UNIVERSIDAD DE ZARAGOZA

Departamento de Ingeniería Electrónica y Comunicaciones

Improvements in Speech Recognition for Embedded Devices by taking Advantage of Lip Reading Techniques

TESIS DOCTORAL

Presentada por:

Jesús Fernando Guitarte Pérez

Dirigida por:

Dr. Eduardo Lleida Solano

Múnich, Junio 2006

UNIVERSIDAD DE ZARAGOZA

Departamento de Ingeniería Electrónica y Comunicaciones

Improvements in Speech Recognition for Embedded Devices by taking Advantage of Lip Reading Techniques

TESIS DOCTORAL

Presentada por:

Jesús Fernando Guitarte Pérez

Dirigida por:

Dr. Eduardo Lleida Solano

Múnich, Junio 2006

This work was supported by a doctoral research grant from Siemens AG.

Resumen

En la presente tesis doctoral la información visual contenida en el movimiento de los labios se ha utilizado para mejorar la robustez frente al ruido de sistemas de reconocimiento de voz en dispositivos con recursos limitados. El sistema aquí descrito reduce de forma significativa la tasa de error en niveles de ruido acústico elevado. Los algoritmos utilizados se caracterizan por su reducido consumo, tanto de tiempo de procesado como de memoria, permitiendo su uso en dispositivos integrados. Los principales aspectos a tomar en consideración en un sistema de lectura de labios son la localización y seguimiento de los labios, la extracción de la información visual y su integración con la información acústica. En el presente trabajo se proponen soluciones a estos tres problemas adecuadas al uso en dispositivos con recursos limitados.

Se ha desarrollado un algoritmo para la localización y seguimiento de los labios. A partir de una clasificación por color, usando contornos horizontales y un modelo sencillo de la cara el algoritmo implementado proporciona la posición de la boca con un consumo muy bajo de recursos. Este algoritmo se ha implementado en un teléfono móvil procesando una tasa de 15 imágenes por segundo en tiempo real. Por otro lado para la extracción de la información visual se han estudiado dos tipos de algoritmos diferentes; uno basado en un modelado de la geometría labial y otro basado en una transformación matemática de los pixeles incluidos en la región de la boca. Se ha mostrado como en dispositivos con recursos limitados el segundo tipo proporciona mejores tasas de reconocimiento al no requerir la extracción precisa del contorno de los labios. Finalmente, se han estudiado tres técnicas para integrar la información acústica y visual, que se diferencian en la posición donde tiene lugar la integración en el proceso de reconocimiento: temprana, tardía e híbrida. Se ha constatado que la última proporciona los mejores resultados de reconocimiento, presentando además ventajas para su implementación en un dispositivo integrado.

El sistema diseñado proporciona resultados muy satisfactorios en la lucha contra los efectos nocivos de ruidos como la interferencia de otro hablante, con reducciones relativas del 38% de la tasa de error. Las técnicas convencionales de reducción de ruido, como la sustracción espectral o el filtrado de Wiener, no consiguen reducir la tasa de error frente a este tipo de interferencia.

Teniendo en cuenta el creciente incremento en el mercado de dispositivos equipados con cámara y la viabilidad de esta tecnología demostrada en esta tesis, la lectura de labios se puede considerar una tecnología adecuada para proporcionar mayor robustez a los sistemas futuros de reconocimiento de voz.

Abstract

In the present doctoral thesis the visual information conveying lip movements was used to improve the speech recognition robustness against noise for embedded devices. Our audio-visual speech recognition system was able to reduce significantly the error rate in acoustically degraded environments. Due to algorithm design constraints, the low complexity requirements of the described system make it suitable for embedded devices. The main aspects in audio-visual speech recognition are the mouth localization, the discriminative video features extraction and the fusion of visual and audio information. These three topics were analyzed from the point of view of an embedded implementation and related solutions are provided in this thesis.

A highly efficient Lip Finding and Tracking algorithm, based on colour classification and horizontal filtering, takes advantage of a very simple geometric face model providing the position of the mouth with very modest memory and computational complexity requirements. This algorithm was implemented in a mobile phone working in real time with a frame rate up to 15 frames per second. For visual feature extraction two different approaches have been studied, one based on a lip geometry modeling and another based on the pixel information of the mouth region. It is shown that the second one is more suitable for the described implementation since it does not require an accurate lip contour extraction, which is still a difficult task for embedded devices. Finally, three different kinds of audio-visual integration strategies: early integration, late integration and hybrid integration (Multi-stream) have been studied. These strategies differ on the location of the information integration in the decision process. It is shown that multi-stream provides the best recognition results for an embedded implementation. Additionally this solution presents important implementation advantages against the other ones. According to the recognition results, this work shows that Lip Reading can be considered as a new modality of Noise Reduction. Our system has provided impressive results to combat non-stationary noise like interfering talker obtaining a relative reduction from 38% of the word error rate. Conventional solutions like spectral subtraction and Wiener filtering failed to increase the recognition performance in interfering talker condition.

Considering the availability increase of visual information in many devices and the feasibility of this technology proved in this work, Lip Reading is a very promising technology to provide more robustness for future speech recognition systems.

Acknowledges

I would like to thank my supervisor at Siemens AG, Klaus Lukas, for his never ending help and encourage. He has always been open for discussions and he has shown me how to transform problems into challenges.

I would also like to express my thanks to the director of this thesis, Professor Dr. Eduardo Lleida, for his support that made this work possible. I gratefully acknowledge the precious opportunity I got to collaborate with Dr. Visvanathan Ramesh and Dr. Nikos Paragios at Siemens Corporate Research, Princeton. I want to thank Alejandro Frangi for the interesting discussions; as well I want to mention my colleagues at the University of Zaragoza who have always offer me their support. I highly appreciate to have taken part in the joint project between BMW Research&Technology GmbH and Siemens AG for Lip Reading, thanking especially Dr. Frank Althoff and Stephan Hoch for the interesting work and discussions. I owe special thanks to Carlos Lucas who has always supported our work at the Siemens Mobile Phones division. I would like also to thank my colleagues at Siemens AG in Munich for the interesting discussions and help especially to Myriam Aparicio, Ewald Frensch, Ankit Jamwal, Xiangpeng Kong, Dr. Joan Mari, Dr. Tobias Schneider, Andreas Schröer, Markus Simon, Dr. Hervé Taddei, and Michael Wandinger. The financial maintenance for this research came from Siemens AG, I am very grateful to the head of CT IC5 EP, Gerhard Hoffmann, who has always supported my work and to Dr. Bernhard Kämmerer, head of CT IC5, for believing in this future technology.

I don’t want to forget my friends in Munich, they have already become a part of my family, especially Jose, Guiller and Wolfgang. También quiero agradecer la ayuda, paciencia y apoyo de mis padres Ernestina y Jesús así como de mi hermano Sebastián.

Contents

Chapter 1

Introduction

1.1 Motivation

In recent years, Automatic Speech Recognition (ASR) has been widely deployed in embedded devices like mobile phones and infotainment systems for car due to convenience and safety reasons. However, especially for these scenarios, often high level of noises occur having a very negative impact on the recognition rate. Current systems are still quite sensitive to noise, although in the last years several techniques have met certain success in degraded environments. There are different kinds of approaches to obtain noise robust speech recognition systems. Some approaches try to find robust speech features that are sensitive to noise [Buera et al., 2004], other approaches use the information coming from a microphone array [Cox et al., 1987], [Cox et al., 1986], [Doclo and Moonen, 2003]. The last ones take advantage of the time delays between the acoustic signals in the different microphones to prevail the signal arriving from a known direction, the direction where the user is placed. This solution provides a spatial filtering in such a way that if the signal and the noise come from different directions significant improvements in the signal to noise ratio (SNR) of the final signal can be achieved, which implies improvements in recognition rate. Another possibility to deal with degraded environments is given by the Noise Reduction techniques that perform a preprocessing on the acoustic signal in order to reduce the quantity of acoustic noise. These Noise Reduction algorithms are widely spread out in commercial devices; the most important ones are spectral subtraction [Martin, 2001], [Martin, 1994] and Wiener filtering [Scalart and Filho, 1996], [Singh and Stern, 2002], [Beaugeant et al., 1998], [Wiener, 1949]. Both solutions have been proved to be very efficient against stationary noise but they fail to work properly against non stationary noise like interfering talker [Singh and Stern, 2002], [Guitarte et al., 2005a].

With the emerging presence of cameras in embedded devices [Tatsuno, 2006] a new information modality is available to achieve more robust speech recognition systems. This technique is called Lip Reading [Sumby, 1954], [Luettin, 1997a], [Meier et al., 2000], [Potamianos et al., 2003], [Nefian et al., 2002]. It uses the additional information that can be found in the lip movement when speaking. This visual information will be combined with the acoustic information in order to improve the recognition rate. Degradation of one of the modalities, for example interfering talker or cross-talk noise for audio or occlusion for visual information, may be compensated to some extent by information from the other modality. Many devices are equipped with cameras to provide the user multimedia connectivity, e.g. mobile phones, and in the near future even cars. An important issue regarding Lip Reading is that the required hardware is already integrated for other purposes: the visual information is in many cases available without any additional cost; we are just proposing to use it. Other solutions like microphone arrays require additional hardware (array of microphones) which benefit is limited to just improve the recognition rate. Lip Reading does not need extra hardware; therefore the balance between the incremental costs and the improvements on the recognition rate will be very positive. Every improvement in recognition rate would justify the use of this solution as it has no extra cost for all devices already equipped with camera.

Up to now we have summarized the economic motivations for the using of Lip Reading but of course there were also scientific reasons which have mainly encouraged us to perform our investigations. First of all, it is well known that deaf people take advantage of the visual cues to communicate, especially mouth movement play an important role [Marschark et al., 1998]. Secondly, the noises that can degrade the acoustic and visual channels are usually not correlated; this means that the visual and acoustic noise sources have in most of the cases nothing to do with each other. The fact that both channels have the same signal but different types of noises can be used to separate speech from noise and it will improve the recognition scores. Only in case of mispronunciations, truncated words or natural language artifacts the audio and visual noises are correlated, in these cases the visual information will not be helpful. Finally, it is showed that there exist complementarities between visual and acoustic signal. The former is more reliable for distinguishing manner and place of articulation when this one is not inside of the throat or mouth, while the latter conveys the voicing and nasalization information. For example phonemes /m/ and /n/ vary only by place of articulation and are acoustically similar but visually different. Thus the use of visual information can improve the performance of the ASR even if the acoustic signal is not disturbed by noise only due to the complementarities of both information sources. However, the information contained on the visual channel is quite limited in comparison to the information of the audio channel. This is the reason why the main challenge is to combine properly the audio and video information.

In the last years speech recognition community has made important efforts to take advantage of the visual information of the lips in order to improve the recognition rates in noisy environments. Significant decreases on Word Error Rate have been obtained by using visual cues combined with audio information. Compared to conventional acoustic recognition, audio-visual speech recognition systems can decrease the Word Error Rate for various signal/noise conditions as it was achieved by Intel [Nefian et al., 2002] and IBM [Potamianos et al., 2003]. Up to now all investigations were addressed to technology development, but without taking special care on resource consumption. This is going to be an important point in our work. We want to show that the use of the visual information is also possible in commercial embedded devices, for this reason also all invasive algorithms that require special make up, headsets or reference points [Huang and Visweswariah, 2005], [Teissier et al., 1999], [Mustafa et al., 2004] will not be considered.

1.2 30 years of Audio-Visual Speech Recognition: Engineering and Psychology

Many technological problems can be solved by observing the nature. For automatic speech recognition it has been always very helpful to study the perception theory in humans [Schmidbauer and Höge, 1991]. For example, how the communication between humans succeeds even despite high level of acoustic noise. Many studies have been performed in this field and it is proved that besides the acoustic information the visual information, like face expressions and lip movements, helps significantly in the perception process especially when the acoustic channel is corrupted by noise. In 1954 [Sumby, 1954] presented a first study in which he showed the contribution of the visual information of face expressions for the understanding in noisy environments. 20 years later an article presented in “Nature” magazine revolutionized the perception theories. In this article the so called McGurk effect [McGurk, 1976] was described. Bimodal fusion of audio and visual stimuli in perceiving speech was demonstrated; when the spoken sound /ga/ is superimposed on the video of a person uttering /ba/, most people perceive the speaker as uttering the sound /da/. This article was the motivation and the starting point for many researchers to use the visual information also in ASR. In 1984 Petajan [Petajan, 1985] reported the first automatic Lip Reading system. Using the video of a speaker’s face together with simple image thresholds, he was able to extract binary (black and white) mouth images, and subsequently, mouth height, width, perimeter, and area as visual speech features. Then he developed a visual-only recognizer based on dynamic time warping to rescore the best two choices of the audio-only system. This method improved ASR for single speaker isolated word recognition task. Pentajan's work generated significant excitement, and soon various sites established research in audio-visual ASR. We should point out the efforts carried out in the Institut Della Molled'Intelligence Artificielle Perceptive (IDIAP) [Luettin, 1997a], Switzerland, as well as the work performed in the Institut de la Communication Parlée (ICP) [Adjoudani and Benoit, 1996], France, where new feature extraction techniques and combination of audio and visual features were investigated. Universities like Carnegie Mellon [Matthews et al., 2002] in USA or the Technische Universität Karlsruhe [Meier et al., 2000] in Germany have been very active in the audio-visual research with important contributions in the last years. Finally important companies have shown their interest in this topic. Both IBM [Potamianos et al., 2003] and Intel [Nefian et al., 2002] have developed their own Lip Reading systems. The objectives of both research groups focused on improving the recognition rate without taking into account restrictions on computational time and system memory limitations. These kinds of systems are very appropriate for server based recognizers but their philosophy is quite opposite to the kind of algorithms that must be implemented on an embedded device. [Adjoudani and Benoit, 1996]

1.3 Thesis Objectives

With this thesis we want to show that the visual information can also be used to improve the speech recognition rate also for embedded devices. Up to now the improvement of the recognition rate through the use of visual information was demonstrated in ideal labor conditions. Until now the previous automatic Lip Reading studies focused on demonstrating that an improvement in recognition rate by using the visual information was possible but without taking special care on implementation considerations that are essential for an embedded solution. The main objective of the present work is to prove that an improvement in recognition rate by using visual information is also possible when the system resources are limited, when the system does not work on ideal laboratory conditions, and when the kind of algorithms used are suitable to be used in a commercial solution. In relation to the last consideration we are not allowed to use intrusive algorithms as for example in [Mustafa et al., 2004] where it is necessary that a set of reference colored points must be placed on the speaker's lips and face. As it can be easily understood these kinds of algorithms can not be commercialized and their use is limited to theoretical research purposes. Assuming that the main advantage of Lip Reading systems is to improve the robustness of Automatic Speech Recognition (ASR) against noise, an important point to assure the feasibility of Lip Reading concerns the study of the audiovisual recognition methods in comparison with conventional

Noise Reduction techniques widely implemented in speech recognition systems. Nowadays there are efficient implementations of Noise Reduction algorithms working on embedded devices. Up to now the comparisons found in the literature showed improvements in the error rate using the visual information taking as baseline the audio channel with high degradations and without Noise Reduction. In this work we want to know whether lip reading can be assumed as a complementary technique to the conventional Noise Reduction. The general objectives of this work have been summarized, now we are going to state the main concrete objectives of the thesis:

- Study of the different solutions for lip localization and tracking. Development and implementation of efficient algorithms able to work on embedded devices. In this study memory and CPU algorithm requirements will be investigated.
- Study of different visual feature extractions techniques, implementation of the most important ones. Comparison of the different techniques to find out which one offers the best results in terms of recognition rate, taking into account that it must work in a resource limited system.
- Study of the integration strategies for the fusion of the visual and acoustic information. Selection of the most appropriate technique for the implementation in an embedded device taking into account recognition performance and resource consumption. An implementation of different integration solutions and an evaluation in terms of recognition rate will be provided.
- Compare our audio-visual speech recognition with conventional Noise Reduction systems (spectral subtraction and Wiener filtering) for different kind of noises. It is of interest to know if when using conventional Noise Reduction systems there is an improvement in recognition rate by adding visual information. As well, we are going to find out which of the solution is the most appropriate for different types of acoustic noise. The combination of Lip Reading with conventional Noise Reduction techniques will be studied.
- Study of the influence of visual noise in recognition rate. We are developing systems that must work not only in laboratory but in normal conditions and the image in such scenarios will be degraded by for example bad illumination, movements that are not able to be tracked by our systems, or simply shadows that change the lip appearance. It is interesting to know how our system reacts to degraded visual information.
- Describe the challenges, open points that after our work must be solved to bring this technology in a commercial solution. We hope that this work has meant an important

step in this direction but of course there are still further issues encouraging future investigations.

1.4 Thesis Outline

The thesis organization is aligned with our system structure, which will be discussed in section 2.3. This system is mainly composed by three different subsystems with its own and independent functionality and this structure is used as basis of our thesis outline. Our work is organized as follows:

In chapter 2 the Lip Reading techniques are going to be presented, as well as a definition of our application scenario. At the end of this chapter a general diagram of our system will be showed in order to clarify its understanding.

In chapter 3 the first module of our general diagram will be explained. The State of the art Lip Finding and Tracking algorithms will be shown. A new solution special for embedded devices will be presented in this chapter together with an evaluation and description of its memory and processing requirements.

Chapter 4 will deal with the second module of our system: the feature extraction. The audio feature extraction used in this work as well as the Noise Reduction techniques will be shown. A comparison of the different visual feature extraction solutions that can be found in the literature will be provided. Two different solutions will be implemented and evaluated.

In chapter 5 the description of our system will be completed with the study of the different fusion strategies and the selection and evaluation of the most appropriate one for an embedded solution. Furthermore in this chapter the visual modelling of the speech will be introduced as well as the synchronization problem between the visual and the acoustic signal.

