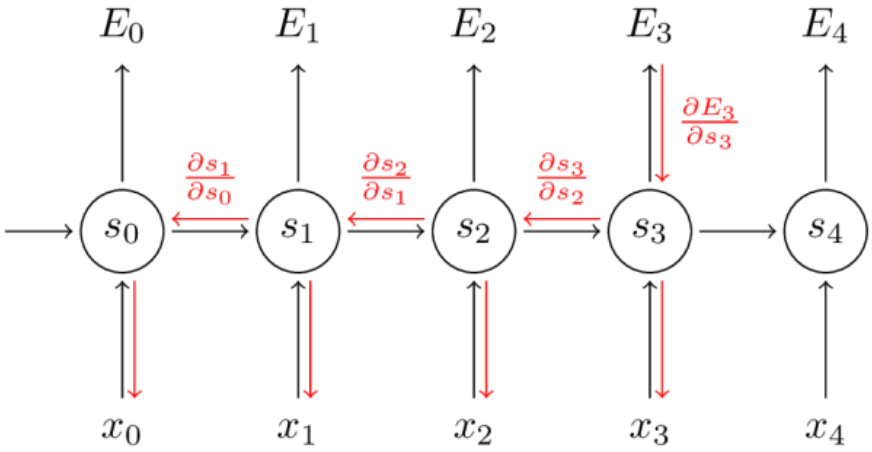


Figure 3.10: Overview of BPTT. E_t is the error function at time step t[6].

Back-Propagation Through Time (BPTT) is a gradient-based technique for training the RNN with the goal to minimize the error of the network outputs as in BP. It is an application

[5] http://www.wildml.com/2015/09/recurrent-neural-networks-tutorial-part-1-introduction-to-rnns/

[6] http://www.wildml.com/2015/10/recurrent-neural-networks-tutorial-part-3-backpropagation-through-time-and-vanishing-gradients/

of the BP training algorithm to the sequential data. This technique works by unrolling all the input time steps, and each time step has one input time step, one output time step and one copy of the network (see Fig. 3.10). At each time step, the errors are calculated and accumulated, and then the network is rolled back to update the weights.

3.4.2 Long Short-Term Memory

In oder to solve the *gradient vanishing / exploding problem* [133] in RNNs, the Long Short-Term Memory (LSTM) was proposed in [140] and set accuracy records in multiple applications domains. Since 2007, LSTM has been applied in various speech recognition tasks [124], [140], [142]–[144] with good performance.

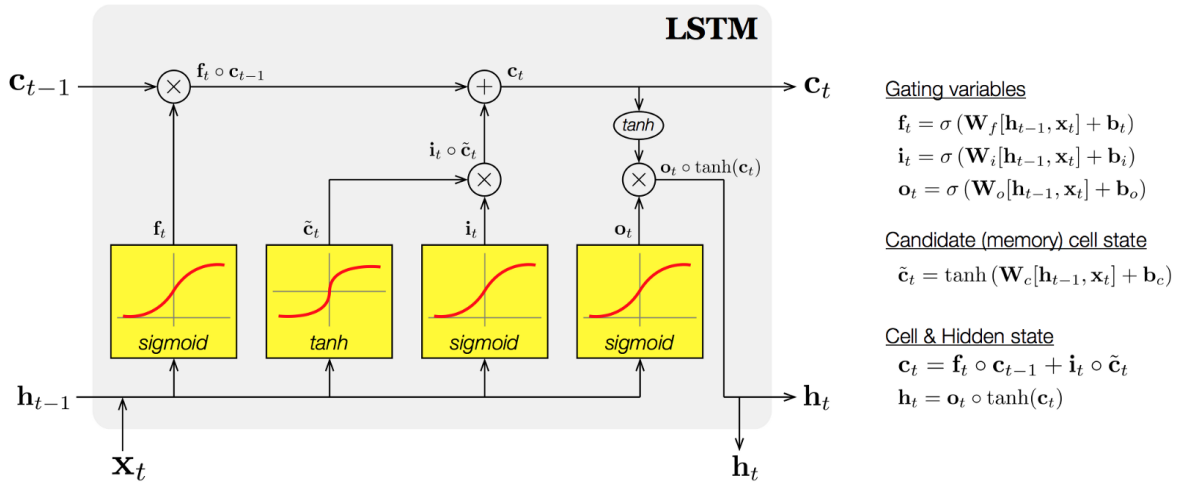


Figure 3.11: A LSTM unit with input, forget and output gates[7].

A LSTM unit is composed of a cell, an input gate i_t, an output gate o_t and a forget gate f_t (see Fig. 3.11), which are composited as follows:

$$\begin{pmatrix} i \\ f \\ o \\ \tilde{c}_t \end{pmatrix} = \begin{pmatrix} \sigma \\ \sigma \\ \sigma \\ \tanh \end{pmatrix} W \begin{pmatrix} h_{t-1} \\ x_t \end{pmatrix} + \begin{pmatrix} b_i \\ b_f \\ b_o \\ b_c \end{pmatrix},$$

$$c_t = f \odot c_{t-1} + i \odot g,$$

$$h_t = o \odot \tanh(\mathrm{c_t}),$$

where i, f, o are gate variables, c_t is a cell responsible for "remembering" the values over arbitrary time interval, h_t is the hidden state and $\tilde{c}_t$ is a candidate variable that transfers the information of input to the current cell c_t.

[7] https://codeburst.io/generating-text-using-an-lstm-network-no-libraries-2dff88a3968

Chapter 4

Multi-modal Fusion in Cued Speech

Contents

4.1 Introduction

In various domains, information about the same phenomenon can be acquired from different types of sensors at the same time. The term *modality* is often used to denote each signal source. In general, a system which is described by several modalities is called a *multi-modal system*. In a multi-modal recognition framework, the fusion problem (i.e., merging these multi-modalities to best represent this phenomenon) is very essential. Audio-visual multi-modality in speech is probably the most intuitive [145], [146], since it uses two of our most informative senses (i.e., audio and vision). In this case, audio and visual modalities are available and provide information about the speech, such as phonetic units, or word sequences.

In a multi-modal system, each modality can be used alone to train a single classifier to recognize the speech classes. However, the single modality can only carry one part of the information. Thus, we expect that combining the multi-modalities will give rise to a multi-modal classifier with superior performance [147]. The key point is how to merge these information from different modalities, especially when they are not synchronized. As we mentioned in Section 1.2.3, the automatic CS recognition is a typical multi-modal recognition problem which has the lips and hand modalities. Therefore, in this thesis, the multi-modal fusion problem is one of the most important challenges. In particular, the multi-modalities (i.e., lips, hand shapes and positions) in CS are asynchronous. This makes the fusion problem in CS recognition more complicated. In the state of the art [10], several fusion strategies are used (see Section 1.2.3) but without taking into account the asynchrony between the lips and hand modalities in CS.

It was reported in [148] that the contextual information, which enables some selective weighting of one or the other modality during the fusion process, is more and more used in AVSR systems and it turns out to be an essential ingredient for improving the robustness. In the speech recognition task, a *context-dependent modeling* [149]–[151] is advantageous to take into account the complexity of the co-articulations and variabilities in speech. This is because of that the dependent context[1] can provide extra information related to the target. In the prior studies [10], [11], context-independent modeling has been applied to the CS recognition[2]. In this thesis, we will incorporate the context-dependent modeling into our CS recognition framework. Besides, the *language model* [152] is also able to improve the robustness of the recognition system. The reason is that the ambiguities (e.g., semantic, syntactic or lexical errors) can be easier to resolve when the information from language model is incorporated.

Moreover, recently, with the popular trend of DNNs, an *attention mechanism* operating across different modalities combined with the *encoder-decoder* is proposed to deal with the multi-modal fusion issue. It uses the attention to fuse the modalities in a context-dependent way. Indeed, we have tried to investigate the LSTM with an attention mechanism to capture the context information and learn the relationship between lips and hand streams automatically. Due to the limited dataset, we finally did not obtain a satisfied phoneme recognition performance in the test set.

In this chapter, we will introduce the HMM, classical fusion methods, as well as the fusion methods in deep learning. Then, a triphone context-dependent modeling based on HMM will be presented. These methods will be applied to the CS recognition in Chapter 8.

4.2 Hidden Markov Model

Before further introduction, we present a brief description of HMM, which has been located in the core position of speech recognition [25]. More details about HMM can be referred to [14]. In this thesis, except that the context-dependent modeling (see in Section 4.4) is based on HMM, it will be also used as the phonetic decoder of CS recognition in Chapter 8.

A diagram of the Hidden Markov Model (HMM) is shown in Fig. 4.1, where we can see that each state has a transition probability to the next state, as well as an emission probability distribution of the possible observation. We now recall three fundamental problems [25] in HMM. (1) The first one is the evaluation problem, i.e., the way to compute the probability of observation given the model parameters. Using a traditional probability calculation based on the Bayesian formula, we have an exponential exploding computation complexity of this probability. Therefore, the *forward and backward* algorithms are used to reduce the computation complexity. (2) The second one is the state decoding issue, i.e., the way to choose a corresponding state sequence which best explains the observations. The solution is the well-

[1] The context is defined as the information about the target observation of audio-visual displays that happen before or after this observation.

[2] It was assumed in [10], [11] that GMMs with a larger number of Gaussian components combined with the first and second derivatives of features are used to take into account a short-term context.

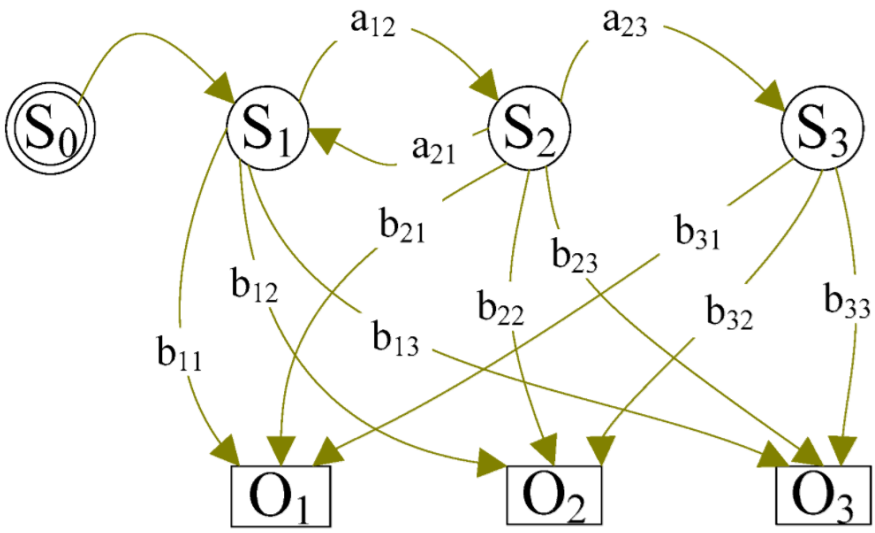


Figure 4.1: A diagram of the HMM. O_i is the observation and S_i is the hidden state. a_{ij} is the transition probability from S_i to S_j, and b_{ij} is the emission probability of O_j at S_i (from [153]).

known *viterbi* algorithm. (3) The third problem is the parameter estimation which aims to optimize the model parameters to best describe the given observation sequence. The solution is the Expectation Maximization (EM) algorithm.

4.3 Fusion techniques for multi-modal speech recognition

As introduced in Section 4.1, the fusion of different modalities is of great significance in the multi-modal recognition problem. In this section, we will introduce some well-known classical fusion strategies, as well as some novel fusion methods based on deep learning.

4.3.1 Classical fusion methods

A classification of the fusion models for AVSR can be seen in [154], which offers an interesting framework to discuss the combination of the lips and hand movements in CS. In [148], [154], four methods are proposed by Schwartz et al. to solve the multi-modal fusion problem. (1) In the Direct Identification (DI) model, the components are concatenated in the same vector, which is then considered in an early classification phase. (2) In the Dominant Recoding (DR) model, one of the modalities is recoded in the dominant one, for instance, the audio modality in AVSR. (3) The transformation of the data flow of each modality into a third common space gives rise to the Motor Recoding (MR) model. (4) In the Separated Identification (SI) model, each modality is first processed with a decision, and the final decision is a combination of all the decisions in a later phase.

Potamianos et al. [147] classified these four basic models into two categories: the fusion of features and fusion of decisions. In the first category, a classifier is applied to a vector that is a concatenation of the audio and visual data or their appropriate transformations. The DI, DR and MR models belong to this category. In the second category, the methods are based on the combination of decisions resulting from the processing of the audio and visual modalities. The SI model belongs to this category.

We now discuss the applications of these models to CS. The MR fusion model refers to a decoding process in the articulatory space. Let us take the vowel recognition in CS as an example. In the audio speech, vowels are identified by their vocal tract shapes, defined by the mandibular open-close, the opening at the lips and the position of tongue inside the vocal tract. In the audio-visual fusion of speech, the tongue which is not always completely visible can be directly derived from the two first formants when the oral vowels are considered. In CS, the hand defines several possibilities for the vocal tract shape. Only one of these possibilities is coherent with the lips. This final vocal tract can thus be submitted to a classification phase. However, this approach needs additional data in the articulatory domain as a dictionary of vocal tract shapes or an articulatory model, and thus it has not been tested. In the DR model, one of the modalities is dominant and the other predicts specific "cues" in the dominant space. In audio-visual speech, the audio is the dominant modality since it carries all the speech information. The lips shapes of the non-dominant modality can predict, for example, the specific spectra in the acoustic domain on the basis of existing correlations between lips and spectral parameters. In CS, neither of the components carry the complete information on speech without ambiguities. Thus, there is no evident dominant component. Moreover, the lips and hand components are complementary to allow complete perception of speech. Thus, no high correlation is expected. These two reasons make the DR model difficult to apply for a successful classification.

In the DI model, the features of all the modalities are concatenated in a vector or matrix, and a classifier is directly applied to these merged data or appropriate transformations. We could apply it to CS, but the resynchronization of lips and hand is necessary. For SI, the *Master-Slave* SI plays an important role. Heracleous et al. [10] implemented this fusion method in the way of *lips first and then hand* for CS phoneme recognition. A vowel-independent and a consonant-independent GMMs were first trained using lips shape parameters only. Then vowel or consonant recognition was realized using the feature vectors corresponding to the vowel or consonant modeling. However, since the hand precedes lips in CS coding, the *hand first and then lips* way [41] give better performance than the *lips first and then hand* fusion.

HMM based fusion techniques have been widely considered in the literature for automatic AVSR [95], [98], [100], [101], [107], [155], [156]. Apart from the above models, Potamianos et al. [147] proposed several HMM based fusion techniques: *early* fusion (fusion of features), *middle* (intermediate) fusion and *late* (decision) fusion. In fact, the early and middle fusions can be seen as DI, and the late fusion is SI. In the early fusion, the audio and visual features are concatenated as one vector, and followed by a transformation to reduce the dimension of the obtained vector [101]. The resulting feature vector is fed to a one stream HMM [157]. In

the middle fusion, multi-stream features are fed to a multi-stream HMM with certain weights, and the output observation likelihood is the product of these signal streams weighted by their proportions [158]. In the late fusion, the audio and visual features are modeled by two different HMMs, and one technique will then be used to combine the output likelihoods of them [101].

In this thesis, we focus on the DI and DI-2 fusion methods in [148], as well as the early and middle fusion methods in [147]. Different from [147], [154], we use the terminologies of *feature-level* fusion, *model-level* fusion and *decision-level* fusion. In [10], the feature-level and model-level fusions combined with HMM-GMMs are used to deal with the multi-modal fusion in CS recognition. In this thesis, we implement the feature-level and model-level fusions in HMM-GMMs decoding frameworks. More precisely, in case of feature-level fusion, one stream HMM will be used and in case of the model-level fusion, multi-stream HMM will be used.

According to [148], we could expect that the feature-level fusion yields better performance than other types, as only this model is able to exploit the joint time variations of the lips and hand streams. However, it still has two problems. The first one is how to weight the inputs in the case of context-dependent fusion. The second one, a major problem, comes from the natural asynchrony between the lips and hand streams. For the first problem, we use a *cross-validation* procedure to choose the best weights for lips and hand experimentally (see Section 8.3.1). For the second one, we propose a novel resynchronization procedure, which first pre-aligns the features from different streams and then merges them (see Section 8.2.3).

4.3.2 Fusion methods based on deep learning

Except for the classical fusion methods in Section 4.3.1, the deep learning based system for the multi-modal fusion becomes more and more popular. In general, it is an end-to-end DNN including the feature extraction, fusion and recognition. In this system, a popular approach to learn the relationships between different modalities is the *attention mechanism*, which has a long history in image recognition [159], [160]. Recently, it has been widely used to improve the ability of neural networks to derive good features, with applications in *speech recognition* [161], *video description* [162]–[164], *image captioning* [165] and *machine translation* [166], [167]. Now we give a brief description of it, and more details can be referred to [167].

Attention mechanism is often incorporated in the *encoder-decoder* [167], which is a neural network based framework to handle the mapping between input and output data, and was recently realized for the machine translation task [168], [169]. The attention mechanism first uses an encoder to process the original sequential data and return a representation feature vector, which incorporates the context information of the sequential data. Then, it scores each context vector to the current hidden state of the decoder.

We could think about applying the encoder-decoder to CS recognition, as this problem can be seen as a multi-modal sequence-to-sequence learning problem. In this process, the CS image sequence is first encoded to a fixed-dimensional observation vector. Then the output sequence, i.e., phoneme or word sequence, is generated from the input vector. In CS recognition, each modality is modeled by an encoder-decoder with an attention, and the weights will

be processed by a multi-modal fusion step. The encoder is modeled as a CNN for lips and hand shapes (ANN for hand positions), while the decoder is modeled as a LSTM network for each modality.

Another recent popular fusion method is the Deep Canonical Correlation Analysis (DCCA) [170], which is to learn the complex nonlinear transformations between two data streams such that the final representations are highly linearly correlated. The Canonical Correlation Analysis (CCA) [171], [172] and Kernel Canonical Correlation Analysis (KCCA) [173] have also been widely used for multi-modal fusion. In fact, the KCCA can be seen as an extension of the CCA since it is correlated with the nonlinear projections. Their other applications include the *natural language processing* [174]–[176], *speech processing* [177], [178], *computer vision* [179] and *multi-modal signal processing*[180], [181].

In [170], it was reported that CCA and KCCA have some problems when tackling with the multi-modal fusion, which is not linearly correlated, while the DCCA does not suffer from these drawbacks. In [182], based on DCCA, CNN-DCCA was proposed for the hearing loss people in AVSR. The audio signal was first converted to the *mel-frequency cepstrum* (Mel) Map, and then CNNs are applied to extract the features on the Mel map. Meanwhile, lips landmarks are detected by Constrained Local Model (CLM) [183], [184] and interpolated by a spline function to form a lips contour. CNNs are then applied to the image of lips contour for feature extraction. Finally, the DCCA is applied to the two streams feature fusion. It is shown that the DCCA can capture the time delay between two streams, and the evaluation confirms that using DCCA outperforms the conventional fusion methods. This study provides an approach to take the advantage of DCCA in our CS case, which will be the future work.

4.4 Context-dependent modeling

As introduced in Section 4.1, the features of lips and hand movements in CS are asynchronous. Since this phenomenon is often correlated with the phonetic context in the word or sentence, the *context-dependent modeling* could possibly help to make the recognition system more robust. This modeling has been widely used in the continuous Audio Speech Recognition (ASR) due to the co-articulation and anticipation problem in nature human speech. For example, in English words, the vowel [eh] has an evident feature difference in the frequency fields (see Fig. 4.2). It has been shown in [148], [185]–[187] that the context-dependent modeling can improve the accuracy of ASR. Similarly, for the CS recognition, the articulatory features are also dependent on the context (e.g., hand rotation, anticipation or the asynchrony). For example, for a given hand position, the hand movement trajectory depends on the corresponding context information (i.e., hand position) of its neighbors, and three modalities may indicate different phonetic targets at the same instant. Therefore, we could expect that the context-dependent modeling could help the continuous CS recognition to have a better performance. It should be noted that the this modeling is not a fusion method but a processing to overcome the co-articulation and variabilities in CS by introducing the contextual information.

[3] https://blog.csdn.net/quheDiegooo/article/details/60873999

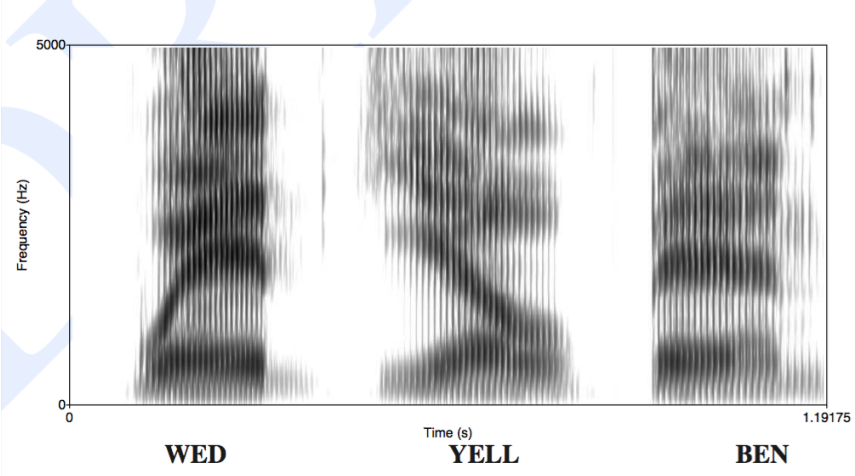


Figure 4.2: Examples of the co-articulation in the speech[3].

In this thesis, we will train a triphone context-dependent modeling, which adds the left and right context information of each phoneme based on our CS data. We follow the steps as introduced in [150]. The first step is to build a HMM system with the tied state to reduce the larger number of HMM parameters in the training. The second step is to build a decision binary tree to search the possible neighbor context (the left and right) for each state. It was reported in [150] that the state tying HMM with decision tree clustering strategy is able to reduce about 10-20% of the state number in HMM without any decrease in the recognition performance.

Part II

Chapter 5

Automatic Inner Lips Parameter Extraction

Contents

5.1 Introduction

Lips feature extraction is an active research topic since lips (especially inner lips) hold significant information on speech production, and plays an important role in AVSR. It is also vital in our CS recognition work, since lips information is an indispensable part in CS. We have reviewed the state of the art on the lips tracking in both CS and other fields in Section 1.2.1. In particular, the method in [10], [11] for lips feature extraction needs to paint colors on the speaker's lips. In this chapter, instead of using the artificial color marks, we propose two new approaches to extract the inner lips parameters (A and B). In fact, these approaches are able to extract the inner lips contour not only in our CS case where the hands may occlude lips but also non-CS case such as lip reading.

The first approach is based on the Constrained Local Neural Fields (CLNF), which was introduced by Baltusaitis et al. [188] in 2013. The experiments in [188] showed that CLNF is more accurate than many previous models, including the Active Shape Model (ASM) [189], Active Appearance Model (AAM) [52] and Constrained Local Model (CLM) [183], [184]. The reason is that CLNF is much more robust to occlusions, rotated faces and different lighting conditions. However, when directly applied to our CS data, CLNF failed in about 41.4% of

all the cases. In fact, the lower inner lips landmarks given by CLNF are often placed much higher than the ground truth. Besides, the two endpoints of inner lips are far away from the ground truth, especially when the lips shape is round. To improve its performance, we propose a Hybrid Dynamic Correlation Template Method (HD-CTM) to correct the CLNF errors of B parameter in inner lips tracking [15] and develop a *periodic spline interpolation* method for A parameter correction. In this thesis, we call these two methods for B and A parameters both as Modified CLNF [16], [190]. These methods were evaluated with three corpora: *MD, DB* and *ChS*. We remark that another possibility is to adjust CLNF by retraining only lips images. However, it needs a large training set, which is not available yet in our CS case.

The second approach is an *adaptive ellipse model* [17], which does not depend on any additional lips landmark. It adopts an ellipse to approach the inner lips area until it finds its optimal position and shape. In this approach, the color based image processing is first applied to delimit preliminary inner lips area. A single discontinuity elimination combined with interrupted region filling is used to obtain a binary inner lips image as complete as possible. After the preprocessing steps, the optimal adaptive ellipse is determined to match the inner lips, finally giving A and B parameters.

In Section 5.2, we will detailedly introduce the first approach, CLNF [188] based inner lips parameter extraction methods, as well as the evaluation results. Then, in Section 5.3, the second approach, adaptive ellipse model, and the evaluation will be presented. The advantages and disadvantages of these two approaches will be discussed in Section 5.4.

5.2 CLNF based inner lips parameter extraction

In this section, we will present our approach to extract the inner lips parameters. It is based on a recent new approach, CLNF, in computer vision, a very powerful tool in tracking the facial feature landmarks. However, a direct application of CLNF to our CS data gives some errors. Therefore, we propose an efficient post-processing method named Hybrid Dynamic Correlation Template Method (HD-CTM) which allows correcting the errors concerning B parameter. Also, we propose a *periodical spline interpolation method* to correct the errors concerning A parameter. After introducing the CLNF model in Section 5.2.1, we describe the HD-CTM for correcting B parameter in Section 5.2.2, and the periodical spline interpolation method for A parameter in Section 5.2.2.7. The evaluation results will be presented in Section 5.2.3.

5.2.1 Constrained Local Neural Fields

The Constrained Local Model (CLM) [183], [184] includes two processes: model-building process and CLM searching process, while the model-building process includes Point Distribution Model (PDM) and patch model. CLM uses patch expert which is fitted to the current feature points to generate a template. Then, a fitting approach is used in the searching process. More precisely, the feature templates are matched to the image using an efficient shape con-

strained search of the template response surfaces. The most popular patch expert and fitting approach in CLM [184] are linear Support Vector Machine (SVM) and Regularized Landmark Mean-Shift (RLMS), respectively.

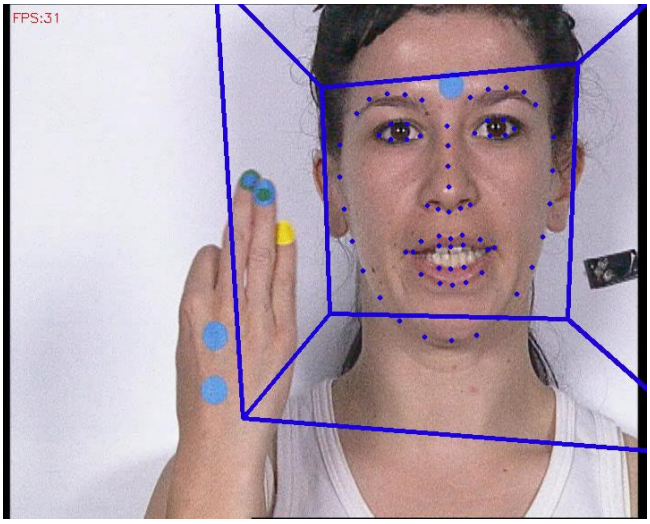


Figure 5.1: Application of CLNF to our database *MD*. Blue carders show the detected face ROI. 68 blue landmarks are located to describe the facial contour and 20 of them are used to describe the lips contour. Only 8 points are used to indicate the inner lips contour.

The Constrained Local Neural Fields (CLNF) is a new instance of the CLM in aspects of the patch expert and fitting approach, as shown in Fig. 5.2. It includes three main parts which are a PDM, patch expert Local Neural Field (LNF) and fitting approach Non-Uniform RLMS. In CLNF, 68 landmarks are placed on each face to describe the facial features, including eight landmarks to describe the inner lips features (see Fig. 5.1). The LNF patch expert is able to capture more complex information and incorporate spatial relationships between neighbor pixels. The Non-Uniform RLMS is the fitting approach which takes into account the reliabilities of the patches by adding weights. More details about CLNF can be referred to [188].

5.2.2 Methodology and Experiment details

In this section, we first introduce our database and present the performance of CLNF to extract the lips parameters on this database. The errors of CLNF when directly applied to our data will be analyzed. In order to correct these errors, we then develop the post-processing approach using HD-CTM for correcting B parameter, and propose the periodic spline interpolation for correcting A parameter.

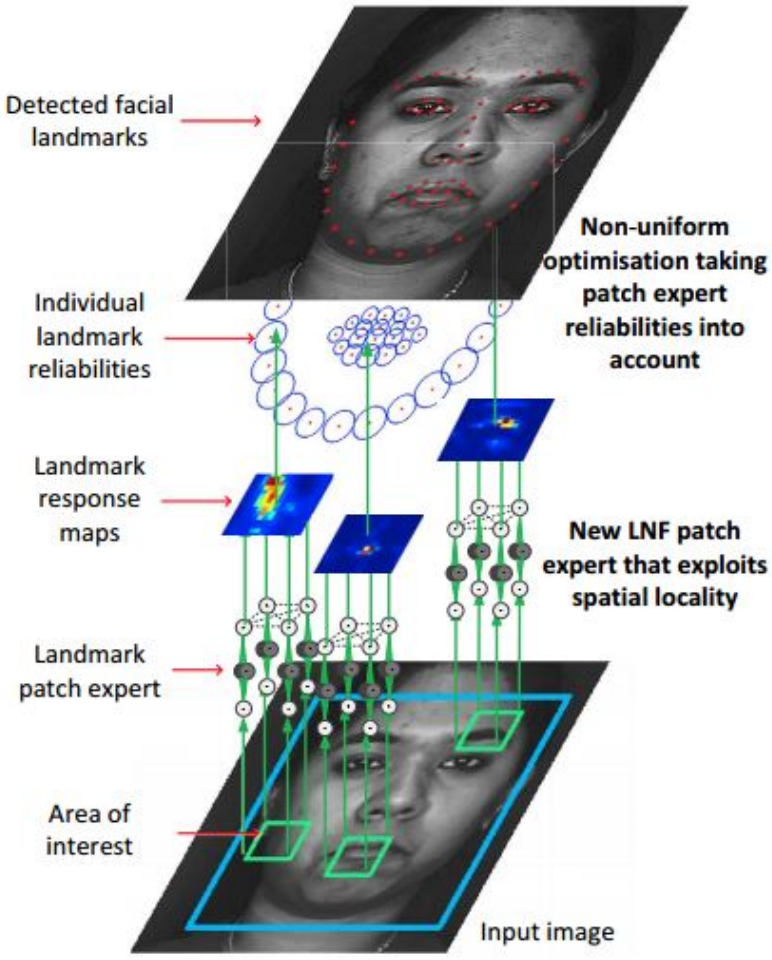


Figure 5.2: Overview of CLNF model. Compared with CLM, two novelties are the new LNF patch and the Non-uniform RLMS optimization (from [188]).

5.2.2.1 Database

The database contains videos of 50 French words made of digits and daily used words. Each word is uttered 10 times by 3 French subjects (*MD*, *DB* and *ChS*): one female CS speaker and two male normal speakers. Words and vowels are annotated with Praat based on speech sound signal. We use the first repetition of three speakers containing 4800 images corresponding to all types of lips shape to evaluate the B parameter. In addition, 3184 images of vowels are extracted from the data of the female CS subject. To evaluate the performance of CLNF model and our proposed model, the ground truth inner lips contour is extracted manually by an expert placing several landmarks on lips. Three visemes (lips shape) are often considered which correspond to the 13 French vowels shown in Table 2.3. The first viseme corresponds to the open vowels, the second viseme corresponds to the open round vowels and the third viseme corresponds to the small-open round vowels.

5.2.2.2 Performance of CLNF for lips extraction on our data

The CLNF is first directly applied to all video of words in the database introduced in Section 5.2.2.1. Among the 68 facial landmarks given by CLNF (see Fig. 5.1), eight landmarks (six of them for inner lips and two for endpoints) are used to describe the inner lips contour (see Fig. 5.1). Based on eight landmarks given by CLNF, we generate the whole inner lips contour using the interpolation. A linear interpolation [191] is used for upper inner lips contour and a spline interpolation [191] is applied to lower inner lips, because the upper inner lips contour is less bending than the lower inner lips contour. The following experiment results show that it does not have much influence on the precision of the A and B parameters. One example of the excellent performance of CLNF is shown in Fig. 5.4(a), where the green curve shows an interpolated inner lips. The A and B parameters are then calculated from the inner lips contour using the classic method in [192] (see Section 2.3.3.1).

The applications of CLNF to different CS subjects are shown in Fig. 5.3. Recall that the main advantage of CLNF is its robustness to the variable lighting conditions, presence of occlusion and head movements. We can see that in most of the cases, CLNF gives a correct estimation of the facial landmarks, but with some errors on the lips. The landmarks of the lower lips are often poorly placed in a much higher position while almost no error is presented for upper lips (see Fig. 5.4(b)). It causes wrong B parameter. This phenomenon can be explained by the fact that the CLNF is based on a dictionary of training images. If the lips shape and appearance are not properly taken into account during the training phase, it may lack the template during the optimization step. In fact, the lower inner lips detection is challenging since lips area is often very complex (tongue and teeth may be visible), and lighting condition is variable.

On the other hand, the two endpoints of inner lips may be poorly placed (see Fig. 5.4(c)). It causes mistaken A parameter. Indeed, from a "geometrical" perspective, two endpoints of inner lips are not false because in this case, the inner contour can be these two endpoints. However, in an articulatory-acoustic point of view, these two points do not define the proper

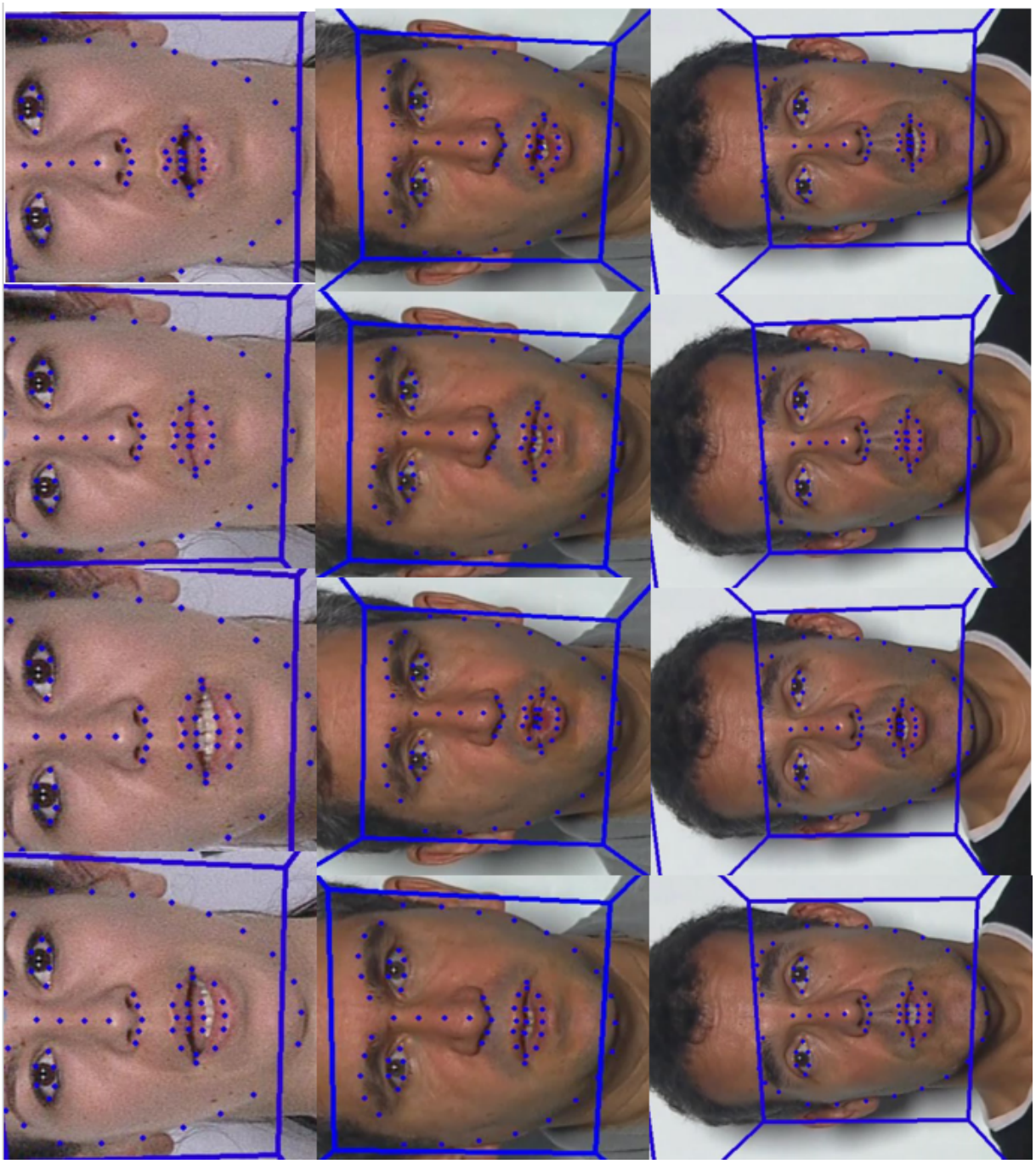


Figure 5.3: Examples of application of CLNF to three speakers in our database. Some good and bad performed cases are shown. For example, the first one is badly performed while the second one is well performed.

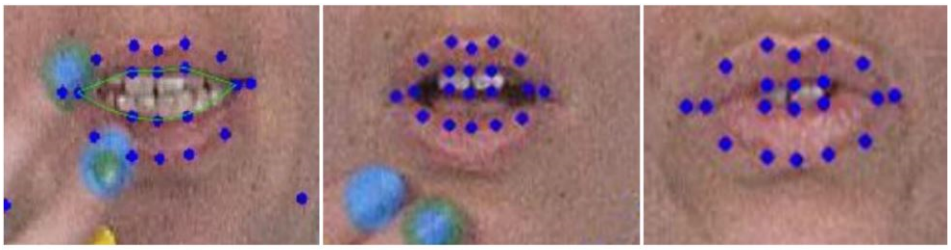
A parameter of inner lips.

In order to evaluate the performance of CLNF, a comparison of B parameter between CLNF and the ground truth is carried out. A statistical distribution of these errors is given in Fig. 5.5, where the error E_{CLNF} is defined as:

$$E_{\text{CLNF}} = B_{\text{CLNF}} - B_{\text{ground truth}}.$$

We first observe that most of the errors (i.e., $E_{\text{CLNF}} > 2$) appear negative. It means that the mistaken CLNF lower inner point is often placed above the ground truth points. Since there is a large proportion of B parameter errors and inner lips height plays a very crucial role in speech production, we have to pay particular attention to B parameter correction. It is shown in Fig. 5.5 that CLNF only obtains about 58.6% accuracy on average (75.2% for speaker *MD*, 52.2% for speaker *DB* and 48.5% for speaker *ChS*).

To see the CLNF performance concerning A parameter, three visemes are plotted using the first repetition of the CS speaker *MD* in A and B parameters plane (see Fig. 5.6). The Gaussian ellipses present the distribution of each vowel. Fig. 5.6(a) corresponds to CLNF landmarks and Fig. 5.6(b) corresponds to the ground truth. Recall that the error of B parameter is also included in Fig. 5.6(a). Compared with the ground truth B parameter, the vertical direction is much dispersed when using the B parameters estimated by the CLNF (for example, the blue ellipse in these two figures). Due to the error of A parameter of CLNF, we observe that the third viseme is considerably shifted to the right compared with the ground truth distribution.



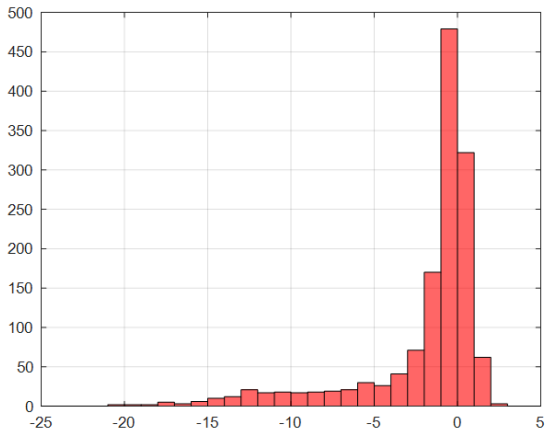
(a) (b) (c)

Figure 5.4: Performance of CLNF on our data. Examples of twenty CLNF landmarks placed in the full lips region. Eight points describe inner lips contour. (a) Good inner lips contour of CLNF even with hand occlusion. Green curve is the inner lips contour obtained by interpolation. (b) Mistaken CLNF landmarks in case of B parameter. (c) Mistaken CLNF: two end landmarks for round inner lips (mistaken A parameter).

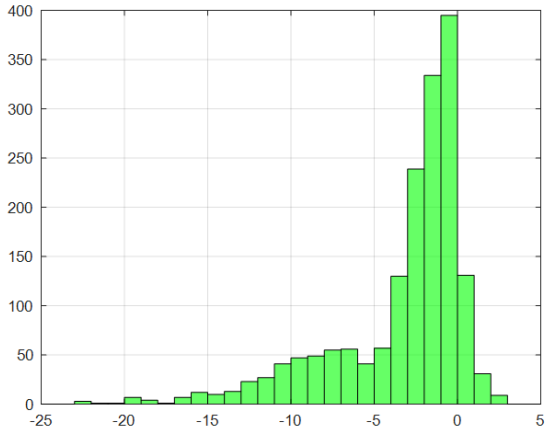
5.2.2.3 Parameter B correction based on hybrid dynamic correlation template model

We now introduce the correction of B parameter in CLNF using our proposed HD-CTM. The principle of this method will be first presented.

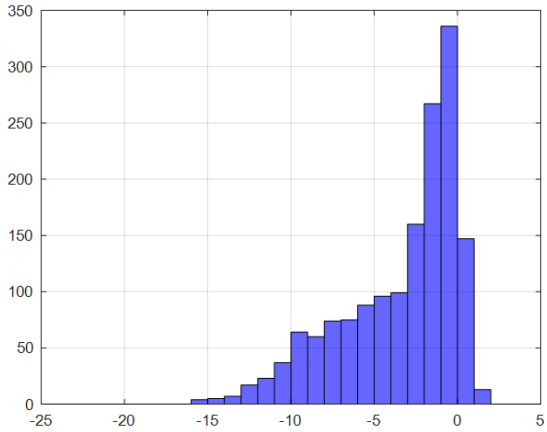
This method is based on the luminance variation along the middle CLNF landmarks of lips. A suitable spline smoothing is first applied to this luminance variation as well as the first



(a)

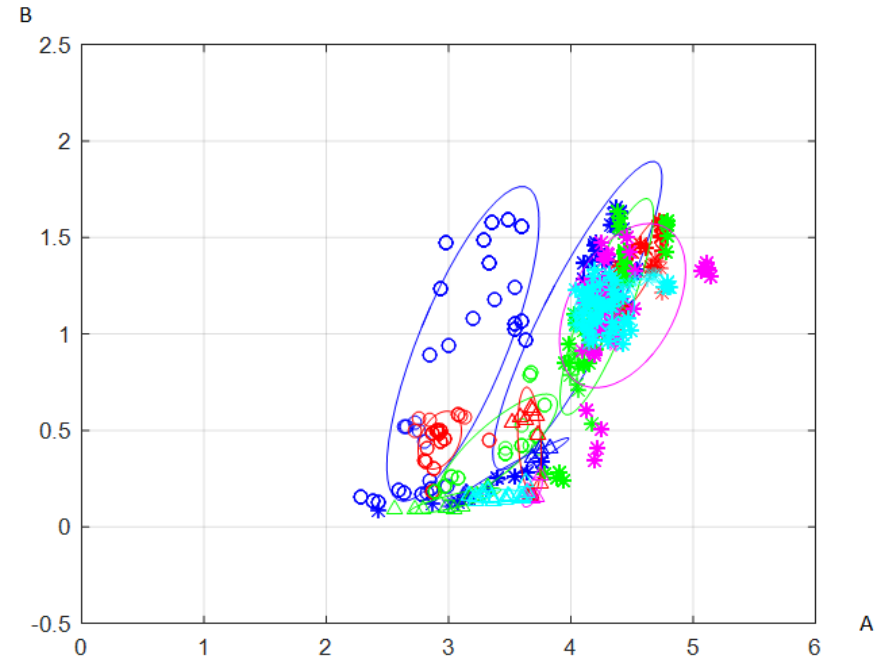


(b)

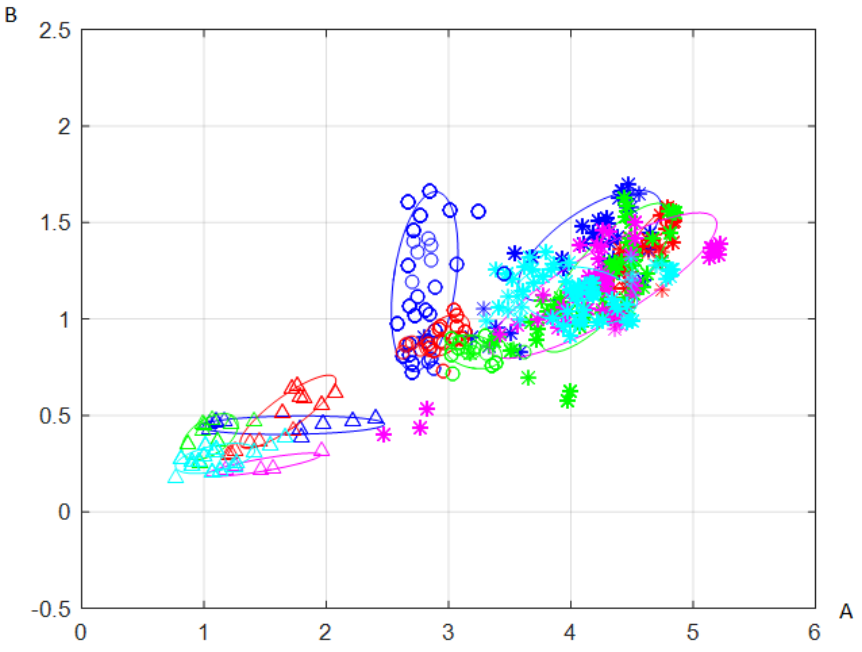


(c)

Figure 5.5: Red, green and blue histograms represent the CLNF errors of three speakers respectively. Abscissa is the error values in pixel (when larger than two pixels, they are considered as mistakes) and y-axis is the frequency of these errors.



(a)



(b)

Figure 5.6: Vowel viseme distributions of (a) CLNF and (b) ground truth. The abscissa is the A parameter (in cm), and y-axis is the B parameter (in cm). Stars correspond to the first viseme. Circles correspond to the second viseme. Triangles correspond to the third viseme. The color order is blue, red, green, magenta and cyan for each vowel of viseme. They correspond to the vowel visemes (V1-V3) in Table 2.3.

derivative curve. The smoothing degree is carefully controlled so that the noise can be removed without losing much useful information. A smoothing coefficient of $p = 0.01$ [193] is used for a good compromise. We may expect to determine the inner lips position by searching the local limit point in the first derivative of the smoothed luminance variation curve. However, this is not always feasible since there are many local limit points (see Fig. 5.7) without a searching interval. Even with a searching interval, the local limit position may be fuzzy or uncertain. Moreover, it is sensitive to the noise and unable to guarantee coherent results for adjacent sequential images. To overcome these problems, we propose the HD-CTM method for the search of limit point .

Even using HD-CTM, inner lower lips detection may still remain challenging since this area may be fuzzy, and several different cases have to be considered. For example, the luminance variation from teeth to lower lips is not the same with that from tongue to lower lips. We call the straight line across the middle inner lips landmark as *middle inner lips line* (see the blue line in left figures of Fig. 5.7). It can be seen in Fig. 5.7(a) that the luminance decreasement from teeth to lower lips corresponds to a local minimum point. However, in Fig. 5.7(b) and Fig. 5.7(c), it becomes complicated to find one particular local minimal point corresponding to the lower inner lips boundary.

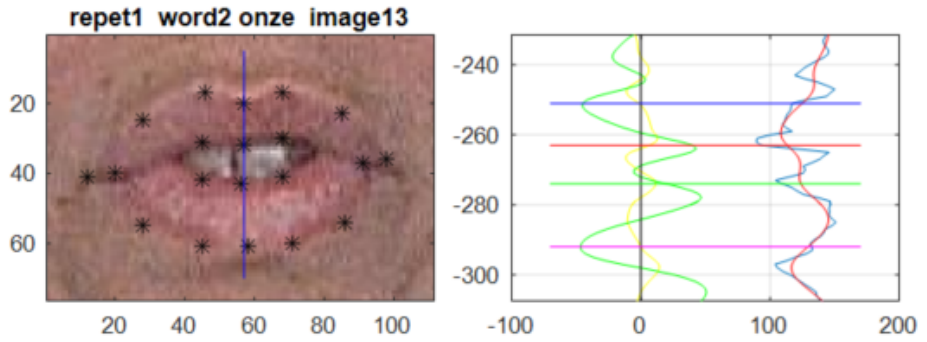
By contrast, in the region of outer lower lips, the luminance variation is less complicated than in inner lower lips region. In fact, the middle landmark (in the vertical sense) of the lower lips is more enlightened and corresponds to a high luminance variation. When the luminance goes down, it decreases rapidly as the color of the chin (the lower part of lower lips) is darker. The first derivative of the luminance curve consequently shows a significant 'V' shape corresponding to the luminance variation. Therefore, as a proposed solution, the HD-CTM is first applied to detect outer lower lips position. Then the inner lower lips position is estimated by subtracting outer lower lips position from the Validated Lower Lips Thickness (VLLT), which will be illustrated in Fig. 5.8. In all, the determination of the inner lower lips is achieved by the combination of HD-CTM to determine the outer lower lips position and the subtraction of VLLT from the outer lower lips.

5.2.2.4 Determination of the lower outer lips position

As discussed above, we first try to determine the lower outer lips position, which contains the following two steps.