Chapter 6 presents an evaluation of the whole system in terms of recognition rate and in terms of requirements. The performance of our system for different acoustic SNR will be showed, as well as a study of the visual noise influence in recognition rate. Furthermore, a comparison of the Lip Reading system with conventional Noise Reduction techniques will be provided. Regarding resources requirements, the memory and computational complexity of the system will be presented.

Chapter 7 will summarise the conclusions and contributions of this work and will provide a view into the future giving possible new investigations in order to solve the problems that are still open.

Chapter 2

Lip Reading Technologies for Embedded Devices

2.1 Auxiliary Lip Reading

Since the 17th century it has been documented that exists useful information about speech conveyed in the facial movements of the speakers [Bulwer, 1648]. Hearing-impaired listeners are able to extract information from the movement of the lips and are capable to understand fluently speech using only the visual cues. Furthermore, even those with normal hearing take advantage of Lip Reading techniques improving significantly the intelligibility, especially in noise degraded scenarios. This is demonstrated by the more frequent mishearing of words on the telephone than in person. Machines try to imitate humans to solve problems, so if humans take use of the visual information this information can also be used for automatic speech recognition systems. The deployment of visual information to improve the communication process between humans has been thoroughly studied since more than 30 years and some definitions are well accepted by the audio-visual research community [Dupont et al., 2000]:

- Lip Reading is defined as the perception of the speech purely based on observing the talkers lip movements. No other visual or acoustic cues are included, only the lip movements.
- Speech Reading is defined as the visual perception of the speech which also includes the observation of facial and manual gestures; this is what the hearing-impaired listeners use to communicate.

- Audio-visual speech perception is defined as the perception of the speech by combining Speech Reading with audition. Here lips, hands, facial expressions and acoustic information are used.

These are the main definitions from the psychology. In this work only the visual information of the lips is used, the face expressions and hand movements are not used. The use of other visual information available in a communication process is not beyond the scope of this doctoral thesis.

It is important to provide a system that gives at least the same good results as the conventional speech recognition system. In the present thesis the system will take advantage of the visual information when this is available but when the system cannot use the visual information it must be able to provide the same results as obtained only with the audio information. In a real situation the visual information is not always to be found in the best conditions because of bad illumination, speaker movement or the impossibility to find the speakers lips due to e.g. a prominent beard. Only if suitable visual features are found, they are given together with the conventional acoustic speech features to a combined recognition process, leading to an "Auxiliary" Lip Reading system. When the visual information is not available, the system will work only with the acoustic information performing the same results as a conventional speech recognition system.

2.2 Speech Recognition Modalities: Embedded Devices

Three different speech recognition scenarios can be defined according to where the recognition takes place: server-based, embedded or distributed recognition. In this thesis the embedded recognition scenario is studied, but in this chapter a brief description of the three scenarios will be provided in order to understand the challenges and limitations of recognition in embedded devices.

2.2.1 Server-based Recognition

In this modality the recognition is implemented in a location remote to the user, the audio-visual signal should be transmitted from the user’s terminal to a server. A coding process has to be carried out. Decoding takes place in the server before recognition. The recognition is performed in the server using as input the transmitted audio-visual information. The bit rate has always been tried to be reduced in order to use the minimum Bandwidth. There is a limitation on the information that will be sent to the server and used on the recognition process, above all in the visual signal. The maximum number of visual frames per second it is expected to be transmitted in a UMTS communication (video telephony) is 15 frames per second. Depending on the Quality of Service (QoS) this number could be reduced until 5 frames per second or even less. This aspect concerns also the acoustic signal in terms of the accuracy. In this scenario there are not high restrictions on the use of memory and computer resources. The recognition process takes place on a server and here enough resources are available. As well as capturing problems, transmission errors and interference are produced in the transmission channel; these aspects can degrade also the network speech recognition. Other degradation is introduced by the coding process, where depending on the source bit rate the speech is codified with more or less accuracy (e.g. Adaptive Multi-rate System), this loss of information could affect the recognition process.

2.2.2 Distributed Speech Recognition

In this modality the recognition is distributed between a terminal and a server. The speech features are extracted at the terminal device and transmitted as data, possibly through an error protected channel, to a network-based recognizer. High resource consuming systems can be implemented (as in server-based recognition systems) and only the interesting characteristics in terms of recognition are sent using less channel resources. That means that we do not have to transmit all voice information and consequently more protection resources can be used in the transmission in order to avoid errors. There exists an ongoing standardization effort by the ETSI, Aurora [Pearce, 2000] that seeks to establish such standards for conventional speech recognition. However, currently there are no standardization activities dealing with distributed audio-visual recognition.

2.2.3 Embedded Recognition (Terminal-based)

An embedded system is a special-purpose system in which the computer is completely encapsulated by the device it controls. An embedded system performs pre-defined tasks, with very specific requirements and with limited resources in comparison with a Personal Computer. In embedded recognition also called terminal-based, the recognition is performed in the user’s terminal device or embedded system. The speech signal is not transmitted through a wireless communication network, so it is unaffected by the transmission channel and coding losses. However, computational and memory resources often have to be constrained due to the cost-sensitive nature of the terminal devices and battery supply.

According to the scenario the quantity of available resources will be different. When the recognition takes place in a server the resources in terms of computational time and memory are quite higher than when the recognition is performed in an embedded device. The prize that should be paid for performing the recognition in a server is the communication channel that links the speaker and the server. Through this channel the acoustic information (or the acoustic features in the case of a distribute speech recognition) should be sent to the server and of course the recognition result must be sent back to the user. There are applications where this channel does not represent an additional cost, for example when someone makes a telephone call for making a reservation, the application itself requires the remote access to the server for making the reservation independently that this is be made using speech recognition or not. There are many cases where the application itself has not the necessity of a server access; in this kind of situations it would be advantageous to perform the whole recognition process in the embedded device in order to save the communication channel and therefore reduce the costs. Examples of these kinds of applications in embedded devices are the command controlling of a navigation system in a car, where there is no access to a server as all information is received in the car with a GPS satellite system. Another application is the use of a Mobile Phone for example in SMS dictation or name dialing. In this thesis we are going to focus on these kinds of situations where the complete recognition takes place in an embedded device.

The range of embedded devices is very wide and it is difficult to characterize the resource limitations of all these devices. We describe the device limitations of two important scenarios for this thesis: mobile phones and car environment.

Mobile Phones

- Memory: It is considered as a rough approximation for a conventional Mobile Phone (not a PDA) an available RAM memory size for Lip Reading of about 450 Kbytes. Nowadays about 32 Mbytes would be available for PDA implementations. We have to take into account that if our investigation succeeds the implementation would not be carried out immediately so it can be expected to have much more memory resources than at present, so that this will not be a very restrictive aspect.
- Processing Power: The number of operations that can be carried out by the DSP in a mobile device is limited by the power consumption. We can take an estimation using the current mobile DSP capacity: 200 MHz [ARM, 2006]. In a mobile phone the speech recognition plays not an essential role for the mobile phone functionality. There are many other processes that must be permanently running. On current multimedia mobile phones many time-consuming processes are run at the same time, this is the reason why some mobile phones make use of two different processors. In any way processing power still remains a high restriction in a mobile phone.
- Image resources:
    - Resolution: (VGA 640x480; CIF 352x288; QCIF176x144)

  - Capture: maximum 30 frames per second.
  - Transmitted: maximum 15 frames per second.

Car Environment

To describe an automotive environment restrictions the processor the SH7770 [Renesas, 2006] has been taken as reference; a single chip solution for car navigation systems. One of the most promising applications of speech recognition in automotive environment is the destination input of the navigation system. Voice applications are also integrated in the same chip used for the car navigation. These are the main Characteristics:

- Memory: In our automotive applications a memory space of 128 Mbytes is being available. From this memory approximately 2-5 Mbytes can be used currently for speech recognition. It is reasonable that for Lip Reading at least 10 Mbytes would be available.
- Processing Power: 400 MHz processor.
- Image resources:
  - Resolution: (854x480 pixels)
  - Capture: 30 frames per second

Mobile phones are much more restrictive than automotive environments in terms of requirements. These limitations will be always taken into account in order to select the algorithms and technologies suitable for our implementation. In section 6.5 a general estimation of our whole system requirements will be presented.

2.3 System Overview

In this chapter a general view of our auxiliary Lip Reading system is provided. All audio-visual recognition systems can be divided into certain functionality modules. First of all the acoustic-preprocessing, which generates from the signal caught at the microphone a set of acoustic features that will be used as input for the recognition. The audio preprocessing will not be beyond the scope of this thesis, although a description of this functionality will be provided in section 4.1. The State of the art of the recognition systems will be used in this work, this means Mel Frequency Cepstral Coefficients (MFCC) and Noise Reduction techniques. As output of our Acoustic Preprocessing a set of acoustic features is provided. In a similar way the camera generates the visual signal, the aim of the second module is to obtain a set of visual features from the image, as it can be seen in Figure 1. This objective is one of the topics of this work and it can be divided in two different tasks. First of all the Region of Interest (ROI) for Lip Reading must be found, this means that the position of the mouth must be automatically found, this task is performed by the Lip Finding and Tracking algorithm. Due to the bidimensional characteristics of the visual signal it is possible to separate in the space domain the part of the signal that is carrying important information for the recognition from the other part of the signal that contains only noise (e.g. background). This spatial separation is not possible to be performed in the unidimensional acoustic signal if there are not multiple sensors (beamforming with an array of microphones [Cox et al., 1987], [Cox et al., 1986]). This is an advantage of the visual processing, the spatial separation of noise and signal. Once the position of the mouth is provided we are able to extract the set of visual features that describe the mouth region, this task is performed by the so called Visual Features Extraction block. The visual features should discriminate between the different visemes. A viseme is the visual equivalent of a phoneme or the minimally distinct, abstract class of a sound in a language using only the visual cues [Chen, 2001], [Lucey et al., 2004].

Finally, when the visual and acoustic sets of features are available both of them must be properly combined and the recognition process must be carried out. The integration of the features and the recognition process are described with the same module because depending on the kind of the integration strategy both processes are very connected.

The next 3 chapters are dedicated to explain profusely each of these functionalities: Lip Finding and Tracking, Feature Extraction and Audio-Visual integration and recognition.

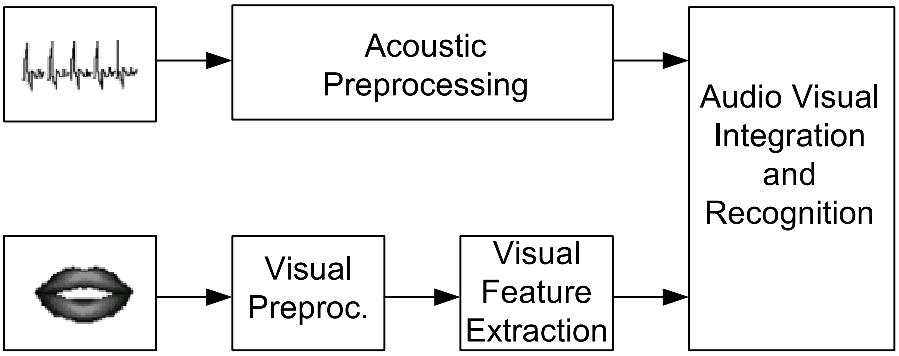
Acoustic Preprocessing
Audio Visual Integration and Recognition
Visual Preproc.
Visual Feature Extraction

Figure 1: General Diagram of our Audio-Visual Speech Recognition System

2.4 Audio-Visual Databases

One of the most important contributions for the success of conventional ASR (using only the audio signal) and for the important improvements obtained in the last years is the availability of a huge quantity of large audio databases in different languages. Collecting all this audio material was a complex process that began in the early eighties with the support of US government agencies (for example the Defense Advanced Research Projects Agency and the National Science Foundation) and of the European Commission or the European Language Resources Association. Different public laboratories, as well as companies and universities took part in consortiums to collect the audio information that later could be shared by the different language research groups.

In contrast to the large quantity of audio databases, it does not exist such a variety of audio-visual databases. There are several reasons that explain this lack of audio-visual databases. First of all, audio-visual speech recognition is a very young research field in comparison with conventional speech recognition. Furthermore, visual speech recognition is a more difficult task than the only audio. Therefore it is normal that companies and universities first focus their resources on the conventional speech recognition. Another reason is the difficulty to record an audio-visual database in comparison with the recording of a pure audio database, this drawback has been solved in the last years with the widely extension of web cams and recording software, but it was a difficult hardware problem in the eighties.

Finally the main reason why there are not enough audio-visual databases is that it has not been constituted consortiums from different public and private organizations to take common advantage of the audio-visual databases. The largest and most important audio-visual databases belong to commercial companies, which difficult the benchmark of the different algorithms. In Table 1 the most important audio-visual databases are summarized, as it can be seen the biggest database (appropriate for Large Vocabularies Continuous Speech Recognition [Potamianos et al., 2004]) belongs to IBM and the use of it is not free for other laboratories.

All speaker independent experiments shown in this thesis have been performed using the CUAVE database. It is a free database that has been used in different works of audio-visual speech recognition [Amarnag et al., 2003], [Gowdy et al., 2004] and it is appropriate for continuous digit recognition. A complete description of the CUAVE database is provided in [Patterson et al., 2002]. CUAVE database consists of 36 individual American English speakers. The selection of the speakers was chosen so that there is an even representation of male and female speakers. Different skin tones and accents are present, as well as other visual features such as glasses, facial hair, and hats. Each speaker was framed including the shoulders and head with a green background. Video is full colour with no aids given for face/lip segmentation. Speakers were standing naturally still (not forced) looking at the camera. The database was recorded with a resolution of 720x480 pixels, but for our work we have used a standard of the mobile phone video sequences with 320X240 pixels every frame, which required a down-sampling of the images. The frame rate used to record was 29.97 frames per second using a 1 Megapixel-CCD, MiniDV camera. From the 36 American English speakers, 20 persons were used for training and 16 for testing in order to test a speaker independent system. The experiments were always connected digits “zero”-“nine” 4 times for every speaker making a total of 640 test numbers and 800 training numbers.

Additionally the system was also evaluated for speaker dependent task, for this purpose we have recorded a total of 378 utterances for training (each utterance is one digit) and 40 utterances of 4 continuous digits for testing of the same speaker in an office environment. Finally, the evaluation of our Lip Finding and Tracking algorithm was performed with an own database recorded in realistic office conditions. CUAVE database offer ideal light conditions and we wanted to test our Lip Finding and Tracking in a more realistic environment. A set of 33 speakers of both sexes, with ages between 25 and 60 years, with different skin colours and with different grades of facial hair has been used for testing our Lip Finding and Tracking in sessions of 10 seconds each (150 frames each speaker). No special light conditions have been used and no reflected markers or special make up were placed on the speaker’s lips.

Database	**Task**	**Language**	**Number of Speakers**
Tulips	“0” to “4” Isolated	English	12
M2VTS	“0” to “9” Isolated	French	37
XM2VTS	“0” to “9” Isolated	English	295
CUAVE	“0” to “9” Continuous	English	36
Potamianos et al.	Isolated Letter	English	49
AMP/CMU	78 Isolated Words	English	10
IBM ViaVoice	LVCSR	English	290
IBM ViaVoice	“0” to “9” Continuous	English	50
SmartKom DB	Command Words	German	60
AV@CAR	Command Words Digits	Spanish	20

BANCA	Digits, Isolated Words	English, French, Italian, Spanish	208
SIEMENS	Commands Words	German	30

Table 1: Most important Audio-Visual Databases

2.5 Recognition Experiments

The audio-visual speech recognition system presented in this work is going to be evaluated in terms of the percentage of errors, substitutions, deletions or insertions. The following terms must be defined:

- Hypothesis: concatenation of words returned as result of the recognition
- Reference: concatenation of words uttered by the speaker
- Substituted words: words that are recognized incorrectly. In the hypothesis and reference different words are found.
- Deleted words: words that are not recognized. They appear in the reference but not in the hypothesis.
- Inserted words: recognized words that were not said. They appear in the hypothesis but not in the reference.
- Correct words: words that are correctly recognized. They will appear in the reference and in the hypothesis in the same order.

Now we can define the percentage of substitutions, deletions, insertions, correct words and the word error rate:

$$Substitutions(\%) = \frac{Number\_of\_Substituted\_Words}{Number\_of\_Reference\_Words} \cdot 100$$

$$Deletions(\%) = \frac{Number\_of\_Deleted\_Words}{Number\_of\_Reference\_Words} \cdot 100$$

$$Insertions(\%) = \frac{Number\_of\_Inserted\_Words}{Number\_of\_Reference\_Words} \cdot 100$$

$$Correct(\%) = \frac{Number\_of\_Correct\_Words}{Number\_of\_Reference\_Words} \cdot 100$$

$$WordErrorRate(\%) = Substitutions + Deletions + Insertions \tag{1}$$

In continuous speech recognition the most important parameter is the *Word Error Rate (WER)* because it conveys the information of all types of errors. In *Correct Words* the influence of *Insertions* cannot be seen. A system can provide many *Insertions* and a very high *Correct Words* rate. In isolated speech recognition the number of insertions is always zero and then:

$$Correct(\%) = 100 - WER(\%) \tag{2}$$

Equation (2) is only true for isolated speech recognition. For continuous speech recognition insertions must be taken into account.

Chapter 3

Lip Finding and Tracking

Not the complete image obtained from the camera is important for Lip Reading, only a small region covering the mouth will be in the interest of the recognition. This is the so called Region of Interest (ROI). The first task to be solved in an automatic Lip Reading system is to find and follow the ROI automatically. This task will be divided into two different processes, when the position of the mouth in the last frame is not known a Lip Finding must be performed. But when this one is known, this information can be used to update the position of the mouth in the current frame, this is called Lip Tracking. Both algorithms will be explained in the next paragraphs.

3.1 Lip Finding

We should now like to address the problem of finding the mouth when there is no information available about its position in the last image. This process can be assumed as an initialization for the tracking. In the section 3.1.1, state-of-the-art Lip Finding algorithms will be presented, this will bring us to introduce in section 3.1.2 our solution for embedded devices.