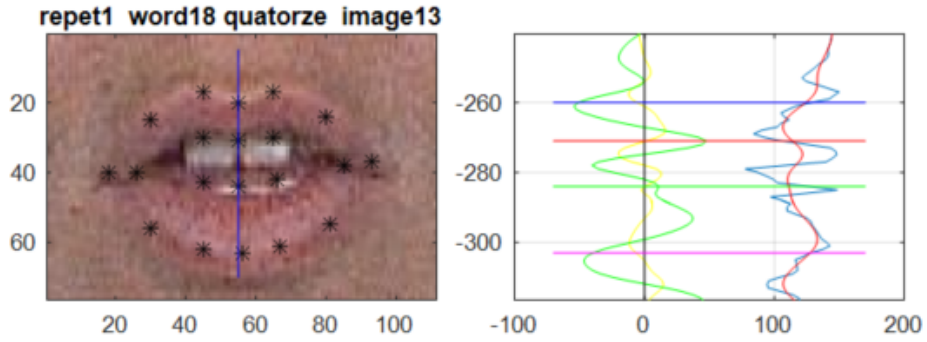
(1) Definition of the template.

Instead of directly finding a minimal local value in the first derivative curve, which is often difficult due to its great sensitivity to noise, a hybrid dynamic template corresponding to a typical derivative variation in the region around outer lower lips position is first established. The template is obtained by training some derivative curves reflecting different lips shapes except for closed lips. Template length L_M is a key parameter since a very small template length makes results sensitive to noise while a very large length reduces the detection precision. If it is badly chosen, it will not be sufficiently pertinent to indicate the 'V' shape of the



(a)

(b)



(c)

Figure 5.7: The left figures show lips ROI with CLNF lips landmarks (20 black stars). The blue line is the middle inner lips line, and all the curves in the right figure are plotted along this blue line. In the right figures, the blue curve is the original luminance variation. The red curve is the smoothed luminance. The green curve is the first derivative of the smoothed luminance. Four straight lines with blue, red, green and magenta color correspond to four middle CLNF landmarks around the blue middle line in the left figures.

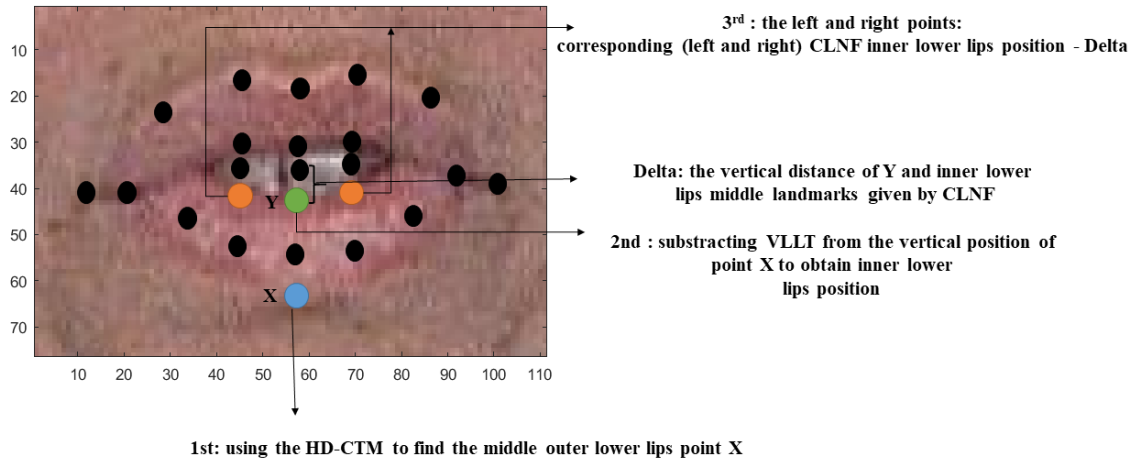


Figure 5.8: Summarized procedure of the proposed method. Note that the inner lips landmarks (black points) are mistakenly placed. Green point X is the middle outer lower lips position estimated by the HD-CTM, and blue point Y is the middle inner lower lips position by subtracting the VLLT from position of the blue point. Two orange points are the left and right key landmarks determined by the 3rd step.

derivative curve. This length is set to 20 pixels experimentally so that a sudden rapid change of outer lower lips position could be well considered. An example of the template is illustrated in Fig. 5.9 by a magenta curve with circle. The template is not necessarily symmetric in our case. In fact, a symmetric shape was tried, but it gave slightly larger errors.

Figure 5.9: This figure is plotted along the middle inner lips line. Blue curve: original luminance variation. Red curve: the smoothed luminance. Green curve: the smoothed first derivative. Magenta curve with circles: hybrid dynamic template. Red curve with stars: correlation values between the template and the first derivative curve in function of the position of the template. Vertical black line around position 301 corresponds to the initial searching position. The bold red line around position 307 located in the maximum correlation value corresponds to the estimated lower outer lips position, and the bold red line around position 287 corresponds to the estimated lower inner lips position. Two cyan lines correspond to the lower inner lips and lower outer position given by CLNF.

In order to increase the capacity of the template to follow the variation of the derivative curve, we use a hybrid dynamical template:

$$m(i) = \alpha m_0(i) + (1-\alpha) m_v(i),$$

where $m_0(i)$ denotes the fixed part of the above template. We denote by $v^{n-1}(i)$ the derivative curve of luminance variation for the previous lips image. The variable part of the template is defined as:

$$m_v(i) = v^{n-1}(i) \quad \text{for } i \in [k_{\text{opt}}^{n-1} + 1, k_{\text{opt}}^{n-1} + L_M],$$

where k_{opt}^{n-1} is the optimal position of template for the previous lips image. α is the weight of the fixed part which is set to be 0.75 experimentally in our case. We find that the performance

of the proposed method is not very sensitive to this value, and a range of α between 0.7 and 0.9 gives comparable results.

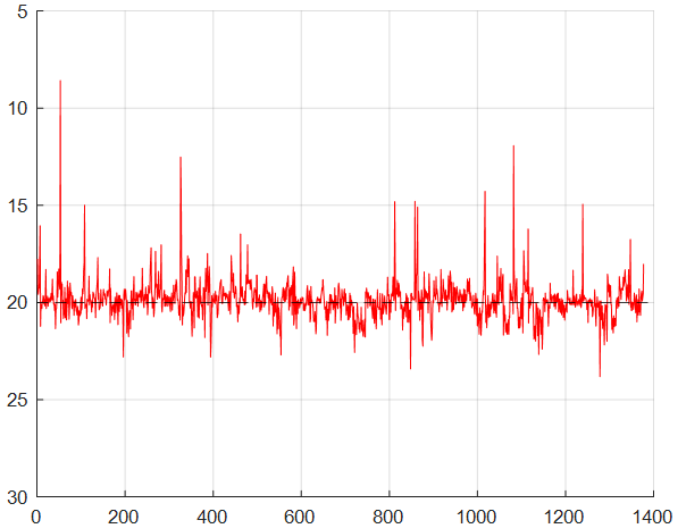


Figure 5.10: The red curve is the distance between the ground truth inner lower lips position and the outer lower lips position obtained by HD-CTM. The abscissas is the number of image frames.

(2) Determination of the optimal outer lower lips.

To determine the optimal position of outer lower lips, correlation values are calculated between the current derivative curve v_i and the template $m_v(i)$ when the template scans through the searching interval. This method using the correlation with template reduces the influence of noise and gives a more consistent result for the adjacent image. The correlation is defined as:

$$c_k = \sum_{i=1}^{L_M} v(k+1)m(i). \tag{5.1}$$

The optimal position of template, which will be considered as the optimal outer lower lips position, is determined as

$$k_{\text{opt}} = \underset{k}{\text{argmax}}(c_k), \tag{5.2}$$

where $k \in [k_0 - \delta_1, k_0 + \delta_2]$ is the searching interval. k_0 is an initial searching position. δ_1 and δ_2 are two parameters determining the length of the searching interval. These parameters (k_0, δ_1 and δ_2) are determined as follows.

- The searching region for the first image.

  For non-open lips, k_0 is the CLNF outer lower middle lips position, while for open lips, k_0 is defined as:

$$k_0 = P_{\text{inner\_lower}} + \text{VLLT}, \tag{5.3}$$

where $P_{\text{inner\_lower}}$ denotes the CLNF inner lower lips position, and $VLLT$ will be detailedly explained in Section 5.2.2.5. In this case, we take $\delta_1 = 3$ and $\delta_2 = 18$ experimentally.

- The searching region for images after the first image.

  The optimal initial search position is estimated from the previous image so that a continuous tracking can be achieved. k_0 is defined as:

$$k_0 = k_{\text{opt}}^{n-1} + \Delta k, \tag{5.4}$$

  where k_{opt}^{n-1} denotes the optimal outer lower position for the previous image, and Δk is an estimated translation of the current outer lower lips position concerning the previous image. To calculate Δk, we take the previous derivative curve $v^{n-1}(i)$ in the interval $[k_{\text{opt}}^{n-1} - 10, k_{\text{opt}}^{n-1} + 10]$, as well as the current derivative curve v_i in the same interval. After calculating the cross-correlation between these two derivative curves, a searching of its maximal value permits to determine Δk. This interval length is reduced when using this automatic tracking method. In this case, we take $\delta_1 = 3$ and $\delta_2 = 6$ experimentally. The details of HD-CTM are shown by an example in Fig. 5.8.

5.2.2.5 Determination of the lower inner lips position

To estimate the inner lower lips based on the outer lower lips, which were determined in Section 5.2.2.4, we first study the distance between the ground truth inner lower lips position and the outer lower lips position obtained by the HD-CTM. It it found that this distance is almost a perfect uniform distribution (see Fig. 5.10) except some errors from the detection using HD-CTM. More importantly, the distance distribution is invariant no matter how the lips shape varies. This distance floats slightly around a constant for each speaker. The distance is 19.9±0.97 pixels for the speaker *MD*, and 19.7±0.89, 16.6±0.83 pixels for other two speakers *DB* and *ChS*. The mean value of this distribution can be regarded as a VLLT, which can be estimated by training their data for a given speaker. For our three subjects, VLLT is set to be 20, 20 and 17 pixels, respectively. The inner lower lips position can then be estimated by subtracting the outer lower inner lips position from the VLLT.

Someone may think of using the "lower lips height" estimated by CLNF landmarks instead of the VLLT. In fact, by comparing them, we find that the "lower lips height" is poorly estimated especially when CLNF gives mistaken lips landmarks. Moreover, the evaluation performance shows that using "lower lips height" obtains a higher Root Mean Square Error (RMSE) (1.49) than the VLLT (1.0).

It should be mentioned that, if the inner lower lips middle position value obtained by Modified CLNF is less than that estimated by CLNF, the initial value is kept. A parallel translation with the same distance as the inner middle lips point is proposed to locate two neighbor inner lower points, which are the left and right points of the middle point (see Fig. 5.8).

5.2.2.6 Closed lips filter based on DCT analysis

If the upper and lower inner lips points given by the CLNF are close to each other, two cases are possible: (1) the lips are not closed, but they are badly placed; (2) these points correctly describe the closed lips (see Fig. 5.11). Therefore, it is not possible to distinguish them only from the CLNF landmarks. However, the real closed lips do not need to be corrected. To eliminate closed lips and remain good results of CLNF, a closed lips detector based on DCT coefficients is developed. The lips ROI is first determined by the 20 landmarks of CLNF which efficiently delimit the lips region and determine a precise center of this region. Then a suitable-sized ROI is determined according to this center (see Fig. 5.12), and the size is 110x75 pixels in our case. By a large number of observations, we find that our detected lips ROI is as precise as that determined by the blue marks on speaker's front in most of the cases. However, when speaker's head rotates or shifts, it is not accurate using the blue mark method, while our proposed method still gives an accurate result. It can be also seen in Fig. 5.12 that in the left one, the lip is not well centered in the ROI, while in the right case, the lip is centered inside ROI. In fact, the proposed method benefits from CLNF which is robust to the rotation or shift of speaker's head.

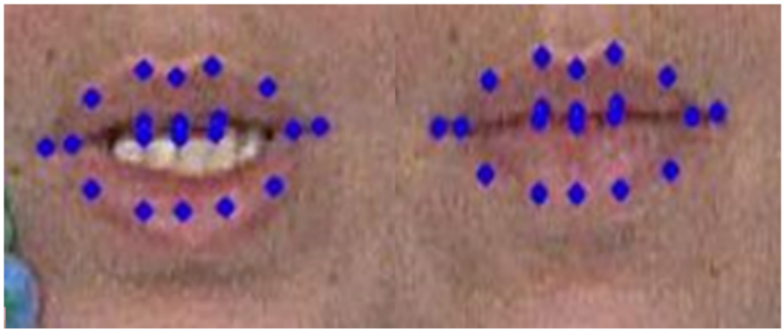


Figure 5.11: An example of ambiguous inner lips detected by CLNF. The left one shows the mistaken CLNF landmarks for an open lip, while the right one corrects landmarks for a closed lip. Note that CLNF landmarks are strongly similar in these two cases.

The DCT coefficients are calculated from the lips ROI, and 10x10 coefficients in the low-frequency region are retained. Ten images of closed lips are chosen to build a *closed lips template* (see Fig. 5.13). For closed lips detection, the euclidean distance is calculated between the DCT coefficients of the given lips ROI and the template. A threshold permits to distinguish the closed and open lips. The threshold is fixed as 80 experimentally for speaker *MD*. By applying this method to all the images, we can extract the closed lips, which are skipped by the HD-CTM.

5.2.2.7 Parameter A correction based on the periodical spline interpolation

Recall that for round lips, the two endpoints determined by CLNF are mistakenly placed from the acoustic point of view. It will cause the wrong parameter A. Therefore, it is necessary to

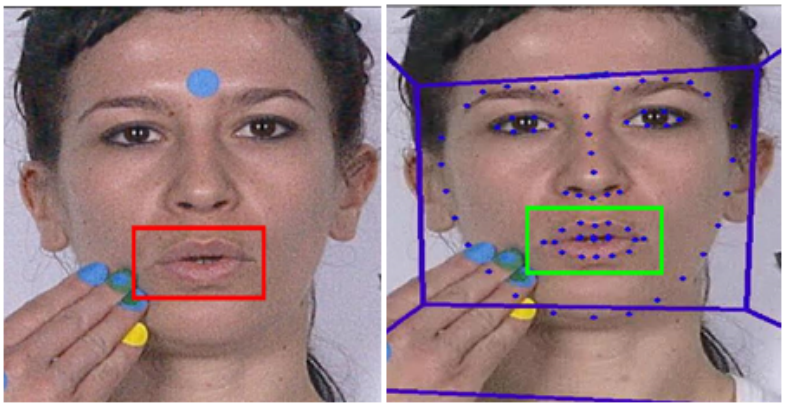


Figure 5.12: Left is the lips ROI determined by a blue mark in the front of the speaker. Right is the lips ROI (same size) determined by a center point estimated from CLNF landmarks.

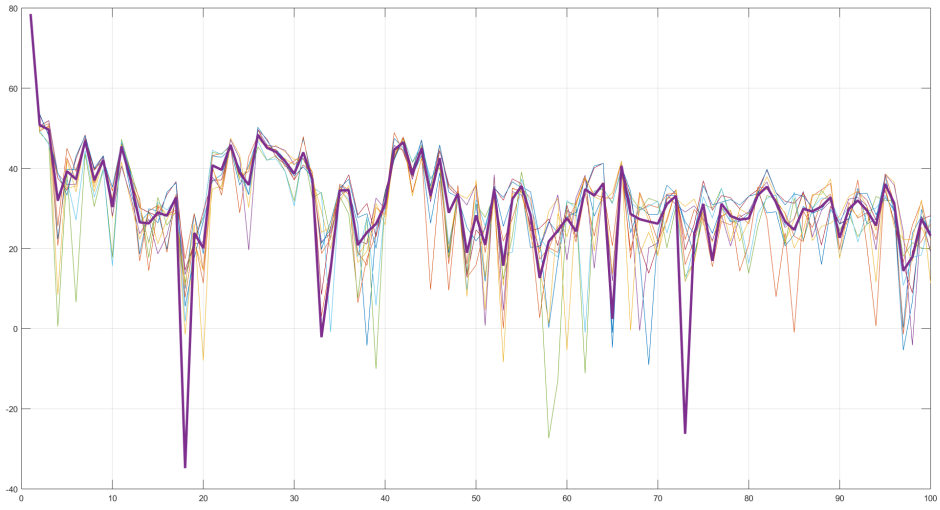


Figure 5.13: DCT coefficients derived from ten closed lips images. Ten curves with different colors correspond to the ten closed lips. Abscissa is the number of coefficients, and y-axis is the DCT coefficient values (in dB). The purple curve shows the mean vector of these DCT coefficients, and it is considered as a model for closed lips detection.

correct these two endpoints. After a large number of observations, we consider that in case of the round lips, the six points from CLNF (three upper points and three lower points) are assumed to be correctly determined in a majority of cases (see Fig. 5.14(a)). We propose to estimate the inner lips contour using the periodical spline interpolation based on these six points. To realize this, these six points are firstly dilated in the vertical scale to form a square (see Fig. 5.14(b)) in order to obtain a regular repartition of these points in the polar coordinate. Cartesian coordinates of these points are then converted into polar coordinates. The center of the polar coordinates is situated in the middle of the two middle landmarks of CLNF inner lips. A spline interpolation is applied to the polar coordinates (see Fig. 5.14(c)). In order to take into account the initial condition of the endpoints, the six points are periodized three times to prepare for a periodical spline interpolation (see Fig. 5.14(d)). Finally, by returning to the original scale, a full contour interpolation can be obtained (see Fig. 5.14(e)).

To apply this method, an automatic round lips detector is necessary to select the round lips from the image sequence which contains all kinds of lips shapes. A similar method as the closed lips filter mentioned in Section 5.2.2.6 is applied to detect the round lips. Firstly, a round lips DCT template is trained from several round lips images. DCT coefficients are calculated from the lips ROI. Secondly, this template scans through all the images, and the distance between current image and template is computed. Lips are considered as round lips if its distance is less than a threshold which is determined experimentally. The performance of the automatic round lips detector is evaluated on 3184 lips images (10 repetitions of the speaker *MD*). Only 42 round lips are mistaken among 467 round lips images (about 9% error rate). We plot the distribution of 3184 lips parameters in the A-B plane. The distribution before correcting A parameter is shown in Fig. 5.15(a), while the one after correcting A parameter is shown in Fig. 5.15(b). After correcting A parameter, we can see that the third viseme (triangle) is shifted to the correct position corresponding to the small A parameter. We observe that some triangles indicating the round lips are not shifted to the correct positions.

5.2.3 Evaluation and Results

To evaluate the performance of the proposed method, the A and B parameters estimated by them are compared with the ground truth and CLNF.

5.2.3.1 Evaluation of B parameter

It is shown visually in Fig. 5.16 that the HD-CTM combined with the back-subtracting of VLLT efficiently corrects B errors of CLNF. Besides, from Fig. 5.17, we see that the estimated B parameters for the image sequence and the ground truth B parameters are very close to each other in most of the cases. By contrast, the CLNF B parameter has an evident difference with the ground truth, especially for the speakers *DB* and *ChS*. One can see that the errors are significantly reduced after using our proposed method.

The accuracy of the proposed method can be measured by the Root Mean Square Error

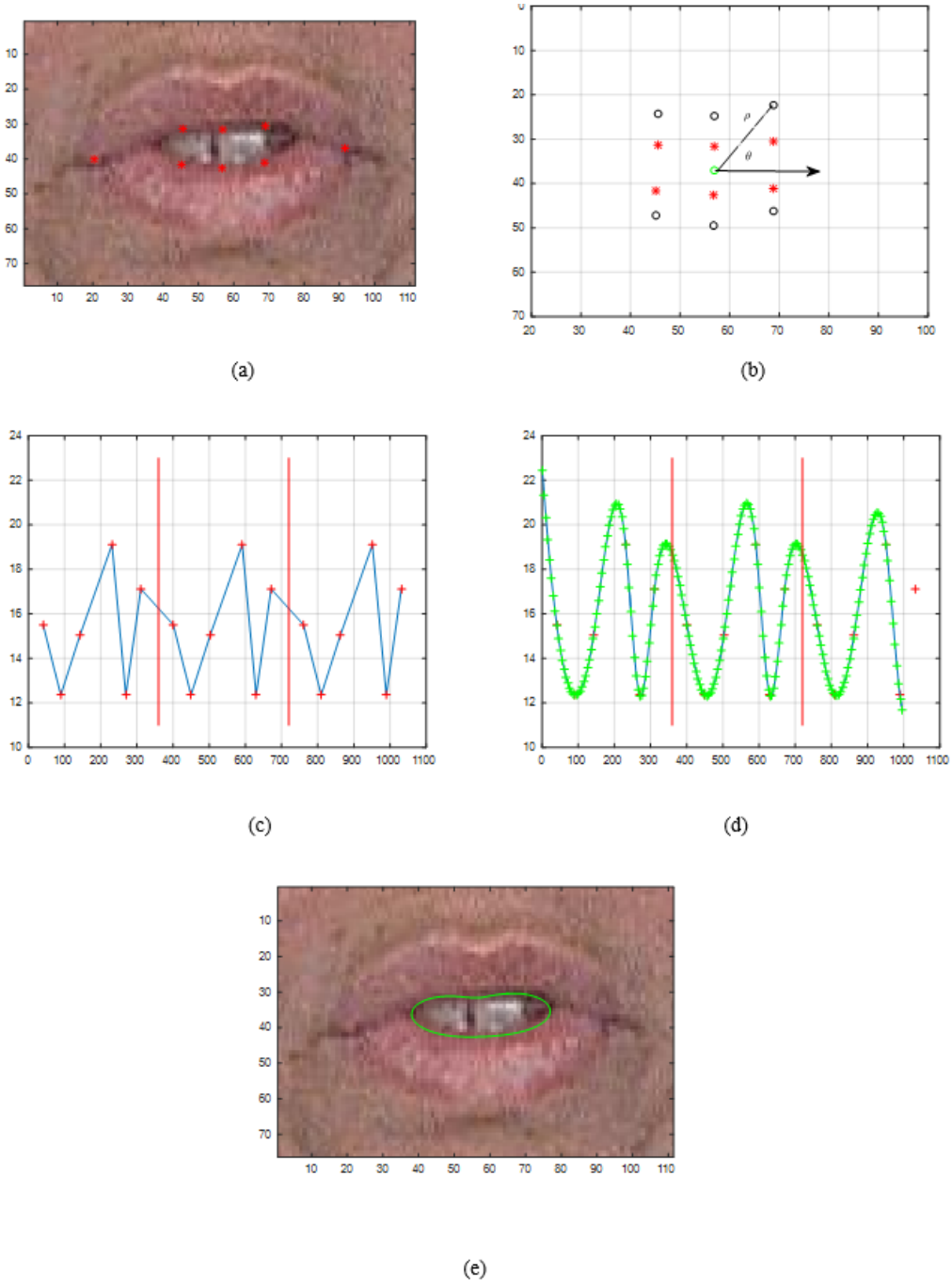
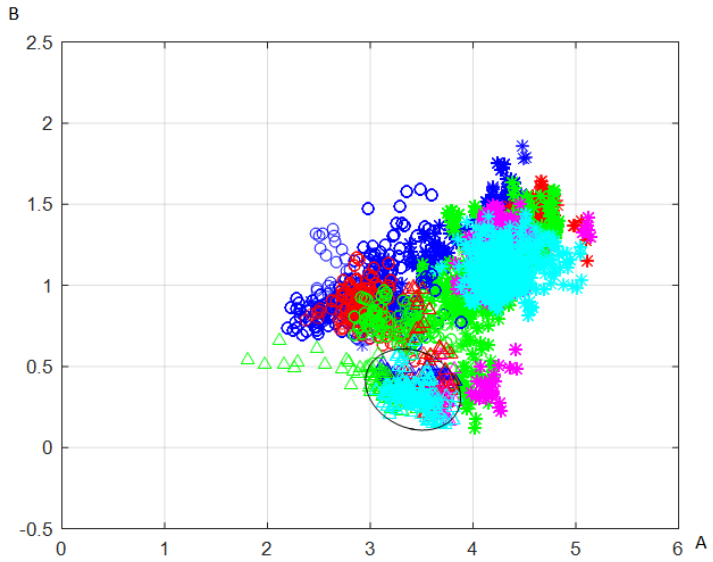
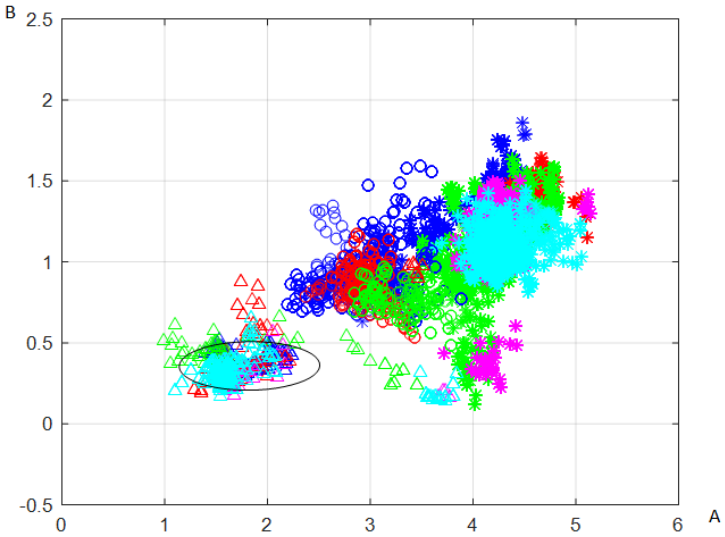


Figure 5.14: Illustration of the period spline interpolation method to correct the A parameter errors for round lips. (a) Speaker's lips with CLNF original landmarks (note that two endpoints of inner lips contour are mistakenly placed). (b) Six center points are plotted with red stars which are dilated in the vertical scale to form a square (black circles). They are then converted into polar coordinates. (c) In polar coordinates, the six points are repeated 3 times. (d) Periodical spline interpolation is realized and only the period inside two red lines (one period) is used to returned to Cartesian coordinates. (e) The full interpolated inner lips contour.



(a)



(b)

Figure 5.15: Performance of the round lips detection. Lips parameters distribution is plotted in the parameter A-B plane (the third viseme is plotted with triangle). (a) CLNF with corrected B parameter, but no correction of A parameter. (b) CLNF with corrected A parameter with the proposed method. B parameter is the same as in (a). The black ellipse shows the distribution of the third viseme which corresponds to the round lips.

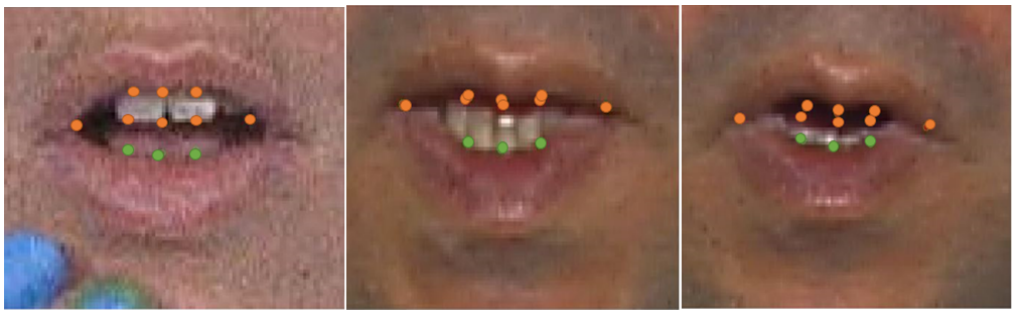


Figure 5.16: Examples of initial mistaken CLNF landmarks (orange points) and corrected landmarks using the proposed method (green points) for inner lower lips.

Table 5.1: RMSE values of B parameter for CLNF and Modified CLNF (in pixels and mm).

RMSE	*MD*	*DB*	*ChS*	Total
CLNF	3.84(2.0mm)	4.02(2.1mm)	3.53(1.8mm)	3.81(2.0mm)
Modified CLNF	1.06(0.6mm)	0.90(0.5mm)	0.94(0.5mm)	0.99(0.5mm)

(RMSE) as follows:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{t=1}^{N}(y_t - \hat{y}_t)^2},$$

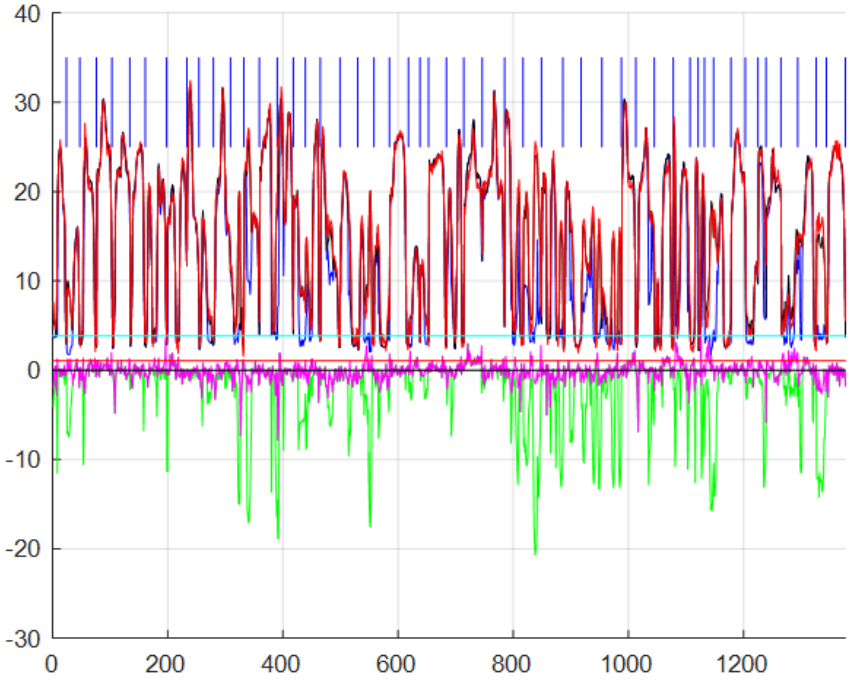
where t is the frame number, N is the total number of frames, y_t is the real value of the target and $\hat{y}_t$ is the estimated value of the target.

The RMSE of the estimated B parameter is shown in Table 5.1. Total RMSE of the errors is reduced to 1 pixel (0.5mm), instead of 4 pixels (2.0mm) when using CLNF. It outperforms the result in [194] which gave about 1.0mm RMSE.

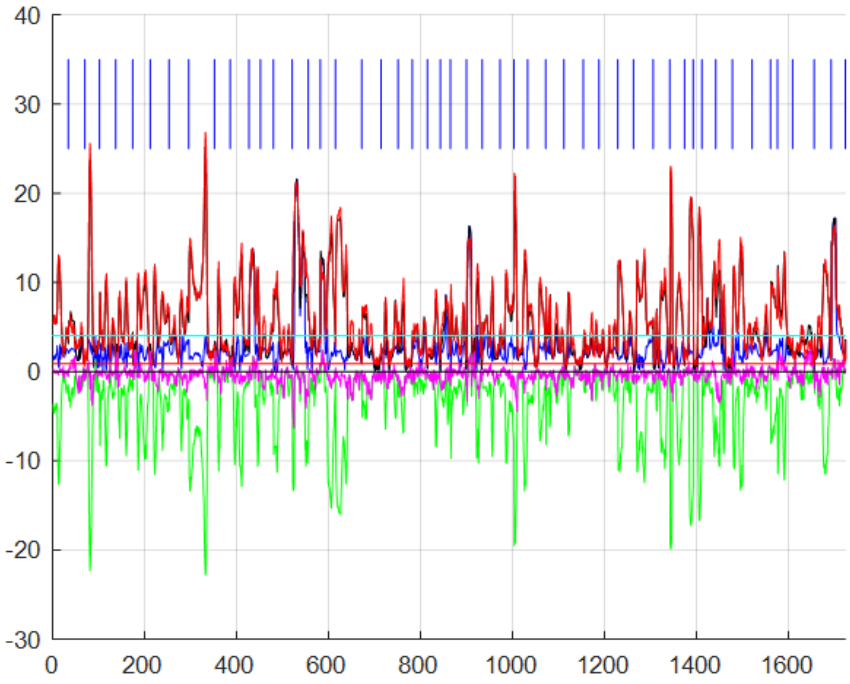
5.2.3.2 Evaluation of *A* parameter

The periodical spline interpolation method efficiently corrects A parameter errors of CLNF, which is visually shown in Fig. 5.18. The evaluation of A parameter is based on the round lips images which are selected using the DCT filter (Section 5.2.2.6). A total 927 round lips images are selected (222 images for the speaker *MD*, 396 images for the speaker *DB* and 309 images for the speaker *ChS*). In Fig. 5.19, the errors between the A parameter estimated by the adaptive ellipse model and the ground truth are shown. We can see that the error is much less than using the CLNF. To further measure the error, we calculate the statistic error (see Table 5.2). We observe that there is a huge bias regarding the mean value for CLNF error which is much greater than the standard deviation (see Fig. 5.19). Therefore, we calculate the RMSE to measure the precision of A parameter. It can be seen from Fig. 5.19 that the huge RMSE of CLNF is significantly reduced, and this is comparable to the state of the art [194].

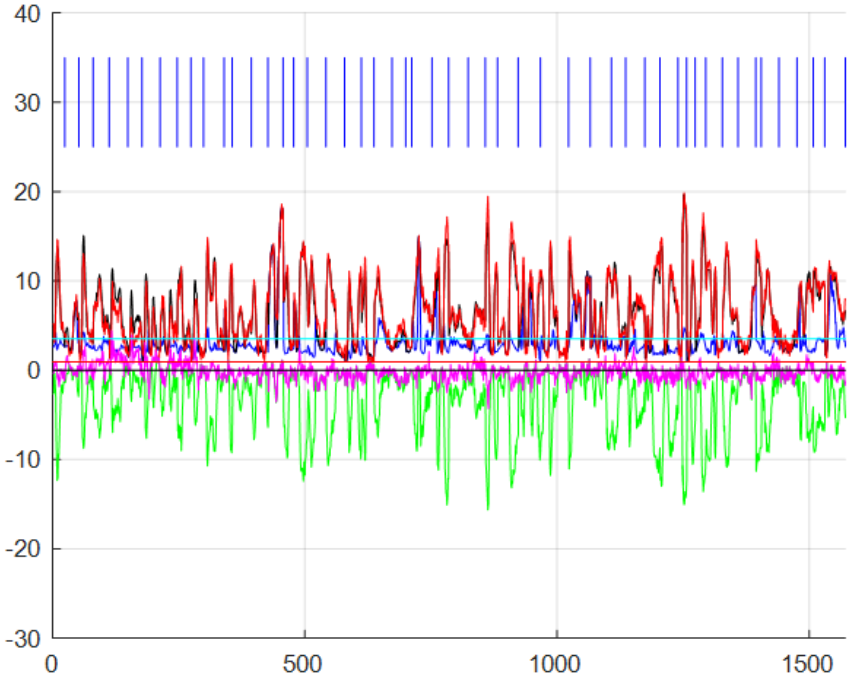
We note that, concerning the error of A parameter, the precision is not as demanding as B parameter from speech production point of view, and the estimation of A parameter is less


40
30
20
10
0
-10
-20
-30
0
200
400
600
800
1000
1200

(a)


40
30
20
10
0
-10
-20
-30
0
200
400
600
800
1000
1200
1400
1600

(b)



(c)

Figure 5.17: Performance of the HD-CTM for correcting B parameter. The figure shows the result of three speakers *MD, DB* and *ChS* from top to bottom. Abscissa is the image frame number and y-axis is the distance measured in pixels. Red curve: the ground truth B parameter (in pixels). Blue curve: CLNF B parameter. Black curve: B parameter estimated by the proposed method (HD-CTM). Green curve: errors of CLNF. Magenta curve: errors of the proposed method. The short blue lines at the top are the boundaries for 50 words in the word database.

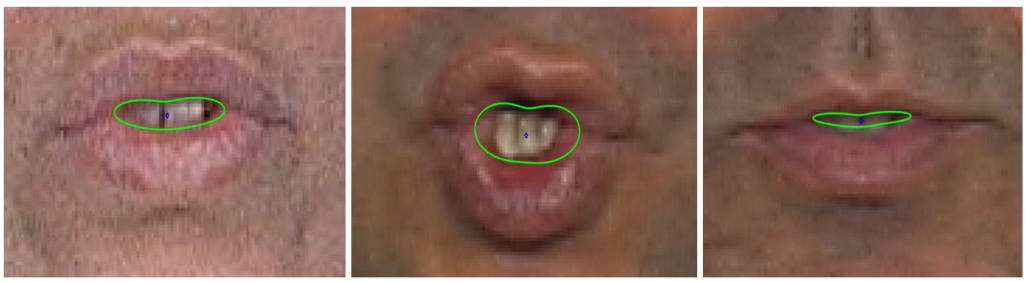


Figure 5.18: Visual result of periodical spline method for round lips. Green curve is the full inner lips contour for round lips, and the blue point is the center of the inner lips landmarks.

Table 5.2: RMSE values of A parameter for CLNF and the periodical spline interpolation method.

RMSE	*MD*	*DB*	*ChS*	Total
CLNF	39.64(22.4mm)	39.94(22.6mm)	32.55(18.4mm)	37.40(21.2mm)
Modified CLNF	8.03(4.5mm)	5.84(3.3mm)	5.07(2.8mm)	6.11(3.5mm)

precise in practice. Meanwhile, comparing the error of B parameter with that of A parameter, we see that the error of B parameter is less than A parameter, which is coherent with the result in [194].

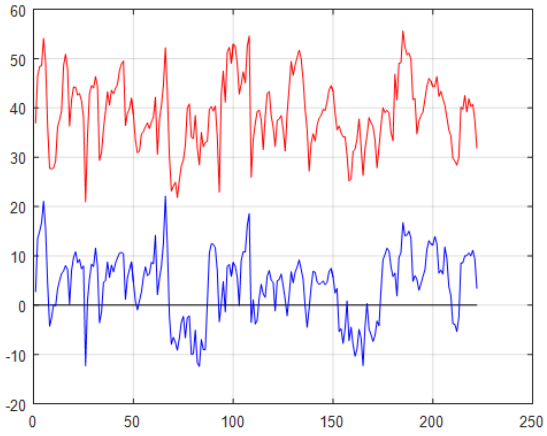
Finally, to evaluate the joint performance of A and B parameters, we study the distribution of three vowel visemes in A and B parameter plane. In Fig. 5.20, three visemes are plotted using the first repetition of the speaker *MD*. The distribution of each vowel is presented by a Gaussian ellipse. We see that three visemes of CLNF are mixed, especially for the third viseme (see Fig. 5.20(a)). After the B parameter is corrected by the HD-CTM, these visemes are well-distributed in the axis B (see Fig. 5.20(b)). After the A parameter is corrected by the periodical spline interpolation, the third viseme is correctly distributed corresponding to the axis A (see Fig. 5.20(c)). It can be seen that the distribution of three visemes corresponds coherently [40] to the ground truth distribution in Fig. 5.20(d).

5.3 Adaptive Ellipse Model

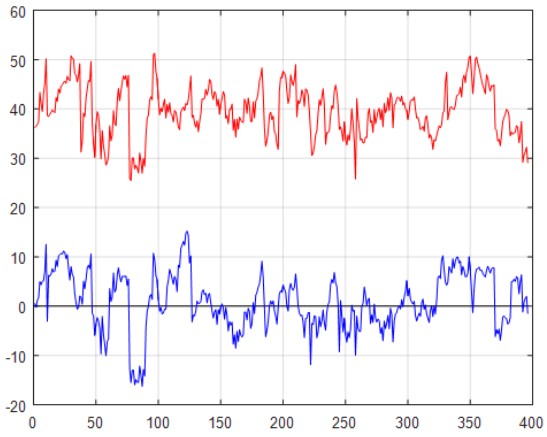
In Section 5.2, we proposed a CLNF based model to estimate A and B parameters of inner lips. In this section, we will explore a new method based on an adaptive ellipse model to estimate A and B parameters without generating a whole inner lips contour. This model is motivated to overcome the following shortcomings of the CLNF based method. Firstly, the CLNF based method depends on the CLNF landmarks. Especially, in case of the round lips, the real inner lips contour is inside the six inner points. It is difficult to estimate a correct A parameter. Secondly, the CLNF based method needs a round lips detector introduced in Section 5.2.2.7. Thirdly, the A parameter given by the CLNF is not only incorrectly placed in the round lips shape case, but also in other cases.

This method can be summarized as follows. An image processing is first realized to segment the inner lips as much as possible. To make the extracted inner lips more complete and connected, a single discontinuity smoothing and an interrupted region filling are applied. Then, an adaptive ellipse is used to match the inner lips and gives the best A and B parameters. An outline of this process is shown in Fig. 5.21.

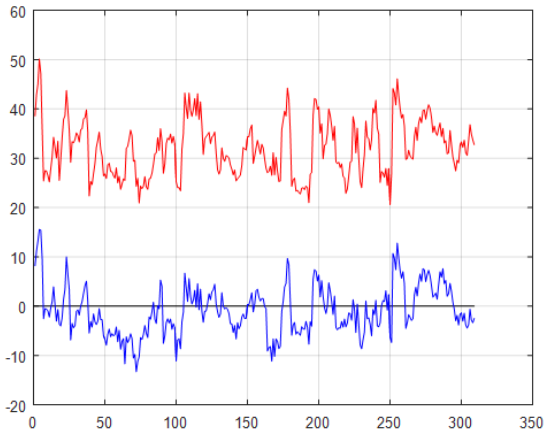
This section is organized as follows. Firstly, the experimental database is introduced in Section 5.3.1.1. Then, we will describe each step of the adaptive ellipse model in Section 5.3.1.4. Evaluation and results on the accuracy and robustness of the method are presented in Section 5.3.2.



(a)



(b)



(c)

Figure 5.19: Performance of the periodical spline method for A parameter. Three figures show the results of three speakers *MD, DB* and *ChS* from top to bottom. Abscissa is the image frame number and y-axis is the distance measured in pixels. Red curve: the error between the values obtained by the CLNF and the ground truth. Blue curve: the error between the values obtained by the periodical spline method and the ground truth.

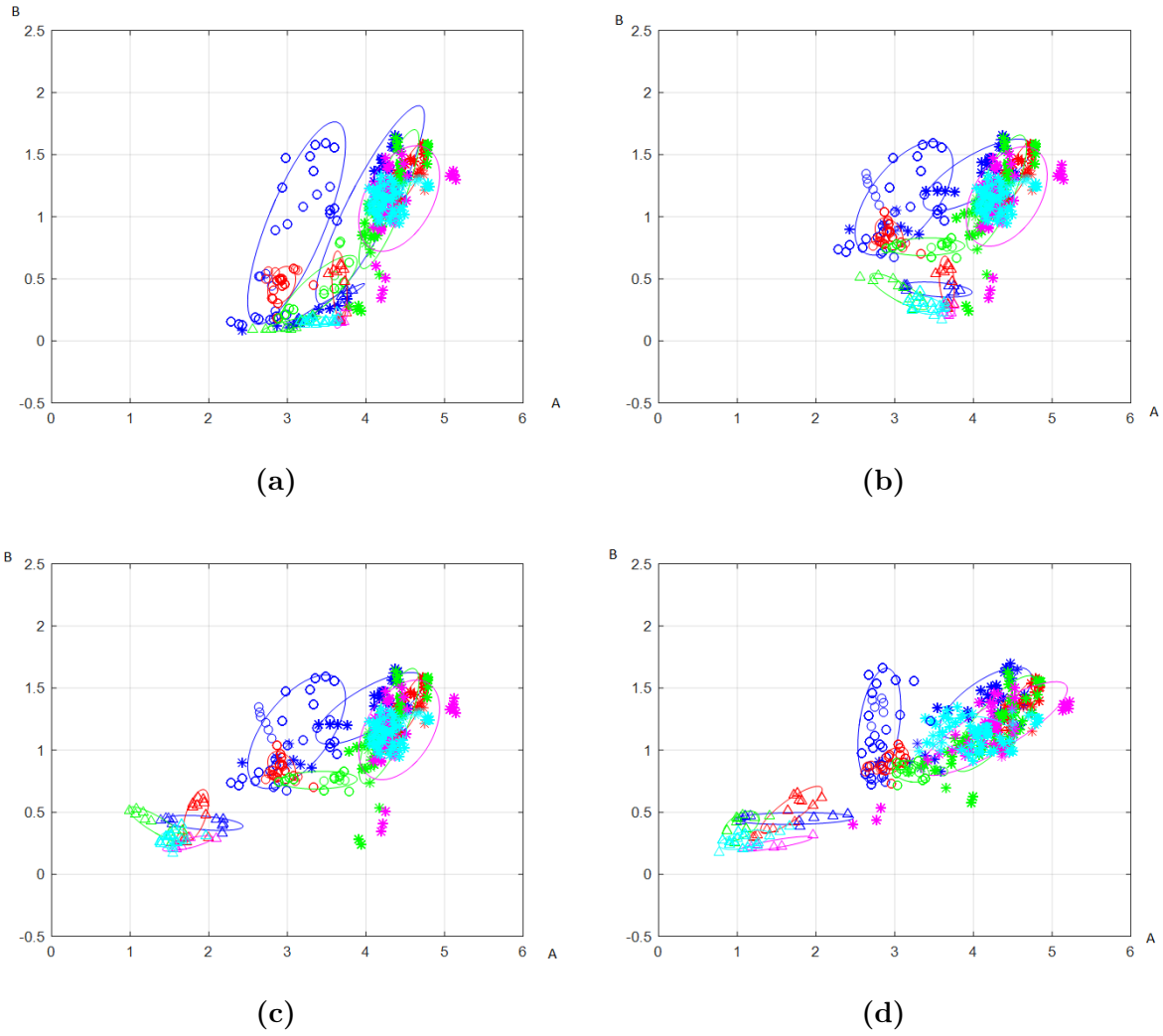


Figure 5.20: Global performance of the proposed methods for both A and B parameters. The figures are plotted in the parameter A-B plane with all the vowels in one repetition for the first subject. (a) CLNF. (b) CLNF with corrected B parameter but no corrected A parameter for the round lips. (c) CLNF with corrected B and A parameters for the third viseme, and automatic detection of the third viseme. (d) The ground truth. Stars correspond to the first viseme, circles to the second viseme and triangles to the third viseme. The color order is blue, red, green, magenta and cyan. They correspond to the vowel order in Table 2.3.

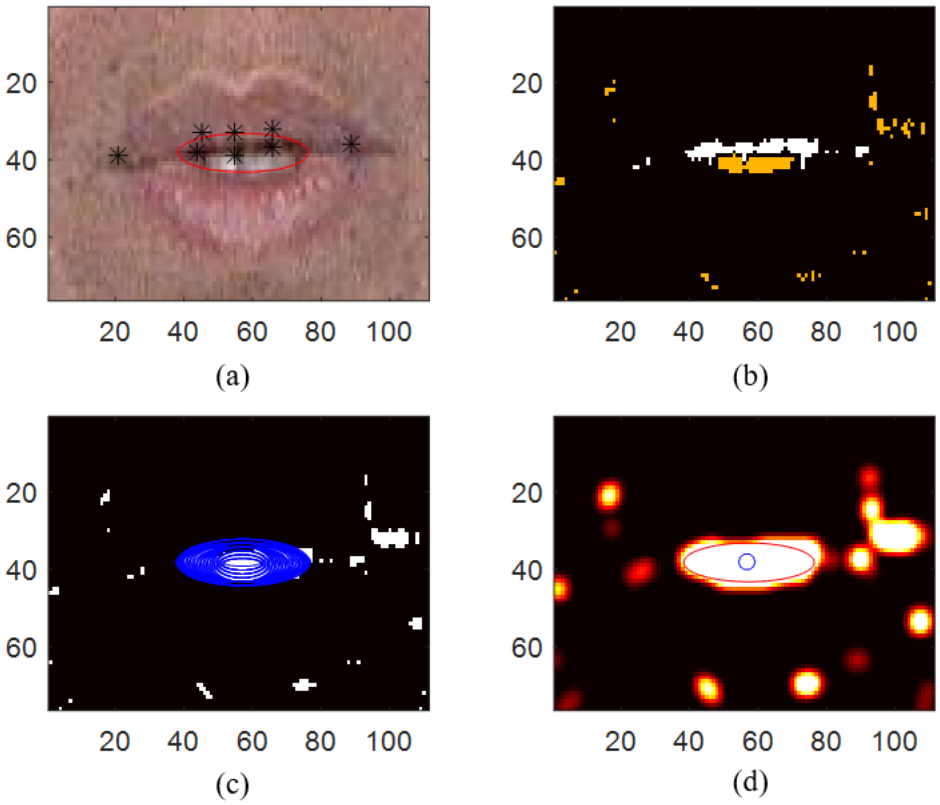


Figure 5.21: Overview of the adaptive ellipse model for inner lips parameter estimation. (a) Raw lips image in ROI with the optimal inner ellipse shown in red. Black stars are the inner lips landmarks given by CLNF (for comparison). (b) Extraction of dark area (white region) and teeth (yellow region) using image processing. (c) Adaptive searching for the optimal position and size of the ellipse. (d) The final optimal ellipse determined after smoothing and scaling post-processing.

5.3.1 Methodology and experiment details

In this subsection, we will first give a brief introduction of the experimental database, and present the details of the adaptive ellipse model.

5.3.1.1 Database

We use the first repetition of the three speakers (*MD*, *DB* and *ChS*) corresponding to all types of lips shapes to evaluate A and B parameters. The number of images in our database for three subjects is 1377, 1744 and 1572, respectively (totally 4693 images). To evaluate the performance of the proposed model, the ground truth inner lips contour is extracted manually by an expert placing several landmarks on lips. In the application of CS vowel recognition, the temporal boundaries of each phoneme are extracted from the audio signal using a conventional ASR system and a forced alignment procedure.

5.3.1.2 Image processing for segmenting inner lips area

In order to estimate the inner lips parameters, the most important thing is to determine the inner region of lips. More precisely, we propose to first extract the inner region of lips instead of directly finding the lips. As we can see, the teeth and darker area inside inner lips have different color properties from lips [47], [195], [196] (see Fig. 5.24), and thus a color based method is proposed to delimit the inner lips and non-inner lips as much as possible. A lips ROI is determined using the landmarks given by CLNF (see Section 5.2.2.6).

We first present an image processing approach to detecting the dark area inside inner lips. In *YCbCr* space, the dark area has a lower luminance, and a threshold of Y value can be used to distinguish it. Now we take the speaker *MD* as an example. In our experiment, a threshold of 70 gives a satisfactory performance. The dark area extraction performances of different thresholds are shown in Fig. 5.22. The effects of using thresholds 40, 70 and 100 are shown in Fig. 5.22(b), (c) and (d), respectively. The best one is the threshold of 70. When the threshold is too small, some dark areas will be omitted, while some extra areas will be detected as dark area if the threshold is too large.

After detecting the dark areas inside inner lips, we now try to extract the teeth. In RGB color space, teeth have a much whiter color than other regions. In comparison, lips have a red-dominant color component. We thus consider the ratio R/G to efficiently distinguish the teeth region. In addition, the luminance of teeth is bright, and can be detected either by Y value or by G + B value. In this thesis, we consider the latter condition. After a large number of observations, we find that a ratio $R/G < 1.25$ (coupled with $G+B > 160$) permits to extract the teeth efficiently. The teeth extraction effects using different thresholds are shown in Fig. 5.23. In order to distinguish the detected teeth region from the dark area, we deliberately show the detected teeth region by different colors. Fig. 5.23(b), (c) and (d) are the effects of thresholds 1.15, 1.25 and 1.35, respectively. We can see that the best performance

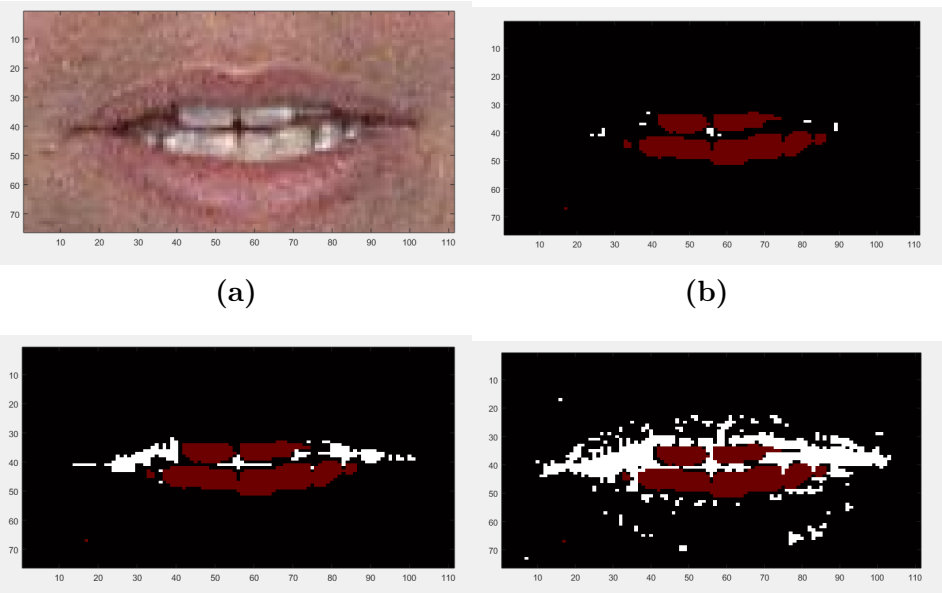


Figure 5.22: Effects of the thresholds for extracting the dark areas of inner lips. In (b)-(d), the white areas correspond to the detected dark areas inside the inner lips.

of teeth detection is obtained with threshold of 1.25 (see Fig. 5.23(c)), while an unsuitable threshold will cause bad performance of teeth detection (see Fig. 5.23(b) and (d)).

It should be noted that the tongue has very similar color as lips and skin. It causes difficulties to segment tongue and extract a complete inner lips region only by the color based approach. After taking into account the dark area and teeth detection, the final effect of inner lips segmentation is shown in Fig. 5.24. We can see that the inner lips area is well segmented except the tongue.

After this preliminary image processing procedure, the pixel value of the detected inner lips region is set to 255 (i.e., white pixel), and 0 (i.e., black pixel) for other regions inside lips ROI. We mention that the image processing is subject-dependent and sensible to the variable lighting conditions. For any speaker, the adjustable parameters are the thresholds for extraction of dark area and teeth, and they can can be determined experimentally.

5.3.1.3 Single discontinuity smoothing and interrupted region filling

The previous image processing allows extracting the teeth and dark areas inside inner lips. However, it is still not enough to form a whole inner contour since tongue and fuzzy invisible teeth are not able to be detected. Matlab function *imfill* can be used to reduce the discontinuity only when it is entirely surrounded by white pixels. But it is not suitable when the extracted region is not connected. In order to solve this problem, we propose two methods: the *single discontinuity smoothing* and *interrupted region filling*.

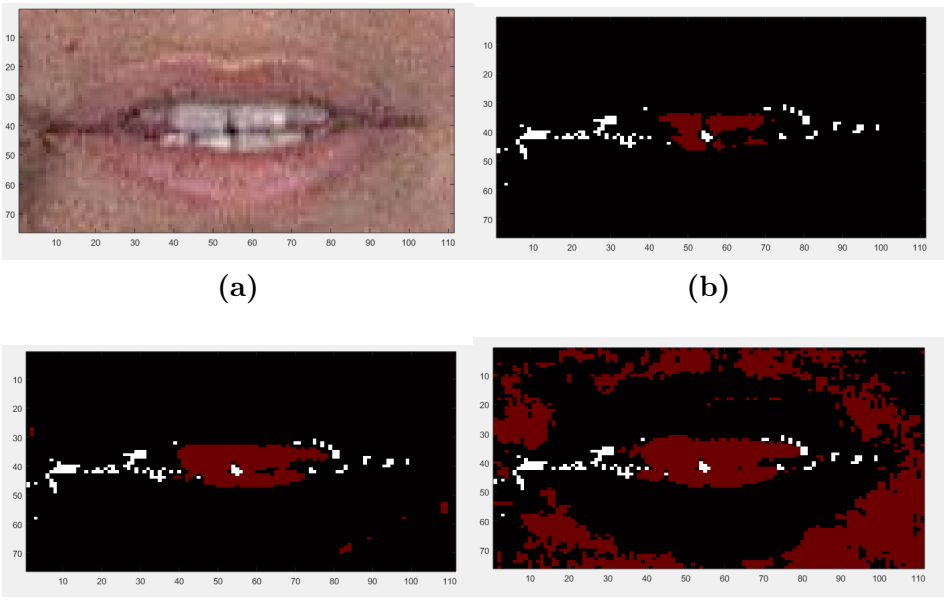


Figure 5.23: Effects of the thresholds for the teeth detection of inner lips. (a) shows the original lips ROI. In (b)-(d), the dark red areas correspond to the detected teeth inside the inner lips area.

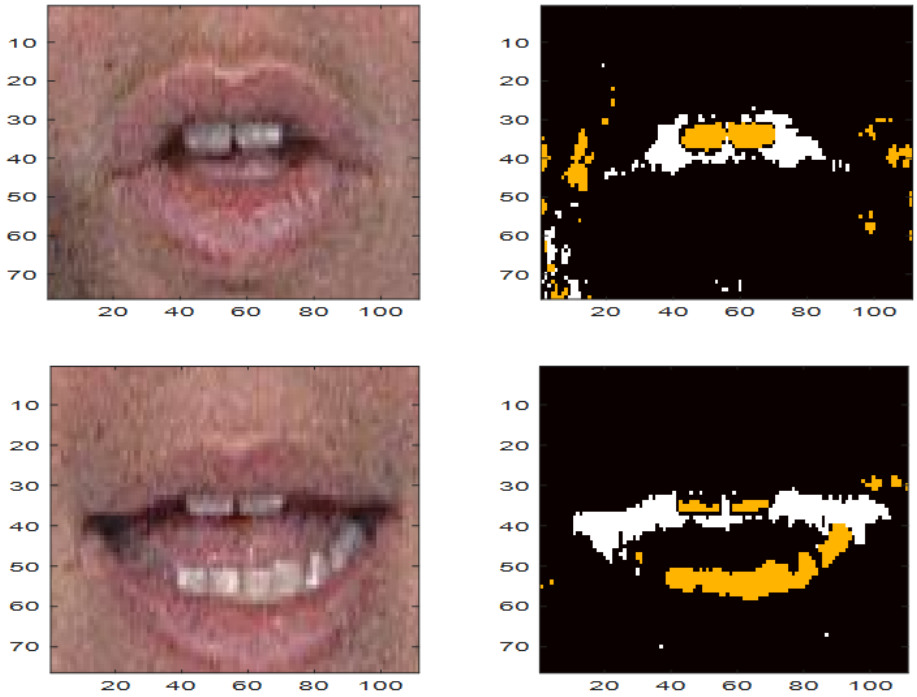


Figure 5.24: Example of teeth and dark area extraction for inner lips region. Left is the original lips ROI, while right is the detected dark area (white part). The yellow area is the detected teeth. The tongue is not detected, due to its similar color with lips.

The single discontinuity smoothing aims at eliminating the single pixel which is not detected inside inner lips. In the ten cases of Fig. 5.25, the central pixel (single rupture) will be set to be "white". This procedure is implemented from top to bottom and then from left to right of the inner lips area. The result of the single discontinuity smoothing is shown in Fig. 5.26.

However, in some cases, one or several blocks of the black pixel remain after the single discontinuity filling (see the bottom image of Fig. 5.26). The interrupted region filling is proposed to solve this problem. It includes the processing in the horizontal and vertical directions. We first examine each row in the lips rectangle ROI, moving from top to bottom. In a row, the orange line in Fig. 5.27 is divided by several black intervals separated by white intervals. Recall that the white part indicates the detected inner lips area, and the black part is the non-inner lips area. If a black interval length is less than the sum of two adjacent white intervals (i.e., $b_i \leq l_i + l_{i+1}$), the black interval will be filled as a white interval. The same procedure is then applied column by column from left to right. The result is shown in Fig. 5.28.

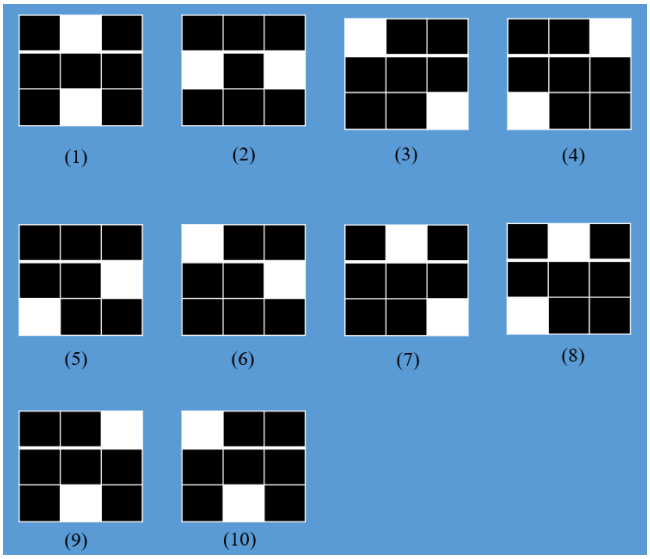


Figure 5.25: Principle of the single discontinuity filling. When the center pixel is black and the surrounding pixels are like these ten configurations, the black center pixel will be changed to white.

5.3.1.4 The process of adaptive ellipse searching

After the above pre-processing steps, a relatively well filled inner lips region is obtained. It is a binary image where the detected inner lips region is white and other areas are black. However, uncontrolled lighting conditions (especially when the hand gets close to face), highly deformable of lips and the appearance of the tongue[1] cause great difficulties to extract a complete inner lips region only by the color based approach.

[1] It has almost the same color with skin.

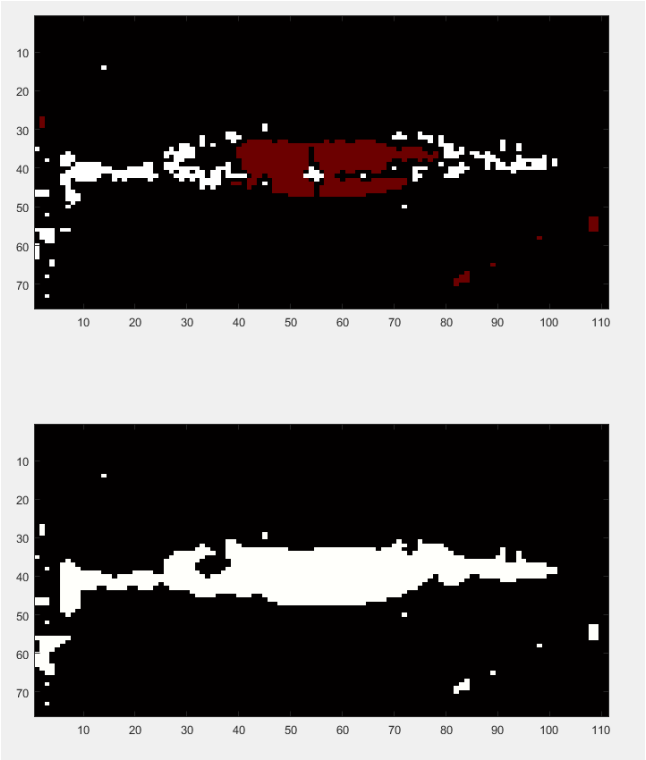


Figure 5.26: One example showing the result of single discontinuity filling. The top one is the raw inner lips region after teeth and dark area detection. The bottom one is the processed image by the single discontinuity filling.

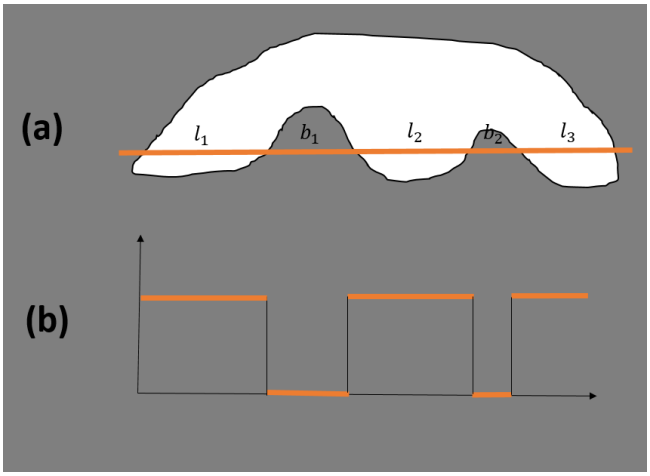


Figure 5.27: Principle of interrupted region filling. The white region in (a) is the inner lips region processed by the single discontinuity filling. We first check this region row by row. The orange line is one such line. The white and black intervals along this line are shown in (b) which are marked as l and b. When the length of l and b satisfies certain criterion, the black interval will be filled as white. This figure shows the processing in the horizontal direction, while the vertical interrupted region filling has the same principle.

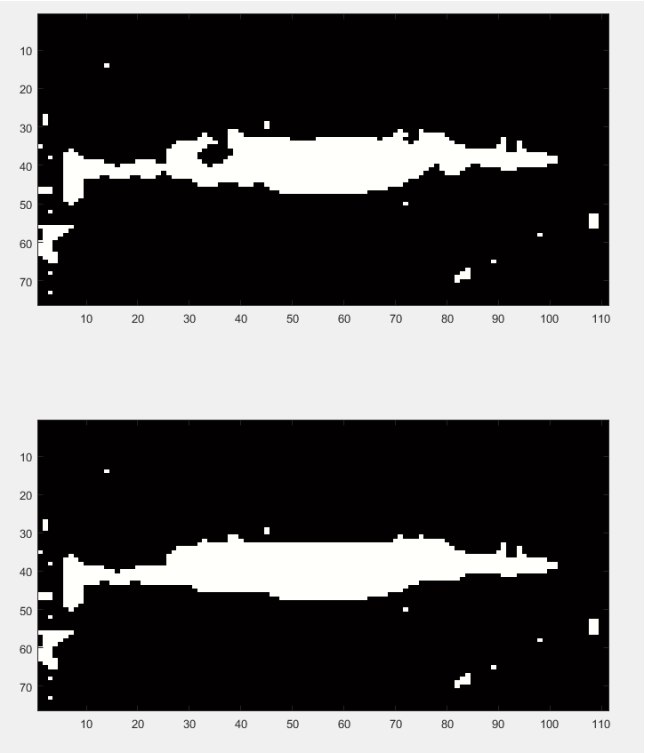


Figure 5.28: One example showing the result of the interrupted region filling. The top one is the inner lips region after single discontinuity filling. The bottom one is the processed image by the interrupted region filling.

Our idea is to fit an adaptive ellipse which can match the detected white area as good as possible. The ellipse starts moving and growing up from the center of the image so that the influence of the noise mentioned above can be efficiently eliminated. It should be noted that we do not claim the inner lips shape is always an ellipse. We just try to find an equivalent ellipse which optimally matches the lips parameters.

The adaptive ellipse model contains the following steps.

(1) Initial ellipse center and radius determination. Firstly, in order to reduce the noise, a small rectangle with size 30×10 is established using the center of lips ROI. This size is then adjusted until the white area in this rectangle takes a significant percentage. More precisely, the small rectangle grows towards left and right by 3 pixels, and then towards up and down by 3 pixels. If the increased white pixels are more than 20% of the increased rectangle area, it continues expanding without exceeding the boundary of lips ROI. Otherwise, it stops. In the end, this rectangle contains the main part of inner lips region.

Now we determine the center of lips region as:

$$x_0 = \frac{\sum\limits_i iP_x(i)}{\sum\limits_i P_x(i)}, \qquad y_0 = \frac{\sum\limits_j jP_y(j)}{\sum\limits_j P_y(j)}, \tag{5.5}$$

where i and j are the pixel index in the detected inner lips area, $P_x(i)$ is the sum of all the horizontal luminance along the ith row and $P_y(j)$ is the sum of all the vertical luminance along the jth column. (x_0, y_0) will be the initial center of the adaptive ellipse. To determine the semi-major axis a and the semi-minor axis b of initial ellipse, we calculate the inertial moments about the two axes as:

$$\sigma_x^2 = \frac{\sum\limits_i (i - x_0)^2 P_x(i)}{\sum\limits_i P_x(i)}, \qquad \sigma_y^2 = \frac{\sum\limits_j (j - y_0)^2 P_y(j)}{\sum\limits_i P_y(j)},$$

and then take

$$a = \frac{\sigma_x}{2}, \qquad b = \frac{\sigma_y}{2}. \tag{5.6}$$

(2) Optimal ellipse searching. By using the center in (5.5) and the initial parameters in (5.6), the initial ellipse is very small compared with the lips region (see Fig. 5.29). This small ellipse starts to move and grow up successively in the four directions (up, right, down and left), one direction by a step. For each step, its radius and the center position (see Fig. 5.30(a)) will be updated using (5.7). They will converge to the optimal position and size which matches the inner lips best. This adaptation can be summarized as follows:

$$\begin{aligned} a_{n+1} &= a_n + \Delta a, \qquad b_{n+1} = b_n + \Delta b, \\ x_0^{n+1} &= x_0^n + \Delta x_0, \qquad y_0^{n+1} = y_0^n + \Delta y_0, \end{aligned} \tag{5.7}$$

where $\Delta a, \Delta b, \Delta x_0$ and Δy_0 are the strides of a, b, x_0 and y_0, respectively.

In each iteration, only a and x_0 (or b and y_0) in one direction are updated at the same time. The signs of Δx_0 and Δy_0 are changed according to the moving direction (for example, Δx_0

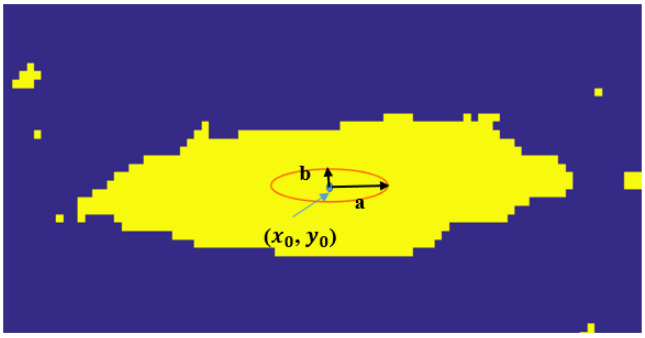


Figure 5.29: Parameters of the initial ellipse. The yellow region is the extracted inner lips area after the previous preprocessing. The center point (x_0, y_0) of the yellow area is used as the initial ellipse center. a and b are the semi-major axis and the semi-minor axis of the initial ellipse, respectively.

is positive if it moves to the right, negative if it moves to the left, etc.). In our experiment, Δx_0 is fixed to 0.5 and Δy_0 is fixed to 0.2. Moreover, Δa is fixed to 0.5 and $b = k\Delta a$, where $k = b_n/a_n$. We denote by S_w the area of the white region in the current ellipse, and by S_e the current ellipse area (see Fig. 5.31). The ellipse expansion will stop if

$$S_w \leq S_e \times 0.7,$$

or the searching region exceeds the lips ROI. When the expansion of one direction stops, the expansion of other directions may continue. The optimal ellipse is obtained until it stops in all the four directions (see Fig. 5.30(b)).

The parameters of final ellipse are used to estimate A and B parameters as follows:

$$A = \gamma \times 2a, \quad B = \gamma \times 2b,$$

where γ is logically equal to $\sqrt{0.7} = 0.84$. However, experimentally, γ is set to be 0.87 to make the estimation results more accurate.

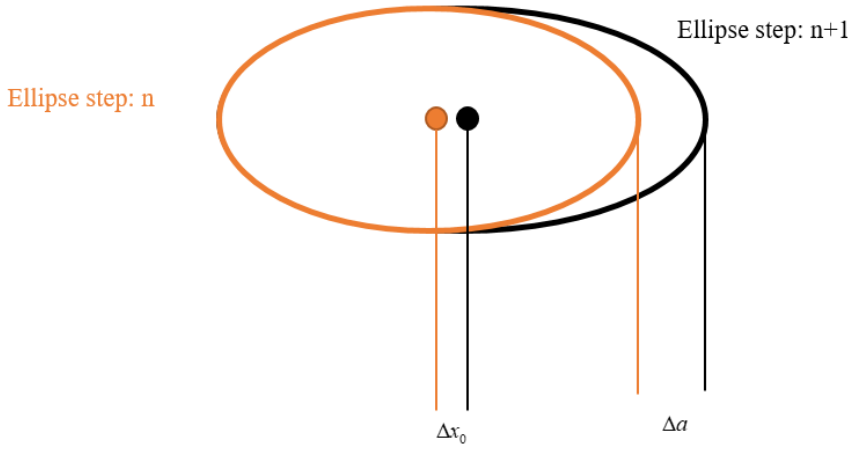
5.3.2 Evaluation and Results

In this subsection, we will evaluate the precision of the A and B parameters estimated by the proposed method. An RMSE is calculated between the predicted values and the ground truth, and a vowel recognition using only lips parameters is carried out.

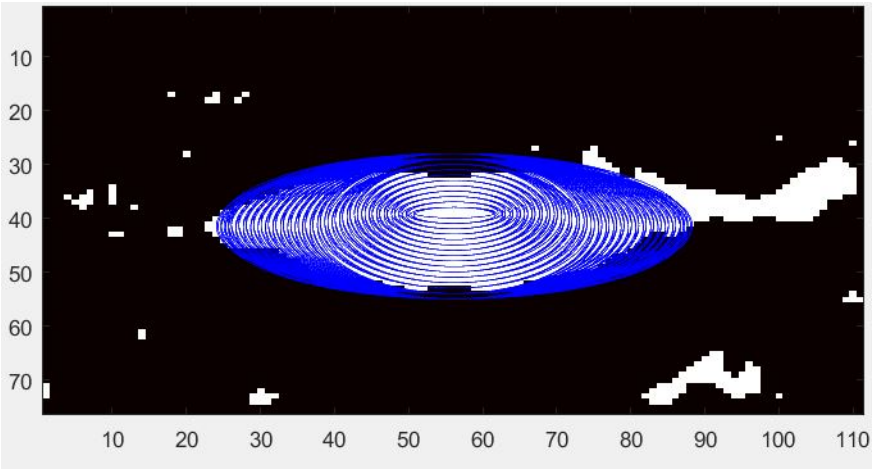
5.3.2.1 Evaluation of the precision of A and B parameters

The adaptive ellipse model efficiently estimates A and B parameters for all kinds of lips shapes, and the performance for three subjects is shown in Fig. 5.32.

The proposed method is evaluated by comparing the estimated values with the ground truth. In Fig. 5.33, we can see that the estimated A and B parameter curves are quite



(a)



(b)

Figure 5.30: Illustration of the ellipse expansion and movement. (a) Expansion and movement along the right direction. From step n to $n+1$, the major axis and center position are updated at the same time. (b) The expansion and movement towards four directions (right, down, left and up). It will stop when it satisfies the stopping criterion.

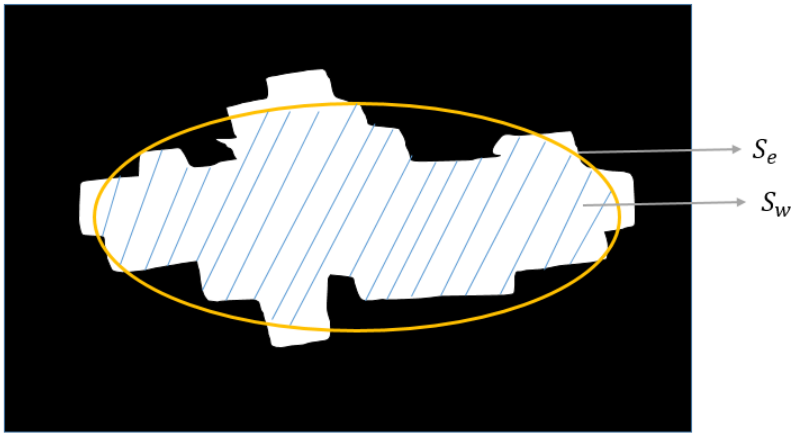


Figure 5.31: Stopping criterion of the Ellipse expansion. The white region is the detected inner lips area. The shaded region is the intersection of white area and orange ellipse. S_e is the ellipse area and S_W is the area of the shaded region inside the ellipse.

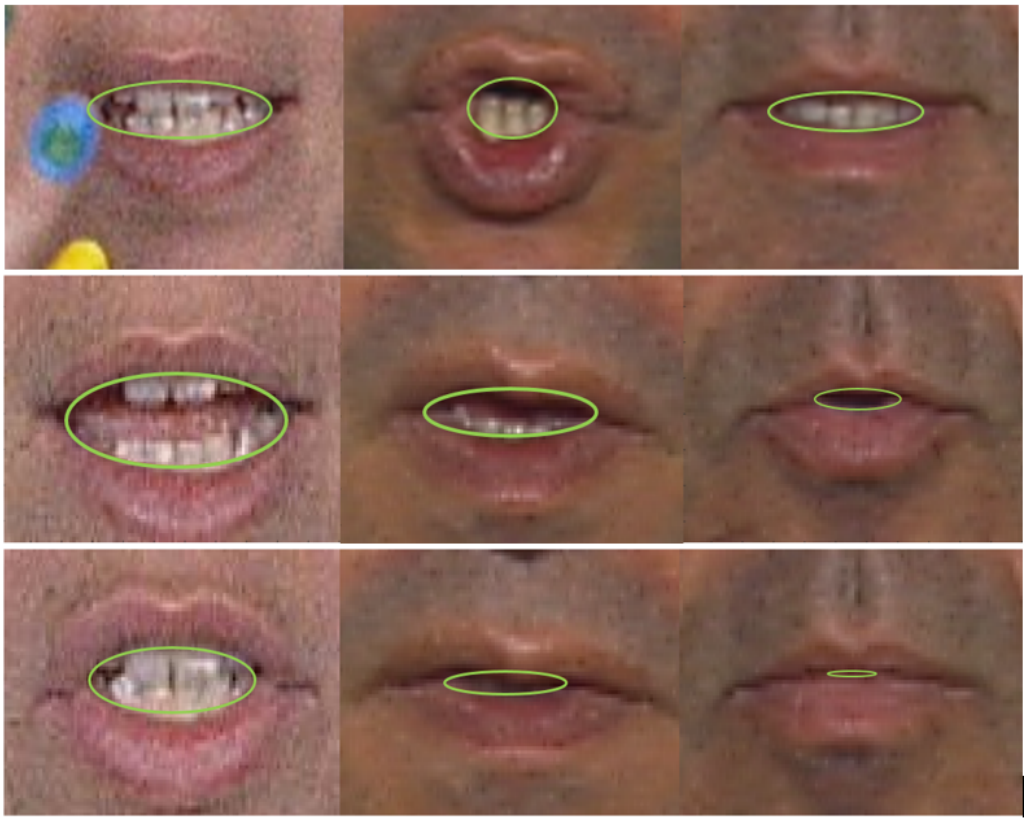
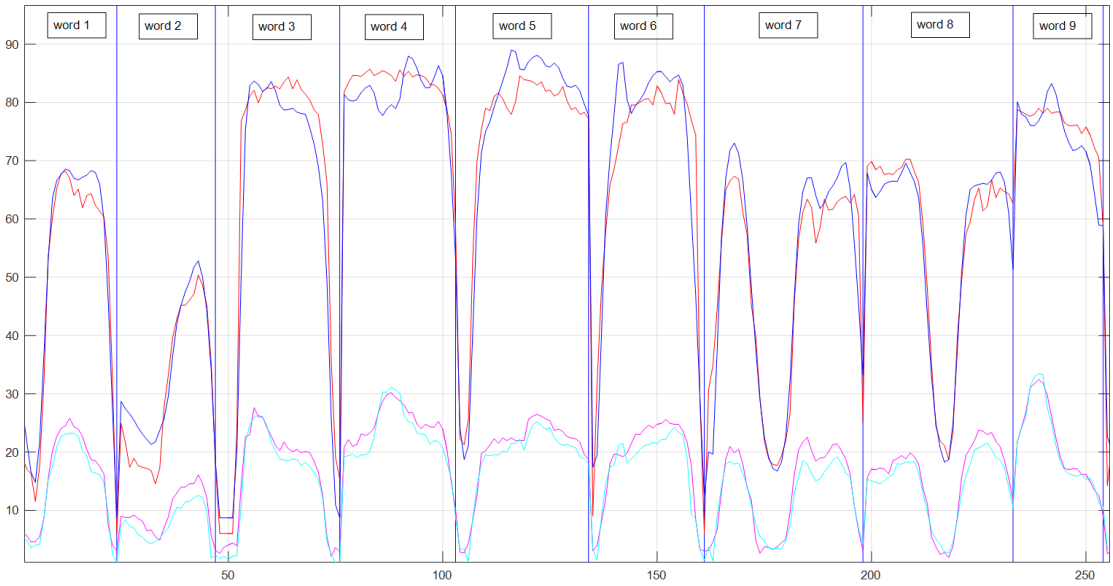
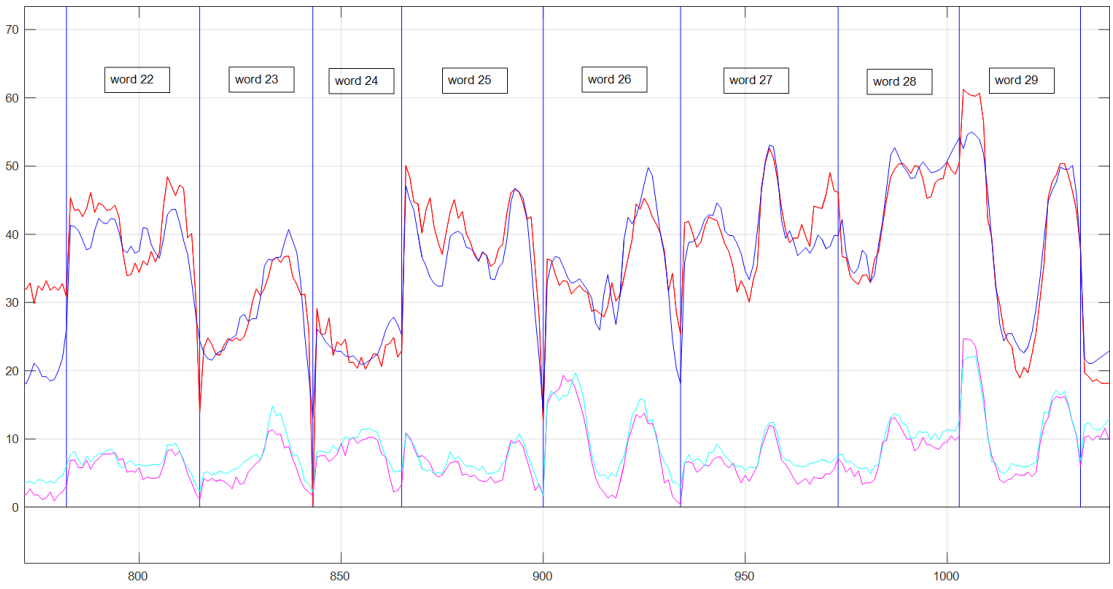


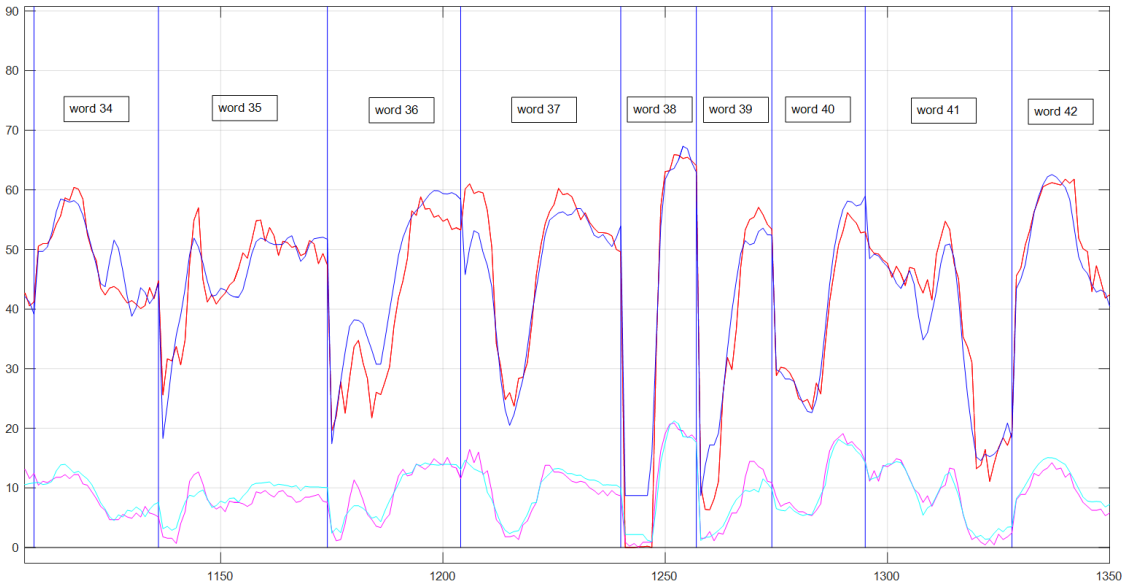
Figure 5.32: Results of the proposed model in different cases (*MD*, *DB* and *ChS* from left to right). The green ellipse is the optimal one which can give a reasonable estimation of A and B parameters for inner lips.


word 1
word 2
word 3
word 4
word 5
word 6
word 7
word 8
word 9
90
80
70
60
50
40
30
20
10
50
100
150
200
250

(a)


word 22
word 23
word 24
word 25
word 26
word 27
word 28
word 29
70
60
50
40
30
20
10
0
800
850
900
950
1000

(b)



(c)

Figure 5.33: Comparison of estimated parameters (A and B) with the ground truth (from top to bottom, the figures correspond to three subjects). The abscissa is the number of images, and y-axis is in pixels. Red curve: the ground truth A parameter. Blue curve: A parameter by the proposed method. Magenta curve: the ground truth B parameter. Cyan curve: B parameter by the proposed method. Blue vertical lines indicate the temporal boundary of each word. For better visualization, we randomly choose several word (small rectangle) intervals of three speakers.

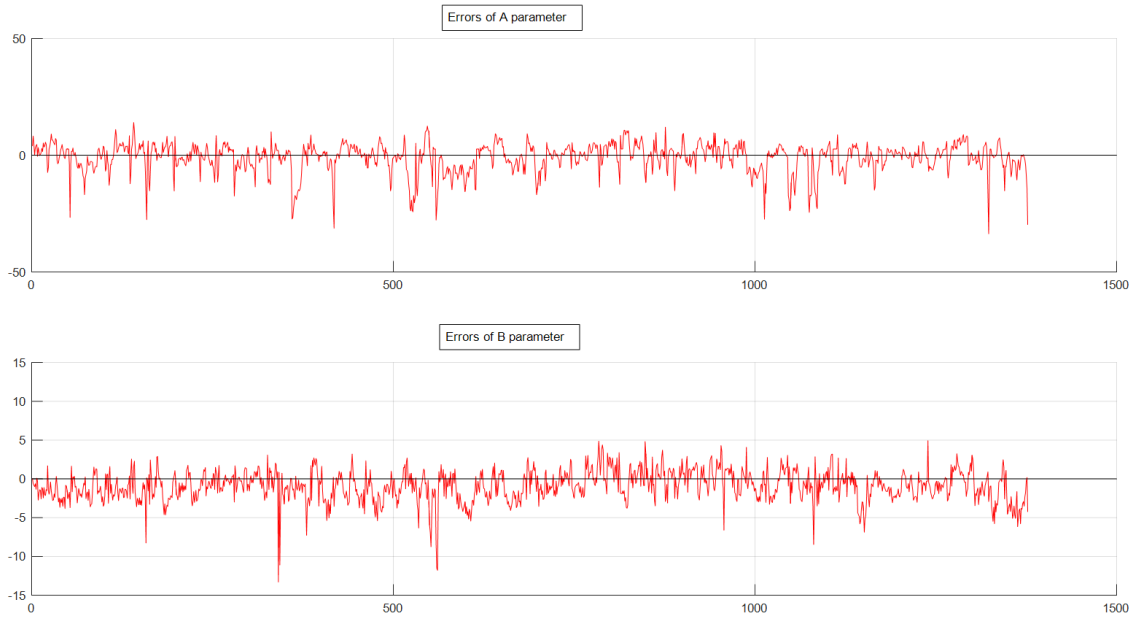


Figure 5.34: Estimation errors (red curves) between (A and B) values of the proposed method and the ground truth for corpus *MD* (total 1377 images in the first repetition).

close to the ground truth curves for all three speakers. In Fig. 5.34, estimation errors of A and B parameters for the CS speaker *MD* are shown. We can see that except few jumped singularities, a global uniform distribution is presented without any evident dependence on the lips shape.

We compare the proposed method with the CLNF approach. Mean values and RMSE of A and B parameters are calculated for these errors. Results are shown in Table 5.3 and Table 5.4. The total RMSE is about 3.37mm for A parameter. It shows a better performance than [194] with the RMSE of 4.5mm. Moreover, the proposed method significantly outperforms the estimation results of CLNF which obtains the RMSE 8.55mm. For B parameter, the RMSE for the adaptive ellipse model is only 0.84mm. It is better than that in [194] with a RMSE 1.0mm, and also outperform the CLNF with RMSE 1.99mm. In summary, our results show a superior performance compared with the state of the art, and also are comparable to the Modified CLNF.

Let's recall that The CLNF is a very powerful algorithm trained on a substantial quantity of images. The main advantage is its insensibility to variations of the luminance, occlusions of the lips and movements of the head. However, CLNF is developed for facial tracking, and it is less precise for the extraction of the lips parameters. From this point of view, the proposed method achieves a very high precision in lips extraction but contains adjustable parameters which depend on the subject and the brightness. Indeed, we sometimes need a high precision extraction algorithm, even if we have to adjust some parameters experimentally.

Table 5.3: Estimation error of A parameter for adaptive ellipse model and CLNF (in mm).

RMSE	*MD*	*DB*	*ChS*	Total error
Adaptive ellipse model	3.54mm	3.50mm	3.06mm	3.37mm
Modified CLNF	4.50mm	3.30mm	2.80mm	3.50mm
CLNF	10.97mm	6.88mm	7.81mm	8.55mm

Table 5.4: Estimation error of B parameter for adaptive ellipse model and CLNF (in pixels and mm).

RMSE	*MD*	*DB*	*ChS*	Total error
Adaptive ellipse model	1.03mm	0.78mm	0.71mm	0.84mm
Modified CLNF	0.6mm	0.5mm	2.0mm	0.5mm
CLNF	2.00mm	2.40mm	2.14mm	1.99mm

Figure 5.35: White filled ellipse determined by the estimated A and B parameters.

5.3.2.2 Continuous CS recognition of vowel based on lips parameter only

In order to further evaluate the performance of the estimated inner lips parameters, French CS recognition based on 13 vowels is carried out using the HMM-GMM recognizer. We use the corpus of first CS speaker with ten repetitions. 80% of the data (randomly chosen) are used for training, and the remaining 20% are used for test (without overlap between the training and test sets). HMM-GMM decoder is built with a standard HMM configuration: context-dependent, three-state, left-to-right, no-skip-phoneme. It is trained with the maximum likelihood estimation based on the EM algorithm. The lips features (A and B parameters) are modeled together with their first derivative. At decoding stage, the most likely image sequence of vowels is estimated by decoding the HMM-GMM state posterior probabilities using the Viterbi algorithm.

Now we show the details of the experiment implementation. An ellipse of white color (with semi-major axis length $A/2$ and semi-minor axis $B/2$) is superimposed on lips region (see Fig. 5.35). PCA is used to extract the good features on raw lips ROI and white filled parametric ellipse lips. For 13 French vowels recognition in CS, 61.8% accuracy on average is obtained using the Modified CLNF, 62% is achieved based on 50 PCA coefficients (explaining 90% variance) in the ellipse, and 59.8% is achieved based on the PCA coefficients in raw ROI. This result is comparable to state of the art [10]. We can see that the recognition score

Table 5.5: Average accuracy of vowel recognition based on the estimated A and B parameters and PCA parameters (30 pca components) in CS. This recognition is conducted by HMM-GMM decoder.

Features	Accuracy
A, B parameters using Modified CLNF	61.8%
PCA on parametric ellipse lips ROI	62.0%
PCA on raw lips ROI	59.8%

using white filled parametric ellipse lips is slightly higher than that using raw images. This confirms the high precision of the estimated A and B parameters. The recognition experiment is subject dependent.

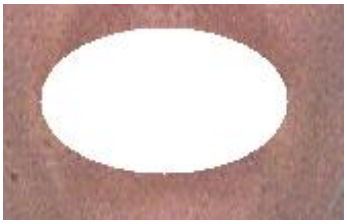


Figure 5.36: An ellipse mask covers the real lips region.

In this study, we may think of that the recognition performance is not only thanks to the white ellipse determined by A and B parameters, since other parts of the image in the ROI also contribute to the recognition performance. In order to verify this point, we mask the whole lips region by a bigger white ellipse for all images (see Fig. 5.36). The vowel recognition decreases to about 30%. This is indeed an interesting result, which may be due to the chin information of the lips ROI or the context-dependent model.

Discussion

As a discussion, we formulate several remarks concerning the above two proposed methods.

(1) In the first proposed method, Modified CLNF combined with the periodical spline interpolation method is based on CLNF lips landmarks. When the real inner lips contour is inside the six inner lips landmarks of CLNF, we cannot expect the proposed method to give a satisfactory inner lips contour (the black inner lips contour in Fig. 5.37). The adaptive ellipse model (the red inner lips contour in Fig. 5.37) can solve this problem since it does not depend on CLNF landmarks.

(2) For the adaptive ellipse model, when the lips have the 'W' shape (see Fig. 5.38), it is difficult to mimic the ellipse of inner lips. However, the proposed method still gives a satisfactory performance.

(3) In both two methods, there are several parameters needed to be optimized by training their data for each subject. Ongoing work is to reduce the subject-dependency of parameters in these methods.

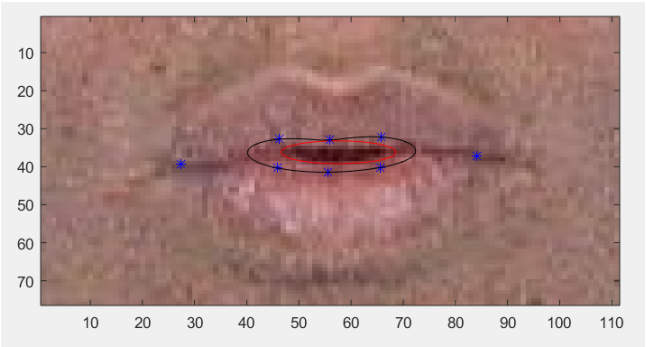


Figure 5.37: An example showing an intrinsic problem of the first proposed method based on CLNF. The blue points are the landmarks given by the CLNF, and the black curve is the inner lips contour obtained by the periodical spline interpolation method. The real inner lips are inside the black inner lips contour, and the red curve shows the estimated inner lips contour by the adaptive ellipse model.

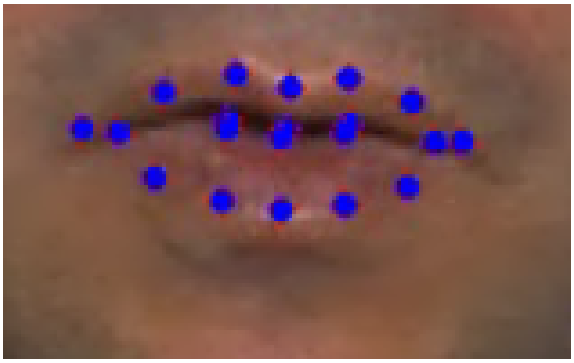


Figure 5.38: An example showing an intrinsic problem of the lips parameter determination. When the lips shape is like 'W' as in this figure, it is difficult to determine the suitable A, B parameters and an ellipse to match the inner lips region.

5.4 Summary

In this chapter, we present two novel automatic methods for estimating the inner lips parameters without any artifice. One is based on the CLNF, which is robust for the facial landmark detection. However, the CLNF presents mistakes in about 41.4% of cases for inner lips tracking. This work aims at correcting CLNF errors by post-processing procedure. We propose two methods to correct B and A parameters, respectively. In the case of B parameter, an efficient method named HD-CTM based on the correlation with a hybrid dynamic template is investigated to detect the outer lower lips position. Then, the inner lower lips position is determined by the back-subtracting of the VLLT. The evaluation of this method on about 4800 images of three speakers confirms its good performance. In fact, RMSE is reduced from

4 pixels (2mm) to 1 pixel (0.5mm). For A parameter, the periodical spline interpolation based on the dilated six CLNF inner lips points is used to estimate the A parameter for round lips. An automatic round lips detector based on DCT coefficient of lips ROI is used to select the third viseme (round lips). This method is tested on 927 round lips images. RMSE is reduced from 21.2mm to 3.5mm using the proposed method. The remaining errors come from the mistaken inner lips landmarks of CLNF.

Another efficient inner lips estimation method based on an adaptive ellipse model is presented. We deal with the parameter extraction of inner lips from video without using any artifice. This method first extracts the inner region of lips with an image processing combined with the single discontinuity elimination and interrupted region filling. Then, an adaptive ellipse model is used to match the inner lips region and gives the best estimation of A and B parameters. The parameter precision is evaluated on 4693 images of three French speakers. The proposed method permits to obtain a RMSE of 3.37mm for A parameter and 0.84mm for B parameter, which outperforms the state of the art. Finally, the CS recognition of 13 French vowels also confirms the superior performance.

Chapter 6

Hand Feature Extraction in Cued Speech

Contents

6.1 Introduction

Hand feature extraction is an essential task in CS recognition. As introduced in Chapter 1, the classical method for hand position and shape feature extraction is based on the artificial marks. Our aim is to get rid of these artifices in this thesis. We notice that in CS, the hand can be seen as a moving foreground when the speaker is coding in the experimental environment. So the CS speaker's hand position tracking can be regarded as a *foreground extraction problem* which can be solved by the Adaptive Background Mixture Model (ABMM) [20]–[22]. After determining the hand position, we build a hand ROI based on it. Then the hand shape feature is extracted using the pixel based methods. In this chapter, we mainly focus on the hand position while the hand shape feature extraction will be introduced in Chapter 8.

In the image processing, the foreground is an integral part of the image, and takes an important advantage in many applications [197]–[200]. A classical and efficient approach for the foreground extraction problem is based on the GMM. A novel method was presented in [201] for the automatic foreground extraction based on the Difference of Gaussian (DoG), which is employed to find the candidate key points. A multi-class statistical model was used in [202] for the tracked objects, but it is a single Gaussian per pixel. In [21], the ABMM was proposed for the foreground (running car) tracking (see Fig. 6.1). ABMMs model each pixel of the image as GMMs instead of modeling all the pixels as one typical distribution. Based on the change of the Gaussian's variance, the Gaussian distribution corresponding to

the background in the image can be determined. It was reported in [21] that this method is robust to the lighting changes of the background.

Therefore, we take advantage of ABMMs [21] and use it for the CS hand position tracking. In Section 6.2, we will give a brief description of the main content of adaptive background mixture model, as well as its application to CS hand position tracking. The evaluation of this method will be presented in Section 6.2.3.