3.1.1 State of the Art: Bottom-Up and Top-Down Approaches

Finding the position of the lips is the first issue to be addressed in a Lip Reading system. In the published literature this problem was sometimes solved by using invasive methods such as head-mounted cameras, reflective markers placed on the speaker’s lips [Mustafa et al., 2004] or blue coloured lips [Teissier et al., 1999]. All these solutions are inappropriate for a commercial implementation in which the costumer should be able to use the audio-visual speech recognition without restrictions. In order to make a Lip Reading system suitable an automatic and non-invasive Lip Finding and Tracking algorithm should be provided. In the literature algorithms used to find the ROI can be classified into two different groups according to the kind of characteristics they process and the “a priori” knowledge they use [Matthews et al., 2002]:

- Data driven or bottom-up strategies: These algorithms start extracting low-level features from all images, and then look for similar features in different images. There is no semantic meaning assigned to the features: Data driven processes do not care about which features are found as long as they really come from similar objects. The system will learn from a set of training images which of the features are important to find the position of the object. The system will learn from a training database. An example of these kinds of techniques is the Neural Networks. An algorithm to find the position of the lips using this approach [Duchnowski et al., 1995] will be studied in 3.1.1.1

- Model driven or top-down strategies: a set of apriori defined features are going to be extracted from the images and tried to be matched into a model. There is apriori knowledge; a training database will be used to adjust the model to the reality. An example of top-down strategies to find the position of the lips is described in [Stiefelhagen and Yang, 1996] and it will be summarized in 3.1.1.2.

3.1.1.1 Bottom-up Approach: Neural Network

In [Duchnowski et al., 1995] a method for automatically finding the position of the lips by using a hierarchical combination of Neural Networks (NN) was reported. This system epitomizes bottom-up approaches. The algorithm, developed by Interactive Systems Laboratories (University of Karlsruhe and Carnegie Mellon University), finds the ROI by using a concatenation of 4 different NN. In this system face finding is the first step, subsequently lips are detected.

To find the face some characteristics of the image are extracted and are used as the input of the NN, these characteristics are colour information, motion and shape of the objects. Colour information is obtained by the Face Colour Classifier (FCC) which groups different hues as skin or not skin. The colour composition of human skin differs surprisingly little across individuals. Brightness dependencies are eliminated by dividing each of the three colour components (R; G; B) by their sum. The motion, computed as the difference between two successive frames is also used as an input of the NN.

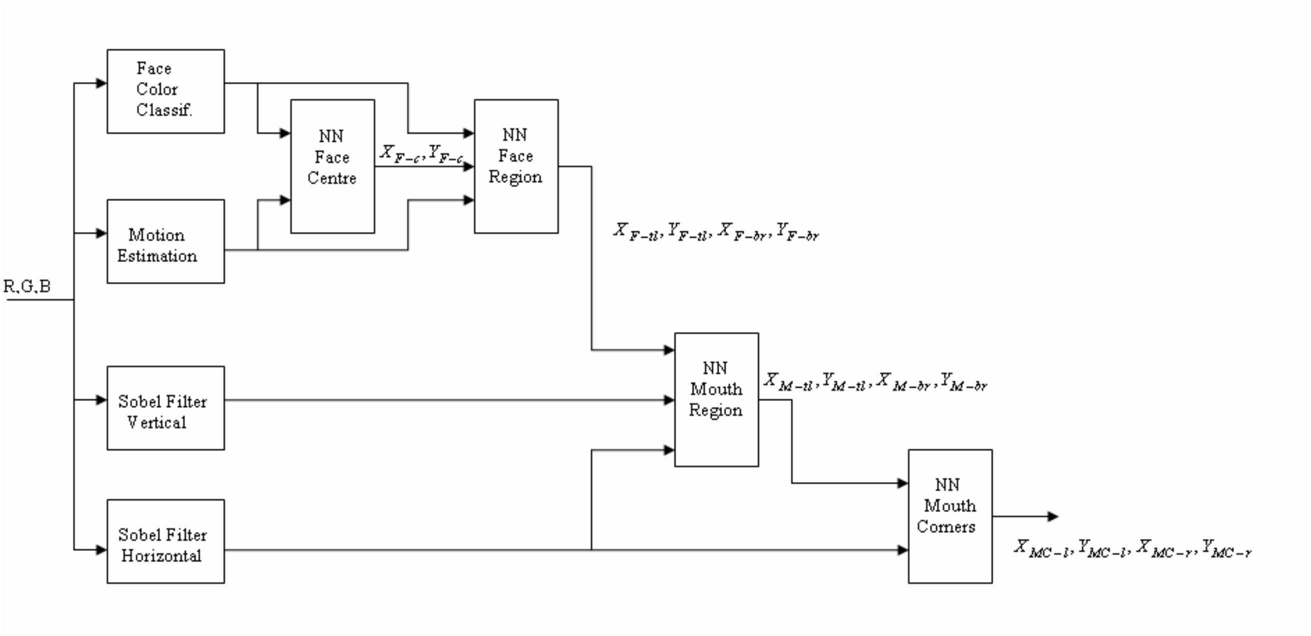

Figure 2: Hierarchical Neural Network System to find the Position of the Mouth.

Two NN are used for finding the position of the face, see Figure 2. The first network receives as input motion and colour analysis information and it estimates the central position of the face, X_{F-c}, Y_{F-c}. The second network uses the area around the central position of the face as input to estimate its size. It provides the coordinates of the region embracing the face $X_{F-tl}, Y_{F-tl}, X_{F-br}, Y_{F-br}$. Networks are trained by using the back-propagation algorithm.

When the position of face is set, a lip-location system is performed. For this system only the face region is given as input to another pair of NN. The first network gives a coarse estimation of the position of the mouth $X_{M-tl}, Y_{M-tl}, X_{M-br}, Y_{M-br}$. The second one locates the two mouth corners $X_{MC-l}, Y_{MC-l}, X_{MC-r}, Y_{MC-r}$ within a region estimated by the first network. In this lip-locator, the gray level image is used and two directional filters (horizontal and vertical Sobel Operators) provide the outlines. The second one uses only a horizontal edge map with higher resolution.

This system offers good results but it consumes many resources, it needs 4 large NN to find the lips. Such requirements make difficult the implementation in an embedded device.

Other approaches are e.g. [Wieghardt, 2001], where Gabor Wavelet Transform is used to find different facial characteristics, and [Viola and Jones, 2004] who have presented a very efficient botton-up approach to detect the position of the face. This system uses a small set of features based on the Haar basis functions the selection of the critical features is performed through a learning process, which is also used to train the classifier. They have obtained impressive results in terms of recognition rate but the implementation in a 200 MIPS Strong ARM processors, similar to ours, is only able with a frame rate of 2 frames per second, which is not enough for Lip Reading.

3.1.1.2 Top-Down approach: Model Based Gaze Finding

In top-down approaches a set of specific features is going to be used. In order to find them apriori knowledge is assumed and different techniques are going to be applied taking advantage of this knowledge. Finally, the position of the found features is going to be evaluated to know if their relative position fulfils a face model.

As an example of this kind of technique a gaze tracking [Stiefelhagen and Yang, 1996] is going to be explained. In this work a set of facial features is searched, once they are found they are used to estimate the pose and track the gaze. The complete system has been outlined in Figure 3.

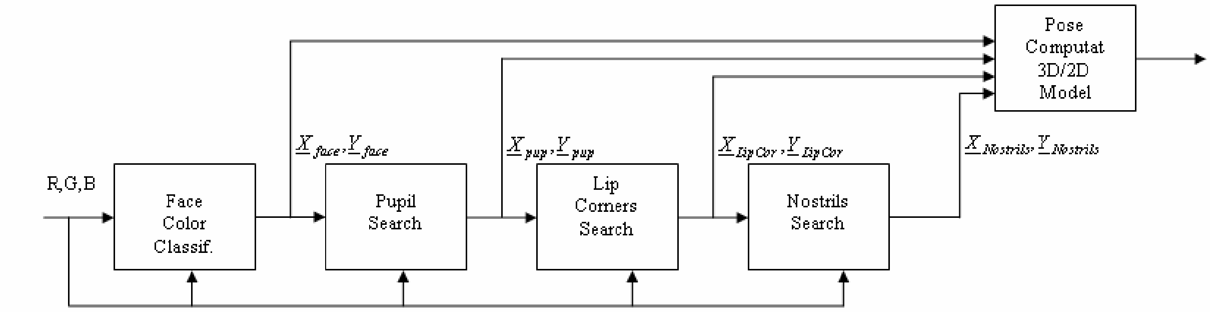
Pose
Computat
3D/2D
Model
$\underline{X}_{face},\underline{Y}_{face}$
$\underline{X}_{pup},\underline{Y}_{pup}$
$\underline{X}_{LipCor},\underline{Y}_{LipCor}$
$\underline{X}_{Nostrils},\underline{Y}_{Nostrils}$
R,G,B
Face
Color
Classif.
Pupil
Search
Lip
Corners
Search
Nostrils
Search

Figure 3: Top-down approach to find the position of the mouth.

First of all, a Face Colour Classifier, a similar system to the one used in the bottom-top approach, is used to provide an estimation of the face region. Each pixel is classified as skin or not skin and the largest connected region of skinned classified pixels is assumed as the face, $\underline{X}_{face},\underline{Y}_{face}$. Once the position of the face is fixed, the two pupils are going to be searched. apriori knowledge is used: the two darkest points in the higher region of the found face that fulfill a geometrical constraint are considered as pupils, $\underline{X}_{pup},\underline{Y}_{pup}$. Once the position of the pupils is known this is used to obtain the mouth searching region. Over this region a horizontal projection of the image is applied to find the vertical position of the line between the lips. To obtain the horizontal boundaries of the lips a smaller search area around the estimated vertical position of the line between the lips is extracted and a horizontal edge operator is applied. The horizontal boundaries of the lips can now be found regarding the vertical projection of this horizontal edge. The vertical line between the lips and the horizontal boundaries of the lips define the position of the lip corners, $\underline{X}_{LipCor}, \underline{Y}_{LipCor}$. Similar to eyes searching, the nostrils can be found by looking for two dark regions that satisfy certain geometrical constraints. Finally, the set of these six features: pupils, nostrils and lip corners are used to find the pose of the face (rotation and translation). This system takes advantage of the top down approach to find the position of different facial features in order to make an accurate estimation of the pose parameters. Our Lip Finding algorithm explained in section 3.1.2, is also a top-down approach but the set of features that we take into account is smaller than the one presented in [Stiefelhagen and Yang, 1996] and all of them are found using the same visual processing. The features used in this thesis are not so small as the ones used in the previous approach (eyebrows will be searched but not pupils, lips but not lip corners). For our purpose an estimation of the mouth region is needed not the gaze orientation. Due to the fact that the features used in our work are larger and therefore easily to be found, our algorithm will gain on robustness.

3.1.2 Lip Finding for Embedded Devices

Several approaches can be found in the literature to solve the lip finding problem. In the last paragraph the most important approaches have been explained; some of them are quite robust. However, they usually require many computing and storage resources like, e.g., those based on large Neural Networks [Meier et al., 2000], [Duchnowski et al., 1995]. The top-down approach shown in the section 3.1.1.2 works by first finding a set of facial features using for each of them different kinds of image processing algorithms (eyes, nostrils and lip corners). In contrast, our Lip Finding for embedded devices extracts a small set of features using for all them the same image processing [Guitarte et al., 2003]. The objective of our Lip Finding is to provide the position of the mouth, to give the coordinates of a box that contains the lips. The box defining the ROI must not be exactly limited by the accurate position of the mouth corners and lip contours but it should be robust. For this reason a set of large enough features was selected. Because of their dimension they are easily found even though without special light conditions. We are not looking for tiny points like mouth corners and nostrils but for large enough regions like lips and the eyebrows. These regions will not be able to provide the accuracy of small points but they will be more robust. The selection of these features was additionally motivated by the fact that they are all horizontal features and therefore only one kind of image processing will be needed to be applied, which implies resource saving. A general schema of the Lip Finding algorithm presented in this thesis can be seen in Figure 4.

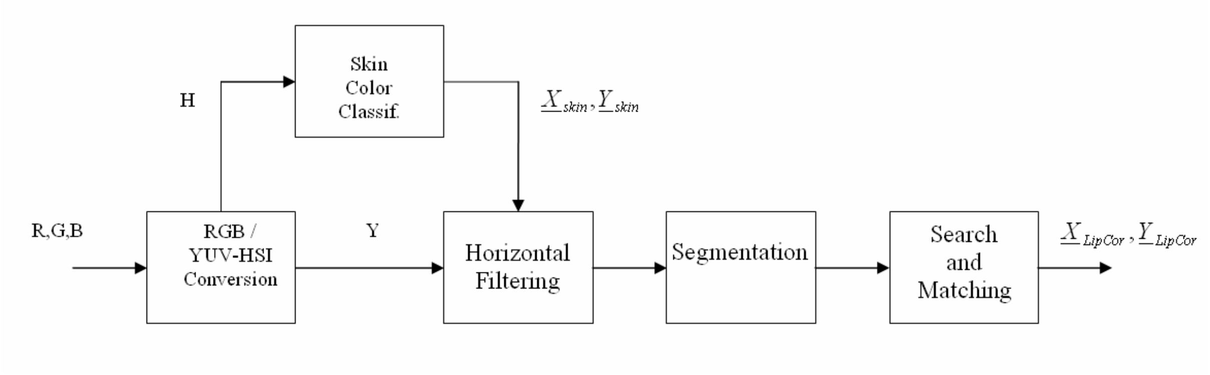

Figure 4:General Schema of our Lip Finding Algorithm for Embedded Devices

Our Lip Finding algorithm for embedded devices is based on a geometric model of the face. Structures of pixels are evaluated in order to know whether their relative positions match a simple prior model of the face. In particular, this model accounts only for the relationships between location of eyebrow(s) and lips.

In order to improve the robustness of the algorithm the colour information is used, all pixels in the image are classified depending on their colour as skin or not skin. Only in the skin labeled pixels the contour-based algorithm is performed. This algorithm starts with a Directional Filtering, which extracts horizontal regions (eyebrows and lips). After Segmentation the system stops working with pixel-based processing and it resumes operating with structures of pixels, called “blobs”. Finally the Search and Matching Process will be accomplished using „blob“ descriptors.

3.1.2.1 Colour Space Conversion

The image information can be shown in different colour spaces. These are mathematical models describing the way colours can be represented as vectors of numbers, typically as three values or colour components. In this work we are going to use three different colour spaces: RGB, YUV and HSV. The image will be normally provided in RGB format, the colour-based algorithm uses the H component of the HSV and finally the contour-based algorithm utilized in our implementation works with the Y component of YUV.

The RGB colour model is an additive model in which red, green and blue (often used in additive light models) are combined in various ways to generate the other colours. The name of the model and the abbreviation "RGB" come from the three primary colours, Red, Green and Blue, this model is used in computer graphics hardware.

YUV model defines a colour space in terms of one luminance and two chrominance components. YUV is used in the PAL and NTSC systems of television broadcasting, which is the standard in much of the world. Y stands for the luminance component (the brightness) and U and V are the chrominance. Y represents the overall brightness, or luminance. In our contour-based algorithm we are going to look for changings on the brightness (contours) this is why this component will be used. The luminance Y, also called luma [Charles, 1996], can be obtained as a linear combination of the gamma corrected tristimulus R, G, B:

$$Y = 0.299 \cdot R + 0.587 \cdot G + 0.114 \cdot B \tag{3}$$

In the HSV (Hue, Saturation, Value) model, Hue (H) specifies the dominant wavelength of the colour, Saturation (S) is the "vibrancy" of the colour, and Value (V) the maximum amplitude of the light waveform. HSV space can be visualized as a cone [Smith, 1982]. In this representation, the hue is depicted as the angle of the colour wheel. The saturation is represented by the distance from the center of a circular cross-section of the cone, and the value is the distance from the pointed end of the cone. Hue component is going to be used to classify each pixel according to its colour information. Hue can be obtained as a non-linear transformation from RGB:

$$H = \begin{cases} 60 \cdot \dfrac{G-B}{\max\{R,G,B\} - \min\{R,G,B\}} + 0, & \text{if } \max\{R,G,B\} = R \text{ and } G \geq B \\ 60 \cdot \dfrac{G-B}{\max\{R,G,B\} - \min\{R,G,B\}} + 360, & \text{if } \max\{R,G,B\} = R \text{ and } G < B \\ 60 \cdot \dfrac{B-R}{\max\{R,G,B\} - \min\{R,G,B\}} + 120, & \text{if } \max\{R,G,B\} = G \\ 60 \cdot \dfrac{R-G}{\max\{R,G,B\} - \min\{R,G,B\}} + 240, & \text{if } \max\{R,G,B\} = B \end{cases} \tag{4}$$

3.1.2.2 Skin Colour Classification

The Skin Colour Classification improves our original contour-based algorithm in order to reduce the high false alarm rate obtained when many horizontal structures appear in the image, e.g. plant leafs in background.

Our contour-based algorithm is applied only in the regions classified as skin colour. The colour classifier based on the Hue component is a very weak classifier, with many false positive (insertions, pixels that do not belong to skin are going to be classified as skin) but with a low number of false negatives (deletions, pixels belonging to skin are not classified as skin). It is known that the colour information is very sensitive to light conditions and moreover any kind of colour skin should be detected, these are the reasons why our colour classifier should not be very restrictive. A further classification will be obtained with the contour-based algorithm applied after the colour classification. In the Skin Colour Classification every pixel will be labeled as skin colour if its Hue value is higher than a lower threshold and smaller than an upper threshold:

$$f(x,y) = \begin{cases} 1 & \text{if } Th_{dw} < H(x,y) \leq Th_{up} \\ 0 & \text{else} \end{cases} \tag{5}$$

These thresholds were experimentally selected. As we have said, the aim of this classifier is not to obtain the right coordinates of the face but just to exclude regions that because of their colour are not possible to be a face.