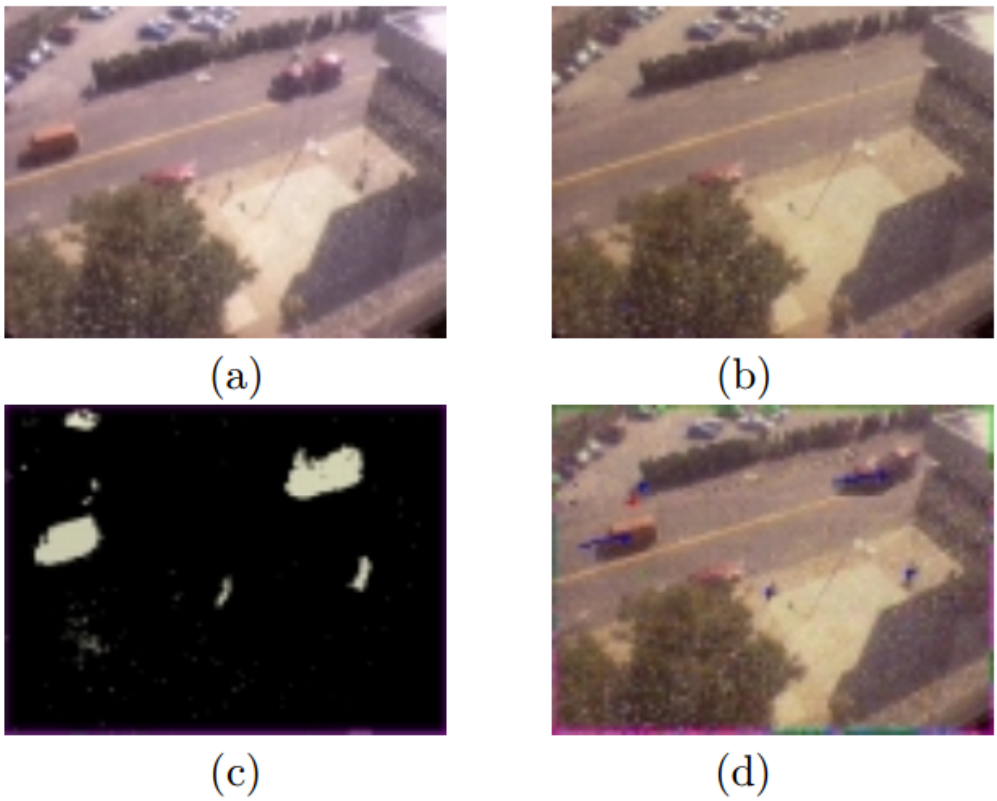


Figure 6.1: The execution of the adaptive background mixture model in car tracking scenario. (a) the current image. (b) the determined background by ABMMs. (c) the detected foreground pixels. (d) the current image with tracked objects (from [21]).

6.2 Adaptive background mixture model for hand tracking

6.2.1 Adaptive background mixture model

The Adaptive Background Mixture Model (ABMM) was proposed in [21] for the real-time segmentation of moving regions in image sequences. In fact, the moving regions can be seen as the foreground, while other regions can be seen as the background. In this model, a simple way is used to evaluate the Gaussians to determine the background, while other pixel values are grouped using the connected components. Then a multiple hypothesis tracker is used to

track these connected components from frame to frame. This process is illustrated in Fig. 6.1. Now we briefly introduce the ABMM. More details can be referred to [21].

Suppose that I is a image sequence, and (x_0, y_0) is a fixed pixel of this sequence. The information of this pixel at time t is

$$\{X_1, ..., X_t\} = \{I(x_0, y_0, j) : 1 \leq j \leq t\}. \tag{6.1}$$

Note that there often exist lighting variations, scene changes, or moving objects in I. We need to model (6.1) by a mixture of more than one Gaussians. More precisely,

$$P(X_t) = \sum_{i=1}^{K} \omega_{i,t} \eta(X_t, \mu_{i,t}, \Sigma_{i,t}), \tag{6.2}$$

where K is the number of Gaussians, η is a Gaussian probability density function, $\omega_{i,t}$, $\mu_{i,t}$ and $\Sigma_{i,t}$ are the weight, mean value and covariance matrix of the i^{th} Gaussian at time t, respectively.

Any new X_t is checked against K Gaussians until a match is found, which means that this pixel value is suited within 2.5 standard deviations of the distribution. If no match is found, the least probable distribution is replaced with a distribution, which has an initially high variance, low prior weight, and takes the current value as the mean value.

Then we study what portion of the mixture model best represents the background in order to determine the Gaussian, which is most likely produced by the background. The pixel values which do not fit the background distribution are considered as the foreground, while others belong to the background.

6.2.2 Automatic hand position and shape ROI determination

Recall that the hand position is obtained by tracking the colors on the subject's hand in the literature, as mentioned in Section 1.2.2. To get rid of these color marks, we now use the above ABMMs to track the hand position automatically.

The hand can be seen as a moving foreground when the CS speaker is coding. In order to guarantee that the hand is the only foreground in the image, we mask the lips (see Fig. 6.2). Therefore, the ABMM is applied to track all the pixels which belong to the hand area, and the gravity center of all these pixels is taken as the hand position. GMMs with five Gaussian components are used to characterize the individual pixel of the image. For each new image frame, the GMM will be updated, and each pixel in the current image is matched with the mixture Gaussian model. If it is matched, this point will be regarded as a background point. Otherwise, it will be classified as a foreground point. Moreover, a rectangle with a suitable size[1] (150×175) for the hand shape is built based on the hand position to form a hand ROI (see Fig. 6.3). The hand ROI is resized to a 64×64 pixel image using cubic interpolation and converted to gray-scale.

[1] This suitable size 150×175 is determined experimentally.

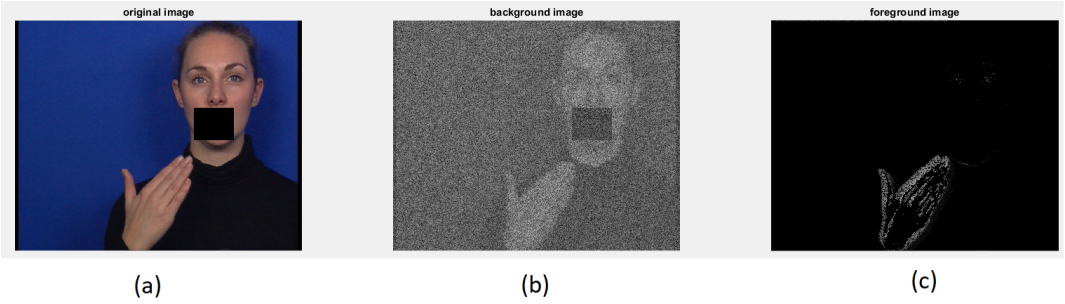


Figure 6.2: Illustration of hand extraction using ABMMs in our data. (a) the raw image with masked lips. (b) the background after applying the ABMMs. (c) white pixels for the foreground. The hand position is taken as the gravity of all the detected hand shape pixels in (c).

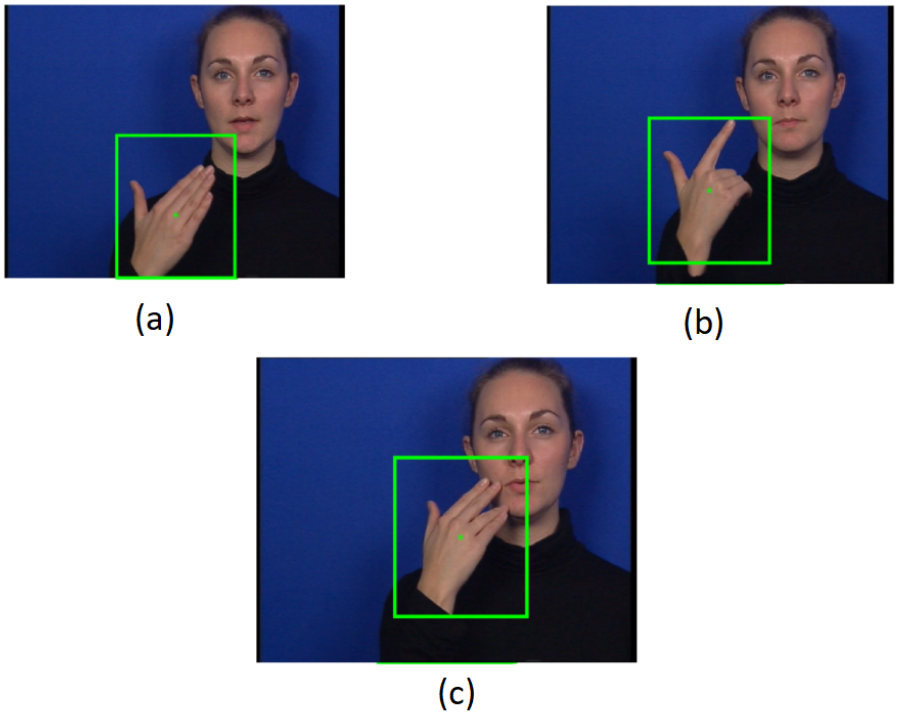


Figure 6.3: Visualization of the hand ROI based on the hand position estimated by ABMMs. The estimated hand position is considered as the center (green point) of this hand ROI (green rectangle).

Table 6.1: Estimation precision (in pixels and mm) of the hand position by the proposed method.

	Hand position (X direction)	Hand position (Y direction)
Mean error	-13.05 (-6.5mm)	23.32 (14.1mm)
Standard deviation	28.11 (11.7mm)	36.64 (18.3mm)

6.2.3 Evaluation of the proposed hand tracking method

A number of 138 sentences from *LM* corpus are used to evaluate the ABMMs for the hand position extraction. First, we compare the obtained hand position with the ground truth hand back position. A good example is shown in Fig. 6.4(a) and Fig. 6.4(c). We can see that the position values estimated by the proposed method are close to the ground truth. A more complicated example is shown in Fig. 6.4(b) and Fig. 6.4(d). However, in some time interval, we can see a significant difference between them. In this case, the estimated value given by the proposed method seems to be difficult to reach the extreme position in both X and Y directions. For all the images in the subset of 138 sentences, the mean error and the standard deviation can be seen in Table 6.1.

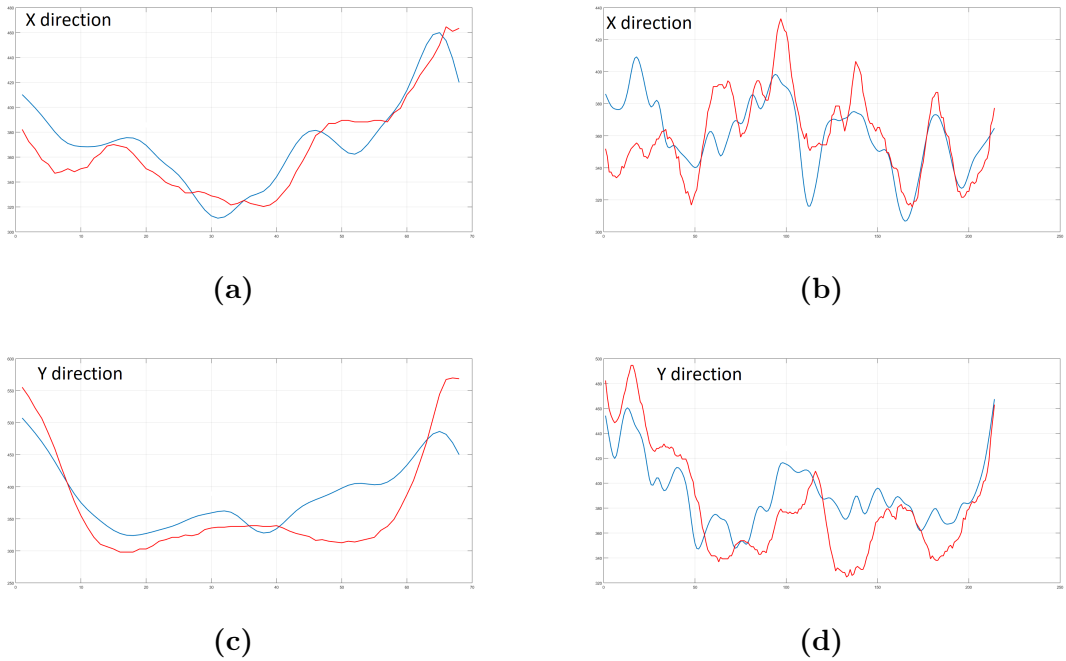
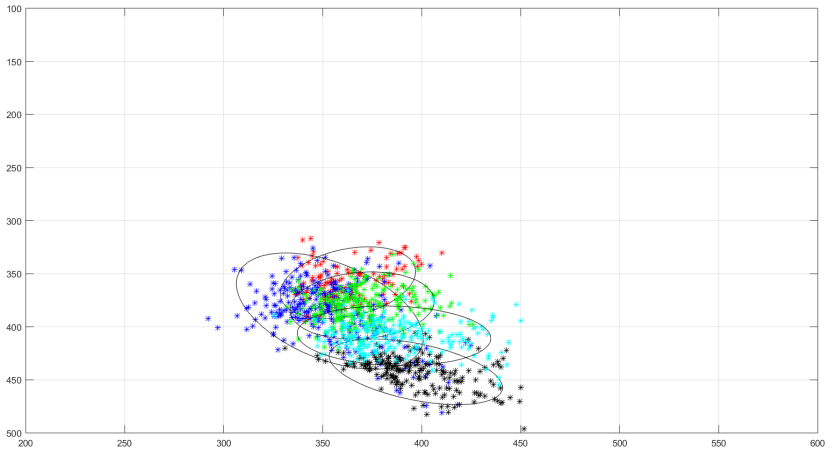


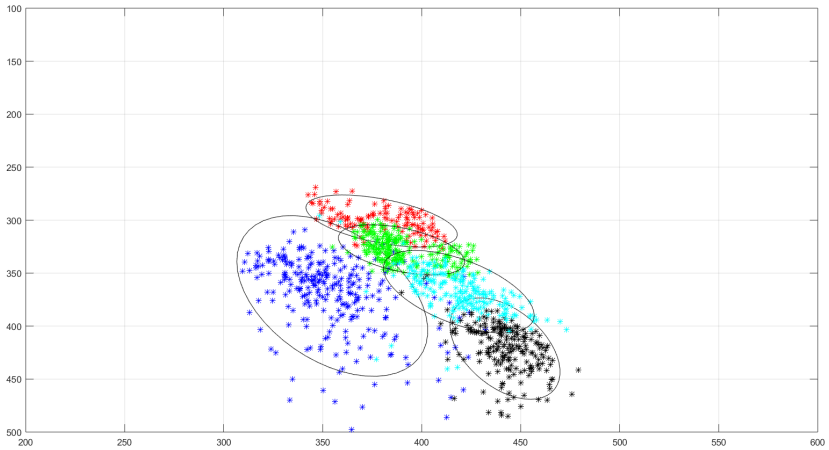
Figure 6.4: Evaluation of ABMMs for hand position estimation. The abscissa is the image frame number, while y-axis is the hand position in pixels. Red curve: the ground truth value. Blue curve: the hand position obtained by ABMMs. (a) and (c) represent the hand position trajectory for one sentence, and (b) and (d) represent the hand position trajectory for the other sentence. (a) and (b) are hand positions in the X direction. (c) and (d) are hand positions in the Y direction.

It is also interesting to see their relative spatial distribution of the hand position. For

this purpose, the five groups of hand position distributions are plotted in Fig. 6.5. As we can see, compared with the ground truth hand position in Fig. 6.5(b), the distribution of the five estimated positions are more confused (see Fig. 6.5(a)). However, we can still see that in Fig. 6.5(a), these five hand position distributions are not in disorder. Their spatial order is basically kept. In another word, apart from the blue points, other four groups of points are toughly distinguishable.



(a)



(b)

Figure 6.5: Hand position distributions with two different extraction methods. (a) the hand positions obtained by the proposed method. (b) the ground truth hand position of the hand back. The ground truth temporal segmentations are used for both cases. Five groups of points correspond to different hand positions. Red points: cheekbone; green points: mouth; black points: throat; cyan points: chin; blue points: side position.

In order to see the influence of the estimated hand position when applying them in the

recognition task, based on the 138 sentences of corpus *LM*, we compare the hand position recognition performances using the estimated hand position with the ground truth hand position (see Fig. 6.6). In this case, we use a simple Gaussian classifier to indicate the divisibility of hand positions. It can be seen in Fig. 6.6 that using the estimated hand position is not as good as using the ground truth hand position in all cases of four different temporal segmentations[2].

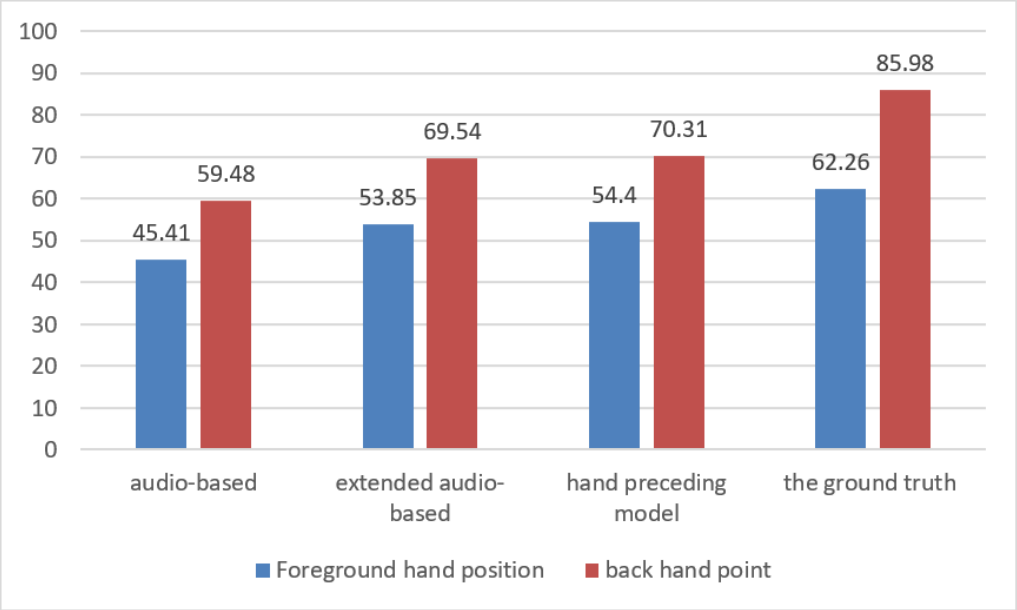


Figure 6.6: Hand position recognition accuracy using the estimated hand position (foreground hand position with blue) and the ground truth hand position (hand back point with red). Different temporal segmentations are used: audio-based, extended audio-based, predicted with hand preceding model and the ground truth segmentation.

6.3 Summary

In this chapter, we describe the application of the ABMMs to extract the hand position of CS speaker. This method is based on a GMM foreground and background extraction approach [20]–[22]. In fact, the speaker's deformable hand is regarded as the foreground in a video. Five Gaussian models are used to characterize the foreground and background pixels in an image. The gravity center of the detected foreground pixels is considered as the hand position. In order to evaluate the performance, we first compare the estimated hand positions with the ground truth, and then perform a hand position recognition using Gaussian classifier. In case of the ground truth temporal segmentation, the recognition results show that using our estimated hand position gives 62.26% while the ground truth hand back position attains 85.98%. Even though this automatic hand position extraction method cannot reach the performance of the ground truth, it permits to track the hand position with a certain precision. On the other

[2] Audio-based temporal segmentation is the temporal segmentation of the raw audio speech signal. The extended audio-based segmentation and that based on the hand preceding model will be introduced in Chapter 7.

hand, based on the proposed method, a hand shape ROI can be built, and the hand shape feature extraction will be introduced in Chapter 8.

Chapter 7

Temporal Study of Hand Movements in Cued Speech

Contents

7.1 Introduction

In CS, lip reading and hand coding work together to make the syllable visible. Hand and lips movements in CS are coherent and complementary to realize an efficient communication system. However, they follow their own movement rules. The hand movement is more related to the speech syllabic cycle, while the lips movement is more related to the phoneme production. These two movements are not well synchronized. Attina et al. [37], [203] investigated the temporal organization of hand and lips movements, and found that the hand reaches its target roughly 200ms (between 171 and 256ms) before the vowel being visible at lips.

The supervised automatic CS recognition system needs a proper alignment and annotation of the data streams. More precisely, in the training step, the boundaries should be provided to indicate which segment in the stream corresponds to a given phoneme. CS recognition is a multi-stream task with the asynchronous data streams. This means that we must realize a proper temporal segmentation for each stream. Note that the lips movement is relatively well

synchronized with the acoustic signal. It is possible to realize the automatic segmentation of lips via the acoustic signal which offers a comparably good quality (see Section 2.3.2). However, performing an automatic segmentation of the hand movement is still an open question, and it is also a challenge in this thesis.

As far as our knowledge, no previous work has explored the automatic method to obtain a proper temporal segmentation of hand movements in CS based on the corpus without any artificial mark. In fact, the prior work [38] investigated the temporal segmentation of hand position stream by using the color marks on subject's hand. The temporal segmentation of hand position is obtained based on the Gaussian modeling of hand positions and the velocity of hand position movement. However, this method needs both position on the hand back and the target finger position, and all these points are given by different color marks [38]. It is not possible to directly apply this method to the database without any artificial mark. We may think to use the deep learning methods like LSTM (see Chapter 3) to capture the asynchrony between these different streams automatically. However, it needs a large amount of training data which is not available in our case.

In this chapter, we focus on the temporal segmentation of the hand movements[1] from a new perspective. It is to determine the instant of the target hand position which indicates the realization of a vowel, and the intervals in which the hand shape is well formed to indicate a consonant. We carry out three studies to efficiently resolve the temporal segmentation problem of the hand movement, which will be used for the automatic CS recognition in Chapter 8.

(1) First, we carry out a detailed analysis of the hand movement in CS, and show that hand movements are organized syllable by syllable. In fact, the hand shape is prepared during the hand moving towards its target position, and reaches its most precise form at almost the same time as the hand position reaching its target. This mechanism allows us to define the best instant to locate the target position for the vowel, as well as the best instant to determine the hand shape for the consonant.

(2) Based on (1), we perform a manual temporal segmentation of a subset of corpus *LM* to establish a preceding model. This model is expected to describe the relationship between the target instants for the hand position and the corresponding acoustic signal. It allows us to predict the target moments for hand position, and segment the hand position stream based on the audio signal. The evaluation shows that a much better vowel recognition score is obtained than using the audio based segmentation.

(3) We are also interested in determining the best instant for locating the hand shape which indicates a consonant. Our analysis shows that the middle instant of this interval is about 60ms before the middle instant of the consonant on the acoustic signal. This allows a segmentation of the hand shape stream based on the audio segmentation.

[1] Here, the hand movements contain the movements of hand position and hand shape.

7.2 Hand movement characteristics in Cued Speech

7.2.1 Lips and hand asynchrony phenomenon

The asynchrony problem of three streams (lips, hand shape and hand position) in CS is a challenging issue. Recall that the hand movement precedes acoustic sound realization 200ms on average for a phoneme [37]. Now we illustrate the hand advance phenomenon by an example of CS where the speaker utters "un petit". In Fig. 7.1(a), the CS speaker points to the cheekbone position on the face for the vowel [ø] with the hand shape corresponding to [p]. The red line corresponds to the instance in the acoustic signal which is neither the position for [ø] nor [p]. In Fig. 7.1(b), the hand starts changing its shape for the consonant [t] while the mouth is still closed for [p] at an instant (see the red line) before the acoustic burst. In Fig. 7.1(c), the speaker shows round lips corresponding to the vowel [ø]. However, the hand shows the shape corresponding to the subsequent [t] consonant and the hand position is already in vowel [i].

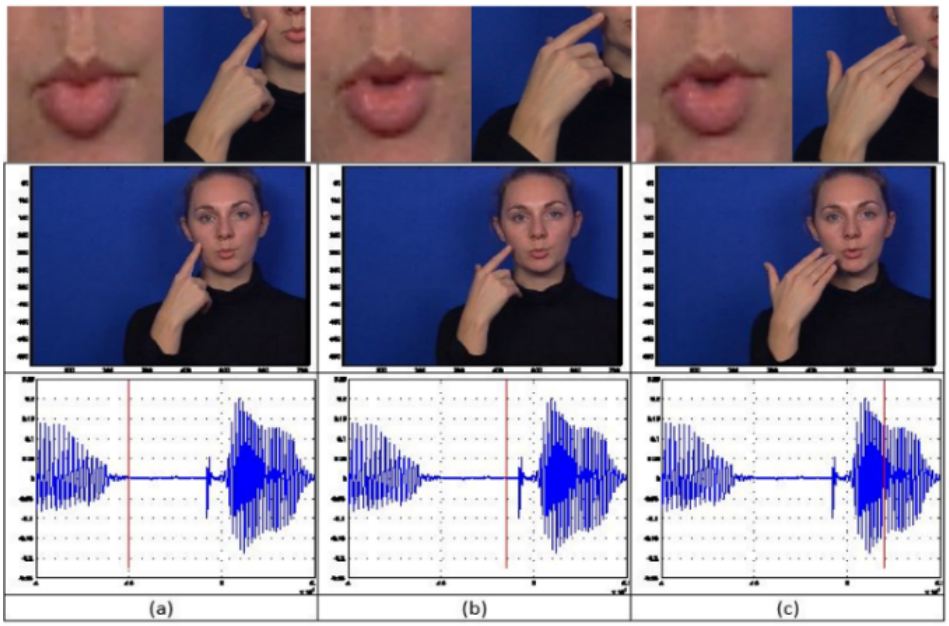


Figure 7.1: Illustration of the asynchrony phenomena in the CS lips-hand movement. The context of the [œ̃ p ø t i] sequence is extracted from the French sentence *un petit*. Top is the lips and hand zoomed from the whole images in the middle row. These images are taken at different instants of the speech signal (bottom) indicated by red lines.

7.2.2 Analysis on temporal organization of the hand movement

The goal of this subsection is to specify several fundamental rules concerning the hand movements in CS, as well as the asynchrony problem between lips and hand movements.

Without loss of generality, we only consider the syllables which consist of a consonant followed by a vowel (i.e., in the form of CV). When the speaker organizes the sentence syllable by syllable, at each target instant, the hand position indicates the concerned vowel while the hand shape indicates the corresponding consonant of the syllable. The movements of hand positions and hand shapes seem to be organized in such a way that the hand shape reaches its final shape at almost the same time when the hand position reaches its target for a vowel. In the duration of two target positions, the hand prepares its shape to indicate the next consonant.

The principle of the temporal organization of hand movement in the unit of the syllable is shown in Fig. 7.2. The rectangles on the top row indicate the time intervals in which the hand reaches its target position. For hand position movements, these intervals are relatively short (about 60ms on average, three images). The rectangles on the bottom row indicate the time intervals in which the hand prepares its shape to indicate the consonant. During this period, the hand shape is almost formed, but the hand continues moving and rotating. Therefore, these intervals are relatively long (about 200ms). In fact, the hand begins to leave the target position once a syllable codes completely, and the fingers move quickly to prepare the next hand shape. We can see that each rectangle on the bottom starts immediately just after the previous syllable.

The definition of different parameters

As described in Section 7.2.1, in CS, the hand reaches its target position before lips pronouncing the corresponding vowel. We are now interested in how long time the hand precedes the lips movement. Now we detailedly study this phenomenon and propose a hand preceding model, which can be used to predict the temporal segmentation of hand movement automatically.

We first give some definitions about the different parameters. As shown in Fig. 7.3, for a vowel in a syllable, the instant corresponding to the middle instant of the vowel in the acoustic speech signal is denoted by t_v, and the target instant for the hand position movement is denoted by t_{tar_v}. Compared with t_{tar_v}, the advance of t_v is denoted by Δ_v. D_v (about 60ms) is the time interval in which the hand reaches its target position, while D_{c_0} is the time interval in which a hand shape is nearly formed but continues moving and rotating. We define the *hand preceding time* as the time difference between the hand target instant and acoustic target instant. In other words, in case of vowel, the hand preceding time (in ms) is

$$\Delta_v = t_v - t_{tar\_v}. \tag{7.1}$$

For the consonant in a syllable, the middle instant of the consonant in the acoustic signal is denoted by t_c. As we just mentioned, t_c necessarily precedes t_v. We consider that the complete hand shape is formed at the same time as the target position t_{tar_v}. However, this best hand shape does not correspond to a single instant but a certain duration naturally. Moreover, this time interval is before t_{tar_v} because after this moment the hand shape begins to change immediately. Let D_c denote the time duration corresponding to the best interval to

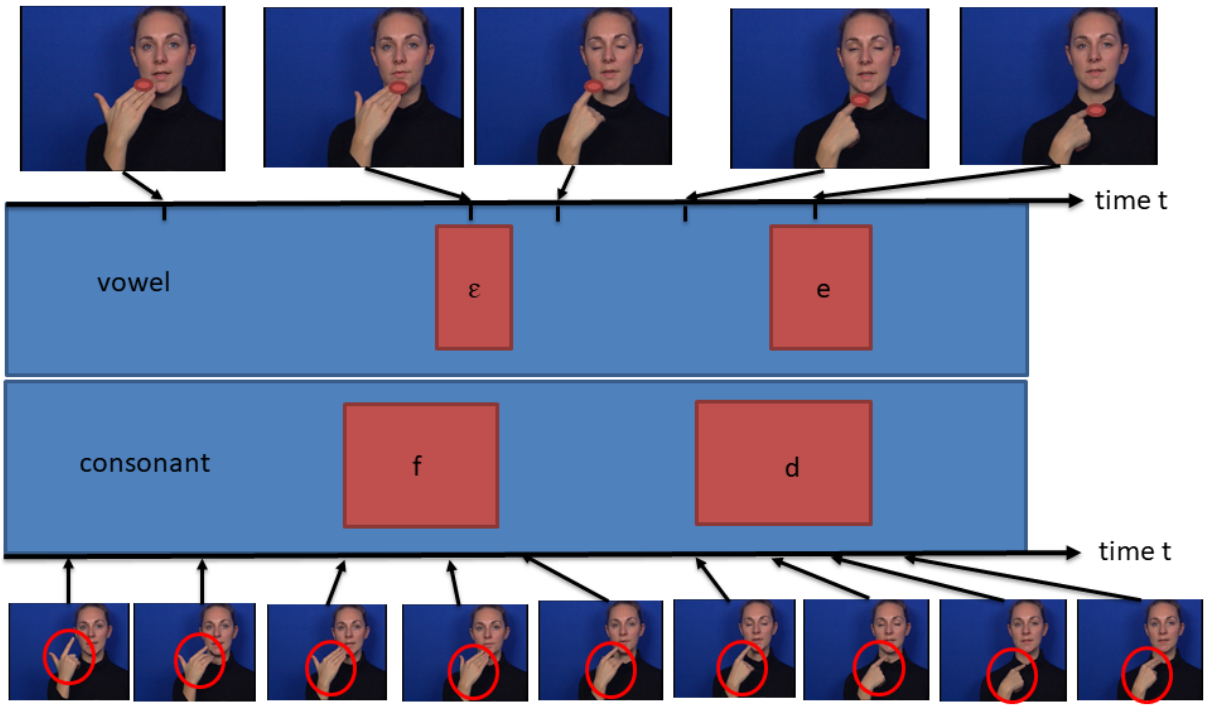


Figure 7.2: Temporal organization of the hand movement (hand shape and hand position). The concerned two syllables are [f ɛ] and [d e]. The top row shows different hand positions, while the bottom row shows different hand shapes. Two middle rows present different instants of the corresponding vowels and consonants.

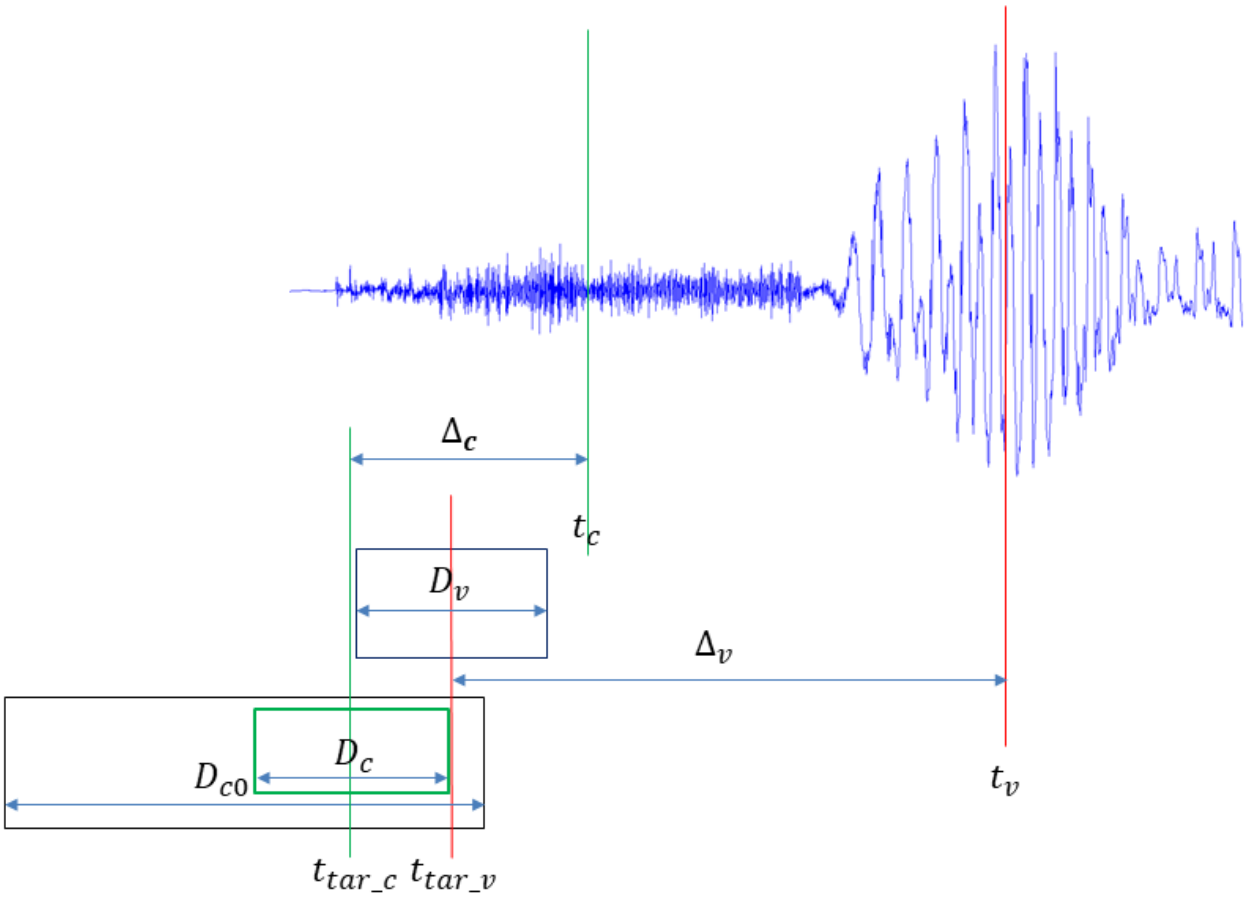


Figure 7.3: Definitions of the target instant and hand preceding time. The blue signal is the audio speech signal for a syllable. The red line t_v indicates the acoustic target instant for this vowel, and the green line t_c indicates the target instant for this consonant. The red line t_{tar_v} indicates the target instant for hand movement: at this moment, the hand reaches its target position for this vowel. The green line t_{tar_c} indicates the best instant for identifying a hand shape corresponding to a given consonant.

identify a given hand shape. The middle of this time interval will be denoted by t_{tar_c} which is the target instant for the hand shape realization.

7.3 Experimental setup

In this section, we will introduce the database for the temporal segmentation of hand stream. The measurement of hand preceding time will also be presented. Using these measurements, a statistical model is built which indicates the hand preceding time according to the vowel instant in a sentence.

7.3.1 Database

The database is composed of two female normal hearing CS speakers *LM* and *SC*. As in Chapter 2, the corpus *LM* is recorded without using any artificial mark, and the corpus *SC* was recorded before with artificial marks (but the marks are not used). The speaker *LM* pronounces and codes a set of 238 French sentences in CS derived from a corpus in [40], [44]. Each sentence is repeated twice resulting in a set of 476 sentences. The corpus *SC* is made of 267 sentences which come from the large database in [38]. We take a subset of the whole database to build the hand preceding model, which contains 138 sentences with 88 short sentences and 50 long sentences from corpus *LM* (totally 1066 vowels), and 44 short sentences (196 vowels) from corpus *SC*.

7.3.2 Measurement of the hand preceding time

This experiment is to establish the relationship between the hand preceding time and vowel instant in sentences. For this purpose, it is necessary to first measure the instant where the hand reaches a vowel target precisely.

It should be mentioned that the accurate automatic CS hand finger tracking in 2D image is still an open problem. Therefore, to obtain the accurate hand movements data, we manually track the 2D position of the target finger of the CS speaker (see Fig. 2.12). The target finger which directly points to the vowel position allows the CS reader to understand what the CS coders want to express. However, the target is not always realized by the same finger since the hand shape is variable during the coding process. We choose the position in the following way: the 2D position of the index finger is used if no middle finger appears. In order to constitute a solid base for studying the hand movement, the point on the hand back (see Fig. 2.12) is also tracked. One advantage is that this hand point is always visible in the CS coding process, so that all the points on the hand back can be collected without any interruption.

To make the manual segmentation easier, we calculate a hand movement velocity curve from the coordinates of the tracked points. This calculation can be done using the target

finger point or the hand back point. In our case, we use the latter one. This procedure was also used in [38] but with blue marks on the speaker's hand. The particularity is that we first apply a suitable smoothing method (spline function with $p = 0.1$ [193]) to x coordinate and y coordinate. This smoothing method gives two speed rates in x and y directions, denoted as v_x and v_y. The velocity of the hand movement in the direction of its trajectory is then calculated as:

$$v = \sqrt{v_x^2 + v_y^2}. \tag{7.2}$$

In Fig. 7.4, an illustration of the velocity of the hand movement is given. Note that we do not need to know the exact value of this speed rate, and only its evolution is sufficient to help us in the segmentation task. This velocity curve of the hand movement offers an important indication to localize the vowel target instants in sentences since in general, the hand moves rapidly between two target positions and moves slowly or even stops at target positions. Thus the minimum value of the curve indicates that the hand reaches its target position. However, not every minimum value of the speed rate corresponds to a vowel target. This phenomenon makes the automatic temporal segmentation more complex. Even in some cases, the vowel target does not correspond to a minimum speed especially in the case of side position.

A manual temporal segmentation of vowels and consonants for each sentence is accomplished by using the movie editor Magix [204], [205]. We translate the velocity curve using (7.2) into a speed rate signal so that it can be visualized in Magix with a perfect synchrony (see the second row in Fig. 7.5). In this way, video, hand movement speed rate and sound can be visualized in one window which benefits the segmentation process a lot. The temporal target interval which contains several images around the hand target position is considered in this segmentation.

The above manual segmentation for vowels allows localizing the target instant t_{tar_v} of the hand movement for each vowel. Thanks to the audio based segmentation localizing each vowel in the acoustic signal, the hand preceding time Δ_v can be calculated for all vowels of the sentences (totally 1066 vowels for LM) by (7.1).

7.4 Hand preceding model for the hand position movement

In this section, we detailedly describe how to establish a hand preceding model for the vowels coded by hand positions [206]. It allows us to know the hand preceding time according to the vowel instant in the sentence. Then we propose a temporal segmentation method for the hand position stream.

7.4.1 Hand preceding model for vowel

The hand preceding time in function of vowel instant of 138 sentences for the subject LM is shown in Fig. 7.6(a). We align all the sentences by their end instead of their beginning

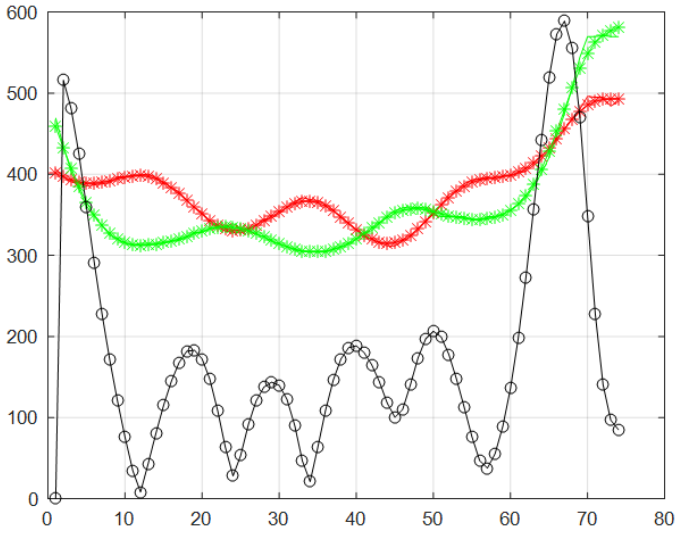


Figure 7.4: Hand movement speed rate of the sentence *Je suis à bout*. The abscissa indicates the image frame number, and y-axis is the hand position. The red curve is the hand position in x coordinate, and green curve in y coordinate. Black curve with circle dots shows the hand movement speed rate.

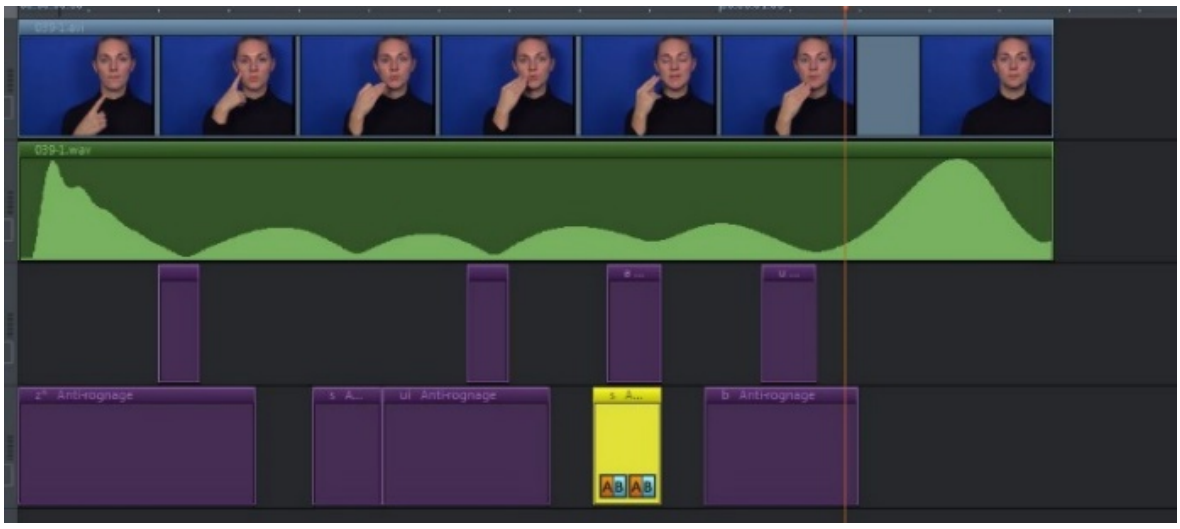


Figure 7.5: The software Magix used to study the hand movement in CS. This example shows the sentences *Je suis à bout*. The first row is the image sequence of this sentence. The second row shows the hand movement speed rate curve. The third row gives the hand target position of vowels: each purple temporal rectangle presents an interval in which the hand reaches its target position to indicate one vowel. The last row gives the temporal interval in which the hand shape is more or less formed to indicate a consonant but the hand continues its movement and rotation.

since we observe a common law of the relationship between the hand preceding time and the vowel position for all the sentences. From the beginning of a sentence to a certain instant (about one second before the end of the sentence), the hand preceding time seems to remain a constant. More precisely, the statistical repartition of the hand preceding time has almost no change from the beginning to about one second before the end. Then the hand preceding time decreases when the vowel instant approaches the end of the sentence. By aligning all the sentences by the end, this phenomenon becomes very evident. Indeed, the distributions of short sentences and long sentences are superposed entirely at the end of the sentence. Besides, the vowels from speakers *LM* and *SC* follow the same repartition as shown in Fig. 7.6(b).

Based on these observations, we build the *hand preceding model*, which contains two parts. The first part is the mean value of the hand preceding time (i.e., 139ms in our case) of all the points from the beginning of the sentence to a turning point. After the turning point, the second part is a linear model which can be obtained by linear regression (a slope of -0.213) of the rest data. The turning point corresponds to the intersection of these two straight lines. In our case, it situates at about 0.88s before the end of a sentence. Based on the above analysis, we see that the hand preceding model fits all the sentences for two subjects.

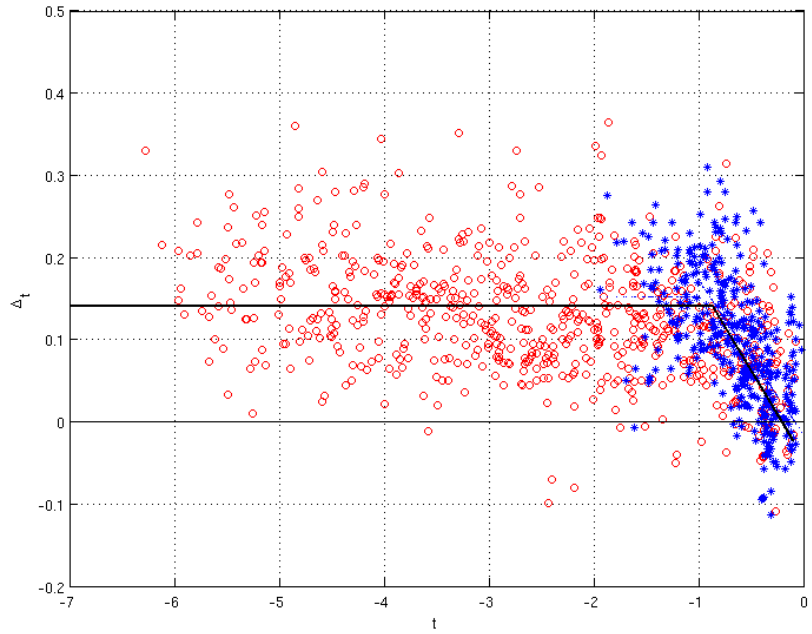
We now propose a temporal segmentation method for the hand position based on the hand preceding model. More precisely, based on the audio based segmentation, each temporal segment for a vowel will be shifted by Δ_t according to the instant of the vowel in the sentence. The Δ_t is calculated using the hand preceding time. An illustration of the segmentation using the hand preceding model is shown in Fig. 7.7.

The audio-extended segmentation

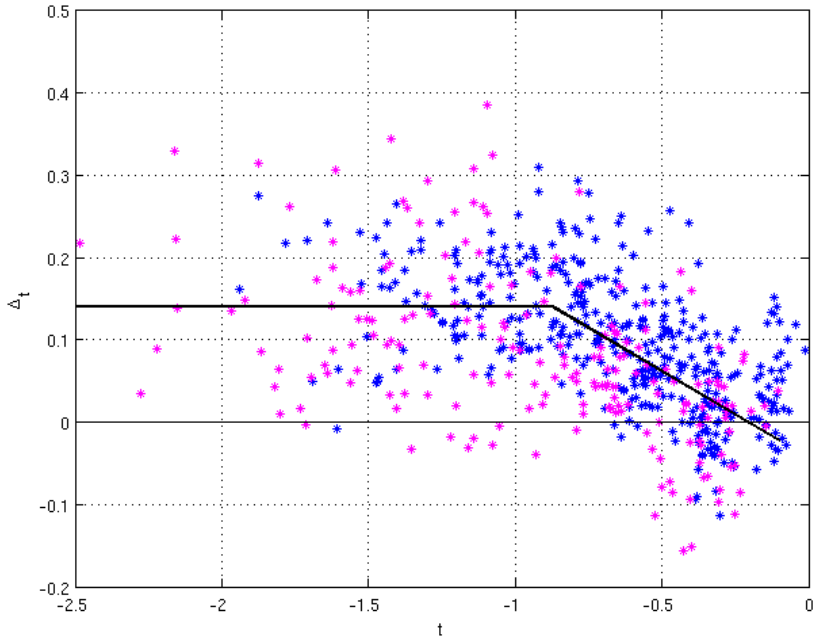
Apart from the above temporal segmentation by the hand preceding model, a simple rule based segmentation called the *audio-extended segmentation* is also proposed.

Recall again that Attina et al. [37] focused on the temporal organization of manual cues of hand and lips. They showed that the instant of the hand reaching its target can vary roughly from 171ms to 256 ms, and is equal to the time length of one syllable on average. Based on this, we investigate a simple procedure to derive the temporal segmentation of the hand stream from the temporal segmentation of the audio stream. As shown in Fig. 7.8, the left boundary of each phoneme (except the first phoneme) is extended to the beginning of the previous phoneme. The boundary of the first phoneme in each sentence keeps it as the audio based segmentation.

In the following experiments, the audio-extended segmentation will be used to compare with the one estimated by the hand preceding model.



(a)



(b)

Figure 7.6: Hand preceding time distribution and hand preceding model. The abscissa is the vowel instant in a sentence. All sentences are aligned at the end, where the instant is 0. Y-axis: the preceding time Δ_v (in seconds). (a) The red circles show the distribution of the 50 long sentences, and the blue stars show the 88 short sentences. The black curve shows the hand preceding model. (b) The blue stars show the 88 short sentences for the subject *LM*, and the magenta stars show the 44 short sentences for the subject *SC*.

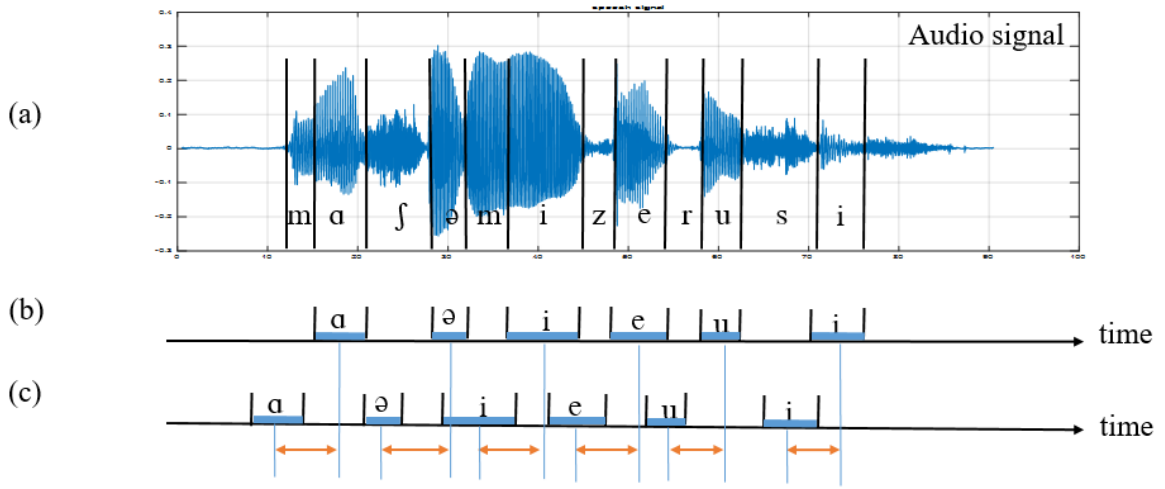


Figure 7.7: Application of the hand preceding model in temporal segmentation. The predicted Δ_t (orange interval) is shown for the sentence *Ma chemise est roussie*. (a) the audio signal. (b) the audio based segmentation. (c) the segmentation predicted by the hand preceding model.

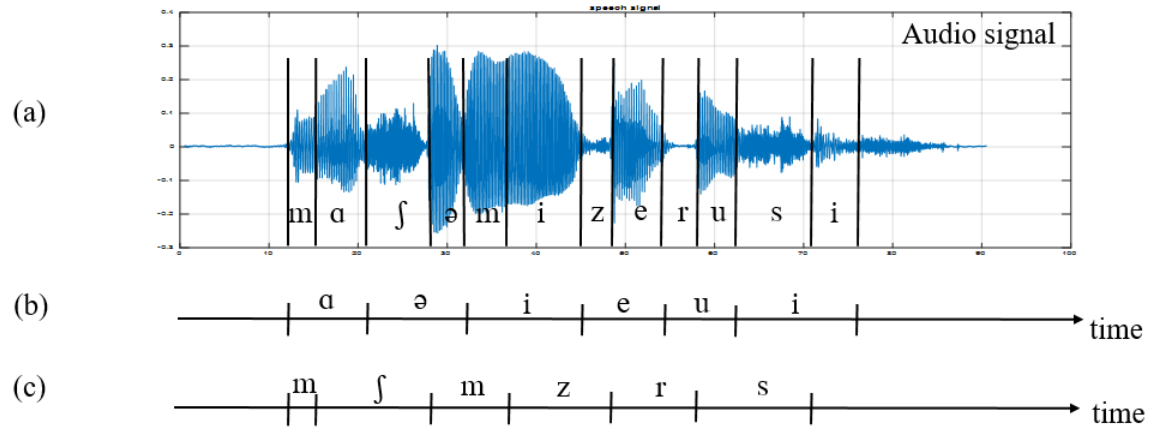


Figure 7.8: The audio-extended segmentation of the sentence *Ma chemise est roussie*. (a) the audio signal with its corresponding phonemes and temporal segmentations. (b) the audio based segmentation. (c) the audio-extended segmentation.

7.4.2 Evaluation of hand preceding model for hand position

To evaluate the performance of the proposed hand position segmentation using the hand preceding model, we compare the predicted temporal segmentation with the following three temporal segmentations:

(1) The ground truth segmentation. During the manual segmentation of the corpus, the target interval for all the vowels of the sentences is determined manually. It constitutes a golden reference for the hand position recognition in CS.

(2) The audio based segmentation. In the history of CS studies [10], [11], the audio based segmentation was applied to the hand position stream. It is necessary to compare the performance of the audio based segmentation with the predicted one in order to see the potential benefits of the proposed method.

(3) The audio-extended segmentation. This simple rule based segmentation was introduced in Section 7.4.1. To guarantee the same experimental conditions, we let the audio-extended segments have the same length as the audio based segmentation.

We compare the boundaries of different segmentations directly and visualize the hand position distribution in a 2D image. Besides, we apply the Gaussian classifier and LSTM to the hand position recognition using different temporal segmentations.

7.4.2.1 A direct comparison with the ground truth

To evaluate the efficiency of the hand preceding model, it makes sense to visualize the predicted temporal segmentation and the manual ground truth segmentation.

In Fig. 7.9, the predicted temporal segmentation of a vowel is compared with the manually determined hand position temporal target interval, as well as the audio based segmentation. From the second and the third rows, we see that they have a coherent match in most of the cases. But there still remain some errors (see the 4^{th} vowel in Fig. 7.9(b)). It is normal that the hand preceding model replaces a large statistical repartition only by its mean value. On the other hand, we observe that the audio based vowel temporal segmentations are far different from other two segmentations, especially at the beginning of sentences. Note that, the predicted segmentation keeps the same length for each vowel as the audio segmentation, while the length of the segment can vary in the case of manual hand based segmentation.

It is useful to see the hand position spatial distributions using different temporal segmentations. We expect that the better temporal segmentation should present a figure which makes five hand position distributions distinguishable and separable.

The distribution of the finger points of 396 vowels (88 short sentences in the corpus LM) using four segmentations is shown in Fig. 7.10. We see that the Gaussian ellipse becomes more and more distinguishable from (a) to (d). The points in (a) have large parts of overlaps

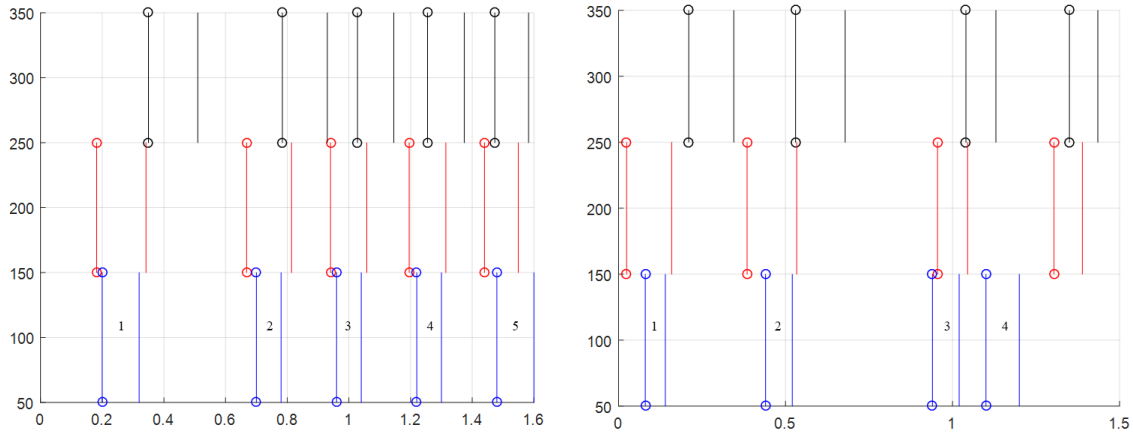


Figure 7.9: Different temporal segmentations of two sentences. For simplicity, the vowel number is only marked in the third row. For each vowel, a line with two circles represents its beginning, and a line without any circle represents its end. Top row (black lines): the audio based segmentations. Middle row (red lines): the predicted segmentations. Bottom row (blue lines): the ground truth segmentations.

for five positions, while the points in (d) are significantly separable. Some improvements in (b) are observed compared with (a). In particular, we find that the throat position and chin position are divided. More importantly, we see that the distribution of the Gaussian ellipse in (c) is very close to that in (d) with only a few intersections between the ellipses. These distributions efficiently illustrate the satisfied performance of the hand preceding model.

7.4.2.2 Hand position recognition in CS using the sub-database

To further evaluate the proposed segmentation method, we apply all these four segmentations to the hand position recognition. A simple multi-Gaussian model is used as a recognizer on this sub-database since the aim is just to evaluate the performance of the predicted segmentation. Five Gaussian models are firstly trained for the five positions. Let $X = (x, y)$ denotes the mean hand position value of a sequence of images in the target time interval. Given any mean hand position X, we calculate the probability of X for each model, and the model with the highest likelihood value is the identified class.

We show the recognition results in Fig. 7.11. These results are obtained using both the finger position and the point on the hand back. Moreover, we compare these two ground truth cases with the automatic tracked hand position estimated by the ABMMs.

Fig. 7.11 gives us rich and interesting information. First, we examine the hand position recognition results using the target finger position. We see that in the case of the ground truth temporal segmentation, the highest score of 96.9% is achieved. This constitutes the golden

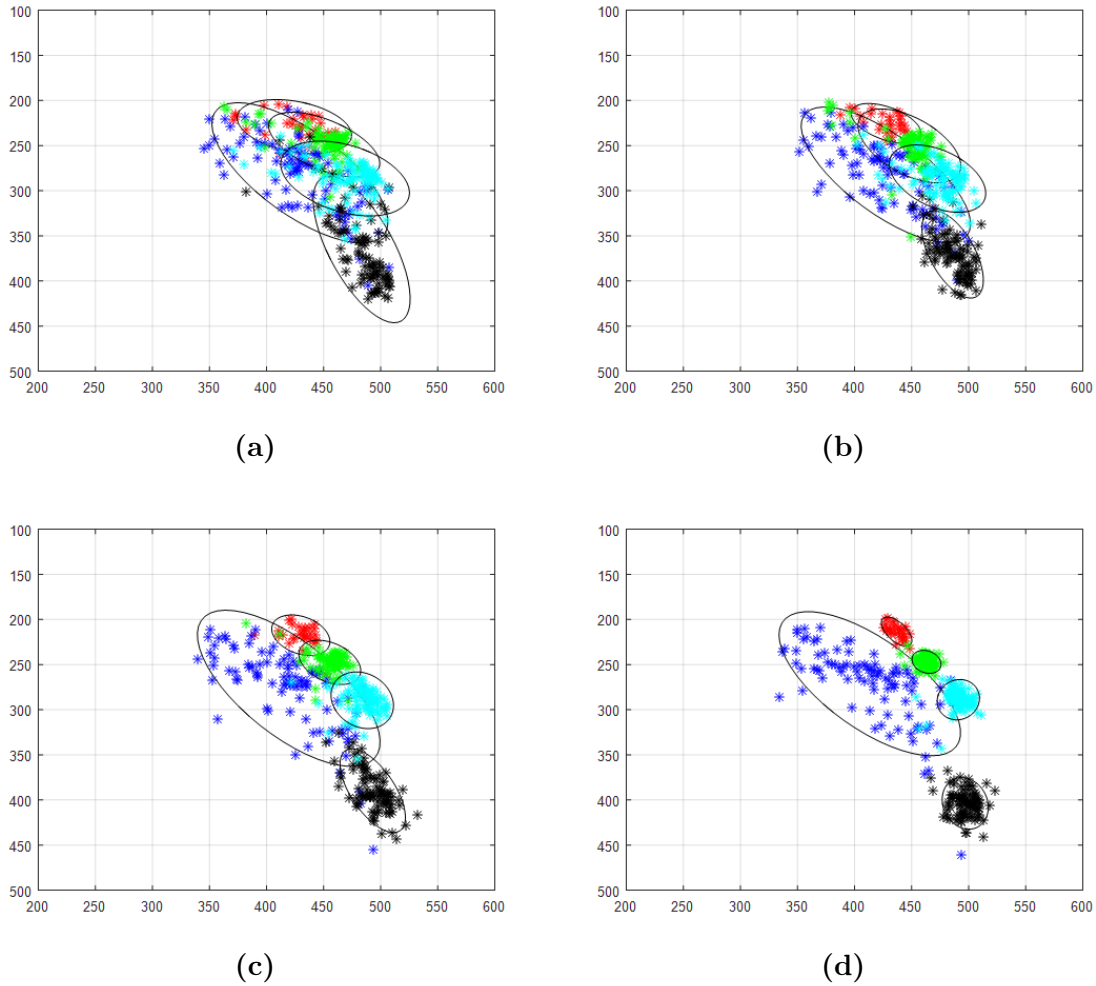


Figure 7.10: Hand position distributions for different temporal segmentations and target finger positions. (a) the audio based segmentation. (b) the extended-audio segmentation. (c) the predicted segmentation by the hand preceding model. (d) the manual segmentation. Five groups of points correspond to different hand positions. Red points: cheekbone; green points: mouth; black points: throat; cyan points: chin; blue points: side position.

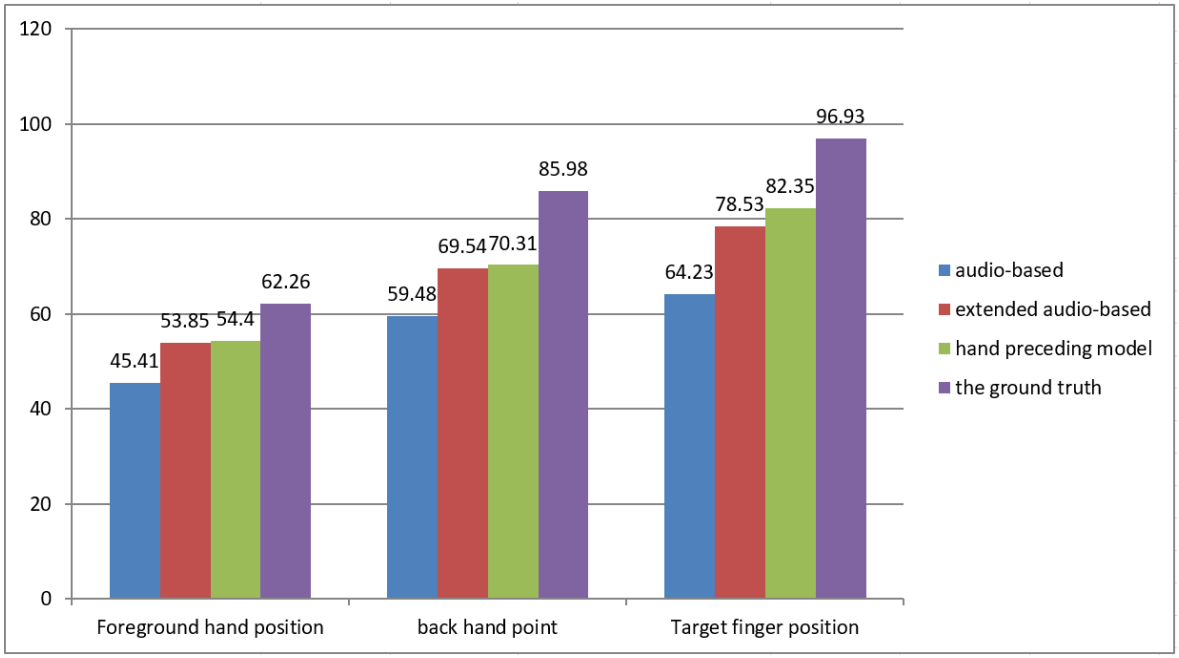


Figure 7.11: Hand position recognition results using multi-Gaussian classifier based on different temporal segmentations.

reference to the hand position recognition in CS. When using the predicted segmentation given by the hand preceding model, a comparable high score of 82.35% is obtained, showing the excellent performance of our proposed method. We observe that the audio-extended method can give a result (78.53%), which does not differ too much from the score using the predicted segmentation. However, when the audio based segmentation is used, the recognition score becomes very low (64.23%). This shows the segmentation based on acoustic signal is indeed not suitable to segmenting the hand position.

Now let us switch to the recognition results using the point on the hand back. The global results have a similar trend with that using the finger but lower accuracy. It is due to the less precision of the hand position on the back compared with the target finger position. Finally, when the hand position is estimated by ABMMs, the scores decrease dramatically. As already discussed in Chapter 6, ABMMs have some errors when tracking the hand position automatically. We can expect a better recognition result when a more robust automatic hand position tracker is applied.

7.4.2.3 Hand position recognition in CS using LSTM based on the whole database

To further evaluate the performance of the proposed method, we apply the hand preceding model to the whole database (476 sentences, about 6000 vowels) of the subject *LM*. LSTM is used for the continuous hand position recognition based on the automatic tracked hand

position[2]. Results are shown in Fig. 7.12. In LSTM, two hidden layers of 500 cells, 200 epoch

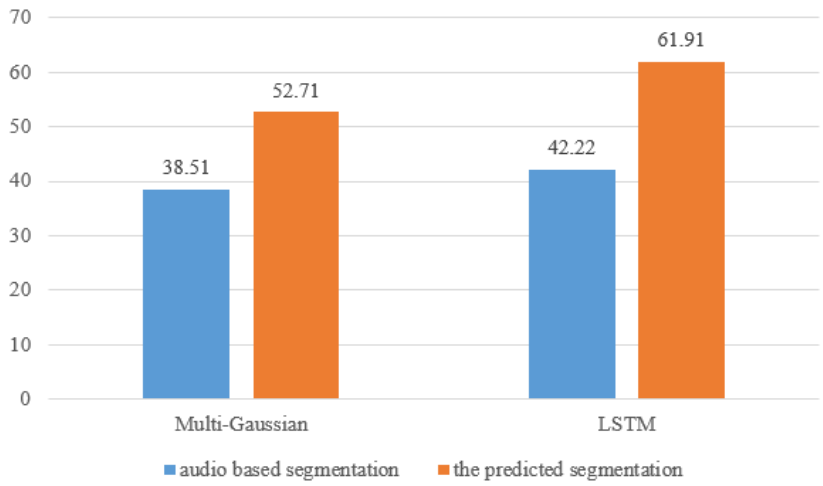


Figure 7.12: Hand position recognition results using the multi-Gaussian and LSTM based on the audio based segmentation and the predicted segmentation, respectively.

are used. It is trained by the BPTT with the cross-entropy cost function. Softmax layer is used to compute the class probability. The final accuracy of LSTM is calculated using max-voting (after softmax layer) which counts the most frequent label as the final label in the corresponding segment. LSTM is implemented using the Keras toolkit [207] based on the GPU-accelerated library.

In Fig. 7.12, the hand position recognition results confirm the advantages of the predicted segmentation for both multi-Gaussian and LSTM. Moreover, LSTM obtains a higher accuracy than multi-Gaussian since it captures the long-term temporal information of the hand movement. More importantly, the proposed temporal segmentation based on LSTM gets the accuracy 61.91%, which almost reaches the upper limit 62.26% by the ground truth temporal segmentation in case of using the automatic tracked hand position (see Fig. 7.11). The continuous hand position recognition score can be further improved when an accurate hand position is provided.

Above all, we see that the proposed segmentation method can significantly improve the recognition performance of the hand position. It is hopeful to adopt this novel temporal segmentation method to the full CS recognition.

7.5 Hand preceding model for the hand shape movement

In Fig. 7.3, we show that the best interval to identify a hand shape is located just before the vowel target instance t_{tar_v}. This segment has a duration D_c and its center t_{tar_c} is considered

[2] Here, we do not have the ground truth hand position for 476 sentences. We only manually track the hand position for 138 among 476 sentences.

as the best instance for a consonant. Now we experimentally determine the hand preceding time for a consonant

$$\Delta_c = t_{tar\_c} - t_c.$$

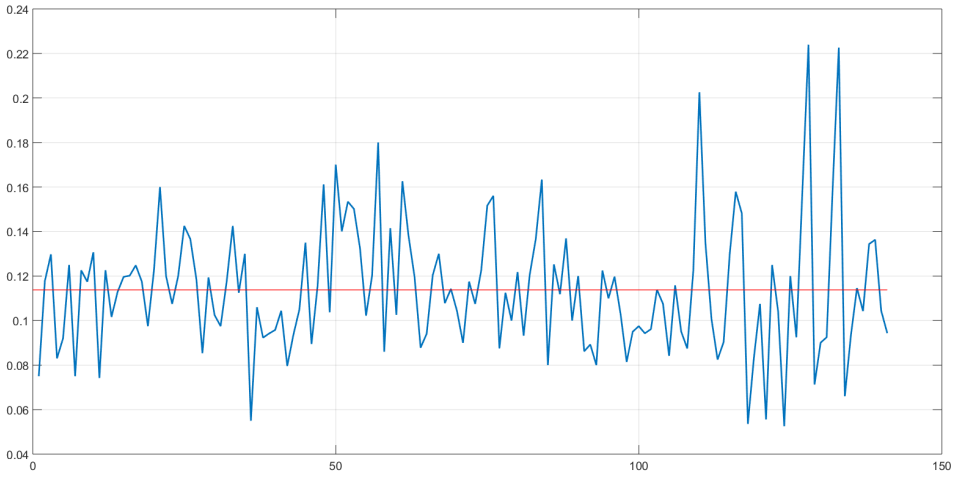


Figure 7.13: Distribution of the time difference between t_c and t_v. The abscissa is the index of vowels, and y-axis is the Δ_{cv} (in seconds). The blue curve is obtained from the vowels randomly extracted from ten sentences in corpus *LM*. The red line is the mean value of all the Δ_{cv}.

Firstly, we perform a statistical study about the time difference

$$\Delta_{cv} = t_c - t_v$$

for the vowel and consonant based on corpus *LM*. We randomly choose 10 sentences which contain about 100 consonants. In Fig. 7.13, Δ_{cv} is plotted for these 10 sentences. Δ_{cv} varies in a broad range, and we only consider its mean value (roughly 110ms). As Δ_v is about 140ms (see Section 7.4.1), we can deduce that t_{tar_v} precedes t_c about 30ms (the difference between 140ms and 110ms). After a large number of observations, we assume that the duration D_c is about 60 ms (3 images). Consequently, t_{tar_c} precedes t_c about 60ms. This is an estimation of Δ_c for consonants.

Then we determine the optimal value of Δ_c experimentally. We perform a hand shape recognition based on the CNNs hand features (see Chapter 8 for more details) and the multi-Gaussian classifier. In the recognition, we use several different segmentations which are derived by shifting the audio based segmentation with different Δ_c (from 0 to 160ms with a step of 10ms). In this way, the Δ_c with the best recognition score will be regarded as the optimal value Δ_c^*.

A hand position recognition experiment using the database of all 476 sentences is conducted, and the results are shown in Fig. 7.14, which gives the recognition score as a function

of the monotonically increasing preceding time Δ_c. We observe a convex curve (red curve) with a local maximum value about 60ms. This is coherent with the theoretical analysis about the Δ_c of hand shape (see Section 7.2.2). Indeed, the peak region of this curve is relatively smooth, but the presence of a clear maximum value confirms that there exists an optimal value of Δ_c.

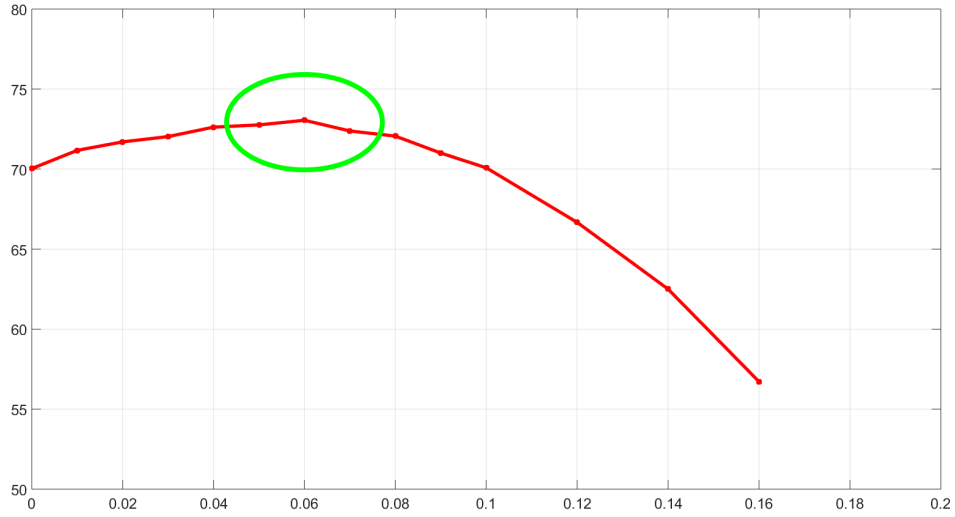


Figure 7.14: Hand shape recognition results using different segmentations in function of Δ_c. The red curve represents the recognition score as a function of Δ_c, and the green circle highlights the optimal recognition score.

The above results also constitute an evaluation of the segmentation method since there exists a common maximum value Δ_c^* (about 60ms). In addition, in Section 8.3.3, this proposed temporal segmentation achieves 84.3% phoneme recognition correctness based on the hand shape and lips information. This result outperforms the state of the art (79.8%) [10].

7.6 Summary

In this chapter, we investigate a specific study concerning the temporal hand movements in CS. As we know, the asynchrony problem of lips and hand movements makes them difficult to share a common temporal segmentation from the audio signal. We show that the speaker's hand movement is essentially organized syllable by syllable. For a typical CV syllable, during the hand movement, the objective of the speaker is to reach a specific position to indicate the vowel. The hand shape changes to indicate the consonant of the syllable. We perform a detailed study by measuring the hand preceding time for vowels and consonants. These measurements show that the hand preceding time for vowels has almost the same distribution (with a mean value of 140ms for LM) from the beginning of a sentence to about 1s before the end. This preceding time then decreases linearly until the end of the sentence. Our measurements permit to

establish a hand preceding model which takes into account this phenomenon. This model allows us to elaborate a segmentation method for CS hand movements using the audio based temporal segmentation. In other words, this method permits to predict the instant where the hand reaches its target position. Our evaluations confirm the superior performance of the proposed method. In fact, hand position recognition score using the predicted segmentation significantly outperforms that using the audio based segmentation, with the Gaussian classifier. Moreover, on the whole dataset, using LSTM achieves a much higher recognition score than using the audio based segmentation. We also observe that the optimal moment to identify a hand shape indicating a consonant is situated about 60ms before the audio signal. It shows that we could obtain the optimal result when this preceding time is used in the hand shape recognition.

Chapter 8

Continuous Cued Speech Recognition based on CNN-HMMs

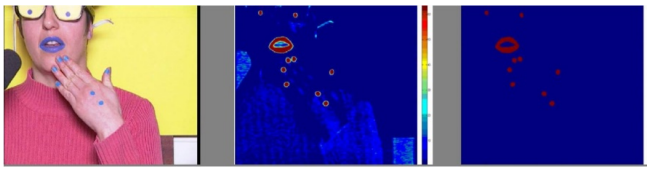
Contents

8.1 Introduction

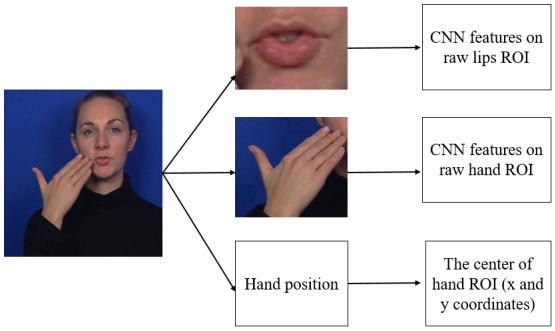
As introduced in Chapter 1, lip reading scarcely reaches perfection in speech recognition due to the ambiguity of the visual pattern. Recall that CS system was invented to make the spoken language visible combining the hand coding with lip reading. The main objective of this chapter is to develop an automatic continuous CS recognition system [208] which transfers the visual video information to the phoneme in the form of text.

Recall that the automatic recognition of CS was explored in [10], [11], [40] with visual artifices to track the lips and hand features (see Fig. 8.1(a)). In this thesis, the **first motivation** is to get rid of these artificial visual marks.

Concerning the recognition stage, the classification of static hand shapes in CS was addressed in [209] using an ANN. In [10], [11], HMM-GMM was used for CS phoneme recognition. In [10], context-independent HMM-GMMs were used to decode a set of isolated phonemes extracted from CS sentences, i.e., the temporal boundaries of each phoneme to recognize in the video were known at the test stage. In [11], the continuous phoneme recognition was also performed by the context-independent HMM-GMMs. However, the dataset in that study was only composed of isolated words repeated several times (not continuous sentences). To the best of our knowledge, no study has addressed the continuous recognition of CS or used the context information of CS. Solving this challenging task is the **second motivation** of the present study.



(a)



(b)

Figure 8.1: Comparison of the feature extraction between the state of the art [10], [11] and this thesis. (a) Lips shape, hand position and shape feature extraction based on blue colors in the state of the art (from [10]). (b) Overview of the feature extraction without using any artifice.

For the first motivation, we explore the benefits of using CNNs to extract the lips and hand features based on raw ROI images[1] (see Fig. 8.1(b)). For the second motivation, the context-dependent HMM-GMM combined with the feature-level and model-level fusions for CS recognition is explored. In fact, another possible approach could use a LSTM to learn the context information between the phonemes and capture the long memory of the CS. In this thesis, we choose to use the context-dependent HMM-GMM.

[1] Raw image here means that no artifice is placed on the speaker's face or hand.

In our case, CNNs are applied in a supervised manner which means that it needs a proper alignment of data in the training step. As we know, in CS, there are three data streams, and they are asynchronous during the CS coding. As a consequence, three different temporal segmentations for them are necessary. The temporal segmentation of audio signal can be used for the lips stream. In this thesis, the hand preceding model (see Chapter 7) is used to predict the temporal segmentation of the hand position and hand shape. It should be noted that different temporal segmentations are only used for the feature extraction by CNNs.

In this thesis, CNNs are combined with an HMM-GMM classifier that models the dynamic feature trajectories for each phonetic context [208]. Recall that in conventional ASR, this combination is often referred to as a *tandem* architecture. This chapter is arranged as follows. Firstly, the experiment database, methodologies, and the experimental implementations will be presented in Section 8.2. Then, experimental metric, evaluation and results will be presented in Section 8.3.

8.2 Methodologies and experimental implementations

In this section, we will first introduce the experimental database in CS recognition. Then the used context-dependent modeling will be presented. Finally, several tandem CNN-HMM architectures will be investigated.

8.2.1 Database

The database for vowel, consonant and phoneme recognition experiments are collected from the normal hearing CS speaker *LM* without using any artifice[2]. A set of 238 French sentences is also derived from a corpus in [44]. Each sentence is repeated twice[3] by the speaker *LM* resulting in a set of 476 sentences (11772 phonemes totally). One example is *Ma chemise est roussie* (in English: *My shirt is scorched*). In this experiment, we use 14 vowels and 18 consonants, since the amount of consonants [ŋ] and [ɲ] is not sufficient in this database. The French CS is described with 8 lips visemes, 8 hand shapes and 5 hand positions. As mentioned in Section 2.3.2, the phonetic transcription is extracted automatically and post-checked manually by carefully listening to and observing the recorded audio signal.

8.2.2 HMM based context-dependent modeling

Recall that the second motivation in this thesis is to realize the *continuous* CS recognition which incorporates the context information. As introduced in Section 4.4, modeling a sufficient amount of contexts is very important for improving the robustness of a continuous CS recognition system. In CS, the articulatory features (i.e., lips and hand) are very sensitive to the

[2] This database is publicly available on Zenodo (https://doi.org/10.5281/zenodo.1206001).
[3] The aim is firstly to increase the size of data and secondly to correct potential errors.

effect of context like the co-articulation, anticipation and variabilities due to the asynchrony.

In this thesis, we model the context information in CS by adding the left and right phoneme context for the current phoneme, which is usually called the *triphone* model (see Fig. 8.2) in ASR. More precisely, 34 HMMs are first built to model 34 French monophones using the same procedure in ASR. Then, using the tree-based state-tying modeling [150] introduced in Section 4.4, we can generate models for triphones (including unseen triphones). Finally, the decision trees are used to find the most possible model for any given triphone.

```
__-a+q    __-i+q
i-q+a     q-a+a~
a-a~+i    a~-i+a
i-a+y     a-y+a
y-a+a     a-q+i
q-i+e^    i-e^+u
e^-u+i    u-i+__
```

Figure 8.2: Examples of the triphone in our context-dependent modeling. Syllable '–' represents the link to the left phoneme, while '+' represents the link to the right one.

8.2.3 A resynchronization procedure of multi-modalities in CS

In fact, it was investigated that the hand reaches its target on average 239ms [37] (based on non sense syllables logatome, like 'mamuma'), and 144.19ms [38] (based on syllables extracted from French sentences) before the vowel being visible at the lips in case of CV syllables, respectively. If we directly concatenate lips, hand shape and position features without any pre-alignment, we can image that the effect of this direct fusion will not be optimal since the asynchrony of lips and hand. In the state of the art [10], [11], a direct feature fusion was applied without taking into account the asynchrony of the multi-modalities in CS. However, it is not suitable for the asynchronous multi-modalities problem. Therefore, in this thesis, we propose a preprocessing resynchronization procedure which is named as Aligned Concatenation (AC). It first pre-aligns the hand shape and position features with lips features, and then concatenates them as a whole feature. The initial idea of AC fusion originates from the Direct Identification (DI) model proposed by Schwartz et al. [154]. In [41], to take into account the asynchrony of lips and hand, Aboutabit et al. applied the SI model to merge the decisions derived first from the hand at hand target instant and then from the lips at lips target instant for **isolated** vowels extracted from sentences. In this thesis, AC fusion is applied to merge lips and hand features in a continuous way for the whole sentence (the transitions of CS movements are included). More precisely, there is no need to indicate the exact target instant of lips, hand shape and positions.

Fig. 8.3(a) shows the audio signal and its alignments and annotation for the sentence *Ma chemise est roussie*. This sentence contains 12 phonemes (6 vowels and 6 consonants). We remark that this audio-based temporal segmentation is directly used for the lips feature stream. For the hand movement, we take the hand position case as an example and show the principles of direct and AC fusions. Fig. 8.3(b) shows the hand movement by plotting the

x coordinate of the hand back point. The aligned hand position feature (see Fig. 8.3(c)) is obtained by positively shifting (delaying) the original hand position with time interval Δ_v. It is equal to 140ms, which is derived from the hand preceding model for vowels (see Chapter 7). Note that 140ms is the average value of the hand preceding model. This may not be the optimal value, but the most reasonable and straightforward one without the ground truth hand preceding time for each vowel. As we can see, the temporal boundary of the vowel [i] (the one after [m]) is $[t_{n-1}, t_n]$. We remark that the red dots represent the time instant when the hand reaches its target position. If we use this boundary to extract the original hand position feature directly, the ground truth hand position does not correspond to the vowel [i]. The direct fusion will not give a good performance. However, when we use this boundary to extract the aligned hand feature, the extracted hand position well corresponds to the vowel [i]. As a consequence, the lips and hand correspond to each other better and this should help the fusion of their parameters for recognition.

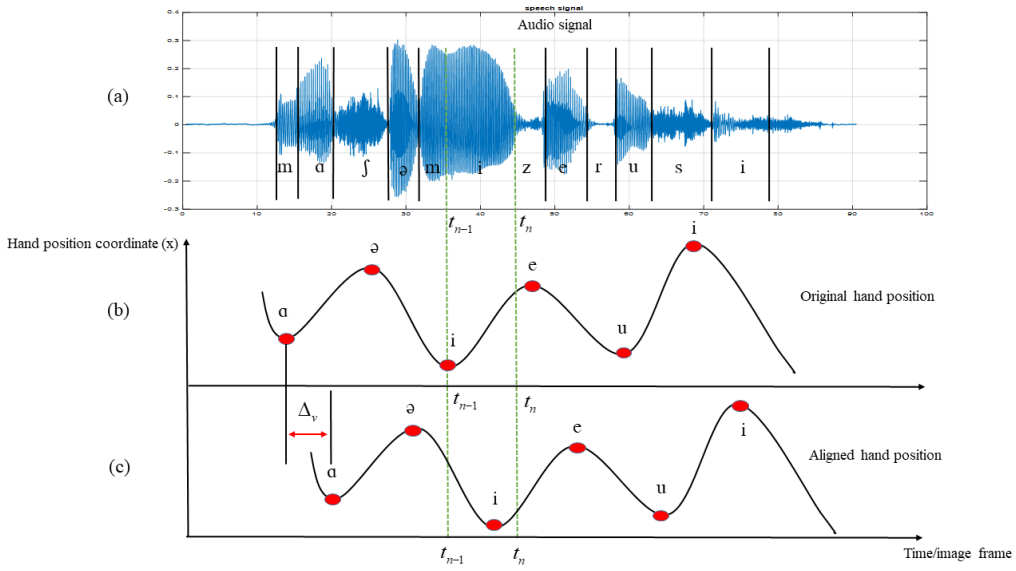


Figure 8.3: Direct fusion and AC fusion. (a) the audio speech with its alignments and phonetic annotation. (b) the original hand position (i.e., x coordinate of the hand back point). (c) The aligned hand position derived by shifting the original hand position in (b) with Δ_v. Two green lines correspond to the temporal boundaries of the vowel [i].

Notably, the aligned hand shape feature follows the same rule as the hand position case. Δ_c is 60ms according to the result in Chapter 7. In fact, hand position stream is more sensitive to the asynchrony problem than the hand shape stream. The reason is that hand position often remains at the target position for a short time interval. If the temporal boundaries are not precise, the corresponding correct hand position features will be missed. However, for the hand shape, the complete hand shape often maintains for a certain time. It makes hand shape stream less sensitive to the precision of the temporal boundary.

Due to the fact that the direct feature fusion without pre-resynchronization is not really suitable for CS case, in the following experiments considering vowel, consonant and phonemes recognition, we will only present the result of AC feature concatenation types in Section 8.3.4.

The effect of using the resynchronization procedure AC will be presented in Section 8.3.5.

8.2.4 The proposed CNN based architecture

As introduced in Section 3.3, a standard CNN contains several *convolutional layers* which are composed of convolutional filtering activation function, pooling, stacked with FC layers using softmax function, and an output layer giving the posterior probability of each class to decode. In this study, we employ 2D CNNs to extract the lips and hand shape features directly from the raw ROI images. Because of the limited size of our current dataset, we only investigate several CNN based architectures which contain two convolutional layers, two pooling layers, two FC layers, and one output (softmax layer). The cross-validation is used to optimize some hyper-parameters for each layer (i.e., the number of filters, the kernel size for the 2D convolutions, the down-sampling factor for the pooling layer, and the number of neurons in the FC layer). Finally, for all the architectures, two convolutional layers with 8 filters, a kernel size of 7×7 pixels, a down-sampling factor of 3 (in both vertical and horizontal directions), the number of epoch 300, the pooling size 3×3, the batch size 2048 and 64 hidden neurons in the FC layer are used.

At the training stage, a mini-batch gradient descent algorithm based on the *RMSprop* adaptive learning rate method (with a learning rate 0.001) and a batch size of 2048 frames, is used to estimate the CNN parameters. The categorical cross-entropy is used as the loss function. Over-fitting is controlled using (1) an early stopping strategy, i.e. 20% of the training set is used as a validation set and the training is stopped when the error on this dataset stops decreasing during 10 epochs, and (2) a dropout mechanism (with a dropout probability of 0.25). All models are implemented using the *Keras* Python library [207], and trained using GPU acceleration.

CNNs are trained in a supervised manner. Therefore, the CNN training process is sensitive to the temporal alignment in its training process The temporal segmentation of the lips is derived directly from the audio signal (the asynchrony between lips and audio is neglected here). However, in CS, the hand generally precedes the lips. To take this phenomenon into account, we apply the hand preceding model (see Chapter 7) to predict the optimal temporal segmentation (with an average hand preceding time 60 ms) of the hand shape from the audio speech signal for each phoneme.

For the automatic recognition of continuous CS, we mainly have the following considerations: (1) one single ROI containing both the lips and hand or two distinct ROI focusing on the lips and hand, respectively; (2) the feature-level and model-level fusion strategies that the lips and hand features are combined within the HMM-GMM decoder; (3) the visual feature extraction techniques: an unsupervised and linear technique based on PCA and a supervised and non-linear technique based on CNN.

Based on these considerations, we propose several architectures S_1-S_3 based on CNN, and

$\mathbf{s}_1$-$\mathbf{s}_3$ based on PCA[4]. The CNN-HMMs architecture in case of S_3 is shown in Fig. 8.6.

(1) In S_1, the single CNN jointly models the lips, hand position and shape. A unique bounding box is set large enough[5] to contain both lips and hand, and it is anchored on the lips ROI. More precisely, the lips and hand are regarded as a global ROI in CS. The audio-based temporal segmentation is used to train CNNs in S_1. CNN is trained with 34 phonetic classes as targets (see Fig. 8.5(a)). CNNs are applied to extract features based on this global ROI. Then, these features will be fed to the one stream HMM-GMM for phonetic decoding.

(2) In S_2 and S_3, each CNN focuses on the lips or hand shape separately. In S_2, three-stream features are concatenated (i.e., feature-level fusion) in a single feature vector (see Fig. 8.5(b)). In S_3, lips and hand information are combined at the state level using a 3-stream HMM-GMM (model-level fusion) (see Fig. 8.5(c)). As for the training in CNN, features are trained with a set of lips visemes, hand shape groups and hand position groups, respectively. More precisely, except for one silence class, there are eight lips visemes defined in Table 2.3. Five hand positions and eight hand shape groups are given by the definition of LPC (see Fig. 1.3).

Besides the lips and hand shape, for S_2 and S_3, the hand position (coordinates of the ROI center) is first extracted automatically by ABMMs. Then these values are processed by a simple feed-forward neural network (ANN) with one single FC layer (with ReLU activation function) and one output *softmax* layer trained with a similar procedure as CNNs.

Hand shape feature extraction based on CNN

In Chapter 6, we introduced the ABMM for hand position feature extraction, and the hand shape ROI was located based on the estimated hand position. Now we have a look at the hand shape feature extraction using CNN. In fact, the hand shape feature extraction in CS is a challenging problem due to the nature of hand coding. For instance, the same hand shape (with the same linguistic meaning) may result in rather different appearances due to the rotation and movement of the hand (see Fig. 8.4). We will use nonlinear CNNs to extract the high-level hand shape features. More precisely, after obtaining the hand shape ROI image by the ABMMs, we feed it to CNNs as the input, and finally extract the visual feature vector, which is the output of the last FC layer (before softmax). In the following CS recognition of this study, lips features are also extracted by the same procedure (CNNs) as the hand shape feature[6].

[4] The only difference between $\mathbf{s}_1$-$\mathbf{s}_3$ and S_1-S_3 is the feature exaction.
[5] The center of the lips and hand ROI is used for the bounding box. The length and width of the bounding box are determined experimentally.
[6] The lips parameters in Chapter 5 can be also used. Here, in order to keep the lips and hand feature dimension at the same level, we use CNN for the lips feature extraction in CS recognition.

Figure 8.4: Visualizations of the same hand shape with different rotations.

HMM-GMM decoder

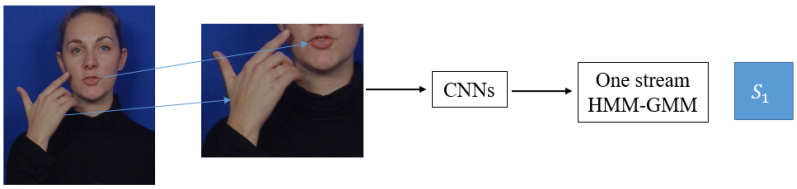
Sequences of visual features extracted using either PCA or CNN (ANN for hand position in S_2 and S_3) are modeled, together with their first derivatives, by a set of context-dependent triphone HMM-GMMs. A standard topology is used with three emitting states (with no connection between the initial and final states). HMM-GMMs are trained using *HTK 3.4* [14]. The number of components of each GMM emission probability iteratively increases from 1 to 4. The parameters are estimated by the EM algorithm. For all the architectures, at the decoding stage, the most likely sequence of phonemes is estimated by decoding the HMM-GMM using the Viterbi algorithm. The model insertion penalty is optimized on the training set. Currently, neither pronunciation dictionary nor language model is used in this study. In fact, we aim at evaluating only the ability of the system to extract the phonetic information from raw data without any prior linguistic knowledge (indeed, the global performance should be significantly higher when using such information).

The implementation details of CNN and ANN

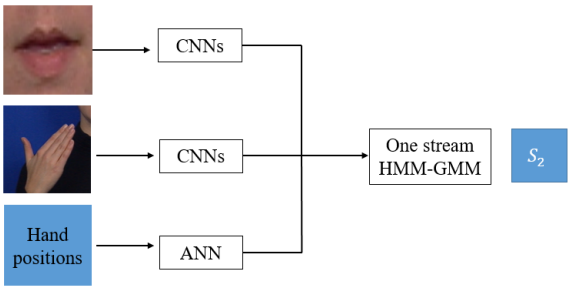
In order to further see how CNN and ANN work for the feature extraction, we now show the implementation schematic diagram.

In cases of S_2 and S_3, the implementation details of CNNs for lips and hand shapes are shown in Fig. 8.7. It shows the layers and dimensions of input and output in each layer. We can see that the inputs (lips and hand shape ROI) are resized to 64×64 and converted to a 2D gray image. They are first fed to a 2D convolutional layer equipped with eight different kernel filters, while the output of this layer is a 64×64×8 tensor. The ReLU activation function will be used to process this tensor, without changing the dimension of the output. Then it will pass the max-pooling layer, dropout layer. After, the above process will repeat once. Then, the flatten layer is used to reshape the output at this node in order to feed it to a FC layer. The dimension of the FC layer is 9 with one silence class. Besides, the implementation details of CNNs for S_1 are shown in Fig. 8.8.

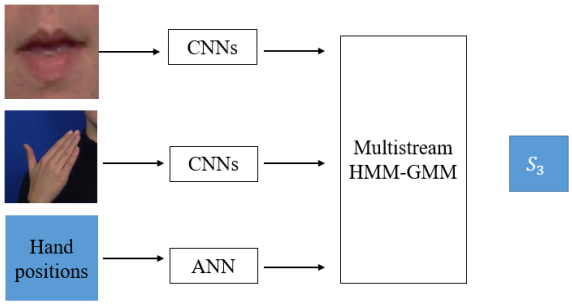
The implementation details of ANN (standard FC layer) for hand position processing can



(a)



(b)



(c)

Figure 8.5: Three different architectures (S_1-S_3) of the continuous CS phoneme recognition based on CNN and HMM-GMM decoder.

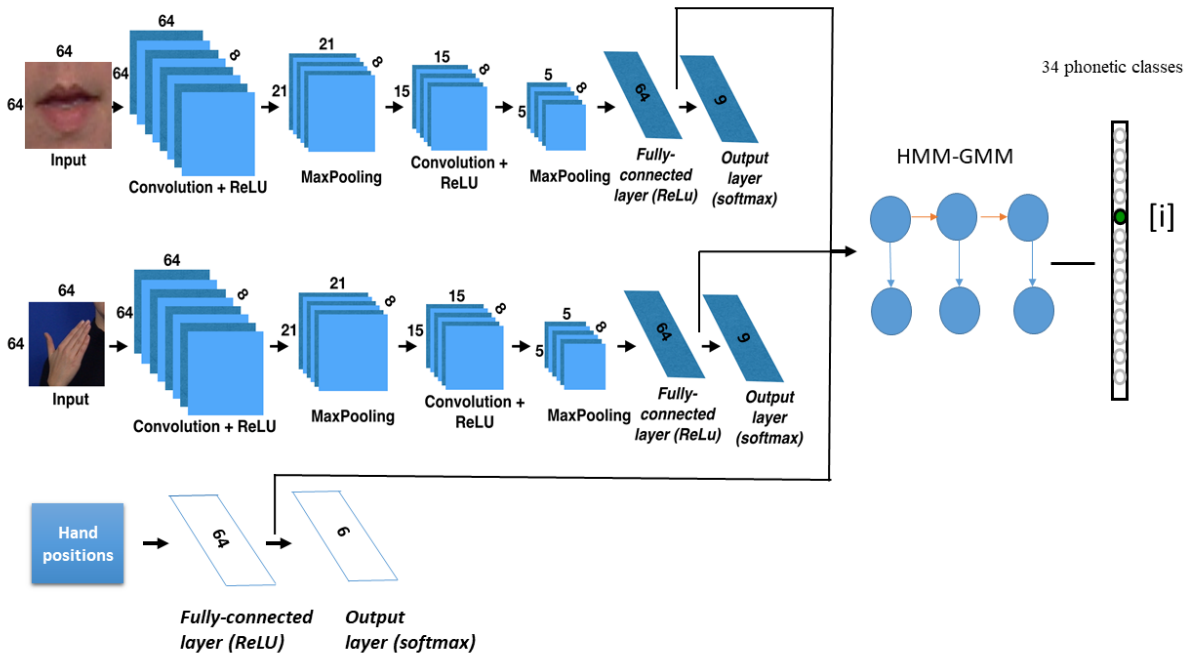


Figure 8.6: CNN-based feature extraction and HMM-GMM decoding in case of S_3. Lips and hand shape features are extracted by CNNs, and the hand position coordinates are processed by the standard ANN.

be seen in Fig. 8.9. The size of the features is resized to 64×64 at the beginning. After softmax layer, it will output the posterior probability of the target classes with dimension 6.

8.2.5 The baseline PCA architectures

As shown in Section 2.3.3.1, PCA technique, also known as the EigenFaces technique [116], is an unsupervised and linear technique which aims at finding a decomposition basis that best explains the variation of pixel intensity in a set of training frames. At the training stage, the PCA is performed on a set of N training frames (in our case, $N = 1000$). The resulting basis vectors are often called *EigenLips* [108] when applying this technique to lips images. At feature extraction stage, each new frame is projected onto the set of these basis vectors. Visual features are defined as the D first coordinates in the decomposition basis. To keep the eigenvectors that carry 85% of the variance, we set $D = 40$ when encoding jointly lips and hand in S_1, $D = 34$ for lips, and $D = 45$ for hand in S_2 and S_3 when considering the lips and hand separately.

PCA-based features are decoded using $\mathbf{s}_1$-$\mathbf{s}_3$ (see Fig. 8.10). The implementation principles of $\mathbf{s}_1$-$\mathbf{s}_3$ are similar with the CNN-HMM (S_1-S_3). It should be noted that PCA is an unsupervised learning method, and thus it is not sensitive when using different alignments of the data streams in the training process.