3.1.2.3 Directional Filtering

From now on only the luminance (Y) component from the YUV colour space will be taken into account, see Figure 5.a. This grayscale image is filtered by a horizontal filter. The horizontal component of the simplified Sobel Operator is used:

$$G_x(n,m) = Y(n-1,m) - Y(n+1,m) \tag{6}$$

After filtering, image thresholding takes place, where the threshold is a fixed percentage of the complete image. A binary image is obtained where all pixels which belong to a contour with a horizontal component are set to “one” (white in Figure 5.c.). To obtain a better contrast, especially in bad light conditions, a histogram equalization is implemented before filtering. A faster execution is run by using the previous frame histogram information for the equalization of the current frame. In the same image scanning a run-length coding (RLC) [Smith, 1997] is created in order to obtain a faster segmentation.

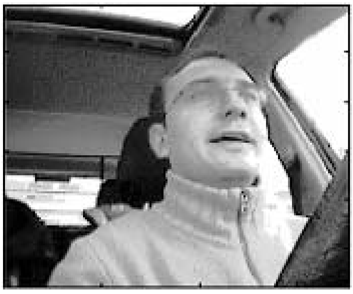

Figure 5.a

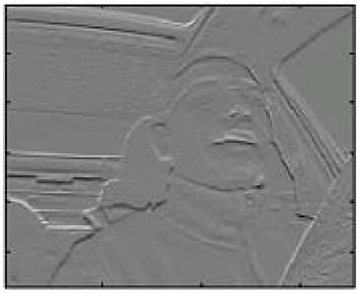

Figure 5.b

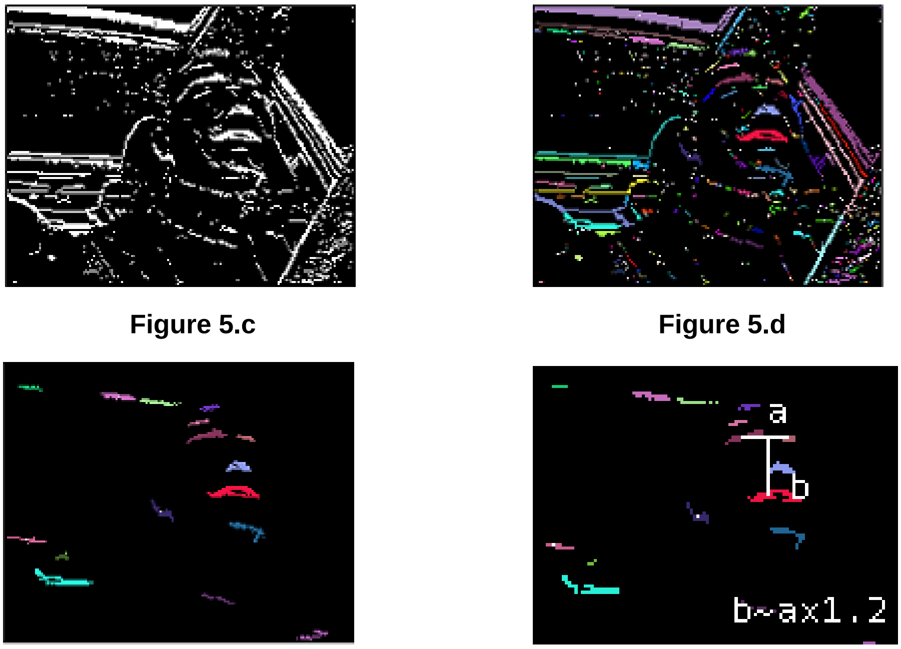

Figure 5.c

Figure 5.d

Figure 5.e

Figure 5.f

Figure 5: Stages of the Lip Finding algorithm. a) Grayscale image, b) horizontal filtering, c) Thresholding, d) Segmentation, e) Blob filtering and f) Matching.

3.1.2.4 Segmentation

A segmented structure is constructed based on the run-length coding. A “blob” is defined as a group of pixels connected according to a specific neighborhood relationship and sharing a common characteristic [Gonzalez and Woods, 2001]. In this case the common characteristic is to belong to the same horizontal contour.

Each “blob” is described only by its area and the coordinates of its center. “Blobs” are filtered according to their area, therefore very small or very large “blobs” will be disregarded. This implies that our algorithm will work for a limited range of distances between the camera and the speaker. In the applications taken into account in this work either the speaker holds the device in his hands or the camera is located at a fixed distance on the dashboard (car environment). The remainder “blobs” after filtering are showed in Figure 5.e.

3.1.2.5 Search and Matching Process

Taking advantage of the restrictions shown in the last paragraph, only approximately 10-25 “blobs” per frame are usually left to take part in the search. The algorithm takes the set of “blobs” which best matches a prior model of a face. The centre of the ideal mouth:

$$\underline{C}_{id} = f\{\underline{C}(e_r), \underline{C}(e_l)\} \tag{7}$$

is computed from the centers of mass of the selected eyebrows. This position is subsequently compared to the location of the nearest “blob” $\underline{C}(m)$. The distance between both centers is considered as a measure of the resemblance to the face model. The set of three “blobs” that minimizes this distance is supposed to be the detected face. When this measure exceeds a certain value the algorithm assumes that no face has been found. This search process is shown in Figure 5.f.

Let $\underline{C}(e_r)$, $\underline{C}(e_l)$, and $\underline{C}(m)$ be the coordinates of the “blob” centers representing the right eyebrow, the left eyebrow, and the mouth, respectively. The objective is to find the three “blobs” e_r, e_l, m that minimize the distance:

$$dist\{\underline{C}_{id}, \underline{C}(m)\}_{e_r.e_l,m} \tag{8}$$

where:

$$\underline{C}_{id} = \begin{cases} C_{id\_x} = \max\{C_x(e_r), C_x(e_l)\} - 0.5 \cdot abs\{C_x(e_r) - C_x(e_l)\} + K \cdot \{C_y(e_r) - C_y(e_l)\} \\ C_{id\_y} = \max\{C_y(e_r), C_y(e_l)\} - 0.5 \cdot abs\{C_y(e_r) - C_y(e_l)\} + K \cdot \{C_x(e_r) - C_x(e_l)\} \end{cases} \tag{9}$$

and where K is a geometrical ratio empirically obtained. It is defined, as shown in Figure 5.d, as $K = b/a = 1.2$.

3.1.2.6 Challenges and Solutions for our Embedded Lip Finding System

We would like to emphasize one of the contributions of this thesis: we have obtained a Lip Finding algorithm that is able to work in real time in an embedded device (implemented in Siemens SX1 Symbian Operating System Mobile Phone). The main important contributions that allow the implementation of such image processing algorithms in a small device will be summarized here:

- The use of simplified version of the Sobel filter combined with a thresholding of the filtered output and a RLC codification. These three operations were performed only in one image scanning reducing the quantity of information in every frame.

- The use of a segmentation structure to describe every image reduced considerably the quantity of information needed in the search process. The use of the RLC coding technique allowed an efficient computation of the segmentation.
- A simplified face model made the search process very fast as well as the use of the colour classification.

In the paragraph 3.3 a profusely evaluation of the Lip Finding and Tracking algorithm will be provided. The main challenges of our algorithm and the respective solutions are touched upon in the next paragraph.

Problems could arise for users with very bushy eyebrows or with glasses. In these cases only one single segment would be recognized as an eyebrow. This situation was taken into account and if there are no segments that satisfy the condition of the face according to the first search algorithm (two eyebrows and one mouth), a second model with a single large eyebrow is assumed and the process is repeated.

Our deterministic algorithm is based on the horizontal segments. The accuracy of our algorithm is related to the quantity of horizontal segments that are going to be taken into account for the search process. When this number is very high the probability that three false segments minimizes the mapping condition of formula (8) increases. This problem was firstly solved by reducing the number of “blobs” that take part in the search according to an area criterion, as it can be seen in Figure 5.d and Figure 5.e. Even though the previous blob reduction, many horizontal regions can appear, e.g. when plant leafs are part of the background, the high number of horizontal structures will surely degrade the performance of the algorithm. This problem was solved by applying a very simple colour classification based on a transformation of the colour space from YUV to HSI (hue, saturation and intensity) explained in section 3.1.2.2. With this implementation the search region is reduced and also the quantity of horizontal contours in background, improving the performance of the Lip Finding. The selection of the threshold for the colour classification should not very restrictive as it was commented in section 3.1.2.2. Colour classification improves the results especially for backgrounds with many horizontal contours. Furthermore, it will not imply an increment on the computing requirements. The increment in the computational charge due to the colour classification will be compensated by the reduction of operations needed for the evaluation of the contour analysis, because the area where the algorithm should be applied will be smaller than without colour classification.

3.2 Lip Tracking

3.2.1 State of the Art: Bayesian Modeling for Tracking

Once mouth is found they must be tracked. The objective is to provide the position of the mouth in every image, when there is not additional information the search must be done along the whole image: Lip Finding. If the position of the mouth is known in the last frame, this can be used to restrict the area where the mouth will be searched, this is called Lip Tracking. In the literature different approaches can be found to perform the tracking. One of them is the so called Bayesian tracking, the mathematical formulation of this problem [Arulampalam et al., 2002] is very similar to the one used for the recognition process. Furthermore, one of its applications is the Particle Filter, which is nowadays one of the most successful tracking techniques. This is the reason why the Bayesian approach will be shown, although for this work a simplified tracking algorithm for embedded devices was implemented, it will be explained in section 3.2.2.

Tracking problem consists in estimating the state of a system (e.g. position of an object) that changes over time using a sequence of noisy measurements. In the Bayesian approach to dynamic state estimation the posterior Probability Density Function (PDF) of the state is estimated based on all available information, including the set of received measurements.

Tracking issue can be mathematically defined. Let's consider the evolution of the state sequence X_k of a target, given by:

$$X_k = f_k\left(X_{K-1}; V_{K-1}\right) \tag{10}$$

where f_k is a function of the state X_{k-1} and V_{k-1} is a process noise sequence. The objective of tracking is to recursively estimate X_k from a set of system measurements:

$$Z_k = h_k\left(X_K; N_K\right) \tag{11}$$

where h_k is a function of the state X_k, and N_k is a measurement noise. In particular we seek to estimate X_k based on the set of all available measurements $Z_{1:k} = \{Z_i, i = 1,...,k\}$ up to time *k*.

It is assumed that the initial PDF of the state vector $p(X_0 \mid Z_0) \equiv p(X_0)$ is available. Then the PDF $p(X_k \mid Z_{1:k})$ may be obtained recursively in two stages: prediction and update.

The prediction stage involves using the system model to obtain the prior PDF of the state at time K via the Chapman-Kolmogorov equation:

$$p(X_k \mid Z_{1:k-1}) = \int p(X_k \mid X_{k-1}) p(X_{k-1} \mid Z_{1:k-1}) dX_{k-1} \tag{12}$$

At time K, a measurement Z_k becomes available, and this may be used to update the prior (update stage) via Baye’s rule:

$$p(X_k \mid Z_{1:k}) = \frac{p(Z_k \mid X_k) p(X_k \mid Z_{1:k-1})}{p(Z_k \mid Z_{1:k-1})} \tag{13}$$

In the update stage the measurement Z_k is used to modify the prior density to obtain the required posterior density of the current state. This recursive propagation of the posterior density is only a conceptual solution. In general it cannot be analytically determined. Solutions do exist in a restrictive set of cases including the Kalman filter. When the analytic solution is intractable Particle Filters can approximate the optimal Bayesian solution.

3.2.1.1 Kalman Filter

If the next set of highly restrictive assumptions can be hold, then the optimal solution to the tracking problem using the Bayesian Modeling is the Kalman filter [Kalman, 1960]:

- The process noise sequence V_k, and the measurement noise sequence N_k can be drawn from Gaussian distributions of known parameters.
- f_k is known and a linear function of X_{k-1} and V_{k-1}
- h_k is known and a linear function of X_k and N_k Under these conditions the equations (10) and (11) can be re-written as:
-

$$X_k = F_k X_{k-1} + V_{k-1} \tag{14}$$

$$Z_k = H_k X_k + N_k \tag{15}$$

F_k and H_k are known matrices defining the linear functions. The covariance of V_{k-1} and N_k are respectively Q_{k-1} and R_k. Here we consider the case when V_{k-1} and N_k have zero mean and are statistically independent. The Kalman filter derived using (12) and (13) can then be expressed as the following recursive relationship:

$$p(X_{k-1} \mid Z_{1:k-1}) = N(X_{k-1}; m_{k-1|k-1}, P_{k-1|k-1}) \tag{16}$$

$$p(X_k \mid Z_{1:k-1}) = N(X_k; m_{k|k-1}, P_{k|k-1}) \tag{17}$$

$$p(X_k \mid Z_{1:k}) = N(X_{k-1}; m_{k|k}, P_{k|k}) \tag{18}$$

where:

$$m_{k|k-1} = F_k m_{k-1|k-1} \tag{19}$$

$$P_{k|k-1} = Q_{k-1} + F_k P_{k-1|k-1} F_k^T \tag{20}$$

$$m_{k|k} = m_{k|k-1} + K_k (Z_k - H_k m_{k|k-1}) \tag{21}$$

$$P_{k|k} = P_{k|k-1} - K_k H_k P_{k|k-1} \tag{22}$$

and where $N(X; m, P)$is a Gaussian density with argument X, mean m and covariance P and:

$$S_k = H_k P_{k|k-1} H_k^T + R_K \tag{23}$$

$$K_k = P_{k|k-1} H_k^T S_k^{-1} \tag{24}$$

are the covariance of the innovation term and the Kalman gain, respectively.

As value for the tracking X_k will be selected to maximize the posterior equation (18). The mean value of the Gaussian maximizes the probability; this is the reason why the estimated tracked value will be given by the mean of the Gaussian. It can be evaluated in a recursive form with equation (21). A detailed description of the implementation of a Kalman filter can be found in [Kalman, 1960]. In order to apply properly the Kalman Filter the restricted conditions given at the beginning of the section must be met. In our case the state evolution function f_k is not known and it can be non-linear as well as the measure function h_k. In such cases approximations of the optimal solutions like Extended Kalman Filter [Wan and

Van der Merwe, 2000] or Particle Filter can be applied. Particle Filter is going to be described as it is one of the most popular approaches at the moment to perform tracking.

3.2.1.2 Particle Filter

When the restricted conditions assumed for the Kalman filter are not met a Particle Filter can be used. This is a technique for implementing a recursive Bayesian filter by Monte Carlo simulations. The main point is to represent the required posterior density function by a set of random samples with associated weights and to compute estimates based on these samples and weights [Arulampalam et al., 2002]:

$$p(X_{0:k} \mid Z_{0:k}) \approx \sum_{i=1}^{M} w_k^i \delta(X_{0:k} - X_{0:k}^i) \tag{25}$$

As the number of the samples becomes very large, this Monte Carlo characterization becomes an equivalent representation of the usual functional description of the posterior PDF. In these conditions, the Particle Filter, using a Sequential Importance Sampling algorithm, approaches the optimal Bayesian estimate. Particle Filter has shown to be very reliable to track complicated movements with occlusions. The Particle Filter algorithm is summarized in the next steps [Bolic 2004]:

1. Initialization *t=0;* Generate N state samples $a_0^{(1)},...,a_0^{(N)}$ according to the prior density $p(x_0)$ and assign the identical weights $w_0^{(1)},...,w_0^{(N)} = 1/N$.
2. At time step t we have N weighted particles $a_{t-1}^{(n)}, w_{t-1}^{(n)}$ that approximate the posterior distribution of the state $p(X_{t-1} \mid Z_{1:t-1})$ at previous time step.
    a. Resample the particles proportionally to their weights, for example keeping only particles with high weights and removing particles with small ones.
    b. Draw N particles according to a dynamic model.
    c. Weight each of the new particles according to its likelihood:

$$w_t^{(n)} = \frac{p(Z_t \mid X_t = a_t^{(n)})}{\sum_{m=1}^{N} p(Z_t \mid X_t = a_t^{(m)})} \tag{26}$$

    d. Give an estimate of the state $\hat{x}_t$ as Maximum a Posteriori (MAP):

$$\hat{X}_t = \arg\max_{x_t} p(X_t \mid Z_{1:t}) \approx \arg\max_{a_t^{(n)}} w_t^{(n)} \tag{27}$$

We show an implementation example of Particle Filter for face tracking [Hamlaoui et al., 2005]. In this Particle Filter the tracked variables convey the pose information P of the face (translation, scaling and rotation) as well as the facial actions C described by the Active Appearance Model [Cootes et al., 2001] (values that control the different expressions of the face):

$$X_t = \{P_t, C_t\} = \{T_t^x, T_t^y, \alpha_t, \theta_t, C_t\} \tag{28}$$

The particles are realizations of these state variables. First of all, according to a giving PDF of the initial position of the face a fix number of particles is produced. These are random hypothesized state and they dependent on the variance of each component. In order to actualize the weightings the formula (26) is used. The observation model consists of the likelihood $p(z_t \mid x_t)$ which indicates that a hypothesized state gives rise to the observed data. In [Hamlaoui et al., 2005] this probability is obtained as a function of the distance between the hypothesized appearance of the face given by the model texture using the facial action coefficients $g_{\mathrm{mod}el}(C_t)$ and the image patch sampled at the hypothesized pose and shape $g_{image}(P_t, C_t)$:

$$p(z_t \mid x_t) = \beta \cdot \exp - d[g_{\mathrm{mod}el}(C_t), g_{image}(P_t, C_t)] \tag{29}$$

where β is a normalizing constant.