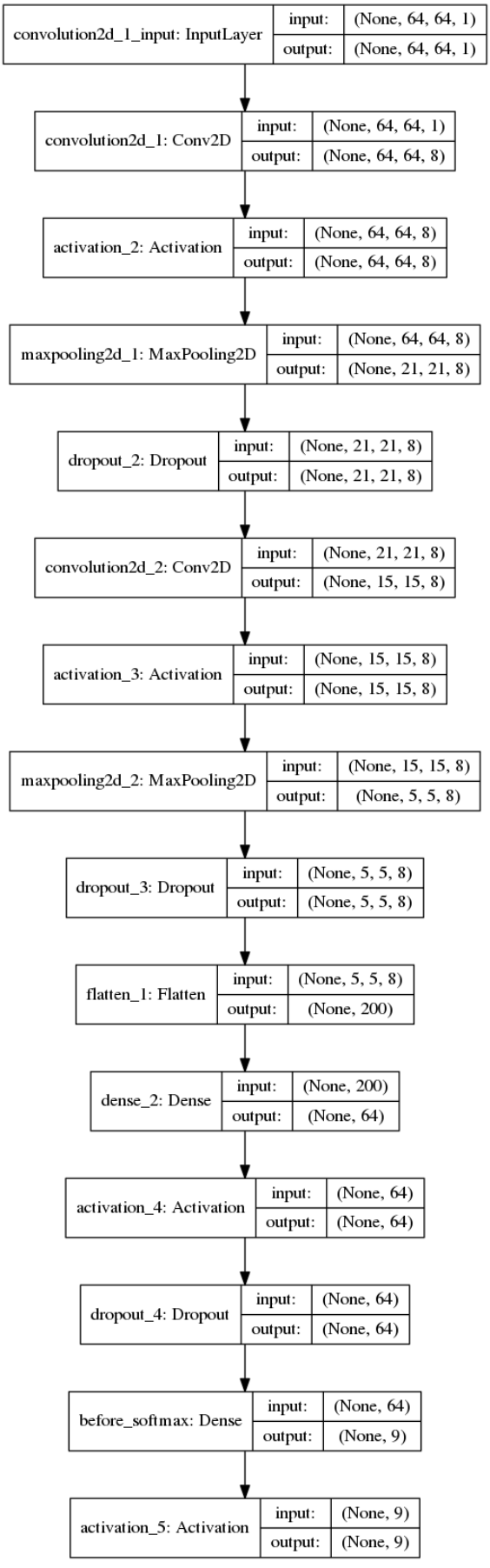


Figure 8.7: Detailed implementation configuration of the CNN architecture for the recognition of nine hand shapes (eight hand shapes + one silence) or nine lips visemes (eight lips visemes + one silence) in cases of S_2 and S_3.

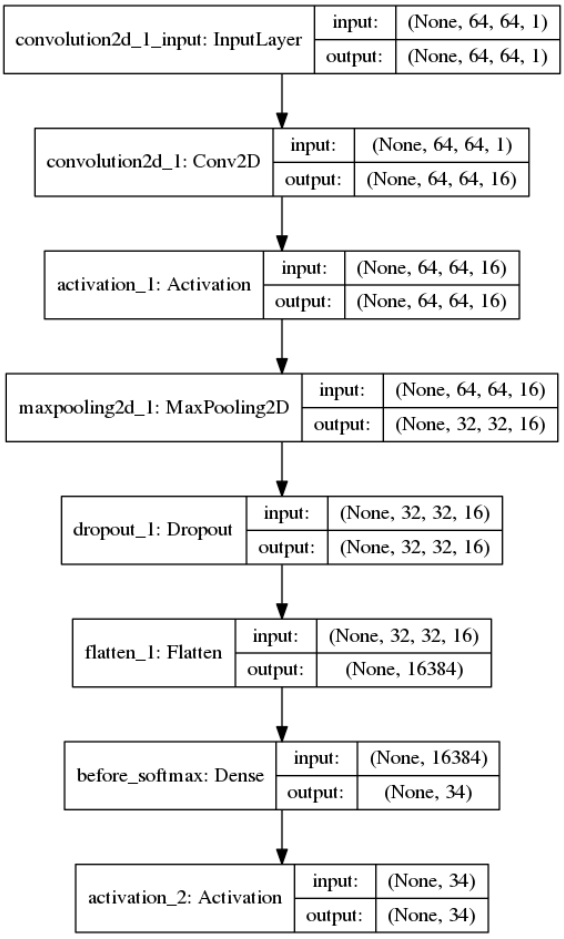


Figure 8.8: The CNN architecture of S_1 for the recognition of 34 phonemes.

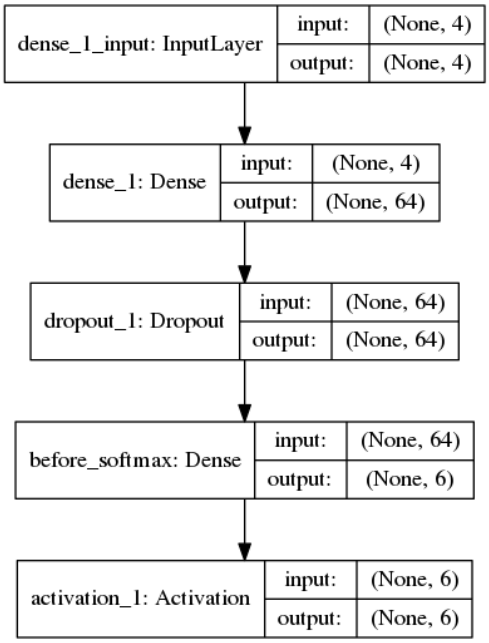


Figure 8.9: Detailed implementation configuration of the ANN architecture for the recognition of six hand positions (five hand positions + one silence) in cases of S_2 and S_3.

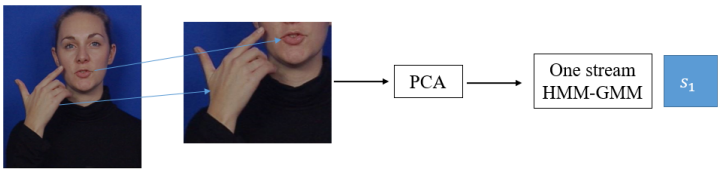
8.3 Evaluation and results

We have developed several architectures for the continuous CS recognition in Section 8.2.4, which will be evaluated from three aspects in this section (see Fig. 8.11). The first phase is about the viseme recognition, i.e., the single stream recognition (see Section 8.3.2), which will offer us meaningful information to understand the contribution of each single stream. The second phase is about the vowel recognition (see Section 8.3.3) and the consonant recognition (see Section 8.3.3). This will allow us to evaluate the fusion of the lips stream and the hand position stream, and the fusion of the lips stream and the hand shape stream, respectively. The last phase is about the phoneme recognition (see Section 8.3.4), which will allow us to see the full recognition performance combining all the three streams.

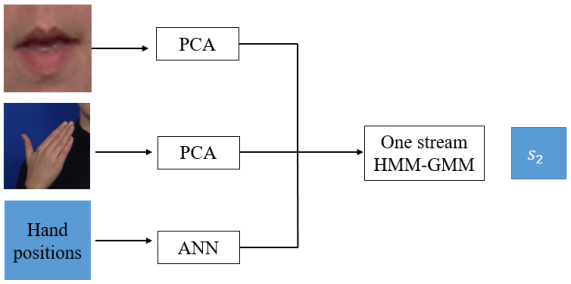
8.3.1 Protocol and metrics

In the following experiments, we randomly choose 80% of the dataset as the training set (with 20% used for validation in CNN), while the remaining 20% as the test set in HMM. Note that the dataset contains the repetitions of some sentences. We allocate the repeated sentences in the training or test set at the same time. For example, if the first sentence (*001-1*) is chosen as a training sentence, its repeated sentence (*001-2*) will also be allocated to the training set. For the CS vowel, consonant and phoneme recognitions, in the HMM training step, three streams share the same temporal segmentation[7]. For the recognition results, two metrics are

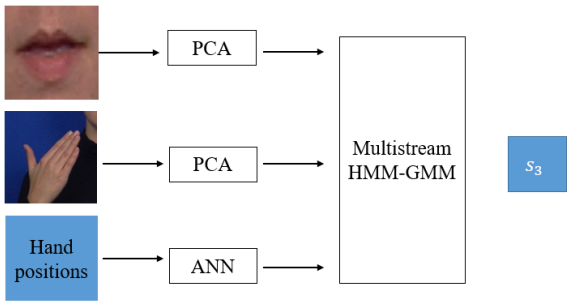
[7] Note that in case of CNNs feature extraction, different temporal segmentations are used for these three streams.



(a)



(b)



(c)

Figure 8.10: Three different architectures ($\mathbf{s}_1$-$\mathbf{s}_3$) of continuous CS phoneme recognition based on PCA and HMM-GMM decoder.

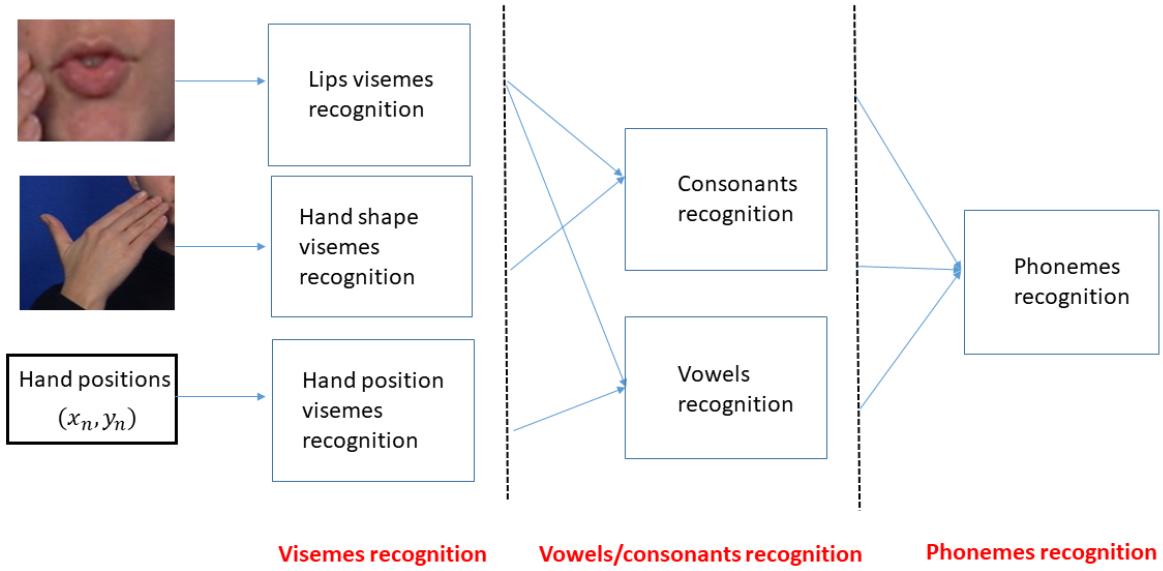


Figure 8.11: Different steps for evaluation of the proposed architecture. The first step is the viseme recognition only using the single stream. The second step is the consonant and vowel recognition using the combinations of two streams. The final step is the full phoneme recognition using all three streams.

used to assess the decoder performances. One is the *correctness* T_c, which only takes into account the deletion error; the other is the *accuracy* T_a, which takes into account the deletion and insertion errors at the same time. More precisely, they can be calculated as follows:

$$T_c = \frac{N - D - S}{N}, \quad T_a = \frac{N - D - S - I}{N}. \tag{8.1}$$

In (8.1), D is the number of deletion errors, S is the number of substitutions, I is the number of insertions and N is the dimension of the test data. In our case, I and D errors are balanced for the accuracy in all the experiments.

The statistical significance of these measurements was assessed by calculating the Binomial proportion confidence interval $\Delta_{95\%}$ using the Wilson formula [210]:

$$\Delta_{95\%} = 2t^2_{95\%} \frac{\sqrt{X(1-X)/N + t^2_{95\%}/(4N^2)}}{1 + t^2_{95\%}/N}, \tag{8.2}$$

with $t^2_{95\%} = 1.96$ for T_a or T_c. It shows that for all the experiments in this thesis the confidence interval $\Delta_{95\%}$ is about 4%. In order to see the variations in the results, we also conduct ten times of the experiments using different training and test sets. The standard deviation is about 2%. We remark that all the experiment results in this thesis do not take into account the CS coding errors. For all the multi-stream recognition experiments, the stream weights are optimized empirically using the cross-validation. For the vowel recognition based on lips and hand position, the ratio is 0.5 both for the lips and hand position. For the consonant recognition based on the lips and hand shape, the ratio is 0.5 both for the lips and hand shape. For the phoneme recognition, the optimal weight is 0.4 for lips, 0.4 for the hand shape and 0.2 for the hand position.

8.3.2 Viseme recognition in Cued Speech

Note that in CS recognition, a single stream cannot give enough information for the phoneme recognition. The maximal information that it can provide is in the viseme level. For example, the hand shape stream only carries the information about eight different hand shapes. It is important for us to realize the recognition of each stream, which will help us to understand the full recognition performance when merging the three streams. We extract the lips and hand shape features by CNN, and the hand position features by ANN. The dimension of lips and hand shape features is 18 (i.e., 9-dimensional original features with its Δ[8]), and the dimension of hand position feature is 12 (i.e., 6-dimensional original features and its Δ). Then the extracted features are fed to the context-dependent HMM for the recognition.

The recognition scores for the single stream in the viseme level are shown in Table 8.1, where a relative homogeneity between the different modalities can be observed. We see that the hand shape recognition obtains a relatively higher score, probably due to the good hand shape features extracted by CNN and the significant contrast of eight hand shapes. Comparably, the

[8] Δ means the first derivative.

lips viseme recognition is around 10% lower than the hand shape recognition. In fact, from the acoustic point of view, the lips shapes do not naturally constitute eight distinguished classes of visemes. In other words, the eight lips visemes (in Table 2.3) may have some inherent inter-class confusions. However, the lips viseme recognition correctness still outperforms the result in [211], [212], which is based on the PCA, DCT and optical flow features. Concerning the hand position, the 61.72% correctness is coherent with the result in Chapter 7. We can expect that if the correct hand position features are available (instead of the automatic hand position), a better recognition score better than 80% can be achieved.

Table 8.1: Results of the viseme recognition of three single streams based on CNN/ANN-HMM architecture. T_c% is the correctness and '—' means that this case does not exist.

T_c%	8 lips visemes	8 hand shapes	5 hand positions
CNN-HMM	65.79	76.34	—
ANN-HMM	—	—	61.72

In order to further analyze the results of the viseme recognition, we compute the confusion matrix[9] for the recognition using these three single streams. Note that for all these confusion matrices, the first element is the silence, the last column corresponds to the deletion error D and the last row corresponds to the insertion error I. A red rectangle is drawn to mark concerned visemes.

First, we consider the case of hand shape recognition. Recall that the hand shape recognition gives a higher recognition score (76.34%). We see that the confusion matrix in Fig. 8.12 (a) is highlighted by the diagonal elements. The dark elements in the diagonal (the last two hand shape visemes) correspond to the less occurrence of the hand shapes, i.e., the No.7 and No.8 shapes, which indicate [g] and [j], [ŋ], respectively. It may be due to that CNN is trained not properly if the data is not enough.

Moreover, in order to understand the behavior of the CNN-based feature extractor, in Fig. 8.13, we show the output of the CNN for the hand shape images of one test sentence(i.e., the sequence of posterior probabilities for all possible hand shape classes). As expected, the posterior probabilities evolve smoothly between consecutive hand targets (with the maximum value achieved when the target is reached). This motivates a explicit modeling of the dynamic of the extracted features, as performed by the HMM-GMM decoder. It is interesting that the CNN seems to be robust to a small intra-class variation, e.g., between shapes No.3 ([d]) and No.5 ([ʒ]) which are both correctly classified as [p], [d], [ʒ] while the hand shape is quite different (due to hand rotations).

Then, we switch to the lips case (see Fig. 8.12(b)). From this figure, we can see that the first three visemes do not have many confusions while the other five visemes have much more confusions. This is because of the fact that, from the speech production point of view, the first three vowel visemes are distributed more contrastively than the consonant visemes.

[9] All the confusion matrices are given in Appendix A.4.

More precisely, the consonant visemes No.2 ([t], [d], [n]) and No.3 ([f], [v]) are easily confused because they have similar lips shapes, as well as No.4 ([k], [g]) and No.5 ([ʃ], [ʒ]).

At last, we discuss the hand position case (see Fig. 8.12(c)). Globally, there are a lot of confusions compared with the above two cases. It is coherent with the relatively low recognition correctness (61.72%). The reason may come from the errors in the automatic hand position features (see Fig. 6.5). We can see that the distribution of five positions using the automatic hand position (see Fig. 6.5(a)) is more confused than the distribution using the ground truth hand position (see Fig. 6.5(b)). Recall that using a Gaussian classifier, the hand position recognition based on the automatic tracked hand position gives 54.4% correctness while using the ground truth hand back position gets higher correctness 70.3%. On the other hand, we see that the hand position recognition score using ANN-HMM (61.72%) is better than the previous one using Gaussian classifier (54.4%). This difference is due to the fact that context-dependent HMM takes into account the context information in the continuous sequence of hand positions.

8.3.3 Vowel and consonant recognition in Cued Speech

After discussing the single stream viseme recognition, we now look at the vowel and consonant recognition by the fused features of two streams. In this section, we use the single stream (feature-level fusion) and multi-stream HMM (model-level fusion) to compare the benefits of different fusion strategies. In addition, it constitutes a good reference for the full recognition when using three-stream features.

Vowel recognition based on the fusion of lips and hand position

First, we mainly discuss the vowel recognition based on the fusion of the lips and hand position. Besides, the single stream used for the vowel recognition is also presented.

Lips and hand position features are merged and fed to one or two-stream HMM. The vowel labels are the targets. Results are shown in Table 8.2. We see that the vowel recognition using the model-level fusion obtains a slightly higher correctness (70.12%) than using the feature-level fusion strategy (68%). However, this vowel recognition correctness is about 10% lower than the prior work [10], since the automatic hand position features usually have some errors. For the single stream, only lips and only hand position give very similar and low correctness. This is normal because lips and hand positions are supposed to be combined to recognize the vowels. As we mentioned above, one stream carries only the viseme-level information.

We now examine the confusion matrix (see Fig. 8.14) of the vowel recognition using the fused (lips and hand position) features. This result is based on the two-stream HMM. The confusion matrix is highlighted by the diagonal elements which correspond to the correctness 70.12%. As expected, the errors mainly come from the vowels {[e], [ɛ] }, {[o], [ɔ] } and {[ø], [œ] }, and the reasons may come from two aspects: (1) the wrong correspondence between

the phonetic transcriptions and the real CS productions, and (2) in French language, the distinction in the production between {[e], [ɛ] }, {[o], [ɔ] } and {[ø], [œ] } is very variable and dependent on speaker's origin (south, center, north of France).

Table 8.2: Vowel recognition results using lips and hand positions in CS. T_c% is the correctness and '—' means that this case does not exist.

T_c%	single stream	two streams (S_2)	two streams (S_3)
only lips	34.88%	—	—
only hand positions	35.71%	—	—
lips + hand positions	—	68.00%	70.12%

Consonant recognition based on the fusion of lips and hand shapes

After reporting the results of the vowel recognition, we now turn to the consonant recognition. Lips and hand shape features are merged and fed to the one or two-stream HMM. The consonant labels are the targets. In Table 8.3, we show the recognition correctness using the fused features based on the one stream (feature-level fusion) and multi-stream (model-level fusion) HMM. Note that, in this case, the model-level fusion of HMM achieves much higher correctness (84.35%) than the feature-level fusion (73.34%). In particular, the correctness 84.35% outperforms the state of the art [10]. Concerning the single stream, using lips and hand shapes obtains a relatively higher score (compared with the lips and hand positions for the vowel recognition) due to the high quality of CNN based hand shape features.

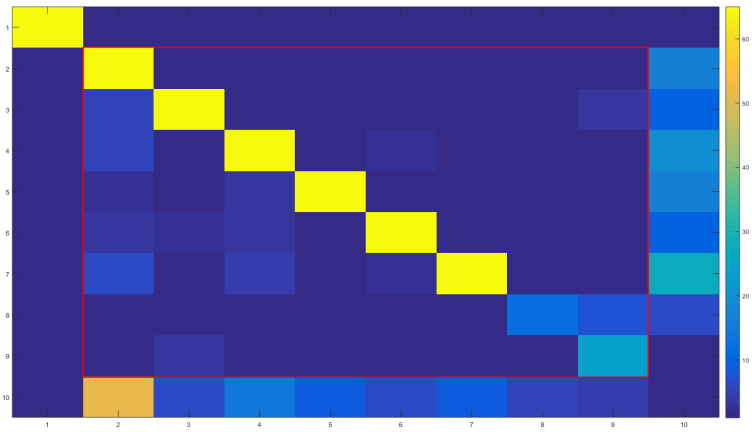
Table 8.3: Consonant recognition using lips and hand shapes in CS. T_c% is the correctness and '—' means that this case does not exist.

T_c%	single stream	two streams (S_2)	two streams (S_3)
only lips	42.25%	—	—
only hand shapes	56.99%	—	—
lips + hand shapes	—	73.34%	84.35%

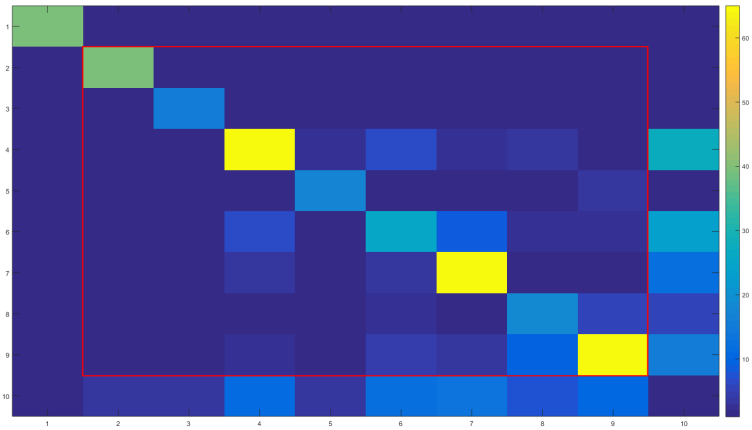
To see the details of the consonant recognition performance, we present the confusion matrix in Fig. 8.15. Globally, it shows a very obvious diagonal structure with few confusions, which is coherent with the high recognition score (84.35%).

8.3.4 Phoneme recognition in Cued Speech

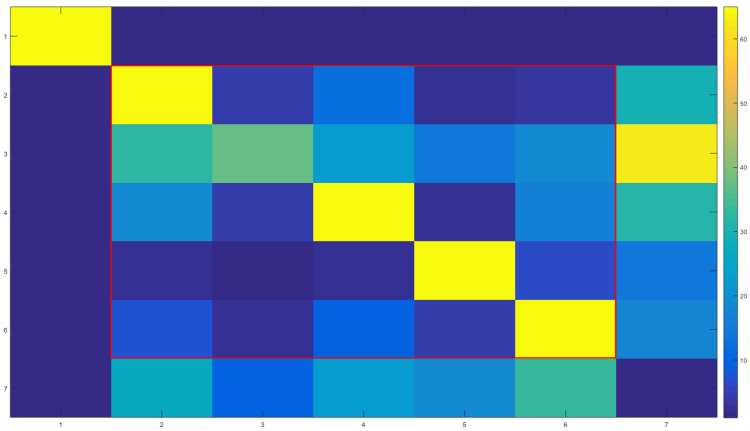
We now pay attention to the three-stream (lips, hand shapes and hand positions) CS phoneme recognition. In particular, we compare the CNN and PCA based methods to show the benefits



(a)



(b)



(c)

Figure 8.12: Confusion matrices for three single stream viseme recognition. (a), (b) and (c) are recognitions based on 8 hand shapes, 8 lips visemes and 5 hand positions. The class "silence" (the first element) is included in the confusion matrix. The red rectangle contains the concerning visemes. The last column corresponds to the deletion error D, and the last row corresponds to the insertion error I. The brighter element corresponds to the higher occurrence in these confusion matrices.

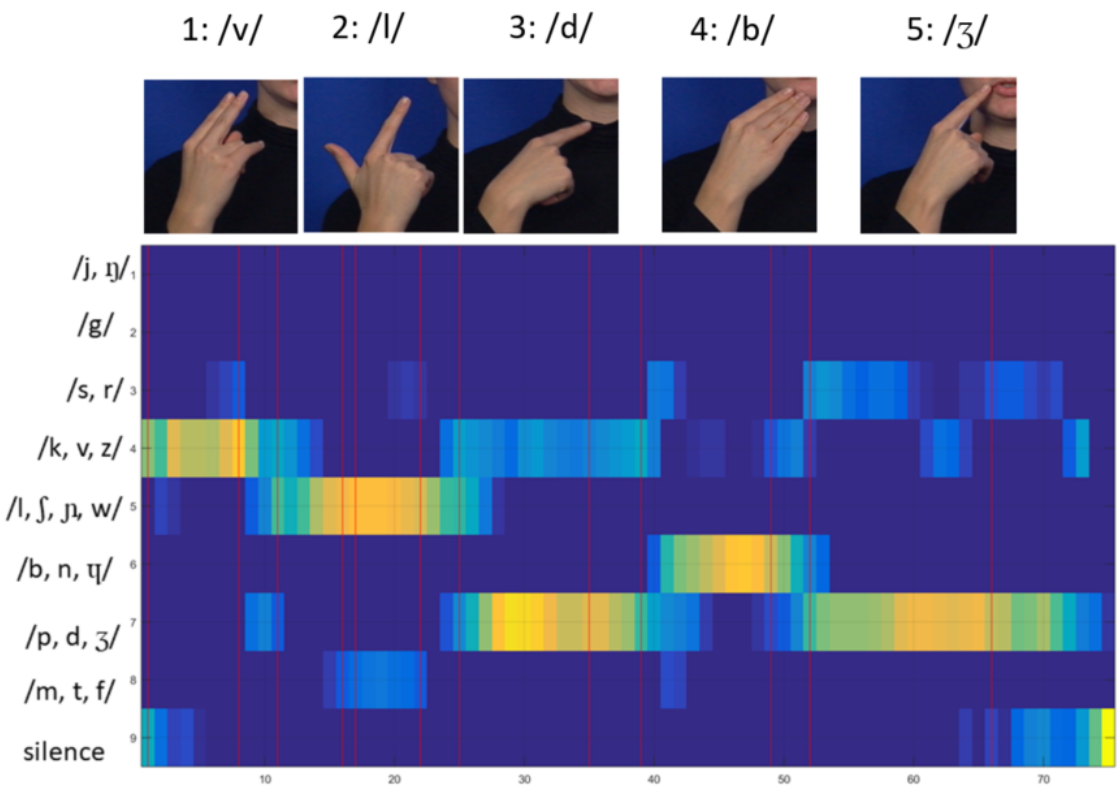


Figure 8.13: Visualization of the representation of CNN softmax layer output. Top: the sequence of target hand shapes (i.e., key frames) for the sentence *voila des bougies*. Bottom: The abscissas is the number of image frame, and y-axis is the target class given by the posterior probability.

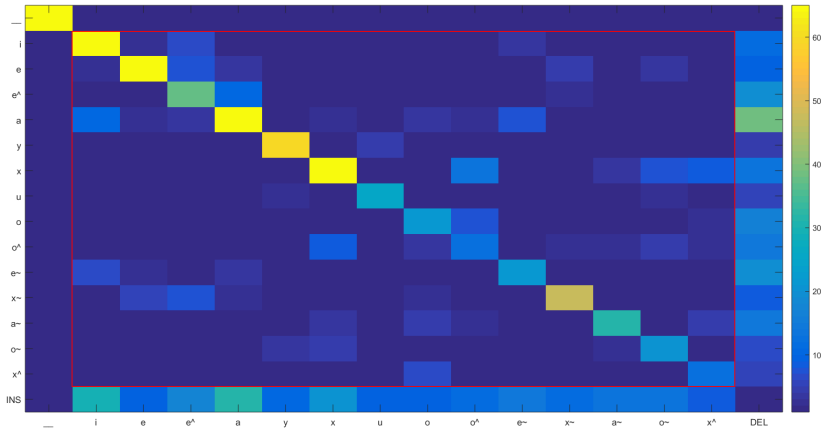


Figure 8.14: Confusion matrix of the vowel recognition based on the HMM model-level fusion of lips and hand positions. The red rectangle contains 14 concerning vowels except the silence, insertion and deletion error elements.

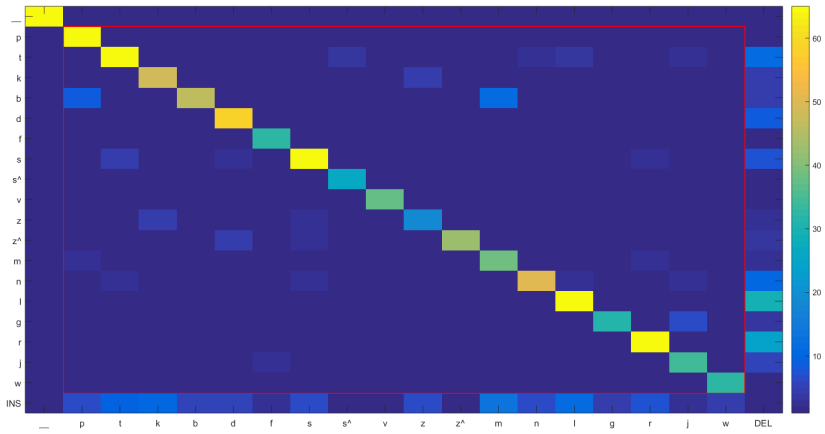


Figure 8.15: Confusion matrix of the consonant recognition based on the HMM model-level fusion of lips and hand shapes. The red rectangle contains 18 concerning consonants except the silence, insertion and deletion error elements.

of the CNN based features.

Firstly, it should be noted that we only show the recognition results using the AC feature fusion (see Section 8.2.3). Compared with the direct fusion, AC fusion (i.e., using the resynchronization procedure for multi-streams) can bring about 1.67% gains for the recognition result (see Fig. 8.18). It confirms that after aligning the hand position and shape features with lips, the fusion will give better performance than direct fusion without any pre-alignment. Moreover, the alignments of hand position and hand shape to lips feature are realized by shifting them with time interval 140ms and 60ms (according to the analysis and results in Chapter 7), respectively. We can naturally image that if the shifted time interval is not the average value (i.e., 140 and 60ms) of the hand preceding models but the ground truth value, a more significant improvement can be obtained.

Then, for Table 8.4, we present some remarks on the recognition results of the phoneme recognition.

Table 8.4: Performances of the proposed CNN and PCA based architectures for automatic continuous CS recognition in terms of correctness T_c% and accuracy T_a%.

$T_c(T_a)$	$S_1/\mathbf{s}_1$	$S_2/\mathbf{s}_2$	$S_3/\mathbf{s}_3$
PCA	45.2(32.3)	50.9(36.0)	51.0(36.5)
CNN	55.0(38.2)	68.3(58.4)	72.7(62.7)

(1) CNN clearly outperforms PCA for all the proposed architectures. This tends to validate the gain of a non-linear and discriminative feature extraction technique for this task. More importantly, the mechanisms for the superior performance of CNNs lie in its powerful non-

linearity and also the robust adjustment for the weakly aligned data. Note that PCA based methods do not have significant differences among s_1, s_2 and s_3.

(2) The model-level fusion strategy outperforms the feature-level fusion. The best performance is achieved using S_3, with 72.7% accuracy. Using S_2, the accuracy is a little bit lower (68.3%) than S_3.

(3) We can see that a much lower accuracy 55.0% of S_1 is obtained. This result seems to show that it is more reasonable to treat lips, hand shapes, and hand positions separately instead of treating them globally. One possible explanation for this unexpected result may be related to the difference of spatial resolution between lips and hand when considering only one ROI (i.e., the hand occupies much more space than the lips) in the image. Extracting a dedicated ROI may help the CNN to better balance the information of lips and hand.

(4) In the best configuration S_3, without exploiting any dictionary or language model, the proposed tandem CNN-HMM architecture can identify correctly about 72.6% of the phoneme (62.7% when considering insertion errors). The performance in terms of T_a is about 10% lower than the one in terms of T_c. Despite the fact that the model insertion penalty is optimized, too many insertion errors remain. Indeed, this issue should be alleviated when using a language model and a pronunciation dictionary. Nevertheless, there may be room for improvements in the way we model the dynamics of lips and hand in continuous CS.

We remark that, in this study, the ground truth phonemes are given by the content of the audio, instead of the actual phonemes encoded by the CS interpreter. A fair comparison with the state of the art [10] is difficult, as they are not based on the same corpus. However, we still get that the proposed tandem CNN-HMM gives a score 72.6% comparable to the result in [10], where visual artifices are used to help the lips and hand tracking for the isolated phoneme recognition.

According to Fig. 8.16, we can see that vowels have more confusions than consonants. From the previous Section 8.3.3, we know that the performance of the hand position (vowel) recognition is not very satisfied. Therefore, we can conclude that these confusions that appear on the vowel part may come from it. For the consonants, quite naturally, some of the substitution errors are made on phonemes with similar lip shapes, such as {[p], [b], [m]} and {[f], [v]}. For other consonants which also have the similar lips shape such as {[t], [d], [n]}, {[k], [g]} and {[ʃ], [ʒ]}, no confusions appears. It shows that hand shape helps the lips to distinguish these consonants. Moreover, It demonstrates the significant benefit of the CNN based feature extractor.

In addition, we tried to use LSTM to replace the HMM-GMM decoder for the continuous CS recognition to capture the long-term memory in the CS. In LSTM, two hidden layers of 500 cells, 200 epoch are used. It is trained by BPTT with the cross-entropy cost function. Softmax layer is used to compute the class probability. LSTM is implemented using the Keras toolkit [207] based on the GPU-accelerated library. The experimental result for the phoneme recognition is shown in Fig. 8.20. We can see that, in the training set (350 sentences), the

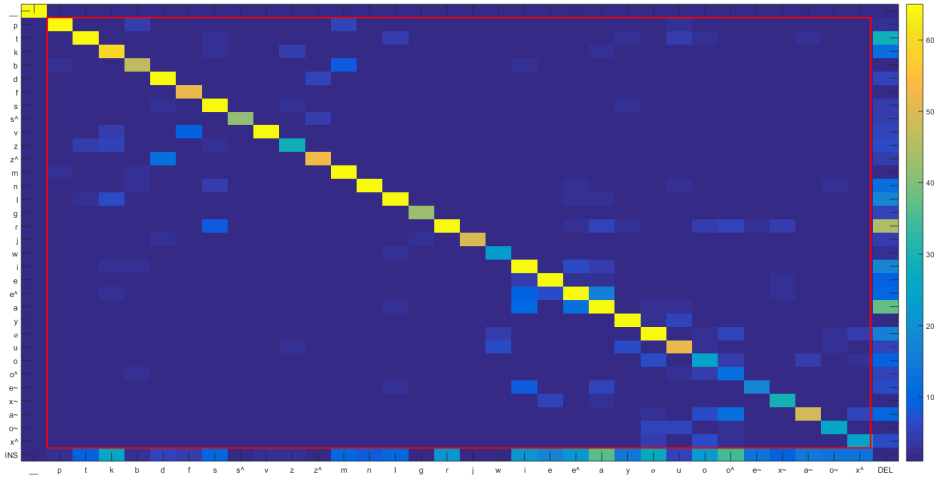


Figure 8.16: Confusion matrix of the CS phoneme recognition. The last row shows the insertion, and the last column shows the deletion errors. The red rectangle contains 33 concerning phonemes except the silence, insertion and deletion error elements.

recognition score is pretty high which means that the model fits well the training data, while in the test set (150 sentences), the recognition score is much lower. This huge difference comes from the over-fitting caused by the limited amount of the data.

8.3.5 Evaluation of the resynchronization procedure

In order to evaluate the proposed resynchronization procedure (i.e., AC concatenation) introduced in Section 8.2.3, we apply the proposed resynchronization procedure to the continuous CS phoneme recognition using $S_{3-resyn}$ (see Fig. 8.17). We compare the result of using $S_{3-resyn}$ with that using S_3 which does not take into account the resynchronization procedure. The results are shown in Fig. 8.18. We analyze the results and draw the following conclusions.

When using the hand position given by the ABMMs and the architecture S_3, the continuous CS phoneme recognition obtains a recognition correctness of 71.0% without using the resynchronization procedure. When the proposed resynchronization is incorporated, the recognition correctness increases to 72.7% (in Table 8.4) which shows a minor improvement (about 1.6%) compared with 71.0% (see 3rd and 4th columns in Fig. 8.18). However, we observe that in the current recognition system, the triphone context-dependent modeling is helpful to correct the recognition errors which are the co-articulation or the asynchrony of the multi-streams [148], [186], [187]. Thus, the use of the context-dependent modeling may hide the effect of the resynchronization procedure. In order to get rid of this effect, we examine the correctness of the CS phoneme recognition without using the context-dependent modeling. In this case, a correctness of only 60.4%[10] is obtained without any resynchronization procedure, while the

[10] This low correctness of 60.4% is obvious since no context-dependent modeling is used.

correctness increases to 64.38% when using the proposed resynchronization procedure (see 1st and 2nd columns in Fig. 8.18). We can see that about 4% improvement is achieved in this case. This improvement is more evident than the case using the context-dependent modeling (1.6%).

There are two reasons which can explain the above weak improvement: (1) only a weight of 20% is applied to the hand position stream. This small weight reduces the importance of the resynchronization procedure. (2) the hand position extracted by the ABMMs may have some errors. This directly reduces the efficiency of the proposed resynchronization procedure since the hand position target can be identified only if the correct hand position is used at a good temporal boundary for vowels. Note that we assume that the CNN-based lips and hand shape features are correct enough in Section 8.3.2.

Concerning the second reason, to use a correct hand position, we use the ground truth hand position instead of the one given by the ABMMs. We manually determine the hand position for all the images in the database. The continuous phoneme CS recognition results using the ground truth hand position are shown in 5th to 8th columns of Fig. 8.18.

When the ground truth hand position is used, without context-dependent modeling and the resynchronization procedure, we obtain a score of 62.83% which is similar to the result 60.4% based on the hand position given by ABMMs. It can be explained by the above first reason. However, when the resynchronization procedure is used, a correctness of 70.1% is achieved, which shows a significant improvement, i.e., 7.3% compared with 62.83% (see 5th and 6th columns in Fig. 8.18).

Concerning the case that using the context-dependent modeling (see 7th and 8th columns in Fig. 8.18), without using the resynchronization procedure, the recognition correctness is 72.04%. However, when combining them in the recognition system, an evidently higher score of 76.63% is obtained (with an improvement of 4.6% compared with 72.04%). It outperforms the state of the art 74.4% [10], which is for the isolated CS phoneme recognition case.

The proposed resynchronization procedure is specifically developed to solve the CS feature fusion caused by asynchrony problem. Its excellent performance is confirmed especially when the used hand position feature is correct.

Furthermore, in order to let the system learn the hand position automatically instead of adding the pre-processed hand position either obtained by the ABMMs automatically or manually. We use CNNs to process the fixed hand ROI[11] to obtain the hand position information (see Fig. 8.19). Because the hand position information is included in a fixed hand ROI, using this system obtains a phoneme recognition accuracy of 73.77% with a standard deviation 0.65. In other words, using the hand position automatically obtained by CNN (processing the fixed ROI) is better than using the hand position obtained by ABMMs (72.67%). However it is still worse than using the ground truth hand position (76.63%).

[11] The fixed hand ROI means a rectangle containing hand shape with the reference of a fixed point in the image.

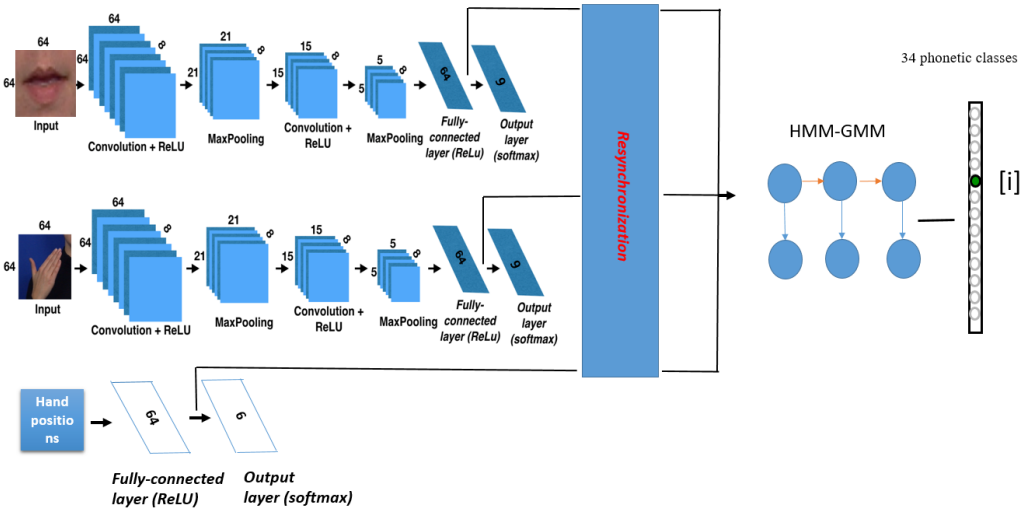


Figure 8.17: The architecture $S_{3\text{-resyn}}$ is the one with the resynchronization procedure rectangle step. Without the resynchronization procedure step, this figure shows architecture S_3 in [208].

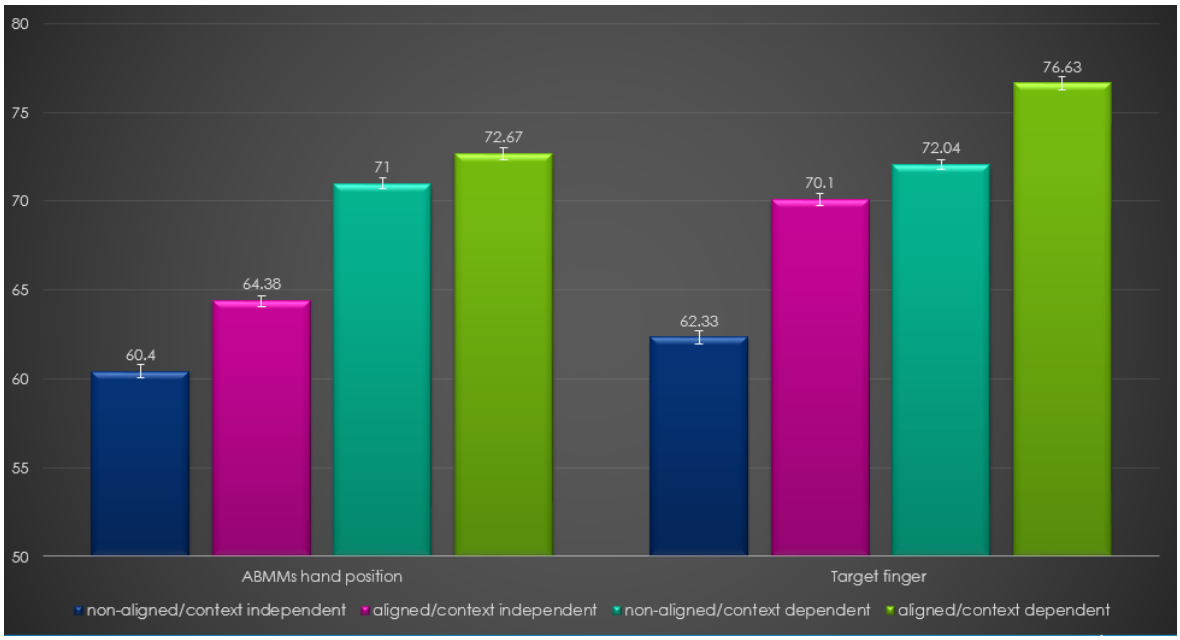


Figure 8.18: The performance of the resynchronization procedure and the context-dependent modeling in continuous CS phoneme recognition.

8.4 Summary

In this chapter, a set of tandem CNN-HMM architectures are proposed for the automatic recognition of CS. Compared with the state of the art, main improvements are the recognition of the continuous CS and the absence of visual artifices used to help the tracking of lips and hand. In the viseme recognition, we find that the hand shape stream gives the highest recogni-

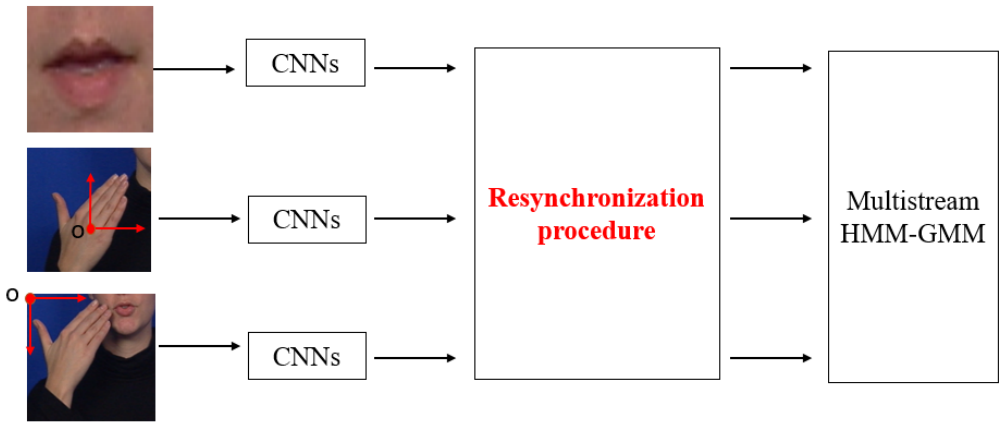


Figure 8.19: Image level architectures of phoneme recognition.The hand position feature is obtained by CNNs based on a fixed image.

tion score mainly thanks to the good hand shape features given by CNNs. This is exactly the main reason of introducing CNNs to CS recognition framework. Combined with resynchronization procedure, the best architecture is based on the model-level fusion strategy within the HMM-GMM decoder. Combined with the feature-level and model-level fusion strategies, a context-dependent HMM-GMM is used to tackle the asynchrony of lips and hand stream in CS, as well as the variations in the continuous CS recognition. A satisfied performance of S_3 (correctness 72.6%) is comparable to the previous work (correctness 74.4%) which is for the isolated recognition case. Moreover, from the recognition results of S_3, we see the significant benefits of CNNs (with correctness 73.0%), compared with the non-discriminative PCA method (correctness 51%). Moreover, an optimal performance 76.6% of the CS continuous phoneme recognition is achieved using the ground truth hand position.

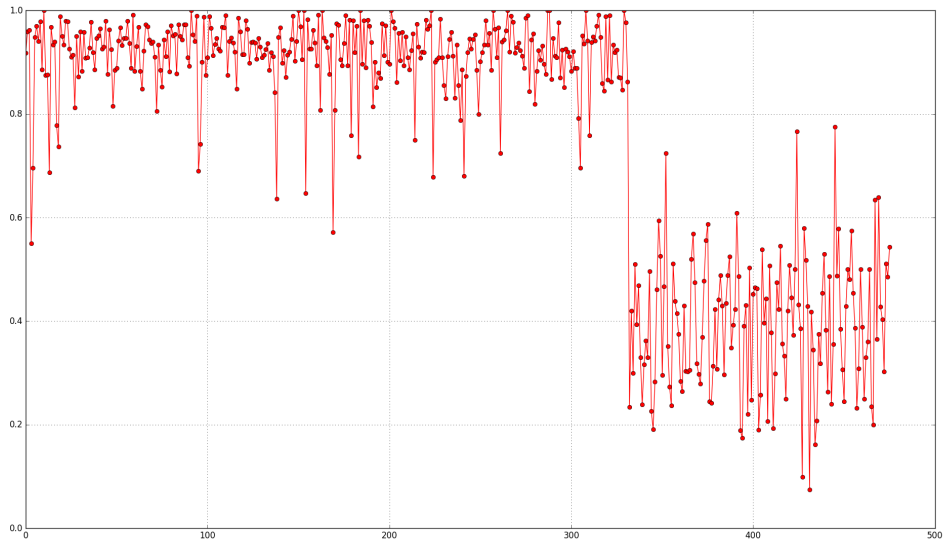


Figure 8.20: Phoneme recognition using LSTM. The previous 350 scores correspond to the training set with higher recognition scores, and the rest 150 scores correspond to the test set with lower recognition scores.

General Conclusion and Perspective

Conclusion

In this PhD thesis, we focus on the automatic continuous Cued Speech (CS) recognition which is situated in the *multi-modal speech recognition*. The feature extraction, multi-modal fusion and automatic speech recognition problems are investigated by the methods in *signal processing, statistics, machine learning* and *deep learning*. To conclude, this thesis mainly contributes to the following three problems.

(1) Automatic CS feature extraction without using any artificial mark.

In the prior work, CS features were extracted by tracking the artificial landmarks on speaker's lips and hand. Our aim is to get rid of these artifices. Recall that the automatic CS feature extraction contains lips, hand position and shape. Now we summarize our work concerning the feature extraction.

Concerning the **lips feature**, we propose two novel methods to extract the inner lips parameters based on the raw images. The first approach, Modified Constrained Local Neural Fields (Modified CLNF), is based on Constrained Local Neural Fields (CLNF) which is robust for facial landmark detection in current computer vision field. As CLNF presents mistakes in about 41.4% of the cases for inner lips tracking, this method is developed to correct these errors by a post-processing procedure in two steps corresponding to B and A parameters, respectively. For B parameter, the Hybrid Dynamic Correlation Template Method (HD-CTM) based on the correlation with a hybrid dynamic template is investigated to detect the outer lower lips position. Then, the inner lower lips position is determined by the back-subtracting of Validated Lower Lips Thickness (VLLT). The evaluation of this method on about 4800 images of three speakers confirms the performance. In fact, the Root Mean Square Error (RMSE) reduces from 4 pixels (2mm) to 1 pixel (0.5mm). For A parameter, the *periodical spline interpolation* based on the dilated 6 CLNF inner lips points is investigated. This method is tested on 467 round lips images among 3184 vowel images. 91% of the vowels in the third viseme are located in the proper position after the correction of A parameter.

The second approach is the *adaptive ellipse model* which is also proposed for the parameter extraction of inner lips without using any artifice. This approach first extracts the inner region of lips with image processing combined with the single discontinuity elimination and the interrupted region filling. Then, an adaptive ellipse is used to match the inner lips region and gives the best estimation of A and B parameters. The precision is evaluated on 4691 images of three speakers (*MD, DB* and *ChS*). This approach permits to obtain RMSE of 3.37mm for A parameter and 0.84mm for B parameter, which outperforms the state of the art [3]. A recognition based on 13 French vowels also confirms the superior performance.