In each iteration only the particles with highest weights survive and new particles are generated by adaptive dynamics [Hamlaoui et al., 2005]. It is conformed by the predicted state in the last frame, an estimation shift for each component of the state and a random value dependent on the variance of each state component. The number of particles in every frame depends on the values of the weights. When the weights are high, which implies a correct matching, a small number of particles is taken into account than when the matching is not very good providing smaller weightings. The implementation of Particle Filter for tracking of faces has provided very good results working even with occlusions (temporal looses of the target in the viewing field). Nevertheless this will not be the prior scenario for Lip Reading, we do not need that our system is able to read our lips when they are not visible. These algorithms are still very high resource consuming, the face tracking proposed in [Hamlaoui et al., 2005] is implemented in a 2.4 GHz PC and is able to proceed only 2 frames per second, which is not acceptable for Lip Reading.

3.2.2 Lip tracking for Embedded Devices

3.2.2.1 Lip Tracking Algorithm

Tracking algorithms showed in the state of the art like Kalman and Particle Filters are probabilistic approaches for predicting the position of an object taking into account its position in the past. These algorithms are able to follow complicated movements even occlusions. The objective of this work is not to go into detail of the tracking algorithms but to generate a system able to work in an embedded device in normal light conditions. The tracking scenario considered in this thesis can be characterized because the tracked are not very fast. In our scenario we are assuming a cooperative speaker that is aware of the recognition task. The difficulty of many tracking systems is due to the fact that the target is not aware of the tracking or even tries to avoid it. This is not the situation studied in this work. The aim of the system is to track the coordinates of a box containing the mouth:

$$X = \{X_{ul}, Y_{ul}, X_{br}, Y_{br}\} \tag{30}$$

where:

- X_{ul} : Horizontal coordinate upper left box corner
- Y_{ul} : Vertical coordinate upper left box corner
- X_{br} : Horizontal coordinate bottom right box corner
- Y_{br} : Vertical coordinate bottom right box corner

Tracking features are the two lips. Upper and lower lips are two relatively large regions in comparison with other spot characteristics like pupils or nostrils. The size of the lips regions makes easy the tracking also in normal light conditions and furthermore a restrictive accuracy requirement does not exist. As it has been said in Lip Finding for a pixel-accurate tracking smaller regions should be considered.

The aim of the Bayesian approach for tracking is to provide the probability density function of the target's position in the current frame knowing a set of measurements of its position in the previous frames $P(X_k \mid Z_{1:k-1})$. The position that maximizes this probability function will be selected as the estimated position of the target. Once the estimation is performed the measurements to update and find the target will be carried out only in a region near to the estimated position. This is the advantage of the estimation in tracking: a prediction of the target location will limit the region where the update must be performed. At this point it is important to define a ratio between the maximum movement of the target between 2 consecutive frames and the square of the mean features area, let's call it Movement Size Ratio (MSR):

$$MSR = \frac{\max(|X_k - X_{k-1}|)}{\sqrt{mean(features\_area)}} \tag{31}$$

An example with a high and a low MSR is provided in Figure 6.a and Figure 6.b respectively. Features used to describe the target in the current frame are represented with a red x-shaped cross, the blue squares represent the regions where the features are going to be searched using an estimation of the position, the size of these regions is dependent on the size of the searched features. The arrows represent different possible movements of the target between two frames and finally the green box represents the whole area where the features will be searched if there is no estimation and just a region around the current position of the target including all possible movements is analyzed. Figure 6.a represents a situation where the MSR is higher than in Figure 6.b, this is the reason why in Figure 6.a the movement vectors are larger and the search regions (blue boxes) smaller. In a situation like the one shown in Figure 6.a the estimation of the position saves many resources as the searching area will be only one of the small blue squares, the one given by the estimation, and not the whole green box. Quite the contrary occurs in Figure 6.b, in this situation the MSR is quite small and the difference between searching the features with movement estimation or not is quite small, even the time required to estimate the new position could be higher than the time needed to evaluate the whole green box.

In our implementation we are looking for upper and lower lips regions. Assuming 15 frames per second and a distance between the camera and the speaker of 20-50 cm, the size our features (lips) is larger than the expected movement of the center of the mouth from one frame to another one. So the MSR in our situation will is producing a result similar to the one observed in Figure 6.b. This is the reason why in our tracking we are not using a position estimation of the features in the future frame and we are just searching them in an area near to the current position (green box).

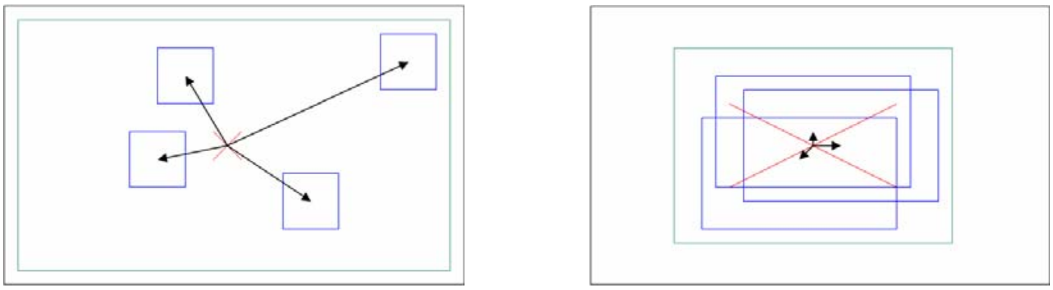

Figure 6.a **Figure 6.b**

Figure 6: Comparison of the searching region using estimation of the position for high Movement Size Ratios (Figure 6.a) and for small MSR (Figure 6.b).

This point can be also shown using the probability density function of the target position taking into account a system with a high MSR (Figure 7.a) and a system with a low MSR (Figure 7.b), in this figures only one dimension has been considered and in blue the feature searching area has been drawn (in this example both systems have the same mean feature size, therefore in the first one the maximum movements are larger that in the second one). In Figure 7.a.1 and Figure 7.b.1 the probability distribution of the position of the target without more information is presented, if there is not more information a prediction is not easy to be made. In Figure 7.a.2 and Figure 7.b.2 the position of the features in the last frame is known, the probability function gives more information of the actual position. As it can be seen when the system has a small MSR the distribution will have a smaller variance as the movements are going to be smaller than in a system with a high MSR. As it has been said, in this example both systems have the same mean feature area. In the system with a small MSR no more information is needed, in fact the form of the probability distribution is not important as the length of the searching features is almost bigger than the variance of the probability function. It will be needed to carry out the feature searching in a region around the position of the measurement features in the last frame Z_{k-1}. In the case of a system with a higher MSR more information can be useful to reduce the searching region. Information of the position in all previous frames can be used to define a new probability function as it is shown in Figure 7.a.3 and Figure 7.b.3. The estimation of this density function can be obtained by the using tracking systems as Kalman and Particle Filters. As we have seen this information is not important in a system as ours with a low MSR.

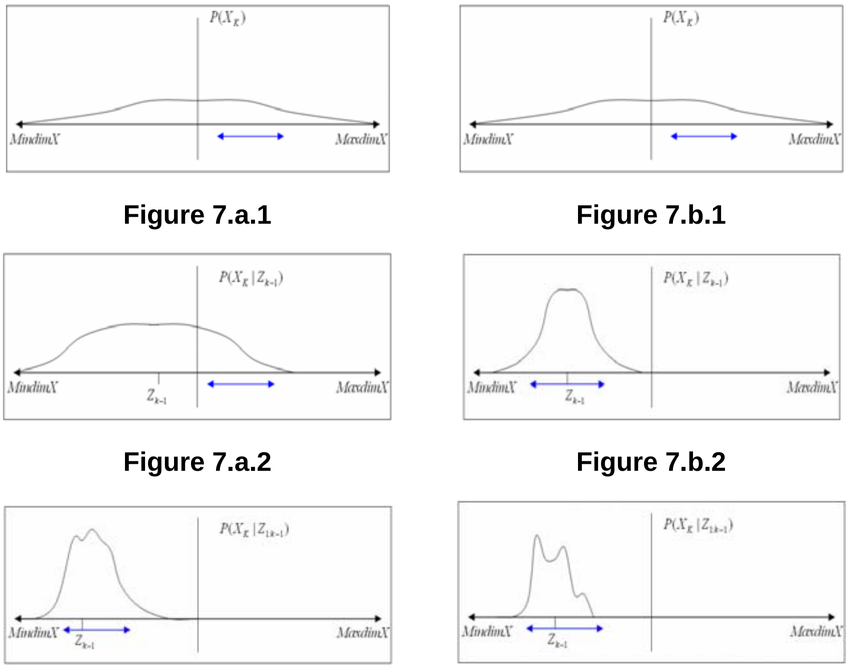

Figure 7.a.1

Figure 7.b.1

Figure 7.a.2

Figure 7.b.2

Figure 7.a.3

Figure 7.b.3

Figure 7: Example of different Probability density Functions for tracking applied to a target where the MSR is high in Figure 7.a.1/2/3 and where this ratio is quite small in Figure 7.b.1/2/3. Using different apriori knowledge.

Our tracking algorithm has been implemented in a Mobile Phone providing good results when the speaker was not making very fast movements (normal situation using an ASR). The influence of the tracking errors on the recognition results was also studied and the results are summarized in chapter 6.

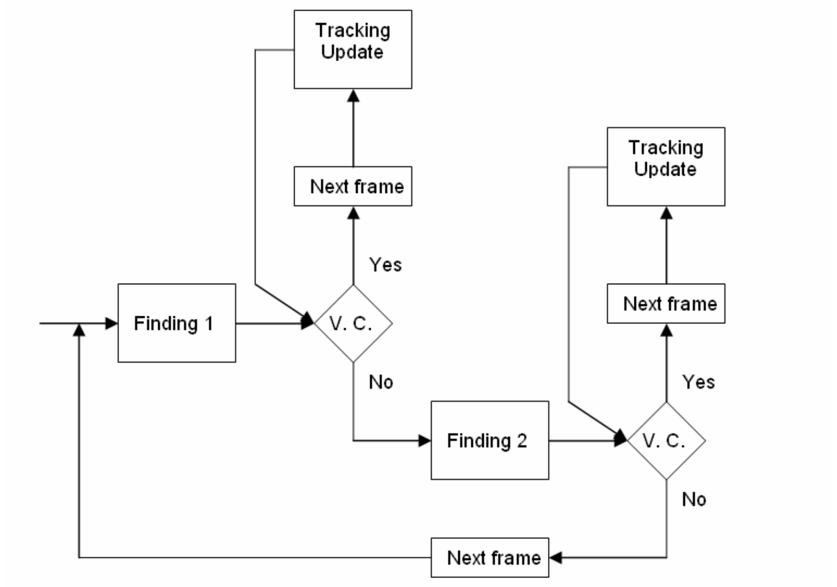

Figure 8: Algorithmic Description of the Lip Tracking for embedded devices.

In Figure 8 a general schema of the Lip Tracking algorithm is described. The system uses the Lip Finding algorithm explained in section 3.1.2 when no previous information of the lip position is available; this functionality is called “Finding 1”, see Figure 8. This happens in the first frame of a sequence or whenever the lips cannot be correctly located in the previous frame. Our Lip Finding algorithm provides a region where the mouth should be located, this region will be the input of a “Verification Criterion” (V.C: in Figure 8). This block is profusely explained in the next chapter, its objective is to asure whether the lips are inside the region or not. When the V.C. provides a positive result (the lips are inside the region) the information of the mouth in the current frame is going to be used to provide a searching region for the next frame. According to the previous discussion and due to the low MSR of our system it is not necessary to make an estimation of the position, just an increment (10%) of the actual region will be enough, as it can be seen in Figure 9. This information can be used to update the lips’ coordinates by inspecting a region close to the last position rather than in the whole image. It has been observed that given approx. 15 frames per second or higher rates, the location of the lips cannot differ too much from one frame to the next one for the application scenarios we are dealing with. For example, in a mobile phone application the device is held by the speaker or in a car environment the relative position of the speaker does not change too much in 1/15 sec. When the result of the “Verification Criterion” (V.C.) is negative another version of the Lip Finding is applied, called “Finding 2”, see Figure 8. In this one, as it was seen in section 3.1.2.6, only one eyebrow is considered. After this it will be proceed in the same way as for “Finding 1”.

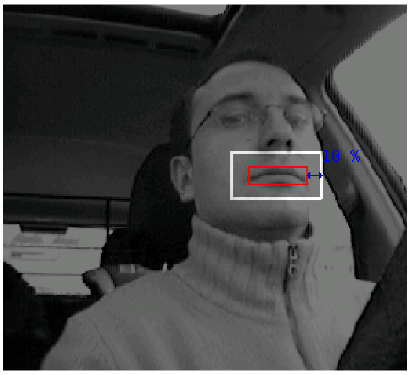

Figure 9: Lip Tracking

3.2.2.2 Verification Criterion

This algorithm receives the position where the lips are supposed to be located from Lip Finding or Lip Tracking. It finds the features that describe the lips. If the typical features of the lips are found, the verification is completed and the lip coordinates are updated. Differences must be pointed out between the feature extraction associated with the Lip Finding and Tracking process and the features extracted to make the Lip Reading (recognition). There are two different set of features, the first one used for the Lip Tracking is a small set of features which only gives information whether the mouth is in the region or not. The set of features used for the Lip Reading (Recognition) is more complex as it must be used to distinguish different visemes, it will be analyzed in the next chapter.

In the Feature Extraction implemented for the Lip Tracking, the upper and lower lip contours are sought. First, the upper lip contour is obtained with a gradient filter that highlights bright-to-dark intensity changes (from top to bottom in the image), see the red square in Figure 10. Then another dark-to-bright filter is applied to obtain the lower contour of the lips blue square in Figure 10.

From each of these two horizontal filters the largest region is going to be obtained as upper and lower lip, each of these regions is going to be described by the coordinates of two points that define the box containing each lip:

- X_{ul}^{U-Lip} : Horizontal coordinate upper left box corner of the Upper Lip
- Y_{ul}^{U-Lip} : Vertical coordinate upper left box corner of the Upper Lip
- X_{br}^{U-Lip} : Horizontal coordinate bottom right box corner of the Upper Lip
- Y_{br}^{U-Lip} : Vertical coordinate bottom right box corner of the Upper Lip
- X_{ul}^{L-Lip} : Horizontal coordinate upper left box corner of the Lower Lip
- Y_{ul}^{L-Lip} : Vertical coordinate upper left box corner of the Lower Lip
- X_{br}^{L-Lip} : Horizontal coordinate bottom right box corner of the Lower Lip
- Y_{br}^{L-Lip} : Vertical coordinate bottom right box corner of the Lower Lip

If both segments fulfill some geometrical properties the verification will be completed, and the lip position is updated. In this implementation, it is assumed that the regions detected as upper and lower lips are correctly describing the mouth when they meet the next set of geometrical properties:

- Both regions detected as lips must be at least 1.5 longer as higher.
- The region detected as upper lip must be over the region detected as lower lip.
- Both regions must be aligned in the horizontal axis.

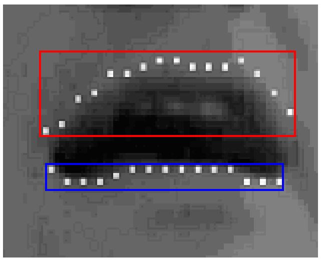

Figure 10: Upper and lower lip detection for Lip Tracking verification criterion

The final result of Lip Tracking is the set of coordinates containing the mouth and given in equation (30), which are obtained from the coordinates of both lips, assuming the reference point (0,0) on the upper right corner of the image:

$$\begin{aligned} X_{ul} &= \min\left\{X_{ul}^{U-Lip}, X_{ul}^{L-Lip}\right\} \\ X_{br} &= \max\left\{X_{br}^{U-Lip}, X_{br}^{L-Lip}\right\} \\ Y_{ul} &= Y_{ul}^{U-Lip} \\ Y_{br} &= Y_{br}^{L-Lip} \end{aligned} \qquad (32)$$

We would like to point out that the Features Extraction used in this implementation can be improved by using other kind of algorithms. Our features are good enough for the tracking but they do not have enough information to allow the recognition of the visemes in Lip Reading. Our Lip Finding and Tracking algorithm has also been used to obtain a real time implementation of the ASM. Providing an approximation of the ROI, the required computational time will be reduced because the ASM algorithm should now fit the lips only in a small region of the image, not in the whole image. It can be said that a coarse Rigid Registration (position, orientation and scaling) of the lips can be obtained by using our algorithm. A Real time implementation of the ASM for the lips was built by using our Lip Finding and Tracking proposal and it will be shown in section 4.2.2

3.2.2.3 Challenges and Solutions for our Embedded Lip Tracking System

As we have seen the implementation of the tracking algorithm is very straight forward and much simpler than other accurate tracking algorithms like Particle Filters and Kalman filter. In spite of this, the Lip Tracking implementation explained in this work meets the requirements of Lip Reading and can be implemented in an embedded device like the Siemens SX1 Symbian Operating System Mobile Phone. In order to obtain better recognition results, a post-processing stabilization block is used after the Lip Finding and Tracking algorithm [Guitarte and Lukas, 2002].

Lip Finding and Tracking takes an image from the camera and gives two points (four coordinates) describing the box where the mouth is found, as giving in equation (30).

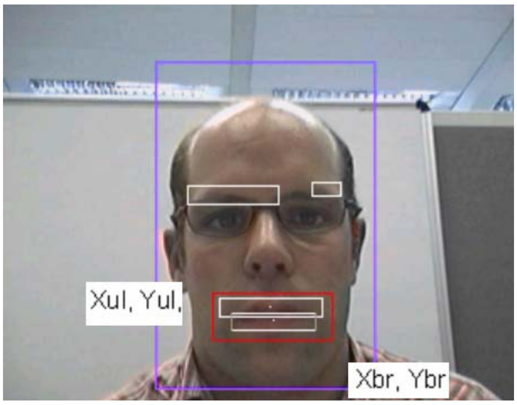

Figure 11: Coordinates obtained from Lip Finding and Tracking, which will be stabilized

The stabilization process is made up of a concatenation of different algorithms, they work on the four coordinates of the two points defining the mouth position X_{ul}, Y_{ul}, X_{br}, Y_{br}.

This stabilization algorithm is made up by a Low Pass Filter. It is a typical mean filter that gives a frame-wise mean value of the current frame and of the neighbors frames. After this filter an X-symmetries implementation tries to avoid the half mouth problem. Due to illumination changes sometimes suddenly only a part of the mouth (right or left) is found, this implies that X_{ul} and X_{br} change their value in a non logical way. We impose the condition that the sign of the change (delta) between the coordinate's values in the previous and in the present frame must be the same, allowing right and left movements. When their signs are not the same, only the minimum absolute value of the delta will be considered for both coordinates, allowing e.g. mouth rounding /o/ viseme.

The algorithm provides a confidence measure by using the verification criterion. This binary value is set to one when the system relies on the position of the mouth and zero when the system was not able to find the mouth. According to the confidence value an interpolative process is performed for all frames where this flag is set to zero. With this implementation we achieve a more robust detection system. This algorithm will be independent on illumination changes that degrade the system for a short period of time. Subsequently, a Median Filter is applied to avoid impulse noises that are not detected by our Lip Finding system. The use of this stabilization process improves the results in terms of recognition rate. However, its implementation cannot be performed in real time, as the whole sequence of images has to be captured, processed completely and then the recognition will be performed. Although for the recognition is not a problem because the result will not be provided until the end of the utterance, for a real time Lip Finding and Tracking without delay this kind of stabilization is not allowed. For this reason in the evaluation of Lip Finding and Tracking algorithm this stabilization has been omitted. A simple dynamical stabilization using only the mean value of the last 5 frames (without future knowledge, and therefore without interpolation) was used.

3.3 Evaluation of Lip Finding and Tracking for Embedded Devices

The Lip Finding and Tracking algorithm was evaluated in office conditions using an own recorded Database. A set of 33 speakers of both sexes, with ages between 25 and 60 years, with different skin colours and with different grades of facial hair have tested the system in sessions of 10 seconds each (150 frames each speaker). No special light conditions have been used and no reflected markers or special make up were placed on the speaker’s lips. In order to test the system, the reference position of the mouth was computed semi-automatically (first of all our algorithm was run and then the results were manually corrected to obtain in all frames the right reference). An error has occurred when the coordinates given by our algorithm are not contained in the reference box which is 5% larger as the accurate box containing the limits of the lips. The distance between the camera and the speaker was between 20 and 50 cm, and people were asked to look at the camera. From the total of 4950 images using the Lip Finding and Tracking algorithm the error rate was 5.8%, as it can be seen in Figure 12.

We have also obtained the results using only Lip Finding without Tracking, that means without using the information about the previous frame. In this situation the error rate raises up to 27.4%, see Figure 12.

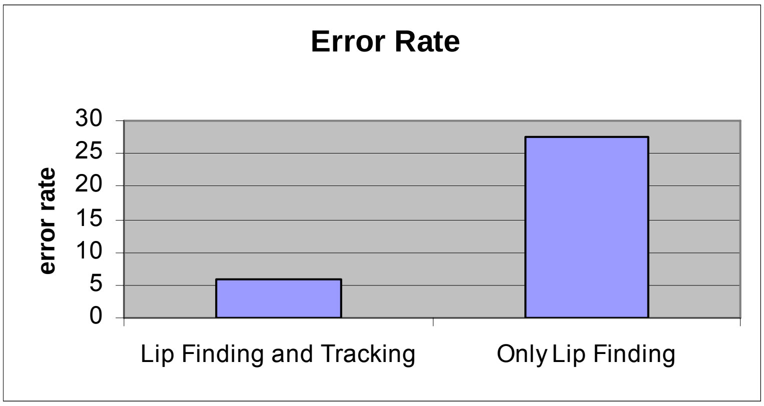

Figure 12: Error Rate for Lip Finding and Tracking

Additionally, to give a better criterion of the generalization ability of the results, this algorithm is applied to different users, it is important to know how the erroneous frames are distributed over the different speakers. Figure 13 shows that the 78.8% of the speakers have an error rate smaller than 5%.

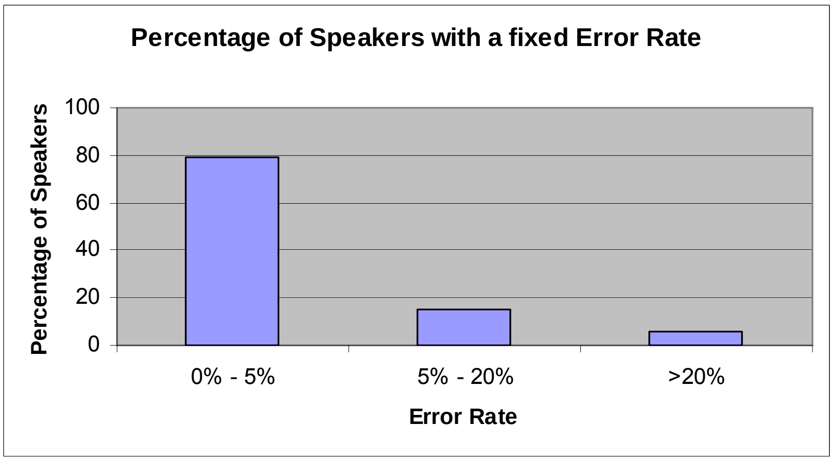

Figure 13: Percentage of Speakers vs. Error Rate

We have classified the error frames on the one hand as false alarm (false positive), when our system has found a structure that does not match the lips, and on the other hand as not found when the system knows that it cannot find the lips in the image and gives therefore no output. The percentage of errors classified like false alarm for the Lip Finding algorithm is 39.2 % and this value is increased to 47.2%, see Figure 14, when Lip Tracking is applied, since some erroneous frames are propagated with the tracking system.

For Lip Reading systems it will be very interesting to have a small false alarm rate because that means a reduction on the video noise. In the same way as we have acoustic noise we will have visual noise when the image information -ROI given by our system- does not match completely the correct lip area. So, as long as we can provide a small false alarm rate we will be able to repeal the visual noise.

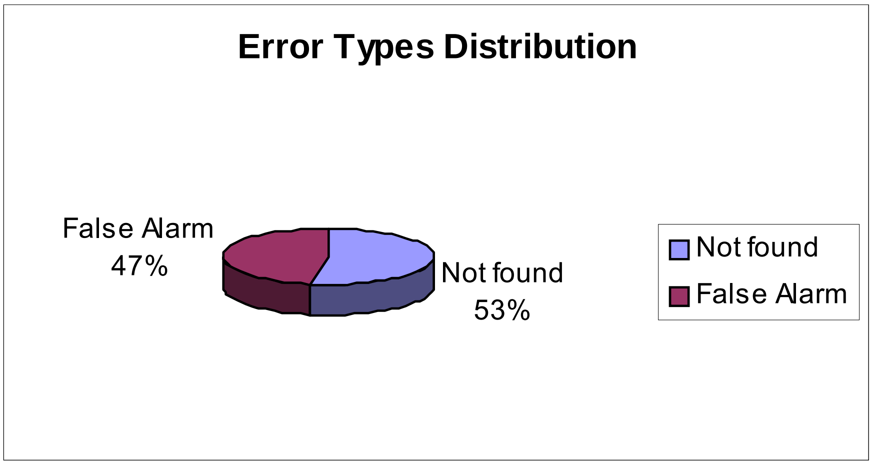

Figure 14: Error type Distribution for Lip Finding and Tracking

The bursts of errors have been checked. A burst of errors happens either when by chance several errors are consecutively found or when the Feature Extraction criterion fails on identifying a false structure. In this situation the wrong structure will be tracked and a burst of false alarms will be caused. The mean length of the burst of errors has been measured in our test set. When the tracking system is used this mean length is 5.05 frames and it decreases to 3.67 frames when the tracking is not applied. As it was expected, the bursts of errors are deeper when tracking is applied because of false structure tracking. However, assuming that the system works with 15 frames every second, the errors can be interpolated in many cases.

In Figure 15 several examples of the performance of Lip Finding algorithm can be seen, the right column shows the different horizontal regions taken into account in order to look for the mouth structure and on the left column the Lip Finding result is shown. In Figure 15.a the nose is found instead of the mouth, which is a common kind of error, due to the different geometry of the faces.

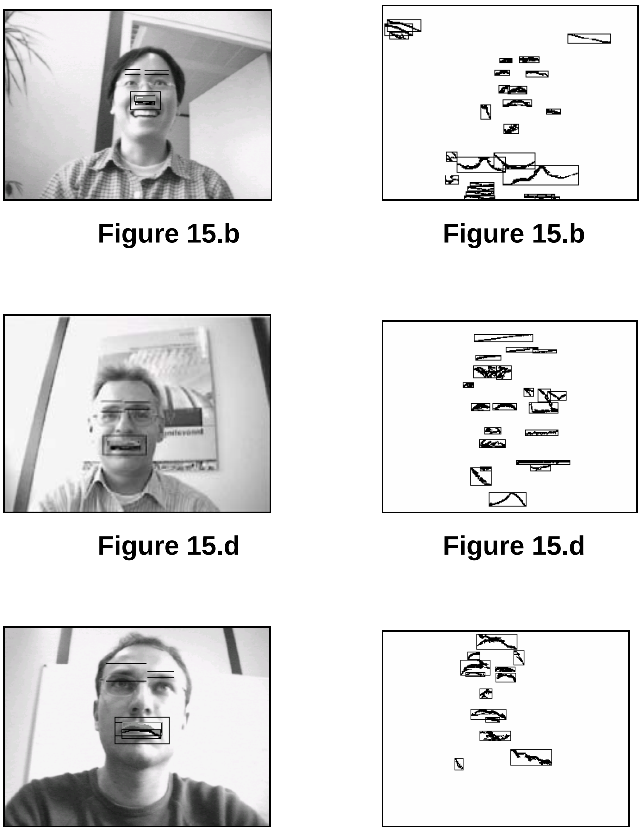

Figure 15.b Figure 15.b

Figure 15.d Figure 15.d

Figure 15.f Figure 15.f

Figure 15: Examples of Performance for Different People, on the Right Column the Different Segments used for Lip Finding are shown.

3.4 System Requirements for Lip Finding and Tracking for Embedded Devices

Since the algorithm is intended for integration in an embedded device, it is important to have low resource consumption. The Lip Finding algorithm saves an important amount of resources by performing the search only between a small amount of regions or “blobs“, as it was indicated in Section 3.1.2.5. Extra savings have been accomplished since the Lip Tracking process can be applied most of the time in comparison to Lip Finding. The former process searches in a small region of approximately 5% of the area of an image.

The fulfillment of the requirements was tested with an emulator for the ARM920T, an exemplary microprocessor suitable for 3G of Mobile Devices (UMTS). ARM920T is a 150 MHz 32-bit RISC CPU processor with 16Kbyte bi-directional cache. The external memory access speed is 150 nsec. for non sequential and 10 nsec. for sequential access. Table 2 lists the algorithm demands tested on ARM920T.

	CPU (MHz)
Lip Finding	**40.0 MHz**
Lip Tracking	**1.8 MHz**

Table 2: Demands of Lip Finding and Tracking.

In the requirements tests a frame rate of 15 frames per second has been used. It must be taken into account that, as long as the lips are being found, only Lip Tracking is applied. Lip Finding is used only when the lips were not found in the previous image and they must be searched within the whole image.

In the sequences used to test the system the percentage of images where Lip Finding was applied was only 5%, so we would obtain a mean CPU use of 4 MHz (2.7% of CPU load). The Code Memory consumption is about 9 Kbytes and 21 Kbytes of linked C standard libraries that may be shared by several software modules and therefore do not increase the memory consumption.

These results were obtained without any kind of platform-specific or assembler optimization of the algorithm. Even in this situation the current software module is compliant with the available resources in an embedded device.

Chapter 4

Feature Extraction

Speech recognition can be understood as a pattern recognition task. Pattern recognition has the objective of mapping a set of measured values $x_1, x_2..., x_L$ to a corresponding pattern (class) ω_k from a given set of patterns Ω [Duda and Hart, 1973], [Höge et al.,2000]:

$$T_x : x_1, x_2, ..., x_L \rightarrow \omega_k \in \Omega \tag{33}$$

The function T_x that defines this mapping is usually not unique; this mapping problem cannot be dealt as a deterministic but as a probabilistic one. Only a certain probability $p(\omega_k \mid x_1, x_2, ..., x_L)$ for given values $x_1, x_2, ..., x_L$ can be assigned to a pattern ω_k. In order to minimize the mapping errors, the Bayesian theory has to be applied. This theory states that the error rate will be minimized when the pattern ω_k with the highest probability $p(\omega_k \mid x_1, x_2, ..., x_L)$ is chosen as the recognized one (principle of maximum a posteriori probability MAP):

$$\omega_k = \arg\max_{\omega_k} p(\omega_k \mid x_1, x_2, ..., x_L); \qquad \omega_k \in \Omega \tag{34}$$

The main problem in pattern recognition is the estimation of the probability function $p(\omega_k \mid x_1, x_2, ..., x_L)$, the dimension L is normally prohibitive. In order to decrease the dimension L of the measurement space, a mapping function F also called feature extraction function is defined:

$$\begin{aligned} F : X_L &\to O_M & ; L >> M \\ X_L &\equiv (x_1, x_2, ..., x_L)^T & ; O_M \equiv (o_1, o_2, ..., o_M)^T \end{aligned} \tag{35}$$

This function maps the measured values $x_1, x_2, ..., x_L$ to so called observations $o_1, o_2, ..., o_L$ where the dimension M of the observations space should be much smaller than the dimension L of the measurement space. The procedure providing this mapping is called feature extraction and the vector O_M is called the feature vector. A ‘good’ feature extraction should lead to a low dimension M and should not increase the probability of error considerably. This is possible if the vector O_M contains the same discriminative information as the measurement vector X_L to separate the patterns ω_k. The vector of observations O_M cannot contain more information than the measurement vector X_L. In the best case, the quantity of discriminative information will be similar. The function F transforms the information in order to make it easily to be classified, generating discriminative features and removing redundant information.

Given an optimal feature vector still the probability $p(\omega_k \mid O_M)$ must be evaluated for constructing a MAP classifier:

$$\omega_k = \arg\max_{\omega_k} p(\omega_k \mid O_M) \tag{36}$$

In the case of many pattern recognition problems, and especially for speech recognition, the dimension of the feature vector O_M is still quite high. This implies that only crude approximations $q(\omega_k \mid O_M)$ of $p(\omega_k \mid O_M)$ can be managed. These approximations are based on easy to handle probability functions, as for example Gaussian distributions. In order to make this approximation close to the reality the observation vector obtained by the function F should have a distribution $p(\omega_k \mid O_M)$ which comes close to the approximated distribution $q(\omega_k \mid O_M)$.

The speech signal will be in many cases corrupted by noise. An optimal feature extraction function should be able to obtain observation vectors which are as less as possible affected by the audio noise.

The main objectives of a feature extraction for speech recognition are:

- Extract relevant information and reject unimportant information for the classification. This will be obtained by transforming the feature space into a new one where each feature has discriminative information.
- Dimension reduction
- The probability function of the features should be able to be approximated by Gaussians.
- The features should be robust against noises

In our audio-visual speech recognition system two different kinds of information are provided: a set of audio and a set of video measurement values. In the audio channel samples of 16 bits are provided with a sample frequency of 8 KHz. This implies an information rate of 125 KBPS. On the video channel, assuming a 320x240 pixel images with 8 bits for each colour component and with a frame rate of 15 frames per second, 27000 KBPS of visual information are generated. The dimensionality of the observation vectors obtained with our audio and visual features extraction will be 12.5 KBPS for the audio channel and 5.2 KBPS for the video channel. As it can be seen in Table 3 a reduction factor of 10 will be obtained for the audio features and a reduction factor of approx. 5000 will be achieved for the visual channel allowing the system to work with a quantity of visual information that is able to be modeled.

	L ***Measurement dimensionality***	***M*** ***Observation dimensionality***
Audio	*125.0 KBPS*	*12.5 KBPS*
Video	*27000.0 KBPS*	*5.2 KBPS*

Table 3: Measurement and Observation dimensionality for Lip Reading in bits per second.

In sections 4.1 and 4.2 the feature extraction functions for audio F_A and video F_V , see equation (35), are profusely studied. The audio feature extraction used in this work is same as the feature extraction implemented in the VSR Very Smart Recognizer ® [Höge et al.,2000], the embedded Siemens Speech Recognition System. The audio feature extraction is not beyond the scope of this work but as it is important for the understanding of the complete system, a brief description will be provided in the paragraph 4.1. VSR offers two different types of Noise Reduction algorithms, these are the wide known spectral subtraction and Wiener filtering. We show them in detail because Lip Reading can be considered as another technique to improve the results of the recognition in noisy environments. In chapter 6 the results of the comparison and combination of Lip Reading with conventional Noise Reduction techniques will be presented.

4.1 Audio Feature Extraction

In this chapter the description of the function F_A is provided [Höge et al.,2000]. The objective is to extract the relevant information for the recognition (observations) from the audio samples (measurements). In Figure 16 the complete chain of algorithms used for the audio feature extraction is shown. Traditional phonetic research claims that all relevant information needed to distinguish phonemes is contained in short time power spectra [Klatt, 1982]. More recent investigations [Peters et al., 1999] claim that under noisy conditions also the phase information of the speech signal is relevant. Nowadays, most recognition systems use short time power spectra. These spectra characteristics are computed via FFT from windowed speech segments, this constitutes the first block of our processing chain in Figure 16. After that, Noise Reduction techniques will be applied in order to make the audio features robust against noises. A set of relevant features for the recognition will be extracted by using a perceptual processing similar to the hearing system in humans. The spectra are smoothed sampled on a mel scale and compressed on a log scale. After that, the distortions of audio channel are eliminated by a Channel Compensation. Finally, and as it was shown in [Yang et al., 1999], the information important to recognize one phoneme is scattered $\pm 100\,ms$ around the center of a phoneme. This context information will be taken into account and the dimension of the final feature vector will be reduced providing discriminative features by the Linear Discrimination Analysis (LDA).