Concerning the **hand position feature**, we propose to use an efficient method to extract

the hand position. This method is based on the Adaptive Background Mixture Model (ABMM) which uses Gaussian Mixture Model (GMM)s to model the background of the image and characterize the foreground and background pixels in an image. The speaker's deformable hand is regarded as a foreground in the video. The gravity center of the detected foreground pixels is considered as the hand position in our case. In order to evaluate the performance, we first compare the estimated hand positions with the ground truth, and then perform a hand position recognition using Gaussian classifier. Even though this method cannot reach the performance of the ground truth, it permits to track the hand position with a certain accuracy (62.26%).

Concerning the **hand shape feature**, the classical methods Active Appearance Model (AAM), Constrained Local Model (CLM) and Kanade–Lucas–Tomasi (KLT) do not work very well since the hand shape keeps changing when the hand moves. In this thesis, instead of extracting the hand contour, we apply a nonlinear Convolutional Neural Network (CNN) to directly capture the high-level hand shape features from the raw images. This feature extractor is able to adjust the weakly temporal segmented context. Compared with the linear Principal Component Analysis (PCA), it achieves much better hand shape features with a satisfied correctness (76.34%). The consonant recognition based on the fusion of lips and hand shape features obtains a correctness of 84.35%, which outperforms the state of the art [10].

(2) Temporal segmentation of the hand movements in CS.

We investigate a specific study concerning the temporal organization of the hand movements in CS, especially concerning its temporal segmentation. As we know, the asynchrony problem of lips and hand movement is a challenging issue for the CS recognition. This phenomenon makes it difficult for the hand to share the same audio-based temporal segmentation with the lips stream. In this thesis, we show that for a typical CV syllable, during the hand movement, the speaker tends to reach a specific position to indicate the vowel, and his hand shape changes in order to prepare the consonant of the syllable. We perform a detailed study by measuring the hand preceding time for hand positions (vowels) and hand shapes (consonants). It is shown that the hand preceding time for vowels has almost the same distribution (with a mean value of 140ms for corpus LM) from the beginning of a sentence to about 1s before the end. Then this preceding time decreases linearly. We also observe that the best moment to identify a hand shape is about 60ms before the target instant of the corresponding audio signal. This allows us to elaborate a segmentation method for CS hand movements using the audio-based temporal segmentation. In case of vowels, this method permits to predict the instant where the hand reaches its target position and also provide a good base for the resynchronization procedure. Our evaluations confirm the superior performance of the proposed method. In fact, the hand position recognition performance using the predicted segmentation significantly outperforms that using the audio based segmentation with the Gaussian classifier and Long Short-Term Memory (LSTM).

(3) Continuous CS recognition which incorporates the multi-modal fusion and context-dependent modeling.

For the automatic continuous CS recognition, we investigate three CNN-HMM tandem architectures S_1-S_3, which are composed of CNN and Hidden Markov Model (HMM) with feature-level and model-level fusion methods. In these architectures, a triphone *context-dependent modeling* is used to capture the context information for continuous CS. Meanwhile, due to the asynchrony problem in CS, a resynchronization procedure Aligned Concatenation (AC) is applied to the three modalities before the HMM modeling. In S_1, a single CNN jointly models the lips, hand position and shape streams, which are regarded as a global Region of Interest (ROI) in CS. Then the audio-based temporal segmentation is used to train the CNN with 34 phonetic classes as targets, and the trained CNN is applied to extract the features. Finally, these features are fed to a one-stream HMM-GMM for phonetic decoding. In S_2 and S_3, three streams are separated independently. We use a CNN for the lips or hand shape stream, and an Artificial Neural Network (ANN) for the hand position. In S_2, three-stream features are concatenated (i.e., feature-level fusion) in a single feature vector and fed to a one-stream HMM-GMM. In S_3, lips and hand information are combined at the state using a three-stream HMM-GMM (i.e., model-level fusion). These two architectures are both trained with a set of lips visemes, hand shape and position groups. In the best architecture S_3, without exploiting any dictionary or language model, the accuracy at phonetic level is 62% (with the correctness of 72.7%). In fact, in these architectures, we find that the context-dependent modeling contributes about 10% in the recognition score. The result of S_3 is comparable to [10], which is for the isolated CS recognition with visual artifices. This architecture is a good candidate for the practical use in CS recognition.

Based on the ground truth hand position and the resynchronization AC procedure, we find that the phoneme recognition correctness increases about 8% compared with the one without this procedure. The phoneme recognition achieves an optimal 76.6% correctness based on AC procedure and context dependent modeling, which clearly outperforms the state of the art [10].

Perspective

Apart from the above contributions, in CS recognition, there is still lots of work deserving more research and attempts. Now we summarize these future work as follows.

(1) Use of a language model to decrease the insertion errors and improve the robustness of the CS recognition system.

As introduced in Chapter 8, we observe a large number of insertion errors in the current CS recognition system. It was shown in [213]–[215] that a language model could help to improve the robustness of a recognition system, as it allows the recognizer to predict the probability of each phoneme given the previous phonemes. Therefore, it will be interesting to apply a language model to our CS recognition system in the future work.

(2) Design of an end-to-end CS recognition system combined with CNNs, RNNs and CTC.

It will be interesting to build an end-to-end CS recognition system by merging the CNNs, LSTM and Connectionist Temporal Classification (CTC) as one global neural unit (see Fig. 2). Firstly, CNNs extract the lips and hand shape features from raw images. Then, the Bi-LSTM will be applied to model the temporal information of the features from CNNs. Finally, a CTC loss layer will be added to avoid pre-aligning the data.

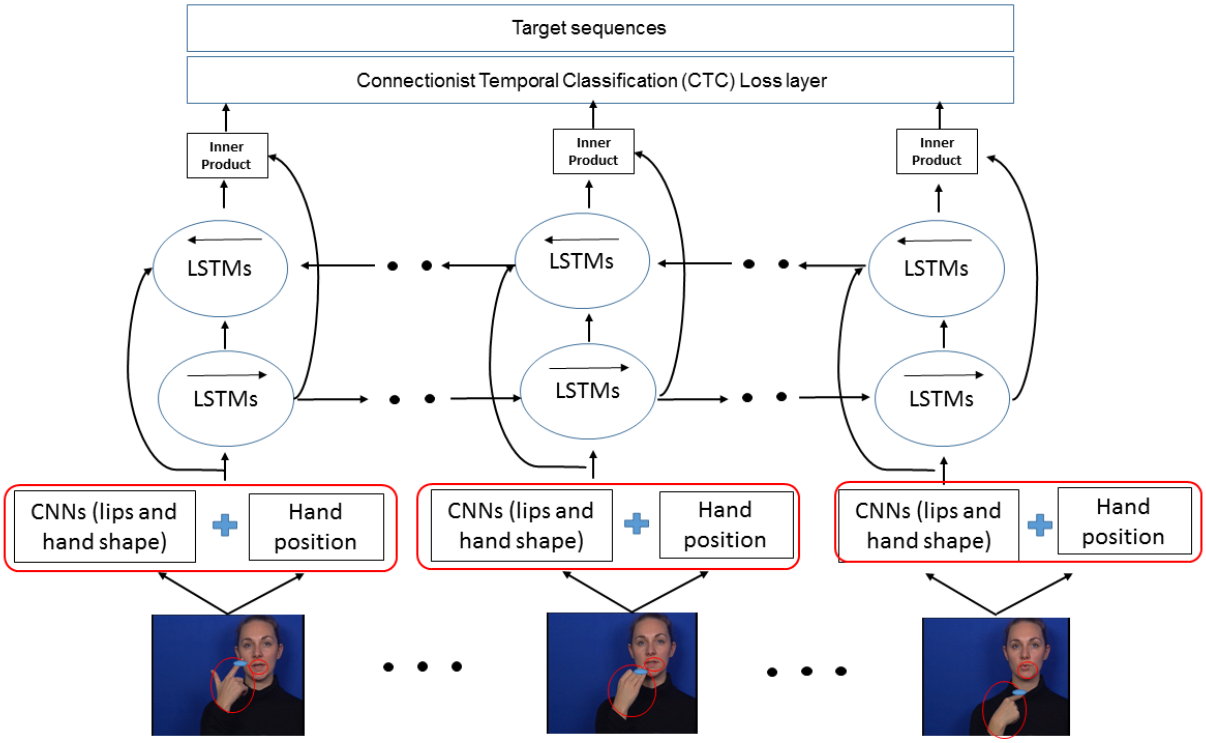


Figure 2: Architecture of the end-to-end CS recognition system in the future work.

(3) The attention mechanism and DCCA for multi-modal fusion in CS. Besides, based on our current tandem CNN-HMM architecture, the intermediate (i.e., the product HMM[12]) and decision fusion will be conducted.

As mentioned in Section 4.3.2, the multi-modal fusion in CS recognition can be seen as a sequence-to-sequence mapping problem. One popular solution is to apply an encoder-decoder framework, which is incorporated with the attention mechanism. Attention to the appropriate modalities, as a function of the context, may help to improve the quality of the CS recognition system. Notably, a recent popular method Deep Canonical Correlation Analysis (DCCA) for feature fusion will be tested in our case concerning the vowel and consonant recognition which just needs the integration of two stream features. However, multi-modal DCCA needs to be developed for the CS phoneme recognition. Moreover, our current tandem CNN-HMM architecture has implemented the feature-level and model-level fusion, but no intermediate and decision fusion. Therefore, one possible future work is to incorporate them into the CNN-HMM architecture.

[12] The product HMM is able to take into account the asynchrony problem of multi-modalities.

(4) Recording more CS database to make a more powerful experimental validation.

It is always important to have a large database in the deep learning framework, as it is able to reduce the over-fitting. In this thesis, we are not able to validate our proposed methods in the corpora of other languages. As mentioned in Chapter 2, the British English CS corpus is in the recording process. Moreover, even though our current database is able to validate the CNN based architectures, as we discussed in Chapter 8, when applying LSTM to CS recognition, we have an inherent over-fitting problem caused by the limited amount of database. A large CS database is a prerequisite for obtaining satisfied results when using the deep learning methods.

(5) An improved method for the hand position feature extraction.

As mentioned in Chapter 6 and Chapter 8, given a correct hand position feature, the correctness of vowel recognition and phoneme recognition can be improved by a large margin. This motivates us to explore a more robust hand tracking method to extract the hand position in CS. As introduced in Chapter 2, the *open-pose* system which incorporates the methods in computer vision and deep learning has good performance in tracking the hand gestures. However, this system cannot be directly used in our current database, since it requires a database with the whole upper body of subjects. It will be meaningful to improve our CS database in the future so that we can take advantage of this system.

(6) The Chinese Cued Speech system.

The CS system has been developed for more than 60 languages in the world[13], but as far as we know, no official research work[14] has been dedicated to the Chinese version of CS. According to the *China Disabled Persons' Federation*[15], about 21 million people have hearing loss out of the 60 million disabled people in China. For the moment, SL is the most popular method for the communication among Chinese deaf people. It will be meaningful if the Chinese Cued Speech (CCS) system can be developed, which will make the deaf people (especially deaf children) in China have better communications. We propose a possible design of this system in Fig. 3, which follows the main criterion that there is no intersection between hand and lips codings (i.e., the phonemes with similar lips shape should be distinguished by different codings). In Fig. 3, eight groups of hand positions and eight hand shapes are used to code vowels and consonants[16], respectively. The phonetic transcriptions of Chinese characters (including 23 consonants, 24 vowels and 16 whole syllables[17]) are shown in Fig. A.1. In the future work, we will focus on the analysis and exploration of the optimal approach to the CCS.

[13] http://www.cuedspeech.org/cued-speech/about-cued-speech.
[14] An analysis of Chinese CS can be referred to https://acroakingdalek.wordpress.com/2015/09/23/cued-mandarin-planting-the-seed/.
[15] http://www.cdpf.org.cn/english/.
[16] It should be noted that the CCS in Fig. 3 is just an initial trial, not the final validated version.
[17] The whole syllables in Chinese language are some fixed matches of some specific initials and finals. The pronunciation of the whole syllable will remain the same as the consonant even after adding a vowel behind the consonant or these syllables are to be read directly without being spelling from consonant to vowel.

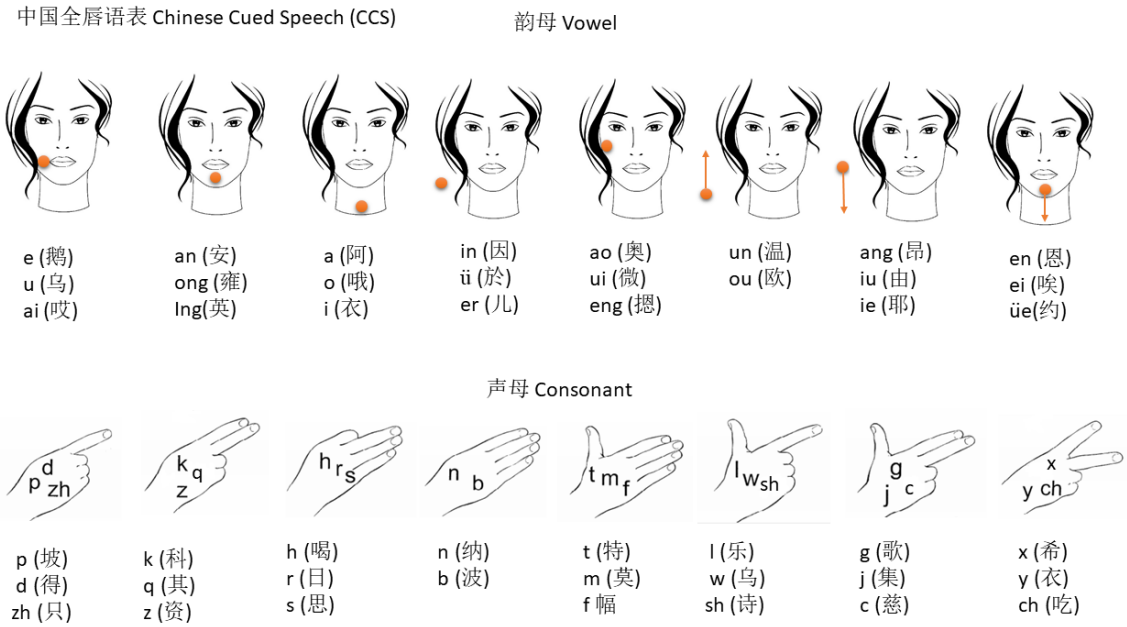


Figure 3: An initial design of the CCS system.

Résumé en Français

Contexte et motivation

La note de World Health Organization (WHO)[18] indique que de nos jours plus de 5% de la population mondiale (environ 432 millions d'adultes et 34 millions d'enfants) souffre d'une perte auditive invalidante, et on estime que ce nombre augmentera de plus de 900 millions d'ici 2050. Par conséquent, il y aura une demande sérieuse de méthodes automatiques pour aider ces personnes à communiquer plus facilement et mieux. En fait, dans la communauté des personnes sourdes utilisant la voie orale, la lecture des lèvres est l'une des principales modalités de perception de la parole, et ses avantages ont été largement admis depuis longtemps ([1]–[3]). Cependant, la parole peut être difficilement perçue si le lecteur de lèvres n'a aucune connaissance du contexte sémantique en raison de l'ambiguïté des lèvres (différents phonèmes ayant des formes aux lèvres similaires). Ainsi les meilleurs systèmes automatiques de lecture labiale atteignent seulement un taux de 76,2% en reconnaissance de mots (base de données *Lip Reading in the Wild* (LRW) [4]).

Pour surmonter cette difficulté et améliorer la capacité de lecture des enfants sourds, Cornett [5] en 1967 a développé le system du Cued Speech (CS) [5]–[9] qui utilise les gestes de la main pour compléter l'information des lèvres afin de rendre tous les phonèmes des langues parlées clairement visibles. Bien que de nombreux phonèmes semblent identiques sur les lèvres (par exemple, [p], [b] et [m]), ils peuvent être distingués grâce au système du CS en ajoutant les informations de la main, ce qui permet aux personnes sourdes utilisatrice de ce système de percevoir complètement la parole à partir des informations conjointes de lèvres et de main.

Pour améliorer les communications entre les personnes malentendantes et les personnes entendantes, il sera utile de développer des systèmes de conversion automatique de la modalité visuelle à la modalité audio et inversement de la modalité audio à la modalité visuelle. Cette thèse porte sur la reconnaissance automatique et continue de la Langue française Parlée Complétée (LPC), version en français du CS, dans la conversion de la modalité visuelle au texte où le texte jouerait le pivot entre le mode visuel et l'audio. Son cadre se situe dans le traitement de la parole multi-modale, et est à la croisée de la Communication Homme-Machine, l'Intelligence Artificielle et la vision par ordinateur.

Enjeux

Un système de reconnaissance du LPC nécessite une reconnaissance automatique sophistiquée pour décoder non seulement les lèvres du locuteur mais aussi les mouvements de sa main. Dans cette thèse, la reconnaissance du LPC est réalisée à partir de trois procédures principales:

[18] http://www.who.int/en/

l'extraction de caractéristiques, la fusion multi-flux (lèvres et main) et la reconnaissance du CS en parole continue, qui peuvent être schématisés par la Fig. 4.

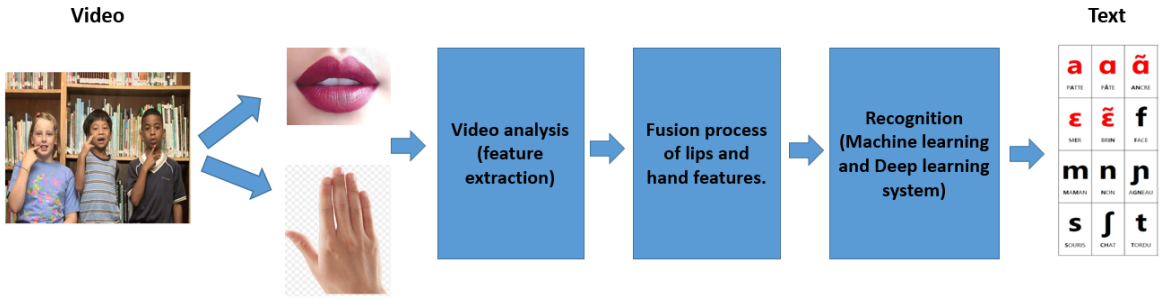


Figure 4: Cadre de la reconnaissance automatique du CS de la modalité visuelle au phonème.

A partir de la Fig. 4, trois enjeux principaux peuvent être résumés comme suit.

(1) Le premier est l'extraction de caractéristiques sans utiliser de marquage de l'information pertinente préalable à l'enregistrement des données. Dans l'état de l'art de la technique sur la reconnaissance du LPC [10], [11], les données vidéo étaient enregistrées avec des artifices (bleu sur les lèvres, pastilles de couleur sur la main et le front) pour faciliter l'extraction des paramètres des lèvres et de la main. A notre connaissance, aucun travail n'a tenté d'extraire des paramètres de lèvres et de la main à partir des images brutes en LPC et CS. Par conséquent, supprimer l'utilisation de ces artifices dans l'étape d'extraction des caractéristiques du LPC est notre premier défi. Et au-delà du seul intérêt de s'affranchir de marqueurs, c'est la possibilité de pouvoir extraire d'autres paramètres plus riche en information.

(2) Le second est la segmentation temporelle des mouvements de la main en LPC. En supposant que le mouvement des lèvres est quasi-synchronisé avec le son audio [12]–[14], il peut être obtenu la segmentation temporelle des lèvres à partir de l'alignement phonétique sur l'audio. Cependant, puisqu'il est établi que les lèvres, la position de la main et la forme de la main sont asynchrones pour le LPC, la main ne peut pas partager la même segmentation temporelle avec le mouvement des lèvres. Dans l'état de l'art sur la reconnaissance du LPC [10], [11], la segmentation temporelle des lèvres, de la position de la main et de la forme a été directement réalisée par alignement sur le signal audio. Ainsi, dans cette thèse, notre deuxième défi est de prendre en compte ce problème d'asynchronie entre ces trois flux, et de proposer des méthodes pour segmenter automatiquement la position de la main et le mouvement de la forme de la main en fonction du temps.

(3) Le troisième est le système de reconnaissance de phonèmes à partir du LPC en parole continue qui prend en compte la fusion des flux caractéristiques désynchronisés. Premièrement, concentrons-nous sur le mot-clé "continu". Ici, nous soulignons que dans cette thèse, une reconnaissance CS continue suit les deux conditions suivantes: (1) la donnée est une phrase (i.e., pas un mot isolé), (2) dans l'étape de test, la limite temporelle n'est pas donnée (c'est-à-dire que chaque image est envoyée au système de reconnaissance). En outre, dans l'étape d'apprentissage, nous utilisons la segmentation temporelle continue

(c'est-à-dire, pas d'espacement entre deux segments successifs). L'état de la technique [10] traitait de la reconnaissance du CS à partir de phonèmes extraits (c'est-à-dire, la limite temporelle des phonèmes dans l'ensemble de test est donnée) dans un corpus de phrases continues. La reconnaissance continue en LPC a été initiée avec [11], mais avec un corpus de mots isolés. De plus, les informations de contexte n'ont pas été modélisées dans [10] et [11]. Dans cette thèse, nous abordons la reconnaissance du LPC en parole continue à partir d'un corpus de phrases continues, dans lequel les phrases sont prononcées et codées normalement par les sujets. Ceci est beaucoup plus difficile que le cas isolé puisque la limite temporelle du phonème n'est pas donnée dans la phase de test. D'autre part comment fusionner les différents flux d'information dans le contexte de parole continue et donc d'une plus grande variabilité intra et inter modalités. En résumé, une reconnaissance LPC robuste qui prend en compte la fusion des différentes modalités, ainsi que l'information dépendant du contexte, est le troisième défi de cette thèse.

Résumé des chapitres

Dans cette thèse, les méthodologies de traitement du signal, de statistique, d'apprentissage automatique et surtout d'apprentissage profond sont utilisées pour résoudre ces enjeux. Nous avons développé dans ce mémoire plusieurs approches qui sont sont ci-après. Cette thèse contient deux parties: Part I et Part II.

Dans Part I (Chapter 1 à Chapter 4), nous présentons le contexte du CS et l'état de l'art de la reconnaissance automatique du LPC, ainsi que les données en LPC qui seront utilisées dans cette thèse. Les méthodologies d'apprentissage profond et la fusion multimodale sont également présentées.

Dans Part II, nous présentons nos principales contributions dans l'extraction de caractéristiques du LPC, l'organisation temporelle et la reconnaissance en parole continue du LPC.

(1) Dans Chapter 5, nous proposons deux nouvelles méthodes nommées Modified Constrained Local Neural Fields (Modified CLNF) [15], [16] et *adaptive ellipse model* [17] pour extraire les paramètres du contour interne des lèvres. D'autre part, les réseaux Convolutional Neural Network (CNN) [18], [19] sont appliqués sur la Région d'Intérêt (ROI) des lèvres brutes pour extraire les caractéristiques de haut niveau extraites des pixels. Pour les caractéristiques de la main (position et forme), nous proposons d'utiliser le Adaptive Background Mixture Model (ABMM) [20]–[22] pour extraire la position de la main, qui est pris comme le centre du ROI de la main. Après le suivi de la ROI manuelle, les CNN sont ensuite appliqués pour extraire les caractéristiques manuelles de la ROI des images brutes.

(2) Dans Chapter 6, nous concentrons sur l'extraction de la position de la main et la ROI de la main dans CS. La position de la main est suivie par les ABMMs, qui modélisent l'arrière-plan de l'image par Gaussian Mixture Model (GMM), et la ROI de la main est ensuite déterminé en fonction de la position de la main.

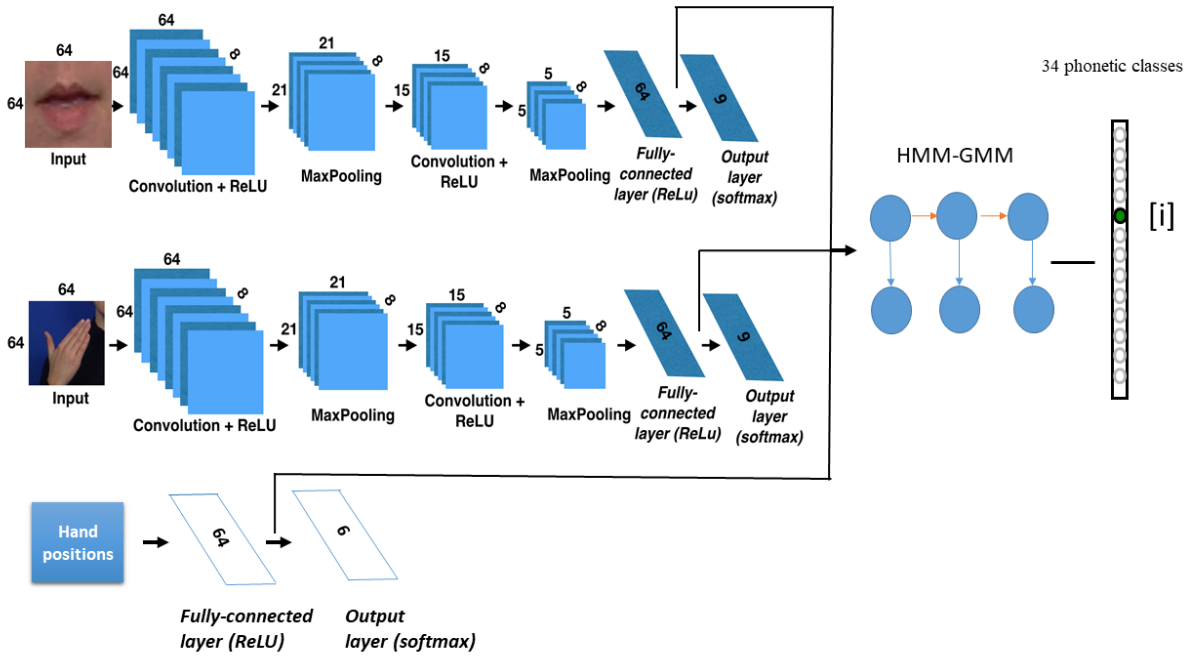


Figure 5: Extraction de caractéristiques basée sur CNN et décodage HMM-GMM dans le cas de S_3. Les Lèvres et les caractéristiques de forme de la main sont extraites par les CNN, et les coordonnées de la position de la main sont traitées par l'ANN. La stratégie de fusion au niveau des caractéristiques ou au niveau du modèle est utilisée en combinaison avec le décodeur HMM-GMM.

(3) Dans Chapter 7, nous proposons un modèle (*hand preceding model*) d'anticipation pour prédire automatiquement la segmentation temporelle du mouvement de la main en explorant la relation temporelle des mouvements des lèvres et l'emplacement des voyelles dans les phrases. Pour évaluer la performance de la méthode proposée, la reconnaissance des positions de la main est réalisée par GMM et Long Short-Term Memory (LSTM) [23] en utilisant différentes segmentations temporelles de la position de la main. Les résultats montrent que l'utilisation de la segmentation temporelle prédite par la méthode proposée améliore significativement les performances de reconnaissance par rapport à celle utilisant la segmentation basée sur l'audio. Pour le flux de forme de la main, nous proposons une segmentation temporelle optimale pour la réalisation de la forme de la main basée sur le modèle d'anticipation de la main.

(4) Dans Chapter 8, la fusion multi-modale asynchrone et la reconnaissance du LPC en continu sont réalisées en général grâce à plusieurs nouvelles architectures en tandem qui combinent CNN, Hidden Markov Model (HMM) [24], [25], différentes stratégies de fusion et modélisation dépendant du contexte. Une nouvelle procédure re-synchronisée Aligned Concatenation (AC) est proposée pour pré-traiter les entités multimodales afin de réduire l'effet de l'asynchronie et garantir la qualité de la fusion. Sans exploiter aucun dictionnaire ou modèle de langage, la meilleure architecture CNN-HMM (dans Fig. 5) proposée contient un modèle dépendant du contexte propre au LPC et peut identifier correctement environ 72.67% des phonèmes en parole continue. Notamment, ce résultat est comparable à l'état de l'art [10], qui était obtenu pour la reconnaissance LPC de phonèmes isolés extraits d'un corpus de phrases et l'utilisation de marqueur d'information. Enfin, cette thèse est le premier travail à traiter de la reconnaissance du LPC en continue à partir d'un corpus de phrases sans aucun artifice.

Publications and Activities

List of publications

1. Li Liu, Thomas Hueber, Gang Feng, Denis Beautemps. Visual recognition of continuous Cued Speech using a tandem CNN-HMM approach. *The Annual Conference of the International Speech Communication Association (INTERSPEECH)*, Sep. 2018, Hyderabad, India.

2. Li Liu, Gang Feng, Denis Beautemps. Automatic Temporal Segmentation of Hand Movement for Hand Position Recognition in French Cued Speech. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Apr. 2018, Calgary, Alberta, Canada.

3. Li Liu, Gang Feng, Denis Beautemps. Inner Lips features Extraction based on CLNF with Hybrid Dynamic Template for Cued Speech. *EURASIP Journal on Image and Video Processing*, 88, 2017.

4. Li Liu, Gang Feng, Denis Beautemps. Inner lips features extraction using Hybrid dynamic template for cued speech. Extend abstract, *BMVC-LRDLM*, 2017, London, UK.

5. Li Liu, Gang Feng, Denis Beautemps. Inner Lips Parameter Estimation based on Adaptive Ellipse Model. *International Conference on Auditory-Visual Speech Processing (AVSP)*, Aug. 2017, Stockholm, Sweden.

6. Li Liu, Gang Feng, Denis Beautemps. Automatic dynamic template tracking of inner lips based on CLNF. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Mar. 2017, New Orleans, USA.

7. Li Liu, Gang Feng, Denis Beautemps. Extraction automatique de contour de lèvre à partir du modèle CLNF. *31e Journées d'Études sur la Parole (JEP-TALN-RECITAL)*, Jul. 2016, Paris, France.

List of activities

1. Sep. 2 - Sep. 6, 2018, *The Annual Conference of the International Speech Communication Association (INTERSPEECH 2018)*, Hyderabad, India.

2. Apr. 15 - Apr. 20, 2018, *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2018)*, Calgary, Alberta, Canada.

3. Dec. 20, 2017, *Sephora Berribi Scholarships for Women in Advanced Mathematics & Computer Sciences* (two winners in France).

4. Oct. 30 - Nov. 10, 2017, *Visiting student*, CVSSP, University of Surrey, UK.
5. Aug. 28 - Sep. 1, 2017. *Data Science Summer School*, Paris, France.
6. Aug. 25 - Aug. 26, 2017. *The 14th International Conference on Auditory-Visual Speech Processing (AVSP 2017)*, KTH campus, Stockholm, Sweden.
7. Jul. 17 - Jul. 21, 2017. *International Summer School on Deep Learning*, University of Deusto and Rovirai Virgili University, Bilbao, Spain.
8. Jun. 28 - Jun. 30, 2017. *19ième conférence francophone sur l'Apprentissage Automatique*, Bâtiment Informatique et Mathématiques Appliquées de Grenoble, France.
9. Mar. 5 - Mar. 9, 2017. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2017)*, New Orleans, USA.
10. Feb. 1, 2017, *Association Francophone de la Communication Parlee (AFCP) Fund for Young Researchers*, France.
11. Nov. 28 - Nov. 29, 2016. *Workshop on Audiovisual Speech Processing and Language Learning*, Universitat Pompeu Fabra, Barcelona, Spain.
12. Dec. 13, 2016. *Journée Machine Learning Optimisation*, Lyon, France.
13. Jul. 4 - Jul. 8, 2016. *31e Journées d'Études sur la Parole (JEP-TALN-RECITAL 2016)*, Paris, France.
14. Apr. 28, 2016. The Best Poster Award of Doctor School in *Journée Doctorant EEATS*, Grenoble, France.

Appendix A

Additional Materials

A.1 Text file of sentences

This text file contains 238 French sentences, which were uttered twice by the subject *LM* using CS.

Ma chemise est roussie.
Il se garantira du froid avec ce bon capuchon.
Voilà des bougies.
Dès que le tambour bat les gens accourent.
Donne un petit coup.
Les deux camions se sont heurtés de face.
Tiens-toi assis.
Annie s'ennuie loin de mes parents.
Il a du goût.
La vaisselle propre est mise sur l'évier.
Elle m'étripa.
Une réponse ambigüe.
Louis pense à ça.
Un loup s'est jeté immédiatement sur la petite chèvre.
Un four touffu.
Nos dalmatiens campaient au camping à la montagne.
Un tour de magie.
Noam Chomsky balaye encore le club ce soir.
Voilà du filet cru.
Huit jésuites très huileux se font un brushing yougoslave.
La force du coup.
L'africa song s'emballe en juillet sur un walkman muet.
Prête lui seize écus.
Je souhaite que sa peau usée ne reçoive jamais cette greffe ridicule.
Vous êtes exclue.
Il abrase chaque jour un pneu ancien avec ses griffes pointues.
Il fait des achats.
Ce fou ordinaire fiche le turban indien dans le bain optionnel.
Chevalier du gué.

Le géologue trouve finalement la houille en vrac dans le gave de Pau.
Le jeune hibou.
Va dans une cave quelconque et caches-y ce drapeau honteux.
Il fume son tabac.
Des ewoks habitent la maison en paille du centre spatial.
Un piège a poux.
La grive perchée sur l'if noir couve toujours ce canif chinois.
L'examen du cas.
Le loup oublie son plan astucieux dans une poche chinoise.
Je suis à bout.
Les keums du wharf rament évidemment dans le paysage.
Elle a chu.
Pose calmement ta dague pointue sur cette étoffe carrée.
Je vais chez l'abbé.
Cette phrase particulière étouffe toute une strophe vertueuse.
Deux jolis boubous.
Il n'arrive nullement qu'une vague surgisse du hors-d'oeuvre.
Une belle rascasse.
La pin-up feind de tomber chez toi mais ne blague jamais.
Il part pour Vichy.
Faire la nouba.
C'est Louis qui joue.
Vous poussez des cris de colère.
C'est ma tribu.
David Bowie s'est rué sur le quai où j'ai organisé ce must.
Gilles m'attaqua.
Tu houspilles ton amant onctueux qui louche réellement.
Pas plus de quatre rubis.
J'ai un scorpion sec dans mon talon aiguille.
Une rocaille moussue.
Le beau ouistiti suit le riche huissier à Waterloo.
Un pied fourchu.
Jean Nohain a chargé Watson de louer le huitième buisson.
C'est lui qui me poussa.
J'ai identifié un mohican dans un western pyrénéen.
La chaise du bout.
Ce buveur balte augmente sa masse veineuse à heure régulière.
Trop d'abus.
Son gant entoure la valise trouvée sur la digue droite.
J'en ai assez.
Zola demande notamment du bon lait à un mage zurichois.
Jean est fâché.
Miss Zazie effectue un travelling heureux sur un machin imposant.
Le pied du gars.

Un pâle zébu agnostique mange normalement une solide pizza.
Vous avez réussi.
La meilleure omelette du Larzac peut rivaliser avec le yachting normand.
Ils n'ont pas pu.
Des rides charmantes aèrent cette robe choisie dans les pages jaunes.
Le vent mugit.
Aux lilas violets européens Corot eugène préfère vingt-et-un oeillets.
Une autre roupie.
Un coup heureux et impétueux modifie un vulgaire pain onctueux en gnome.
Deux beaux bijoux.
Tu ris beaucoup.
Mon père m'a donné l'autorisation.
Vend-on un cake intact à Hong-Kong.
L'avoué a besoin d'un joint sous huitaine.
La sueur suinte du thon huileux.
Tout winipeg attend Wendy sur le parking ouest.
Bud et Buck font un bon whist à Maubeuge.
Youri fouette l'ail ionique de Kohoutek.
Bing j'ai heurté le puits dans la lueur.
J'avais honte car la fille huait les who.
Vuitton fait cuire dix wapitis goûteux.
Young fait un petit huit avec un joueur noueux.
Li-peng met du nuoc-mam dans son amuse-gueule.
J'ai Eugène au téléphone qui cueille joliment du gui.
Ivanhoe a fait un bug au huitième essai.
Tu huiles l'étui du buzzer de deux watts.
C'est Hervé qui fuit dans un yacht en leasing.
J'ai étudié le parking huit à Plancoet.
Walid a hué les Pink Floyd à Rouen.
Eh oui les forums de l'accueil sont chouettes.
La famille ouistiti a éternué sous les dolmens.
Nous jouons aux billes dans les ruines muettes.
Le balai a fait un looping sur la toundra.
Ce tuyau a voyagé très haut chez les martiens.
J'ai huilé un rayon du train huit à l'équinoxe.
Les caïds jouent au ping-pong avec l'équipe de Bosnie.
J'ai eu les symptômes de la presbytie en huit jours.
Une agrafe géante a pu heurter son beau hors-bord.
De mauvaises gens privent Victor de sa coiffe bretonne.
Le vase zen a perdu aussi un anneau en roche grise.
La houle lave les hublots d'une case déserte.
Le photographe garantit un gag tordu au goût incertain.
Le bateau heurta les housses du hublot un peu humides.
La feuille fut sertie avec une dent usée de la biche docile.

Le prof mielleux triche souvent à ce jeu idiot.
Ce jazz rythmé est un cadeau inespéré.
Le veau heureux attend Eudes dans le hameau indien.
L'âne bègue voit que la vache de Joseph se vexe.
Lagaffe fabrique une ruche carrée si tu y coopères.
Rêves-y car l'extase vient de cette bague gracieuse.
Il élague curieusement la houpe qui est récalcitrante.
Le camp hostile coordonne le putsch dans la cohue.
Cette pêche fameuse a vu onduler l'endive blanche.
Il se lève chaque jour et attend Hercule qui oublie.
Quand je soulève ma hache le banc ondule.
Un zébu heureux ne touche jamais au houblon.
Ce chant hideux rase son héros venu en hâte.
Jean heurta une cuve large pleine de gouache verte.
Le vent établi sèche bien le houx où crèche mon hibou.
Tchang ôte sa toge cintrée d'une main innocente.
Un très bon vin en bouteille exige un planning idoine.
Don juan drague finalement une jeune fille mal faite.
A eux la soif zoologique du bourgeon ouvert.
Au yen la tache pénible de ce prêt embarrassant.
En haut la guêpe pense aux heures.
Objectez à Neuilly contre le gaz nocif des hommes.
L'anglaise lui offre ce qu'elle a au doigt ou à l'oreille.
Elle joue uniquement avec la neige chantante.
Oudini ignore le train où doit se produire le spectacle.
Il est parti illico en avion ou en gondole.
Il gobe douze fèves et bêche tout mon jardin.
La caisse seule a enflé sur le ring en bois.
Votre crêpe chaude vise bien le haut du feu.
Tailles-en un bien haut et travaille chaque nom.
Fernand oublie de moudre son café.
L'abeille n'engrange pas de miel sur un chemin.
Cherche où est le thon obtus que je trouve sot.
Eole aide sa robe fendue à se soulever.
Bashung oublie aussi qu'il lègue quelque chose.
Je passe chercher ce que j'ai lu avec vous.
Un zoom ferait ce que neuf demis pensent faire.
Le fou immerge son aiguille et brode finement.
Chaque bout du rail carré est une tige ténue.
Un argument élogieux échappe bien au rosbif.
Le malade guéri attrape mon solide microbe.
Cette dame veut galber un tube vertical.
Nous traquions bien Euler pendant son footing urbain.
J'ai vu un holding important sur un terre-plein escarpé.

Pain et pudding gallois aident le petit hussard oublieux.
Une bouteille de riesling heurta le balcon humide.
Ce jeu invite un type joueur et une dame riche.
Une vache normande dirige rarement un jumping zélé.
Le Viking honteux a mal chuté sur cette petite nappe.
Le moteur du boeing ronronne dans la brouette.
Le pape vient en yamaha dans une bourgade curieuse.
Le rotring exige une page carrée dans une feuille verte.
Le lapin utilise son yoyo et a besoin d'aide.
Le dumping l'incite à jeter les prunes tombées.
Les yétis mal rasés ont la bouille pâteuse.
Ils oublièrent chuck dans un tube carré.
Léon range le parking vendéen où on aime zoner.
Le king charmeur porte une chemise rouge foncée.
Yasmine aime ton standing japonais.
Gaspard blague mollement sur le leasing omniprésent.
Nous draguions le torrent pour trouver des crabes noirs.
Eux aussi aiment la tripe glorieuse un peu euphorique.
Oeuvrez pour l'ove du globe bleu des yeux.
Mes juges vont manger ce fichu yaourt à la truelle.
Ce soldat un peu honteux fait un job glorieux.
Cet ?il globuleux porte une lentille luisante.
La sage baleine zoophile n'a aucune patte valide.
Le prieur brade tout centime gagné.
La caille revient sans eux dans l'herbage gourmand.
Une guenon heureuse a vu un balcon ombragé.
Chaque garçon aime que le soleil brille.
Il y a un truc qui ondule dans la cage murale.
Tapes-en au noir sur une petite zone.
La fausse reine en tailleur agace Guy.
Nous tuons chaque chiot qui a été heureux.
Flambes-y une crêpe bretonne de gamme moyenne.
Chaque zéro est un looping tordu.
Un nain heurta une bogue charnue un onze janvier.
Une tombe ming ne passe jamais pour un karting belge.
Un homme jeune ne tombe pas pendant cette java.
La foule a afflué quand mon neveu heurta le RER.
Le thon heurta un bleuet.
Il a été heurté par un pêcheur.
Intonnes un u ou un euh à intervalles réguliers.
Ceux des gueux bigleux veulent libérer Bob Taylor.
Où était oxymel.
Le jeu ôtait illico au parfum oublié un fin bouquet d'embruns.
Le cousin chinois du tribun évalue au juge autrement le tissu invendu.

Le CE isole les engins communs aux deux charlots.
Une québécoise pleurnicheuse brandit Euclide lors des réunions.
Moreau étale immanquablement un déficit commun à la queue de l'UE.
Aladin élève chacun en symbiose avec le vieil ouzbek.
Chacun ignore son CE un peu un moment.
Avec un aplomb imparable nous avons chacun un CE énergique.
Cette énergie insensée grève un quinzième de Ugines.
Sur le zing chacun interprète l'atlas humblement posé sur l'ancien jabot.
Sa tape un peu impolie heurta Bernache un peu trop violemment.
Sylvain ne suit pas le parfum imprévu.
Ce cabot ombrageux fête son accession au pouvoir.
Un noir de jais évoque le front eurasien.
Ce suspect heurta le bibelot ancien un peu lourdement.
Le bedeau euphorique secoue l'anneau un jour par an.
Jojo heurta le défunt et le tua.
A jeun Antoine le heurte et cet accident le hantera.
Le LPE insiste et les PME ont signé.
Regardes il zigzague un peu vite.
Un huit dans l'eau a huilé l'un des tiroirs.
Railles un bourrin oisif.
Prends-le Euclide.
Tailles huit brins ouatés.
Je m'huile le corps dans ce lieu iodé.
Jourdain rajoute un pneu huileux.
Il se ouate le teint rebelle.
Antoine avait ouint son numéro huit.
J'ai reçu ton dessin hier.
Quantum suédois ou rituel wolof.
La secoueuse fait des percings linguaux.
Les gangs infligent des bings et des bangs périlleux sur une île.
La horde de hors-la-loi alpague bientôt l'épave galloise.
Ce soldat un peu honteux fait un job glorieux.
J'ai oublié le message.

A.2 Text file of words

This text file contains 50 French words, which were uttered ten times by the subjects *MD, DB* and *ChS*.

annuler.
aout.
au-revoir.
avril.

bonjour.
changer.
cinq.
decembre.
deux.
dix.
dix-huit.
dix-neuf.
dix-sept.
douze.
fevrier.
heure.
huit.
janvier.
juillet.
juin.
mai.
mars.
neuf.
novembre.
octobre.
onze.
quatorze.
quatre.
quinze.
rendez-vous.
seize.
sept.
septembre.
six.
treize.
trente.
trente-et-un.
trois.
un.
vingt.
vingt-cinq.
vingt-deux.
vingt-et-un.
vingt-huit.
vingt-neuf.
vingt-quatre.
vingt-sept.
vingt-six.

vingt-trois.
zero.

A.3 Phonetic transcriptions of Chinese characters

In Chinese language, there are 23 consonants, 24 vowels and 16 whole syllables (see Fig. A.1).

Figure A.1: Phonetic transcriptions (consonants, vowels and reading the whole syllables) in Chinese[1].

[1] https://www.pinterest.com/pin/417568196678494389/

A.4 Numerical details of confusion matrices

1. Confusion matrix (10×10) for the hand shape recognition in Fig. 8.12(a).

$$\begin{bmatrix}
190 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 156 & 0 & 3 & 2 & 3 & 3 & 0 & 1 & 26 \\
0 & 9 & 164 & 0 & 1 & 1 & 2 & 1 & 5 & 15 \\
0 & 8 & 0 & 105 & 1 & 4 & 2 & 1 & 3 & 31 \\
0 & 4 & 1 & 6 & 203 & 1 & 0 & 3 & 1 & 26 \\
0 & 5 & 4 & 6 & 0 & 120 & 3 & 1 & 0 & 15 \\
0 & 11 & 3 & 7 & 1 & 4 & 185 & 0 & 1 & 44 \\
0 & 1 & 1 & 0 & 0 & 0 & 0 & 20 & 12 & 11 \\
0 & 2 & 5 & 0 & 1 & 0 & 0 & 0 & 37 & 2 \\
0 & 82 & 10 & 23 & 13 & 10 & 13 & 8 & 7 & 0
\end{bmatrix}$$

2. Confusion matrix (10×10) for the lips viseme recognition in Fig. 8.12(b).

$$\begin{bmatrix}
190 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 188 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 4 \\
0 & 0 & 73 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 5 & 8 & 369 & 13 & 31 & 14 & 18 & 7 & 131 \\
0 & 2 & 0 & 4 & 82 & 0 & 1 & 2 & 19 & 9 \\
0 & 4 & 6 & 29 & 3 & 123 & 40 & 14 & 11 & 112 \\
0 & 4 & 5 & 17 & 1 & 15 & 614 & 5 & 2 & 60 \\
0 & 5 & 1 & 4 & 2 & 11 & 1 & 89 & 26 & 24 \\
0 & 3 & 5 & 11 & 7 & 23 & 17 & 44 & 313 & 75 \\
0 & 15 & 15 & 53 & 15 & 60 & 66 & 34 & 49 & 0
\end{bmatrix}$$

3. Confusion matrix (7×7) for the hand position recognition in Fig. 8.12(c).

$$\begin{bmatrix}
192 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 131 & 6 & 19 & 3 & 5 & 44 \\
0 & 48 & 56 & 34 & 22 & 27 & 93 \\
0 & 27 & 6 & 128 & 3 & 24 & 47 \\
0 & 3 & 1 & 4 & 167 & 9 & 22 \\
0 & 11 & 3 & 14 & 6 & 143 & 26 \\
0 & 40 & 14 & 34 & 27 & 50 & 0
\end{bmatrix}$$

4. Confusion matrix (15×15) for vowel recognition in Fig. 8.14.

$$\begin{bmatrix}
190 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 75 & 2 & 6 & 1 & 0 & 0 & 0 & 0 & 1 & 3 & 0 & 1 & 1 & 0 & 11 \\
0 & 2 & 78 & 7 & 3 & 0 & 1 & 0 & 1 & 0 & 0 & 4 & 0 & 3 & 0 & 9 \\
0 & 1 & 1 & 37 & 10 & 0 & 0 & 1 & 1 & 0 & 1 & 2 & 0 & 0 & 0 & 19 \\
0 & 10 & 2 & 3 & 113 & 0 & 2 & 0 & 3 & 2 & 7 & 1 & 1 & 0 & 1 & 38 \\
0 & 0 & 0 & 0 & 0 & 59 & 0 & 4 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 4 \\
0 & 0 & 0 & 1 & 0 & 1 & 113 & 0 & 0 & 13 & 0 & 0 & 3 & 7 & 8 & 13 \\
0 & 0 & 0 & 0 & 0 & 2 & 0 & 25 & 1 & 0 & 0 & 0 & 0 & 2 & 1 & 5 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 21 & 7 & 0 & 1 & 1 & 1 & 2 & 16 \\
0 & 0 & 0 & 0 & 0 & 1 & 8 & 0 & 3 & 12 & 0 & 2 & 2 & 4 & 2 & 14 \\
0 & 6 & 2 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 21 & 0 & 0 & 0 & 0 & 19 \\
0 & 0 & 5 & 7 & 2 & 0 & 0 & 0 & 2 & 0 & 0 & 47 & 0 & 0 & 0 & 8 \\
0 & 0 & 0 & 0 & 0 & 0 & 3 & 0 & 4 & 2 & 0 & 0 & 31 & 1 & 4 & 14 \\
0 & 0 & 0 & 0 & 0 & 3 & 4 & 0 & 0 & 0 & 0 & 0 & 2 & 20 & 0 & 6 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 6 & 0 & 0 & 0 & 0 & 0 & 12 & 5 \\
0 & 29 & 9 & 17 & 31 & 10 & 20 & 9 & 9 & 11 & 14 & 11 & 13 & 13 & 8 & 0
\end{bmatrix}$$

5. Confusion matrix (20×20) for consonant recognition in Fig. 8.15.

$$\begin{bmatrix}
190 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 71 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 71 & 0 & 0 & 0 & 0 & 0 & 3 & 0 & 0 & 0 & 0 & 2 & 3 & 0 & 1 & 2 & 0 & 11 \\
0 & 0 & 0 & 48 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 4 \\
0 & 8 & 0 & 0 & 46 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 11 & 1 & 0 & 0 & 1 & 0 & 0 & 4 \\
0 & 0 & 0 & 0 & 0 & 58 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 8 \\
0 & 0 & 0 & 0 & 0 & 0 & 32 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 4 & 0 & 0 & 2 & 0 & 76 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 7 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 26 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 37 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 4 & 0 & 1 & 0 & 2 & 0 & 0 & 18 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 \\
0 & 0 & 0 & 0 & 0 & 4 & 0 & 2 & 0 & 0 & 0 & 42 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 3 \\
0 & 2 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 38 & 1 & 0 & 0 & 2 & 0 & 0 & 2 \\
0 & 0 & 2 & 0 & 0 & 1 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 50 & 2 & 0 & 0 & 2 & 0 & 10 \\
0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 132 & 0 & 0 & 0 & 1 & 29 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 31 & 0 & 6 & 0 & 3 \\
0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 116 & 0 & 0 & 24 \\
0 & 0 & 0 & 1 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 34 & 0 & 5 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 32 & 1 \\
0 & 6 & 9 & 10 & 5 & 5 & 2 & 6 & 0 & 1 & 6 & 0 & 13 & 6 & 11 & 4 & 7 & 2 & 4 & 0
\end{bmatrix}$$

6. Confusion matrix (34×34) for phoneme recognition in Fig. 8.16.

.

$$\begin{bmatrix}
190 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 59 & 0 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 2 \\
0 & 0 & 53 & 1 & 1 & 1 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 1 & 3 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 1 & 2 & 0 & 3 & 2 & 1 & 1 & 1 & 2 & 0 & 1 & 21 \\
0 & 0 & 1 & 44 & 0 & 0 & 0 & 2 & 0 & 1 & 3 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 2 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 9 \\
0 & 2 & 0 & 0 & 35 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 6 & 1 & 0 & 0 & 1 & 1 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 64 & 0 & 0 & 0 & 0 & 0 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 4 \\
0 & 0 & 0 & 0 & 0 & 0 & 38 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 2 & 0 & 84 & 0 & 0 & 2 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 31 & 0 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3 \\
0 & 0 & 0 & 3 & 0 & 0 & 7 & 1 & 0 & 53 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 \\
0 & 0 & 3 & 4 & 0 & 0 & 0 & 2 & 0 & 0 & 21 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 5 \\
0 & 0 & 0 & 0 & 0 & 9 & 0 & 0 & 0 & 0 & 0 & 39 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3 \\
0 & 2 & 0 & 0 & 2 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 58 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 2 & 1 & 0 & 3 & 0 & 0 & 1 & 0 & 0 & 53 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 1 & 1 & 0 & 2 & 0 & 1 & 0 & 0 & 0 & 2 & 0 & 9 \\
0 & 0 & 2 & 5 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 113 & 0 & 0 & 0 & 1 & 0 & 0 & 2 & 2 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 13 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 32 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 6 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 126 & 0 & 0 & 0 & 1 & 2 & 4 & 2 & 1 & 0 & 3 & 4 & 2 & 3 & 0 & 0 & 1 & 33 \\
0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 2 & 0 & 37 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 17 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 \\
0 & 0 & 0 & 2 & 2 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 93 & 1 & 5 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 13 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 3 & 57 & 2 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 7 \\
0 & 1 & 0 & 2 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 7 & 5 & 55 & 12 & 0 & 0 & 0 & 1 & 1 & 1 & 2 & 0 & 0 & 0 & 8 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 & 2 & 0 & 1 & 0 & 0 & 8 & 0 & 9 & 108 & 0 & 2 & 2 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 28 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 84 & 2 & 4 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3 & 0 & 0 & 0 & 0 & 0 & 97 & 0 & 2 & 4 & 0 & 0 & 0 & 2 & 3 & 12 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 5 & 0 & 0 & 0 & 0 & 5 & 1 & 38 & 2 & 0 & 0 & 0 & 0 & 2 & 0 & 4 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 5 & 1 & 18 & 3 & 0 & 0 & 3 & 0 & 2 & 7 \\
0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 3 & 9 & 0 & 0 & 0 & 0 & 0 & 4 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 & 0 & 0 & 0 & 0 & 6 & 0 & 1 & 4 & 0 & 0 & 0 & 0 & 0 & 13 & 0 & 0 & 0 & 1 & 5 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & 0 & 2 & 0 & 0 & 0 & 1 & 0 & 1 & 22 & 0 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 2 & 0 & 5 & 9 & 0 & 0 & 36 & 1 & 4 & 8 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 4 & 4 & 0 & 2 & 0 & 0 & 0 & 19 & 0 & 1 \\
0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 3 & 0 & 5 & 2 & 0 & 0 & 1 & 1 & 18 & 5 \\
0 & 2 & 7 & 19 & 2 & 4 & 3 & 8 & 1 & 2 & 2 & 1 & 8 & 6 & 8 & 1 & 16 & 0 & 1 & 15 & 13 & 16 & 27 & 11 & 20 & 4 & 17 & 26 & 10 & 7 & 11 & 10 & 10 & 0
\end{bmatrix}$$

Bibliography

[1] W. H. Sumby and I. Pollack, "Visual contribution to speech intelligibility in noise," *The Journal of the Acoustical Society of America*, vol. 26, no. 2, pp. 212–215, 1954 (cit. on pp. 1, 173).