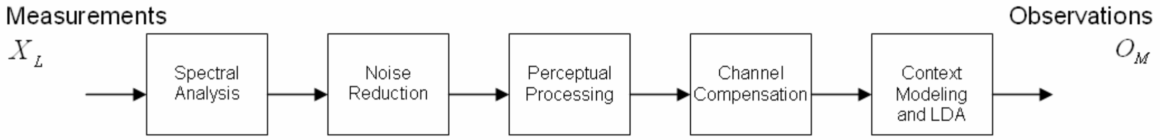

Figure 16: Architecture of the audio feature extraction

In Figure 16 the complete structure of the audio feature extraction algorithm is shown. As it can be seen, it is composed by several sequential sub modules which will transform the measurement values X_L in the observation vector O_M also called feature vector. In the next paragraphs we are going to explain each of these modules.

4.1.1 Spectral Analysis

The speech signal delivered from the Analog/Digital converter has a sample rate of 8 KHz, each sample has been linear coded with 16 bits. In a first step, the speech samples x_n are pre-emphasized in order to lift the higher frequencies. The pre-emphasized filter is implemented with a first order FIR filter.

$$u_n = x_n - p x_{n-1} \tag{37}$$

The emphasized samples are windowed using a Hamming window. Every window will have a duration of *32 ms* (256 samples, assuming 8 KHz sampling rate) and every window is shifted *15 ms*. As the frame length is longer than the frame shift, consecutive frames contain common information, this is called overlapping windows.

For the windowed sequence an 8 order FFT will be calculated, which implies that a block of 256 values has to be processed. The FFT input values between the Hamming length and the FFT block length will be filled up with Zeros (zero padding). Because we are dealing with FFT transforms of real functions the short term power spectrum is symmetrical. For this reason only 128 values from the FFT must be calculated for every frame. In our implementation the phase information will not be used.

$$X_n = \left(X_n(f_0),\ldots,X_n(f_{127})\right) = \left(|\, x_n(f_0)\,|^2,\ldots,|\, x_n(f_{127})\,|^2\right) \tag{38}$$

4.1.2 Noise Reduction

There are many sources of distortion that corrupt the speech signal and degrade the recognition rate of the system. One of the most important distortions is the background noise. Different types of noises can be distinguished: car noise, public noise (street, airports, train stations…), office noise, interfering talker, industrial noise, etc. The most popular algorithms to reduce the noise from the audio channel are spectral subtraction and Wiener filtering (RLS), both of them are implemented in our recognizer. The objective of Lip Reading is to provide a more noise robust system; this is the reason why it will be interesting to compare Lip Reading with the conventional approaches to deal with noisy environments. This comparison will be given in chapter 6. Now, a theoretical background of conventional Noise Reduction techniques is going to be provided.

4.1.2.1 Spectral subtraction

The spectral subtraction algorithm and the noise estimation used in our implementation is based on [Martin, 1994]. The distortions are assumed to be additive noise η superposed to the undisturbed speech signal s:

$$x_n = s_n + \eta_n \tag{39}$$

where the noise and the signal are statistically independent, $E[x\eta] = 0$, the operator $E[\ ]$ denotes the expectation.
Assuming further that η is a null mean signal:

$$E[x^2] = E[s^2] + E[\eta^2] \tag{40}$$

and in the spectral domain it can be written as:

$$|x^n(f)|^2 = |s^n(f)|^2 + |\eta^n(f)|^2 \tag{41}$$

From now on this formula will be written using a simplified notation in this way:

$$X_n(f) = S_n(f) + N_n(f) \tag{42}$$

where $X_n(f), S_n(f), N_n(f)$ denote the short term power spectra of x, s, η for the hamming window centered in time n.

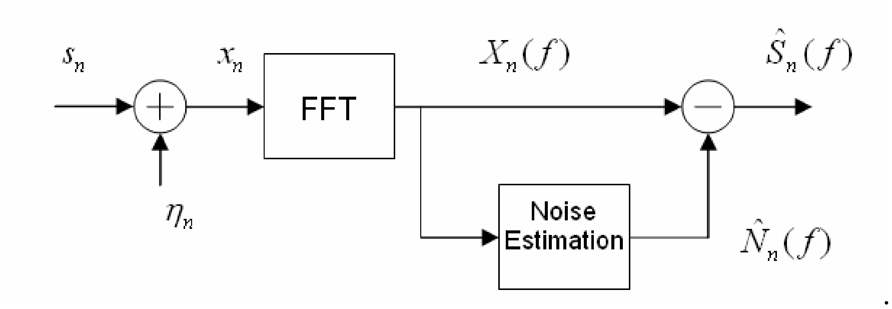

Figure 17: Spectral Subtraction Noise Reduction

Given an estimate $\hat{N}_n(f)$ for $N_n(f)$, less disturbed speech spectra $\hat{S}_n(f)$ can be derived by the method of spectral subtraction:

$$\hat{S}_n(f) = X_n(f) - \hat{N}_n(f) = S_n(f) + N_n(f) - \hat{N}_n(f) \approx S_n(f) \tag{43}$$

For this solution only an estimate for the additive noise $\hat{N}_n(f)$ must be found. The power spectrum $N_n(f)$ is estimated by minimum of the smoothed power spectrum of the corrupted signal within a moving interval with fixed width D. This algorithm is based on the observation that for each frequency band the smallest value of the $\overline{X_n(f)}$ that is observed in a sufficiently large number of consecutive frames corresponds only to the noise.

$$\overline{X_n(f)} = \alpha \cdot \overline{X_{n-1}(f)} + (1-\alpha) \cdot X_n(f) \tag{44}$$

$$\hat{N}_n(f) = \beta \cdot \min_{n \in I_n} \overline{X_n(f)}; \quad I_n = [n-D, n] \tag{45}$$

4.1.2.2 Wiener Filtering (RLS)

This is a Noise Reduction method that uses the Recursive Least Square (RLS) in the frequency domain [Scalart and Filho, 1996], [Beaugeant et al., 1998], [Wiener, 1949].

In this approach it is assumed that an estimation of the speech signal without noise $\hat{S}_n(f)$ can be obtained by filtering the signal (multiplication in frequency or convolution in time domain) with an adaptive RLS filter as it can be seen in Figure 18 and in the formula:

$$\hat{S}_n(f) = X_n(f) \cdot G_n^{RLS}(f) \tag{46}$$

A Least Square (LS) error function will be defined as:

$$J_{LS}[e_n(f)] = \sum_{l=0}^{n} \beta_{l,m} \, | e_l(f) |^2 \tag{47}$$

where the error is defined as:

$$e_n(f) = \hat{S}_n(f) - S_n(f) \tag{48}$$

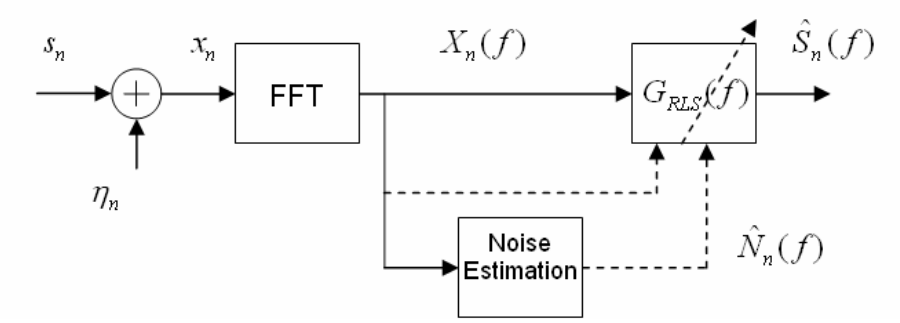

Figure 18: RLS Noise Reduction

The solution of the equation $\partial J_{LS} / \partial G_n^{RLS} = 0$ provides for each frame n and frequency f the filter $G_n^{RLS}(f)$ minimizing the error function J_{LS}. This leads to the most general formula of weighting rule derived from a recursive implementation of the RLS criterion in frequency domain:

$$G_n^{RLS}(f) = \frac{\sum_{l=0}^{n} \beta_{l,n} \, | S_l(f) |^2}{\sum_{l=0}^{n} \beta_{l,n} \, | S_l(f) |^2 + \sum_{l=0}^{n} \beta_{l,n} \, | N_l(f) |^2} \tag{49}$$

The value of $G_n^{RLS}(f)$ obtained in (49) is transformed in such a way that the term $| S_l(f) |^2$ is "replaced" by the noisy signal $| N_l(f) |^2$. Accordingly the following expressions will be obtained for our RLS implementation:

$$G_n^{RLS}(f) = \frac{E_n^X(f)}{E_n^X(f) + O \cdot E_n^{\hat{N}}(f)} \tag{50}$$

$$E_n^X(f) = \beta_{SignalEnergy} \cdot E_{n-1}^X(f) + X_n(f) \tag{51}$$

$$E_n^{\hat{N}}(f) = \beta_{Noise} \cdot E_{n-1}^{\hat{N}}(f) + \hat{N}_n(f) \tag{52}$$

In these recursive equations the noise can be estimated by using a similar approach as we have presented in the spectral subtraction. There will be still three parameters defining the values of the forgetting factors for the corrupted signal $\beta_{SignalEnergy}$ and for the noise β_{Noise}, as well as a Noise Overestimation factor O. These parameters will be fixed for every scenario (car kit, hands free, office). When a Voice Activity Detector is available, the equation (49) can be directly used to estimate the actual $G_n^{RLS}(f)$.

4.1.3 Perceptual Preprocessing

In the feature extraction theory for speech recognition two main approaches can be found in the literature [Schmidbauer and Höge, 1991]. A first one, Linear Prediction Coefficients (LPC) based on the production of the speech were through a statistical analysis the main formants of the speech describing the vocal tract are obtained. The second method, cepstral analysis takes as basis the auditory processing and perception theory. The latest has been found to achieve better recognition performances.

In the so called cepstral analysis, first of all the power spectrum is processed by a mel scale filter bank, where the spectral energies are smoothed in frequency domain by triangular functions, as can be seen in formula (53), providing a total of $J_{Cep} = 15$ different frequency outputs one for each triangular filter. This band filter analysis is similar to the one performed inside of the ear:

$$\underset{i=1 \ldots J_{Cep}}{w_n(f_i)} = \sum_{j=-M}^{M} X(f_i + j \cdot \Delta f) \cdot a_{ij} \tag{53}$$

where a_{ij} are the coefficients of a sampled triangular function.

In a second step the smoothed power spectra are transformed into a mel-cepstral domain, which is basically a logarithmical compression and a DCT transformation, as it can be seen:

$$u_{n,k} = \sum_{j=1}^{J_{Cep}} \log_{10} w_n(f_i) \cdot \cos \frac{\pi \cdot k \cdot (2 \cdot j + 1)}{2 \cdot J_{Cep}} \qquad k = 1 ... K_{Cep} \tag{54}$$

In addition to these 12 cepstral values, an energy value of the total frame is going to be evaluated. This will be a value of the difference in logarithmic scale between the total energy in the current frame and the smoothed total energy value from all the previous frames in the utterance. In this way, the frame energy will be normalized to the mean energy value and will not be dependent on the total energy of the signal (that can be dependent on the speakers voice volume, distance to microphone, microphone type, etc.). At the output of the perceptual processing block 13 feature coefficients for every frame will be provided.

4.1.4 Channel Compensation

The spectrum of the speech can be changed due to different characteristics of the transmission channel, as for example the acoustic of the place where the speaker is located or the resonance of the device where the microphone is embedded. All these transmission distortions, when they are linear, can be modeled by a transfer function, which transforms the spectrum of the original signal $S_n(f)$ to a modify spectrum $\hat{S}_n(f)$:

$$\hat{S}_n(f) = S_n(f) \cdot H_n(f) \tag{55}$$

The transfer function can change with the time. When distortions cannot be modeled as linear, they must be assumed as additional noise and they will be tried to be eliminated by Noise Reduction systems. Now we are dealing only with linear distortions of the signal. As we have explained in the previous paragraph, a logarithmical scaling of the frequency components is performed. So that the equation (55) will be converted in:

$$\log_{10}(\hat{S}_n(f)) = \log_{10}(S_n(f)) + \log_{10}(H_n(f)) \tag{56}$$

In this model the cepstral features have an offset depending on the channel. This offset can be estimated using a Maximum Likelihood Estimator and it will be subtracted from the cepstral features. This offset can change along the time, so that it will be updated.

4.1.5 Context Modeling and Linear Discrimination Analysis

As it was shown in [Yang et al., 1999], not only the short term power spectrum is important for the recognition. The variation over time of this characteristic will also play an important role for the recognition. Furthermore, the information scattered over *100 ms* around the centre of the phoneme is significant for its characterization. Although this contextual information is included in the acoustic modeling using Hidden Markov Models (HMM), better results can be obtained when this information is additionally used in the Feature Extraction. For this reason the feature vector obtained from the previous pre-processing steps, which was firstly composed of 13 coefficients, is extended by adding the first and second derivatives of all its components. In this way, a new vector with 39 features is obtained for every frame. In a second step the current vector of 39 features is concatenated with the one of the previous frame generating a so called super vector, which contains inter frame information. In our case, the super-vector includes information of two frames, the dimension of our vector will be 78 components every frame. This dimension is reduced by a Linear Discrimination Analysis (LDA). Finally, only the first 24 most discriminative output coefficients of the LDA are considered as final components of the observation vectors, which will be directly used in the recognition process. The evolution of the feature dimension is summarized in Figure 19.

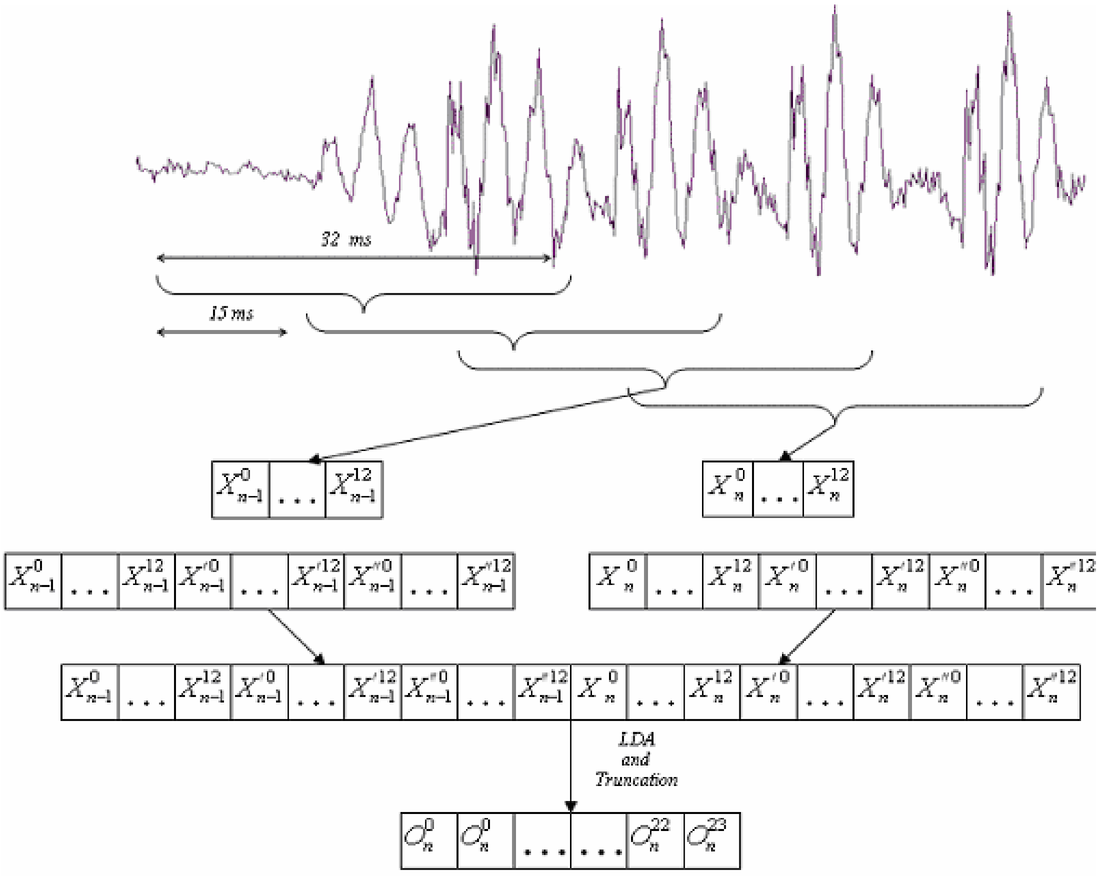
32 ms
15 ms
X^0_{n-1} . . . X^{12}_{n-1}
X^0_n . . . X^{12}_n
X^0_{n-1} . . . X^{12}_{n-1} X'^0_{n-1} . . . X'^{12}_{n-1} X''^0_{n-1} . . . X''^{12}_{n-1}
X^0_n . . . X^{12}_n X'^0_n . . . X'^{12}_n X''^0_n . . . X''^{12}_n
LDA
and
Truncation
O^0_n O^0_n . . . . . . O^{22}_n O^{23}_n

Figure 19: Context dependent information and LDA dimension reduction

Linear Discriminative Analysis is a dimension reduction process that will be used not only in the audio feature extraction, but also in the visual feature extraction. This is the reason why we study it a bit more profusely.

LDA will reduce the dimension of $\underline{X}^n$ without loosing the discriminative powwer of the features. LDA does not increase the discriminate capability of the features but the it helps that easy to implement classifiers with lower feature dimensionality achieve high recognition rates.