[2] Q. Summerfield, "Use of visual information for phonetic perception," *Phonetica*, vol. 36, no. 4-5, pp. 314–331, 1979 (cit. on pp. 1, 173).

[3] C. Benoît, T Lallouache, T Mohamedi, A Tseva, and C. Abry, "Nineteen (± two) French visemes for visual speech synthesis," in *The ESCA Workshop on Speech Synthesis*, 1991 (cit. on pp. 1, 35, 167, 173).

[4] J. S. Chung and A. Zisserman, "Lip reading in the wild," in *Proceedings of the Asian Conference on Computer Vision (ACCV)*, 2016, pp. 87–103 (cit. on pp. 1, 173).

[5] R. O. Cornett, "Cued speech," *American Annals of the Deaf*, vol. 112, no. 1, pp. 3–13, 1967 (cit. on pp. 1, 7, 10, 13, 173).

[6] G. H. Nicholls and D. L. Mcgill, "Cued speech and the reception of spoken language," *Journal of Speech, Language, and Hearing Research*, vol. 25, no. 2, pp. 262–269, 1982 (cit. on pp. 1, 12, 13, 173).

[7] C. J. LaSasso, K. L. Crain, and J. Leybaert, *Cued speech and cued language development for deaf and hard of hearing children*. Plural Publishing, 2010 (cit. on pp. 1, 7, 11, 173).

[8] C. LaSasso, K. Crain, and J. Leybaert, "Rhyme generation in deaf students: The effect of exposure to cued speech," *The Journal of Deaf Studies and Deaf Education*, vol. 8, no. 3, pp. 250–270, 2003 (cit. on pp. 1, 173).

[9] C. LaSasso, K. L. Crain, and J Leybaert, "Research and theory support cued speech," in *Odyssey*, vol. 5, 2003, pp. 30–35 (cit. on pp. 1, 173).

[10] P. Heracleous, D. Beautemps, and N. Aboutabit, "Cued speech automatic recognition in normal-hearing and deaf subjects," *Speech Communication*, vol. 52, no. 6, pp. 504–512, 2010 (cit. on pp. 2, 3, 11, 12, 15, 16, 20–22, 27, 39, 57, 58, 60, 61, 67, 107, 131, 137, 139, 140, 142, 156, 157, 161, 163, 168, 169, 174, 175, 177).

[11] P. Heracleous, D. Beautemps, and N. Hagita, "Continuous phoneme recognition in cued speech for french," in *Proceedings of the European Signal Processing Conference (EUSIPCO)*, 2012, pp. 2090–2093 (cit. on pp. 2, 15, 16, 20, 21, 23, 39, 58, 67, 131, 139, 140, 142, 174, 175).

[12] D. Rybach, C. Gollan, R. Schluter, and H. Ney, "Audio segmentation for speech recognition using segment features," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2009, pp. 4197–4200 (cit. on pp. 2, 13, 174).

[13] S. Tranter, K. Yu, G Everinann, and P. C. Woodland, "Generating and evaluating segmentations for automatic speech recognition of conversational telephone speech," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 1, 2004, pp. I–753 (cit. on pp. 2, 13, 174).

[14] S. J. Young and S. Young, *The htk hidden markov model toolkit: Design and philosophy*. Cambridge University, 1993 (cit. on pp. 2, 58, 146, 174).

[15] L. Liu, G. Feng, and D. Beautemps, "Extraction automatique de contour de lèvre à partir du modèle clnf," in *Actes des 31èmes Journées d'Etude de la Parole (JEP)*, 2016 (cit. on pp. 3, 4, 68, 175).

[16] ——, "Automatic tracking of inner lips based on clnf," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 5130–5134 (cit. on pp. 3, 4, 68, 175).

[17] ——, "Inner lips parameter estimation based on adaptive ellipse model," in *Proceedings of the International Conference on Auditory-Visual Speech Processing (AVSP)*, 2017 (cit. on pp. 3, 4, 68, 175).

[18] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, 2012, pp. 1097–1105 (cit. on pp. 3, 47, 175).

[19] S. Lawrence, C. L. Giles, A. C. Tsoi, and A. D. Back, "Face recognition: A convolutional neural-network approach," *IEEE Transactions on Neural Networks*, vol. 8, no. 1, pp. 98–113, 1997 (cit. on pp. 3, 175).

[20] K. Kim, T. H. Chalidabhongse, D. Harwood, and L. Davis, "Real-time foreground–background segmentation using codebook model," *Real-time Imaging*, vol. 11, no. 3, pp. 172–185, 2005 (cit. on pp. 3, 4, 111, 117, 175).

[21] C. Stauffer and W. E. L. Grimson, "Adaptive background mixture models for real-time tracking," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 2, 1999, pp. 246–252 (cit. on pp. 3, 4, 111–113, 117, 175).

[22] D. R. Magee, "Tracking multiple vehicles using foreground, background and motion models," *Image and Vision Computing*, vol. 22, no. 2, pp. 143–155, 2004 (cit. on pp. 3, 111, 117, 175).

[23] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: Continual prediction with lstm," *Neural Computation*, vol. 12, pp. 2451–2471, 1999 (cit. on pp. 3, 177).

[24] S. R. Eddy, "Hidden markov models," *Current Opinion in Structural Biology*, vol. 6, no. 3, pp. 361–365, 1996 (cit. on pp. 3, 17, 177).

[25] C. M. Bishop, *Pattern recognition and machine learning*. Springer, 2006 (cit. on pp. 3, 17, 58, 177).

[26] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015 (cit. on pp. 4, 15, 47, 49, 50, 53).

[27] M. Kemouche and N. Aouf, "A gaussian mixture based optical flow modeling for object detection," in *Proceedings of the International Conference on Imaging for Crime Detection and Prevention (ICDP)*, 2009 (cit. on p. 4).

[28] G. Potamianos, C. Neti, G. Gravier, A. Garg, and A. W. Senior, "Recent advances in the automatic recognition of audiovisual speech," *Proceedings of the IEEE*, vol. 91, no. 9, pp. 1306–1326, 2003 (cit. on pp. 4, 19).

[29] J. Jeffers and M. Barley, *Speechreading (lipreading)*. Charles C. Thomas Publisher, 1980 (cit. on p. 8).

[30] P. B. Denes, "On the statistics of spoken english," *The Journal of the Acoustical Society of America*, vol. 35, no. 6, pp. 892–904, 1963 (cit. on p. 10).

[31] R. O. Cornett, "Cued speech, manual complement to lipreading, for visual reception of spoken language. principles, practice and prospects for automation," *Acta Oto-Rhino-Laryngologica Belgica*, vol. 42, no. 3, pp. 375–384, 1988 (cit. on pp. 10, 13).

[32] J. L. Singleton and M. D. Tittle, "Deaf parents and their hearing children," *Journal of Deaf studies and Deaf education*, vol. 5, no. 3, pp. 221–236, 2000 (cit. on p. 12).

[33] B. R. Clarke and D. Ling, "The effects of using cued speech: A follow-up study," *Volta Review*, vol. 78, no. 1, pp. 23–34, 1976 (cit. on p. 12).

[34] D. Ling and B. R. Clarke, "Cued speech: An evaluative study," *American Annals of the Deaf*, vol. 120, no. 5, pp. 480–8, 1975 (cit. on p. 12).

[35] J. Alegria, B. L. Charlier, and S. Mattys, "The role of lip-reading and cued speech in the processing of phonological information in french-educated deaf children," *European Journal of Cognitive Psychology*, vol. 11, no. 4, pp. 451–472, 1999 (cit. on pp. 12, 13).

[36] J. Leybaert, "Phonology acquired through the eyes and spelling in deaf children," *Journal of Experimental Child Psychology*, vol. 75, no. 4, pp. 291–318, 2000 (cit. on p. 13).

[37] V. Attina, D. Beautemps, M.-A. Cathiard, and M. Odisio, "A pilot study of temporal organization in cued speech production of french syllables: Rules for a cued speech synthesizer," *Speech Communication*, vol. 44, no. 1, pp. 197–214, 2004 (cit. on pp. 13, 14, 119, 121, 128, 142).

[38] N. Aboutabit, D. Beautemps, and L. Besacier, "Hand and lip desynchronization analysis in french cued speech: Automatic temporal segmentation of hand flow," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 1, 2006, pp. I–I (cit. on pp. 13, 14, 120, 125, 126, 142).

[39] S. Young, G. Evermann, M. Gales, T. Hain, D. Kershaw, X. Liu, G. Moore, J. Odell, D. Ollason, D. Povey, *et al.*, "The htk book," *Cambridge University*, vol. 3, p. 175, 2002 (cit. on p. 13).

[40] N. Aboutabit, "Reconnaissance de la langue française parlée complété (lpc): Décodage phonétique des gestes main-lèvres," PhD thesis, Grenoble INPG, 2007 (cit. on pp. 13, 14, 27, 39, 90, 125, 139).

[41] N. Aboutabit, D. Beautemps, and L. Besacier, "Automatic identification of vowels in the cued speech context," in *Proceedings of the International Conference on Auditory-Visual Speech Processing (AVSP)*, 2007, p. 8 (cit. on pp. 13, 20, 39, 60, 142).

[42] R. O. Cornett, R. Beadles, and B. Wilson, "Automatic cued speech," in *Proceedings of the Research Conference on Speech Processing Aids for the Deaf*, 1977 (cit. on p. 14).

[43] P. Duchnowski, D. S. Lum, J. C. Krause, M. G. Sexton, M. S. Bratakos, and L. D. Braida, "Development of speechreading supplements based on automatic speech recognition," *IEEE Transactions on Biomedical Engineering*, vol. 47, no. 4, pp. 487–496, 2000 (cit. on p. 14).

[44] G. Gibert, G. Bailly, D. Beautemps, F. Elisei, and R. Brun, "Analysis and synthesis of the three-dimensional movements of the head, face, and hand of a speaker using cued speech," *The Journal of the Acoustical Society of America*, vol. 118, no. 2, pp. 1144–1153, 2005 (cit. on pp. 14, 27, 125, 141).

[45] Z. Ming, "Spectral parameters to cued speech parameters mapping: Multi-linear and gmm approaches (applied to french vowels)," PhD thesis, Université Grenoble Alpes, 2013 (cit. on pp. 14, 26, 38).

[46] D. Beautemps, L. Girin, N. Aboutabit, G. Bailly, L. Besacier, G. Breton, T. Burger, A. Caplier, M.-A. Cathiard, D. Chêne, *et al.*, "Telma: Telephony for the hearing-impaired people. from models to user tests," 2007, pp. 201–208 (cit. on p. 14).

[47] S. Stillittano, V. Girondel, and A. Caplier, "Lip contour segmentation and tracking compliant with lip-reading application constraints," *Machine Vision and Applications*, pp. 1–18, 2013 (cit. on pp. 15, 94).

[48] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *International Journal of Computer Vision*, vol. 1, no. 4, pp. 321–331, 1988 (cit. on p. 15).

[49] A. L. Yuille, P. W. Hallinan, and D. S. Cohen, "Feature extraction from faces using deformable templates," *International Journal of Computer Vision*, vol. 8, no. 2, pp. 99–111, 1992 (cit. on p. 15).

[50] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham, "Active shape models-their training and application," *Computer Vision and Image Understanding*, vol. 61, no. 1, pp. 38–59, 1995 (cit. on p. 15).

[51] T. Cootes, E. Baldock, and J Graham, "An introduction to active shape models," *Image Processing and Analysis*, pp. 223–248, 2000 (cit. on p. 15).

[52] T. F. Cootes, G. J. Edwards, and C. J. Taylor, "Active appearance models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 6, pp. 681–685, 2001 (cit. on pp. 15, 67).

[53] M. Hlaváč, "Lips landmark detection using cnn," 2016 (cit. on p. 15).

[54] K. Noda, Y. Yamaguchi, K. Nakadai, H. G. Okuno, and T. Ogata, "Audio-visual speech recognition using deep learning," *Applied Intelligence*, vol. 42, no. 4, pp. 722–737, 2015 (cit. on p. 15).

[55] Y. Li, Y. Takashima, T. Takiguchi, and Y. Ariki, "Lip reading using a dynamic feature of lip images and convolutional neural networks," in *Proceedings of the IEEE/ACIS International Conference on Computer and Information Science (ICIS)*, 2016, pp. 1–6 (cit. on p. 16).

[56] T. Burger, A. Caplier, and S. Mancini, "Cued speech hand gestures recognition tool," in *Proceedings of the European Signal Processing Conference (EUSIPCO)*, 2005, pp. 1–4 (cit. on pp. 16, 17).

[57] M. Gonzalez, C. Collet, and R. Dubot, "Head tracking and hand segmentation during hand over face occlusion in sign language," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2010, pp. 234–243 (cit. on p. 17).

[58] M. M. Zaki and S. I. Shaheen, "Sign language recognition using a combination of new vision based features," *Pattern Recognition Letters*, vol. 32, no. 4, pp. 572–577, 2011 (cit. on p. 17).

[59] H. Cooper, E.-J. Ong, N. Pugeault, and R. Bowden, "Sign language recognition using sub-units," *Journal of Machine Learning Research*, vol. 13, no. Jul, pp. 2205–2231, 2012 (cit. on p. 17).

[60] R. J. Schalkoff, *Artificial neural networks*. McGraw-Hill New York, 1997, vol. 1 (cit. on p. 17).

[61] N. C. Camgöz, A. A. Kındıroğlu, and L. Akarun, "Sign language recognition for assisting the deaf in hospitals," in *Proceedings of the International Workshop on Human Behavior Understanding (HBU)*, 2016, pp. 89–101 (cit. on p. 17).

[62] N. C. Camgöz, S. Hadfield, O. Koller, and R. Bowden, "Subunets: End-to-end hand shape and continuous sign language recognition," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017 (cit. on p. 17).

[63] H. Cooper and R. Bowden, "Learning signs from subtitles: A weakly supervised approach to sign language recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009, pp. 2568–2574 (cit. on p. 17).

[64] J. Forster, O. Koller, C. Oberdörfer, Y. Gweth, and H. Ney, "Improving continuous sign language recognition: Speech recognition techniques and system design," in *Workshop on Speech and Language Processing for Assistive Technologies (SLPAT)*, 2013 (cit. on p. 17).

[65] J. Forster, C. Schmidt, O. Koller, M. Bellgardt, and H. Ney, "Extensions of the sign language recognition and translation corpus rwth-phoenix-weather," in *LREC*, 2014, pp. 1911–1916 (cit. on p. 17).

[66] O. Koller, H. Ney, and R. Bowden, "Deep hand: How to train a cnn on 1 million hand images when your data is continuous and weakly labelled," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 3793–3802 (cit. on p. 17).

[67] O. Koller, J. Forster, and H. Ney, "Continuous sign language recognition: Towards large vocabulary statistical recognition systems handling multiple signers," *Computer Vision and Image Understanding*, vol. 141, pp. 108–125, 2015 (cit. on p. 17).

[68] L. Pigou, S. Dieleman, P.-J. Kindermans, and B. Schrauwen, "Sign language recognition using convolutional neural networks," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2014, pp. 572–578 (cit. on pp. 17, 18).

[69] H. Wang, X. Chai, X. Hong, G. Zhao, and X. Chen, "Isolated sign language recognition with grassmann covariance matrices," *ACM Transactions on Accessible Computing*, vol. 8, no. 4, p. 14, 2016 (cit. on p. 17).

[70] M. Mahmoud, T. Baltrušaitis, and P. Robinson, "Automatic analysis of naturalistic hand-over-face gestures," *ACM Transactions on Interactive Intelligent Systems*, vol. 6, no. 2, p. 19, 2016 (cit. on pp. 17, 18).

[71] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields," arXiv:1611.08050, 2016 (cit. on pp. 18, 41).

[72] T. Simon, H. Joo, I. Matthews, and Y. Sheikh, "Hand keypoint detection in single images using multiview bootstrapping," arXiv:1704.07809, 2017 (cit. on pp. 18, 41).

[73] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh, "Convolutional pose machines," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 4724–4732 (cit. on pp. 18, 19, 41).

[74] M. Wand, J. Koutník, and J. Schmidhuber, "Lipreading with long short-term memory," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 6115–6119 (cit. on p. 19).

[75] T. Hueber and G. Bailly, "Statistical conversion of silent articulation into audible speech using full-covariance hmm," *Computer Speech & Language*, vol. 36, pp. 274–293, 2016 (cit. on p. 19).

[76] E. Tatulli and T. Hueber, "Feature extraction using multimodal convolutional neural networks for visual speech recognition," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 2971–2975 (cit. on p. 19).

[77] I. Matthews, T. F. Cootes, J. A. Bangham, S. Cox, and R. Harvey, "Extraction of visual features for lipreading," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 2, pp. 198–213, 2002 (cit. on p. 19).

[78] G. Zhao, M. Barnard, and M. Pietikainen, "Lipreading with local spatiotemporal descriptors," *IEEE Transactions on Multimedia*, vol. 11, no. 7, pp. 1254–1265, 2009 (cit. on p. 19).

[79] M. Gurban and J.-P. Thiran, "Information theoretic feature extraction for audio-visual speech recognition," *IEEE Transactions on Signal Processing*, vol. 57, no. 12, pp. 4765–4776, 2009 (cit. on p. 19).

[80] G. Papandreou, A. Katsamanis, V. Pitsikalis, and P. Maragos, "Adaptive multimodal fusion by uncertainty compensation with application to audiovisual speech recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, no. 3, pp. 423–435, 2009 (cit. on p. 19).

[81] ——, "Multimodal fusion and learning with uncertain features applied to audiovisual speech recognition," in *Proceedings of the Workshop on Multimedia Signal Processing (MMSP)*, 2007, pp. 264–267 (cit. on p. 19).

[82] V. Pitsikalis, A. Katsamanis, G. Papandreou, and P. Maragos, "Adaptive multimodal fusion by uncertainty compensation," in *Proceedings of the International Conference on Spoken Language Processing (ICSLP)*, 2006, pp. 2458–2461 (cit. on p. 19).

[83] P. Lucey and S. Sridharan, "Patch-based representation of visual speech," in *Proceedings of the HCSNet Workshop on Use of Vision in Human-Computer Interaction (VisHCI)*, 2006, pp. 79–85 (cit. on p. 19).

[84] Z. Zhou, G. Zhao, X. Hong, and M. Pietikäinen, "A review of recent advances in visual speech decoding," *Image and Vision Computing*, vol. 32, no. 9, pp. 590–605, 2014 (cit. on p. 19).

[85] Y. M. Assael, B. Shillingford, S. Whiteson, and N. de Freitas, "Lipnet: End-to-end sentence-level lipreading," arXiv:1611.01599, 2016 (cit. on p. 19).

[86] M. Cooke, J. Barker, S. Cunningham, and X. Shao, "An audio-visual corpus for speech perception and automatic speech recognition," *The Journal of the Acoustical Society of America*, vol. 120, no. 5, pp. 2421–2424, 2006 (cit. on p. 19).

[87] S. Gergen, S. Zeiler, A. H. Abdelaziz, R. M. Nickel, and D. Kolossa, "Dynamic stream weighting for turbo-decoding-based audiovisual asr," in *Proceedings of the Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2016, pp. 2135–2139 (cit. on p. 19).

[88] J. S. Chung and A. Zisserman, "Out of time: Automated lip sync in the wild," in *Proceedings of the Asian Conference on Computer Vision (ACCV)*, 2016, pp. 251–263 (cit. on p. 19).

[89] P. Heracleous, N. Aboutabit, and D. Beautemps, "Lip shape and hand position fusion for automatic vowel recognition in cued speech for french," *IEEE Signal Processing Letters*, vol. 16, no. 5, pp. 339–342, 2009 (cit. on p. 20).

[90] ——, "Hmm-based vowel and consonant automatic recognition in cued speech for french," in *Proceedings of the IEEE International Conference on Virtual Environments, Human-Computer Interfaces and Measurements Systems (VECIMS)*, 2009, pp. 33–37 (cit. on pp. 20, 35).

[91] N. Aboutabit, P. Heracleous, and D. Beautemps, "Hand shape coding for hmm-based consonant recognition in cued speech for french," in *Proceedings of the International Conference on Speech and Computer (SPECOM)*, 2009, pp. 1–4 (cit. on p. 20).

[92] G. Gibert, "Conception et évaluation d'un système de synthèse 3d de langue française parlée complétée (lpc) à partir du texte," PhD thesis, Grenoble INPG, 2006 (cit. on p. 27).

[93] C. Benoit, T. Lallouache, T. Mohamadi, and C. Abry, *A set of french visemes for visual speech synthesis*, 1992 (cit. on p. 30).

[94] F. Béchet, "Lia phon: Un systeme complet de phonétisation de textes," *Traitement Automatique des Langues*, vol. 42, no. 1, pp. 47–67, 2001 (cit. on p. 31).

[95] A. Adjoudani and C. Benoit, "On the integration of auditory and visual parameters in an hmm-based asr," in *Speechreading by Humans and Machines*, Springer, 1996, pp. 461–471 (cit. on pp. 35, 60).

[96] Z. Ming, D. Beautemps, G. Feng, and S. Schmerber, "Estimation of speech lip features from discrete cosinus transform," in *Proceedings of the Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2010, pp. 1612–1615 (cit. on p. 35).

[97] Y. Zhang, S. Levinson, and T. Huang, "Speaker independent audio-visual speech recognition," in *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, vol. 2, 2000, pp. 1073–1076 (cit. on p. 35).

[98] S. Dupont and J. Luettin, "Audio-visual speech modeling for continuous speech recognition," *IEEE Transactions on Multimedia*, vol. 2, no. 3, pp. 141–151, 2000 (cit. on pp. 35, 36, 60).

[99] T. J. Hazen, K. Saenko, C.-H. La, and J. R. Glass, "A segment-based audio-visual speech recognizer: Data collection, development, and initial experiments," in *Proceedings of the International Conference on Multimodal Interaction (ICMI)*, 2004, pp. 235–242 (cit. on p. 35).

[100] G. Potamianos and H. P. Graf, "Discriminative training of HMM stream exponents for audio-visual speech recognition," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 6, 1998, pp. 3733–3736 (cit. on pp. 35, 36, 60).

[101] C. Neti, G. Potamianos, J. Luettin, I. Matthews, H. Glotin, D. Vergyri, J. Sison, and A. Mashari, "Audio visual speech recognition," IDIAP, Tech. Rep., 2000 (cit. on pp. 35, 36, 60, 61).

[102] A. V. Nefian, L. Liang, X. Pi, X. Liu, and K. Murphy, "Dynamic bayesian networks for audio-visual speech recognition," *EURASIP Journal on Advances in Signal Processing*, vol. 2002, no. 11, p. 783 042, 2002 (cit. on pp. 35, 36).

[103] J. Barker and X. Shao, "Energetic and informational masking effects in an audiovisual speech recognition system," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 17, no. 3, pp. 446–458, 2009 (cit. on p. 35).

[104] D. Gibbon, I. Mertins, and R. K. Moore, "Audio-visual and multimodal speech-based systems," in *Handbook of Multimodal and Spoken Dialogue Systems*, Springer, 2000, pp. 102–203 (cit. on p. 35).

[105] T. Chen and R. R. Rao, "Audio-visual integration in multimodal communication," *Proceedings of the IEEE*, vol. 86, no. 5, pp. 837–852, 1998 (cit. on p. 35).

[106] A. W.-C. Liew, S. H. Leung, and W. H. Lau, "Lip contour extraction from color images using a deformable model," *Pattern Recognition*, vol. 35, no. 12, pp. 2949–2962, 2002 (cit. on p. 35).

[107] C. Bregler, H. Hild, S. Manke, and A. Waibel, "Improving connected letter recognition by lipreading," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 1, 1993, pp. 557–560 (cit. on pp. 36, 60).

[108] C. Bregler and Y. Konig, "Eigenlips for robust speech recognition," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 2, 1994, pp. 669–672 (cit. on pp. 36, 148).

[109] P. Duchnowski, U. Meier, and A. Waibel, "See me, hear me: Integrating automatic speech recognition and lip-reading," in *Proceedings of the International Conference on Spoken Language Processing (ICSLP)*, 1994 (cit. on p. 36).

[110] N. Li, S. Dettmer, and M. Shah, "Lipreading using eigensequences," in *Proceedings of the International Workshop on Automatic Face and Gesture Recognition*, 1995, pp. 30–34 (cit. on p. 36).

[111] N. M. Brooke, "Talking heads and speech recognisers that can see: The computer processing of visual speech signals," in *Speechreading by Humans and Machines*, Springer, 1996, pp. 351–371 (cit. on p. 36).

[112] M. J. Tomlinson, M. J. Russell, and N. Brooke, "Integrating audio and visual information to provide highly robust speech recognition," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 2, 1996, pp. 821–824 (cit. on p. 36).

[113] G. I. Chiou and J.-N. Hwang, "Lipreading from color video," *IEEE Transactions on Image Processing*, vol. 6, no. 8, pp. 1192–1195, 1997 (cit. on p. 36).

[114] M. S. Gray, J. R. Movellan, and T. J. Sejnowski, "Dynamic features for visual speechreading: A systematic comparison," in *Advances in Neural Information Processing Systems*, 1997, pp. 751–757 (cit. on p. 36).

[115] J. Luettin and N. A. Thacker, "Speechreading using probabilistic models," *Computer Vision and Image Understanding*, vol. 65, no. 2, pp. 163–178, 1997 (cit. on p. 36).

[116] M. Turk and A. Pentland, "Eigenfaces for recognition," *Journal of Cognitive Neuroscience*, vol. 3, no. 1, pp. 71–86, 1991 (cit. on pp. 36, 148).

[117] R. C. Gonzalez, R. E. Woods, *et al.*, *Digital image processing*, 2002 (cit. on p. 36).

[118] S. Nakamura, H. Ito, and K. Shikano, "Stream weight optimization of speech and lip image sequence for audio-visual speech recognition," 2000 (cit. on p. 36).

[119] P. Scanlon and R Reilly, "Feature analysis for automatic speechreading," in *Proceedings of the IEEE Workshop on Multimedia Signal Processing (MMSP)*, 2001, pp. 625–630 (cit. on p. 36).

[120] I. Daubechies, *Ten lectures on wavelets*. SIAM, 1992, vol. 61 (cit. on p. 36).

[121] C. Tomasi and T. Kanade, "Detection and tracking of point features," 1991 (cit. on p. 41).

[122] J. Shi and T. Carlo, "Good features to track," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 1994, pp. 593–600 (cit. on p. 41).

[123] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013 (cit. on p. 47).

[124] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Networks*, vol. 61, pp. 85–117, 2015 (cit. on pp. 47, 55).

[125] Y. Zhang, M. Pezeshki, P. Brakel, S. Zhang, C. L. Y. Bengio, and A. Courville, "Towards end-to-end speech recognition with deep convolutional neural networks," arXiv:1701.02720, 2017 (cit. on p. 47).

[126] R. Collobert and J. Weston, "A unified architecture for natural language processing: Deep neural networks with multitask learning," in *Proceedings of the International Conference on Machine Learning (ICML)*, 2008, pp. 160–167 (cit. on p. 47).

[127] A. Kumar and B. Raj, "Deep cnn framework for audio event recognition using weakly labeled web data," arXiv:1707.02530, 2017 (cit. on p. 47).

[128] D. Ciregan, U. Meier, and J. Schmidhuber, "Multi-column deep neural networks for image classification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012, pp. 3642–3649 (cit. on p. 47).

[129] A. Graves, A.-r. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013, pp. 6645–6649 (cit. on p. 48).

[130] H. Soltau, H. Liao, and H. Sak, "Neural speech recognizer: Acoustic-to-word lstm model for large vocabulary speech recognition," arXiv:1610.09975, 2016 (cit. on p. 48).

[131] K. Gurney, *An introduction to neural networks*. CRC Press, 1997 (cit. on p. 48).

[132] S. Ruder, "An overview of gradient descent optimization algorithms," arXiv:1609.04747, 2016 (cit. on p. 49).

[133] S. Hochreiter, Y. Bengio, P. Frasconi, J. Schmidhuber, *et al.*, *Gradient flow in recurrent nets: The difficulty of learning long-term dependencies*, 2001 (cit. on pp. 50, 55).

[134] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," arXiv:1502.03167, 2015 (cit. on p. 53).

[135] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014 (cit. on p. 53).

[136] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Nature*, vol. 323, no. 6088, pp. 533–536, 1986 (cit. on p. 53).

[137] A. Graves, M Liwicki, S. Fernandez, R. Bertolami, H. Bunke, and J Schmidhuber, "A novel connectionist system for improved unconstrained handwriting," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 5, pp. 855–868, 2009 (cit. on p. 53).

[138] H. Sak, A. Senior, and F. Beaufays, "Long short-term memory recurrent neural network architectures for large scale acoustic modeling," in *Proceedings of the Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2014 (cit. on p. 53).

[139] X. Li and X. Wu, "Constructing long short-term memory based deep recurrent neural networks for large vocabulary speech recognition," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 4520–4524 (cit. on p. 53).

[140] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997 (cit. on pp. 53, 55).

[141] F. J. Pineda, "Generalization of back-propagation to recurrent neural networks," *Physical Review Letters*, vol. 59, no. 19, p. 2229, 1987 (cit. on p. 54).

[142] S. Fernández, A. Graves, and J. Schmidhuber, "An application of recurrent neural networks to discriminative keyword spotting," in *Proceedings of the International Conference on Artificial Neural Networks (ICANN)*, 2007, pp. 220–229 (cit. on p. 55).

[143] A. Graves and J. Schmidhuber, "Offline handwriting recognition with multidimensional recurrent neural networks," in *Advances in Neural Information Processing Systems*, 2009, pp. 545–552 (cit. on p. 55).

[144] D. Gillick, C. Brunk, O. Vinyals, and A. Subramanya, "Multilingual language processing from bytes," arXiv:1512.00103, 2015 (cit. on p. 55).

[145] D. Lahat, T. Adali, and C. Jutten, "Multimodal data fusion: An overview of methods, challenges, and prospects," *Proceedings of the IEEE*, vol. 103, no. 9, pp. 1449–1477, 2015 (cit. on p. 57).

[146] A. K. Jain, R. P. W. Duin, and J. Mao, "Statistical pattern recognition: A review," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 1, pp. 4–37, 2000 (cit. on p. 57).

[147] G. Potamianos, C. Neti, J. Luettin, and I. Matthews, "Audio-visual automatic speech recognition: An overview," *Issues in Visual and Audio-Visual Speech Processing*, vol. 22, p. 23, 2004 (cit. on pp. 57, 60, 61).

[148] J.-L. Schwartz, P. Escudier, and P. Teissier, "Multimodal speech: Two or three senses are better than one," *Language and Speech Processing*, pp. 377–415, 2009 (cit. on pp. 58, 59, 61, 62, 162).

[149] K.-F. Lee, "Context-dependent phonetic hidden markov models for speaker-independent continuous speech recognition," in *Readings in Speech Recognition*, Elsevier, 1990, pp. 347–365 (cit. on p. 58).

[150] S. J. Young, J. J. Odell, and P. C. Woodland, "Tree-based state tying for high accuracy acoustic modelling," in *Proceedings of the Workshop on Human Language Technology*, 1994, pp. 307–312 (cit. on pp. 58, 63, 142).

[151] G. E. Dahl, D. Yu, L. Deng, and A. Acero, "Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 1, pp. 30–42, 2012 (cit. on p. 58).

[152] Y. Bengio, R. Ducharme, P. Vincent, and C. Jauvin, "A neural probabilistic language model," *Journal of Machine Learning Research*, vol. 3, no. Feb, pp. 1137–1155, 2003 (cit. on p. 58).

[153] A. Luo, S. Chen, and B. Xv, "Enhanced map-matching algorithm with a hidden markov model for mobile phone positioning," *ISPRS International Journal of Geo-Information*, vol. 6, no. 11, p. 327, 2017 (cit. on p. 59).

[154] J.-L. Schwartz, J. Robert-Ribes, and P. Escudier, "Ten years after summerfield: A taxonomy of models for audio-visual fusion in speech perception," *Hearing by Eye II: Advances in the Psychology of Speechreading and Auditory-Visual Speech*, pp. 85–108, 1998 (cit. on pp. 59, 61, 142).

[155] M. E. Hennecke, D. G. Stork, and K. V. Prasad, "Visionary speech: Looking ahead to practical speechreading systems," in *Speechreading by Humans and Machines*, Springer, 1996, pp. 331–349 (cit. on p. 60).

[156] P. Teissier, J. Robert-Ribes, J.-L. Schwartz, and A. Guérin-Dugué, "Comparing models for audiovisual fusion in a noisy-vowel recognition task," *IEEE Transactions on Speech and Audio Processing*, vol. 7, no. 6, pp. 629–642, 1999 (cit. on p. 60).

[157] L. R. Rabiner and B.-H. Juang, *Fundamentals of speech recognition*. PTR Prentice Hall Englewood Cliffs, 1993, vol. 14 (cit. on p. 60).

[158] J. Luettin, G. Potamianos, and C. Neti, "Asynchronous stream modeling for large vocabulary audio-visual speech recognition," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 1, 2001, pp. 169–172 (cit. on p. 61).

[159] H. Larochelle and G. E. Hinton, "Learning to combine foveal glimpses with a third-order boltzmann machine," in *Advances in Neural Information Processing Systems*, Curran Associates, Inc., 2010, pp. 1243–1251 (cit. on p. 61).

[160] M. Denil, L. Bazzani, H. Larochelle, and N. de Freitas, "Learning where to attend with deep architectures for image tracking," *Neural Computation*, vol. 24, no. 8, pp. 2151–2184, 2012 (cit. on p. 61).

[161] D. Bahdanau, J. Chorowski, D. Serdyuk, P. Brakel, and Y. Bengio, "End-to-end attention-based large vocabulary speech recognition," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016, pp. 4945–4949 (cit. on p. 61).

[162] L. Yao, A. Torabi, K. Cho, N. Ballas, C. Pal, H. Larochelle, and A. Courville, "Describing videos by exploiting temporal structure," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2015, pp. 4507–4515 (cit. on p. 61).

[163] C. Hori, T. Hori, T.-Y. Lee, Z. Zhang, B. Harsham, J. R. Hershey, T. K. Marks, and K. Sumi, "Attention-based multimodal fusion for video description," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 4203–4212 (cit. on p. 61).

[164] H. Yu, J. Wang, Z. Huang, Y. Yang, and W. Xu, "Video paragraph captioning using hierarchical recurrent neural networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 4584–4593 (cit. on p. 61).

[165] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *Proceedings of the International Conference on Machine Learning (ICML)*, 2015, pp. 2048–2057 (cit. on p. 61).

[166] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," arXiv:1409.0473, 2014 (cit. on p. 61).

[167] K. Cho, A. Courville, and Y. Bengio, "Describing multimedia content using attention-based encoder-decoder networks," *IEEE Transactions on Multimedia*, vol. 17, no. 11, pp. 1875–1886, 2015 (cit. on p. 61).

[168] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," arXiv:1406.1078, 2014 (cit. on p. 61).

[169] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in Neural Information Processing Systems*, 2014, pp. 3104–3112 (cit. on p. 61).

[170] G. Andrew, R. Arora, J. Bilmes, and K. Livescu, "Deep canonical correlation analysis," in *Proceedings of the International Conference on Machine Learning (ICML)*, 2013, pp. 1247–1255 (cit. on p. 62).

[171] H. Hotelling, "Relations between two sets of variates," *Biometrika*, vol. 28, no. 3/4, pp. 321–377, 1936 (cit. on p. 62).

[172] T. W. Anderson, *An introduction to multivariate statistical analysis*. Wiley New York, 1958 (cit. on p. 62).

[173] D. R. Hardoon, S. Szedmak, and J. Shawe-Taylor, "Canonical correlation analysis: An overview with application to learning methods," *Neural Computation*, vol. 16, no. 12, pp. 2639–2664, 2004 (cit. on p. 62).

[174] A. Vinokourov, N. Cristianini, and J. Shawe-Taylor, "Inferring a semantic representation of text via cross-language correlation analysis," in *Advances in Neural Information Processing Systems*, 2003, pp. 1497–1504 (cit. on p. 62).

[175] A. Haghighi, P. Liang, T. Berg-Kirkpatrick, and D. Klein, "Learning bilingual lexicons from monolingual corpora," *Proceedings of ACL-08: HLT*, pp. 771–779, 2008 (cit. on p. 62).

[176] P. Dhillon, D. P. Foster, and L. H. Ungar, "Multi-view learning of word embeddings via cca," in *Advances in Neural Information Processing Systems*, 2011, pp. 199–207 (cit. on p. 62).

[177] R. Arora and K. Livescu, "Multi-view cca-based acoustic features for phonetic recognition across speakers and domains," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2013, pp. 7135–7139 (cit. on p. 62).

[178] F. Rudzicz, "Adaptive kernel canonical correlation analysis for estimation of task dynamics from acoustics," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2010, pp. 4198–4201 (cit. on p. 62).

[179] T.-K. Kim, S.-F. Wong, and R. Cipolla, "Tensor canonical correlation analysis for action classification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007, pp. 1–8 (cit. on p. 62).

[180] M. E. Sargin, Y. Yemez, E. Erzin, and A. M. Tekalp, "Audiovisual synchronization and fusion using canonical correlation analysis," *IEEE Transactions on Multimedia*, vol. 9, no. 7, pp. 1396–1403, 2007 (cit. on p. 62).

[181] M. Slaney and M. Covell, "Facesync: A linear operator for measuring synchronization of video facial images and audio tracks," in *Advances in Neural Information Processing Systems*, 2001, pp. 814–820 (cit. on p. 62).

[182] Y. Takashima, T. Takiguchi, Y. Ariki, and K. Omori, "Audio-visual speech recognition for a person with severe hearing loss using deep canonical correlation analysis," in *Proceedings of the Conference on Challenges in Hearing Assistive Technology (CHAT)*, 2017, pp. 77–81 (cit. on p. 62).

[183] D. Cristinacce and T. F. Cootes, "Facial feature detection and tracking with automatic template selection," in *Proceedings of the IEEE International Conference on Automatic Face and Gesture Recognition*, 2006, pp. 429–434 (cit. on pp. 62, 67, 68).

[184] ——, "Feature detection and tracking with constrained local models," in *Proceedings of the British Machine Vision Conference (BMVC)*, vol. 1, 2006, p. 3 (cit. on pp. 62, 67–69).

[185] K.-F. Lee, "On large-vocabulary speaker-independent continuous speech recognition," *Speech Communication*, vol. 7, no. 4, pp. 375–379, 1988 (cit. on p. 62).

[186] R Schwartz, Y. Chow, O. Kimball, S Roucos, M Krasner, and J Makhoul, "Context-dependent modeling for acoustic-phonetic recognition of continuous speech," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 10, 1985, pp. 1205–1208 (cit. on pp. 62, 162).

[187] T. Hueber, E.-L. Benaroya, G. Chollet, B. Denby, G. Dreyfus, and M. Stone, "Visuo-phonetic decoding using multi-stream and context-dependent models for an ultrasound-based silent speech interface," in *Proceedings of the Annual Conference of the International Speech Communication Association (INTERSPEECH)*, 2009, pp. 640–643 (cit. on pp. 62, 162).

[188] T. Baltrusaitis, P. Robinson, and L.-P. Morency, "Constrained local neural fields for robust facial landmark detection in the wild," in *Proceedings of the IEEE International Conference on Computer Vision Workshops (ICCVW)*, 2013, pp. 354–361 (cit. on pp. 67–70).

[189] T. F. Cootes and C. J. Taylor, "Active shape model search using local grey-level models: A quantitative evaluation," in *Proceedings of the British Machine Vision Conference (BMVC)*, vol. 93, 1993, pp. 639–648 (cit. on p. 67).

[190] L. Liu, G. Feng, and D. Beautemps, "Inner lips feature extraction based on clnf with hybrid dynamic template for cued speech," *EURASIP Journal on Image and Video Processing*, vol. 2017, no. 1, p. 88, 2017 (cit. on p. 68).

[191] C. Bregler and S. M. Omohundro, "Surface learning with applications to lipreading," in *Advances in neural information processing systems*, 1994, pp. 43–50 (cit. on p. 71).

[192] M. Lallouache, "Un poste "visage-parole". acquisition et traitement de contours labiaux," *Actes des Journées d'Etude de la Parole (JEP)*, 1990 (cit. on p. 71).

[193] G. Feng, "Data smoothing by cubic spline filters," *IEEE Transactions on Signal Processing*, vol. 46, no. 10, pp. 2790–2796, 1998 (cit. on pp. 76, 126).

[194] L. Revéret and C. Benoît, "A new 3d lip model for analysis and synthesis of lip motion in speech production," in *Proceedings of the ESCA Workshop on Audio-Visual Speech Processing (AVSP)*, 1998 (cit. on pp. 87, 90, 106).

[195] M. Shemshaki and R. Amjadifard, "Lip segmentation using geometrical model of color distribution," in *Proceedings of the Iranian Conference on Machine Vision and Image Processing (MVIP)*, 2011, pp. 1–5 (cit. on p. 94).

[196] C.-C. Chiang, W.-K. Tai, M.-T. Yang, Y.-T. Huang, and C.-J. Huang, "A novel method for detecting lips, eyes and faces in real time," *Real Time Imaging*, vol. 9, no. 4, pp. 277–287, 2003 (cit. on p. 94).

[197] J. Starck, A. Maki, S. Nobuhara, A. Hilton, and T. Matsuyama, "The multiple-camera 3-d production studio," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 6, pp. 856–869, 2009 (cit. on p. 111).

[198] Q. Chen, D. Li, and C.-K. Tang, "Knn matting," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 9, pp. 2175–2188, 2013 (cit. on p. 111).

[199] S. M. Yoon and G. Yoon, "Alpha matting using compressive sensing," *Electronics Letters*, vol. 48, no. 3, pp. 153–155, 2012 (cit. on p. 111).

[200] S. M. Prabhu and A. Rajagopalan, "Natural matting for degraded pictures," *IEEE Transactions on Image Processing*, vol. 20, no. 12, pp. 3647–3653, 2011 (cit. on p. 111).

[201] Y. Yuan, Y. Liu, G. Dai, J. Zhang, and Z. Chen, "Automatic foreground extraction based on difference of gaussian," *The Scientific World Journal*, vol. 2014, 2014 (cit. on p. 111).

[202] C. R. Wren, A. Azarbayejani, T. Darrell, and A. P. Pentland, "Pfinder: Real-time tracking of the human body," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 780–785, 1997 (cit. on p. 111).

[203] V. Attina, M.-A. Cathiard, and D. Beautemps, "Temporal measures of hand and speech coordination during french cued speech production," in *Proceedings of the International Gesture Workshop*, 2005, pp. 13–24 (cit. on p. 119).

[204] D. V. Abreu, T. K. Tamura, D. G. Keamy Jr, R. D. Eavey, *et al.*, "Podcasting: Contemporary patient education," *Ear, Nose & Throat Journal*, vol. 87, no. 4, p. 208, 2008 (cit. on p. 126).

[205] J. Naylor, *Magix movie edit pro 2014 revealed*. Dtvpro Publishing, 2014 (cit. on p. 126).

[206] L. Liu, G. Feng, and D. Beautemps, "Automatic temporal segmentation of hand movements for hand positions recognition in french cued speech," in *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018, pp. 5130–5134 (cit. on p. 126).

[207] F. Chollet *et al.*, *Keras*, https://github.com/fchollet/keras, 2015 (cit. on pp. 135, 144, 161).

[208] L. Liu, T. Hueber, G. Feng, and D. Beautemps, "Visual recognition of continuous cued speech using a tandem cnn-hmm approach," in *Interspeech, 2018*, 2018, pp. 2643–2647 (cit. on pp. 139, 141, 164).

[209] A. Caplier, L. Bonnaud, S. Malassiotis, and M. G. Strintzis, "Comparison of 2D and 3D analysis for automated cued speech gesture recognition," in *Proceedings of the International Conference on Speech and Computer (SPECOM)*, 2004 (cit. on p. 140).

[210] R. V. Hogg and E. A. Tanis, *Probability and statistical inference*. Pearson Educational International, 2009 (cit. on p. 154).

[211] L. Cappelletta and N. Harte, "Viseme definitions comparison for visual-only speech recognition," in *Proceedings of the European Signal Processing Conference (EUSIPCO)*, 2011, pp. 2109–2113 (cit. on p. 155).

[212] ——, "Phoneme-to-viseme mapping for visual speech recognition," in *Proceedings of the International Conference on Pattern Recognition Applications and Methods (ICPRAM)*, 2012, pp. 322–329 (cit. on p. 155).

[213] R. Rosenfeld, "Two decades of statistical language modeling: Where do we go from here?" *Proceedings of the IEEE*, vol. 88, no. 8, pp. 1270–1278, 2000 (cit. on p. 169).

[214] J. R. Bellegarda, "Exploiting latent semantic information in statistical language modeling," *Proceedings of the IEEE*, vol. 88, no. 8, pp. 1279–1296, 2000 (cit. on p. 169).

[215] S. Watanabe, T. Iwata, T. Hori, A. Sako, and Y. Ariki, "Topic tracking language model for speech recognition," *Computer Speech & Language*, vol. 25, no. 2, pp. 440–461, 2011 (cit. on p. 169).