The LDA is a linear transformation, which transforms an input vector $\underline{X}$ to an output feature vector $\underline{O}$.

$$\underline{O} = \underline{\underline{A}} \cdot (\underline{X} - \underline{\hat{X}}) \tag{57}$$

Where $\underline{\hat{X}}$ is the mean vector of the input feature vectors and $\underline{\underline{A}}$ the so called LDA matrix.

The objectives of the LDA transformation are [Bauer, 2001]:

- To obtain output features which are not correlated to each other “De-correlated features”.
- The variance of the features that belong to the same phoneme (segment) will be normalized. All the distributions will have a similar variance.
- Maximize the discriminative capability when only the first components of the output vector are considered.

To achieve these objectives the so called Scatter-Matrix are used and the matrix that optimize the previous criteria is found [Hojas, 1994].

In the speech recognition for embedded devices the previous advantages introduced by the LDA have an important relevance [Bauer, 2001]. First of all, the fact that the features will be at the output decorated reduces the error performed when these features are described by only diagonal covariance matrixes. The input components of the vector $\underline{X}$ are correlated because the mel filters are overlapped. Feature vectors are described in the recognition process by Gaussians with diagonal covariance matrixes in order to reduce the number of parameters. This assumption can only be made when features are not correlated, fact that is achieve with the LDA. Furthermore, in our embedded implementation only one variance is used for the description of the emission probabilities, this assumption will be right only in the case that all these variances are similar, which is also achieved with the LDA. Finally, the LDA will order the components of the output vector according to their discriminative, in such a way that a truncation is possible. Only the first 24 first coefficients are going to be taken, a reduction of 54 coefficients per frame is obtained without decreasing the discriminative capability of the system.

4.2 Visual Feature Extraction

In this section a description of the function F_V , that transforms the visual measurement (image received from the camera) into the so called observation vector (visual features), is provided. As it was explained in the introduction, the visual channel receives 27000 KBPS of information. All this information cannot be directly used in the recognition. For the visual channel and due to its high dimensionality, the dimension reduction is even more important than for the audio channel. The feature extraction approach that will be showed in this paragraph achieves a reduction factor of 5000 between the dimension of the measurement and the observation vector. As comparison value, the reduction factor in audio was 10. Once the mouth region is found, an appropriate set of lips features must be extracted. These features are used directly in the recognition. Therefore, they must content viseme discriminative information. The selected features should maximize the recognition rate and the extraction techniques must be suitable for an embedded implementation. Lip finding and tracking can be considered as a part of the visual feature extraction, in this thesis it was analyzed independently in chapter 3.

4.2.1 State of the Art of Visual Feature Extraction

Several approaches of visual feature extraction for Lip Reading can be found in the literature, they can be classified according to the type of information source they process: shape-based and appearance-based [Matthews et al., 2002].

In the shape-based approaches, features describe contours of figures, in such a way that a geometrical description of them is provided; these techniques are also called contour-based approaches. The appearance-based techniques use appearance information (texture, grey level information, colours) these approaches are also called pixel-based approaches.

The Lip Finding and Tracking algorithm is essential in the pixel-based approaches because it allows a reduction of the image used for the recognition, because in this approach all pixels of the ROI are used for the recognition. A reduction factor of approx. 30 is obtained just by analyzing only the region of the mouth and not the whole image. In the case of the shape-based approaches, many of these algorithms convey itself a tracking of the geometrical form. Our Lip Finding and Tracking algorithm makes the geometrical based approach faster and more efficient, as it provides an initial location where the model should be placed.

Finally, in the shape-based approaches, there is more apriori knowledge. It is known that a shape similar like a lip must be found. This approach assumes that the shapes of the lips convey the most important information for the visual recognition. In the appearance-based approaches, all visual information inside the region given as mouth from the Lip Finding and Tracking is considered as important. In appearance-based approaches it will be a task for the recognition system to use all these features properly and take advantage of the important features. It can be said that in the shape-based approaches more specific features will be extracted than in appearance-based, where the recognition system must be able to deal with a higher dimension vector of non-specific features.

One implementation of each approach is performed and a comparison in terms of recognition results is provided in order to find the most convenient feature extraction for the system. For each approach, different solutions can be found in the literature. In the next paragraph a summary of the main solutions in each group (shape and appearance-based algorithms) is summarized in order to find out the best one to be implemented.

Shape-Based Feature Extraction

In this approach, a model of the visible speech articulators, mainly lip contours, is built and its configuration is described by a small set of parameters. A very easy shape-based model was used in one of the first Lip Reading System [Petajan, 1985]. They used main features of the mouth like height, width of the mouth cavity, together with area and perimeter. This simple model was very sensitive to shades and bad illumination. A modeling of the mouth contour by using Fourier Affine Invariant Descriptors was provided in [Gurbuz et al., 2001], where pose parameter independent were obtained. Other approaches to describe the contour of the mouth are e.g. B-Splines [Dalton et al., 1996] were a set of control points adapts a curve to the contour of the lips. A different solution was given by another parametric curves like Snakes or Deformable Templates [Chiou and Hwang, 1997] which can elastically be deformed in order to close the real lip contour minimizing an error energy. In this approach there is not compression of the information and it cannot be very robust to shades because it minimizes the error according to the information in the current image; shades can easily degrade the result. There is no apriori knowledge about how a mouth can be deformed, only the initialization of the template. This apriori knowledge of the mouth deformations is used in the so called Active Shape Models (ASM) [Luettin et al., 1997]. In this approach, a set of training images are used to extract the mean mouth and to obtain the main modes of variation. Once the contours of the mouth are extracted, the mode combination that minimizes the error is provided. In this way, the influence of shade contours not belonging to the mouth is minimized as this deformation was not seen in the training set. Furthermore, the adaptation of the contours to the model will be performed in a normalized space were the rigid transformation parameters will not be included. In this way the system will be scale, rotation and translation independent. The set of parameters that will be given as observation vectors are the coefficients that multiply the different modes in which the mean mouth can be deformed. ASM is selected as shape-based feature extraction method and it will be implemented. A further description of this method and its implementation is given in the next paragraph.

Appearance-Based Feature Extraction

In this approach, the entire bitmap of the mouth area is considered as interesting for Lip Reading. Appropriate transformations of its pixels values are used as visual features. This approach follows the philosophy of simplifying the feature extraction of the image relying on the later processing of the recognition engine (HMM) to develop appropriate internal representations of these low level features.

A very popular approach consists in using linear transformations of the mouth region. The most common transformations are the Discrete Cosine Transformation (DCT) [Nefian et al., 2002], discrete wavelet [Potamianos et al., 1998] or the Karhunen-Loève transformation (KLT) [Bregler and Konig, 1994]. These approaches achieve energy compression in a small set of coefficients but there is no a compression based on more discriminative features for the recognition. This is the reason why they are usually used in combination with the Linear Discriminative Analysis (LDA) [Bauer, 2001] which generates a new set of coefficients which are ordered depending on their discriminative power. Other pixel-based approaches can be found in the literature. One of them is the *eigensequences* [Dupont et al., 2000], [Bregler and Konig, 1994] where pixels of the ROI are considered as vectors, their eigenvectors and *eigenvalues* are obtained. The highest *eigenvalues* are used to characterize the image and they are used in the recognition process.

Another solution is the Multiscale Spatial Analysis [Matthews et al., 2002], [Matthews, 1998] a non-linear image decomposition method. It is also called sieve filter, it transforms the image into a granularity domain. This describes the image in terms of granules that have position, amplitude and scale attributes. A visual feature vector is formed using only the scale information in an attempt to define a feature set that is intensity and position invariant. The advantage of all these pixel-based approaches is that they do not need many resources to find the suitable characteristics. This process is easier than the shape-based, as the system relies on the next subsequent block to extract the most important characteristics.

Comparison of feature extraction algorithms

We have summarized the shape and appearance-based algorithms in

Table 4, showing their advantages (+) and disadvantages (-). We have introduced a new criterion to make the classification: the quantity of apriori knowledge invested on the extraction process. Features generated with the lowest quantity of apriori knowledge are placed on the top of the

Table 4, whereas the ones with the highest amount of previous knowledge can be found at the bottom. For this work, one technique of each group was chosen. As shape-based approach ASM [Luettin et al., 1997] was selected to be implemented, because it uses a high amount of apriori knowledge which allows an improvement of the robustness of the system in real conditions (bad illumination). Moreover, this solution provides pose invariant features. As appearance-based approach Discrete Cosine Transformation (DCT) is proposed. It is the most popular Lip Reading features extraction technique [Potamianos et al., 1998], providing good results. Furthermore, due to the emerging of video applications in mobile devices, efficient DCT algorithms are available; these algorithms are already implemented for video coding purposes.

Shape-Based Feature Extraction	Appearance-Based Feature Extraction
Height, Width, Perimeter, Area [Petajan, 1985] *+ Very simple, it contains important Information* *- Sensitive to shades and bad illuminations*	***Linear transformations (DCT) [Potamianos et al., 1998], [Amarnag et al., 2003]*** *+ Simple and direct, it exists efficient embedded algorithms* *- Image normalization must be done. They are shift sensitive*
Fourier Affine Invariant Descriptors [Gurbuz et al., 2001] *+ Independent of the rigid transformation parameters* *- Low apriori knowledge, sensitive to bad illumination*	***PCA, Eigenlips [Dupont et al., 2000]*** *+ Use apriori knowledge, it is more noise robust* *- A very accurate alignment of the images is required*
B-Splines, Bezier Curves [Dalton et al., 1996] *+ Simple contour parameterization* *- There is not apriori knowledge about the mouth shape*	***Optical Flow [Mase and Pentland, 1991]*** *+ It focuses on the dynamics, the most important information* *- Very dependent on light variations*
Snakes, Deformable Templates [Chiou and Hwang, 1997] *+ Elastic contour with a high level of Freedom* *- No apriori knowledge about deformations*	***Multiscale Spatial Analysis (MSA) [Matthews et al., 2002]*** *+ Robust on bad illuminations* *- Morphological filters are highly resource consuming*
Active Shape Models (ASM) [Luettin et al., 1997] *+ Rigid Trans. Independent, information compression, robust* *- Conventional implementations are highly resource consuming*	***Active Appearance Models (AAM) [Matthews et al., 2002]*** *+ Combines the shape with the appearance model info* *- Training must be representative of all illumination variation*

Table 4: Overview of Shape-Based and Appearance-Based Feature Extraction Algorithms

4.2.2 Active Shape Models

As shape-based approach Active Shape Models (ASM) has been chosen for the implementation. In ASM, apriori knowledge of the plausible mouth deformations is learnt in a training process [Cootes and Taylor, 2000], [Cootes et al., 2001]. A set of points (landmarks) must be consistently located in the mouth contours of the training set. For each image of our training set, a vector with the coordinates of the landmarks will be defined:

$$X = (x_1, \ldots, x_n, y_1, \ldots y_n)^T \tag{58}$$

Rigid transformation dependencies are firstly removed by using Procustes Analysis [Cootes et al., 2001], which aligns each shape, so that the sum of distances of each shape to the mean is minimized:

$$D = (X - \overline{X})^T \cdot (X - \overline{X}) \tag{59}$$

Let's assume that all the landmarks are aligned, after that, Principal Component Analysis (PCA) is applied on these aligned points. PCA computes the main variation modes of the points. This allows the deformations to be described only by a small set of parameters. Let's define the covariance matrix of the aligned landmarks as:

$$S = (X - \overline{X}) \cdot (X - \overline{X})^T \tag{60}$$

It is interesting to show that in (59) the distance that will be minimized is a scalar value, meanwhile in formula (60) S is a covariance matrix. The eigenvectors ϕ_i and the corresponding eigenvalues λ_i of the landmark coordinates covariance matrix S are computed and sorted so that $\lambda_i > \lambda_{i+1}$. If Φ contains the t eigenvectors corresponding to the largest eigenvalues, a set of points describing the mouth contour X can be approximated by:

$$X = \overline{X} + \Phi \cdot b \tag{61}$$

where $\Phi = (\phi_1 \mid \phi_2 \mid \ldots \mid \phi_t,)$ and b is a t-dimensional vector given by:

$$b = \Phi^T \cdot (X - \overline{X}) \tag{62}$$

b coefficients describe the different variations of the mouth with respect to its mean value, as it can be seen in [van Ginneken et al., 2002], where a detailed description of the ASM with optimal features can be found.

According to the mathematical formulation of the ASM, two different processes must be carried out. First of all, an offline training process is performed, the objective of the training is to find the matrix Φ . After that, when the recognition process of an image takes place, a set of visual features will be extracted online from the image. These are the b coefficients obtained by applying the equation (63), they will be the basis for the observation vector.

ASM training is performed in the same way as it is showed in [Cootes and Taylor, 2000]. A set of 140 different images of 33 different speakers have been taken for training, the outside contour of the upper and the lower lip was marked with a set of 15 points. This landmarking was manually performed. As first reference, a set of 7 points describing the main characteristics of the lips are set, red points in Figure 20, then the secondary points are placed, these are the green ones in Figure 20. This is what we call model 1. However, in the matching process and also in training, where this process was manually performed, we have realized that it was quite difficult to find the outside contour of the lower lip. This is the reason why we have decided to obtain a new model (model 2, Figure 20.b) where the outside contour of the upper lip and the inside contour of the lower lip are considered, see Figure 20.

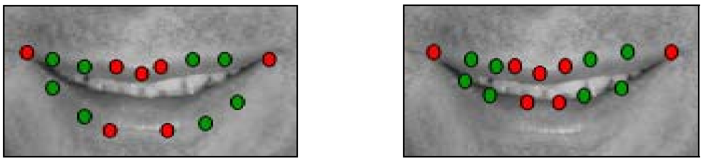

Figure 20.a **Figure 20.b**

Figure 20: ASM lip models. Figure 20.a Model1 and Figure 20.b Model2.

It is important to say that in the training process we were working with 30 points obtained from the 15 manually marked landmarks. We have doubled the number of points by simple linear interpolation. It must be pointed out that the interpolation will only add redundant information in the ASM training, but due to the complex matching process (finding the position of the markers in a new image) the algorithm will work better if more points are considered.

In the training process, first of all, every mouth must be aligned in order to reject the dependences on scaling, translation and rotation. This will be obtained by using the so called “Procustes Analysis” [Cootes and Taylor, 2000] with an iterative approach. An initial value for the normalized mean mouth is assumed and this will be updated in several iterations of the alignment. Let’s call the set of non-aligned coordinates X and the coordinates of the mean normalized mouth X'. We assumed that the set of X points have been centered in *(0,0)*, which is achieved by subtracting the mouth center of gravity to each coordinate. Now we want to scale and rotate our shape X by (s,θ) so as to minimize $|(s \cdot A \cdot X) - X'|$, where A performs a rotation of a shape X by θ. It is a linear transformation using the similarity case.

$$T\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \alpha & -\beta \\ \beta & \alpha \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix} \tag{63}$$

where $s^2 = \alpha^2 + \beta^2$ and $\theta = \tan^{-1}(\beta / \alpha)$.

In this way a Rigid Transformation has been carried out, see Figure 21. The translation, rotation and scaling are taken out of the geometric representation in such a way that a set of “normalized mouths” has been provided.

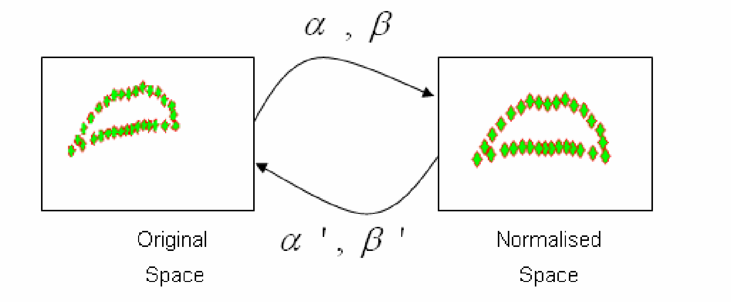

Figure 21: **ASM rigid transformation coefficients.**

The values of α and β are obtained from the coordinates values X' and X using the followings equations:

$$\alpha = (X \cdot X') / |X|^2 \tag{64}$$

$$\beta = \frac{\sum_{i=0}^{n}\left(x_i \cdot y_i^{'} - y_i \cdot x_i^{'}\right)}{|X|^2} \qquad (65)$$

Once all training shapes are aligned with an original mean mouth, a new mean mouth is obtained from all aligned images and the process is repeated until the mean aligned mouth remains quasi constant between one iteration and the next one. As result, a mean aligned mouth will be generated. It's important to point out that this iterative process takes place only in the offline training and it is not necessary in the online alignment.

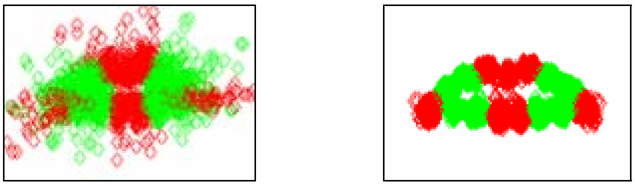

Figure 22.a **Figure 22.b**

Figure 22: ASM lip models alignment: Figure 22.a training set before the alignment Figure 22.b training set after the alignment.

In Figure 22 all training mouths before and after the alignment are shown. Once all the mouths are aligned PCA (Principal Component Analysis) is applied to the coordinates and the different modes that describe the different forms of the lips will be obtained. First of all, the covariance matrix S is evaluated and then the eigenvectors and eigenvalues of this matrix are obtained. In Figure 23, the reconstructed mouths with equation (61) are showed. Φ contains the eigenvectors from the highest eigenvalues of matrix S. In each column of Figure 23, the result of applying equation (61), setting all components of b to zero but one with a different value is shown. In this way, the influence of each mode can be seen. The first mode conveys information about the opening of the mouth, the second one about the shape of the lower lip; the third carries more information about the upper lip. The quantity of information for modes higher than the 5th is very small and in many cases it conveys more noise than information. This is the reason why we have used only the first five modes to perform the recognition. A particular mouth will be obtained as a linear combination of the first five modes.

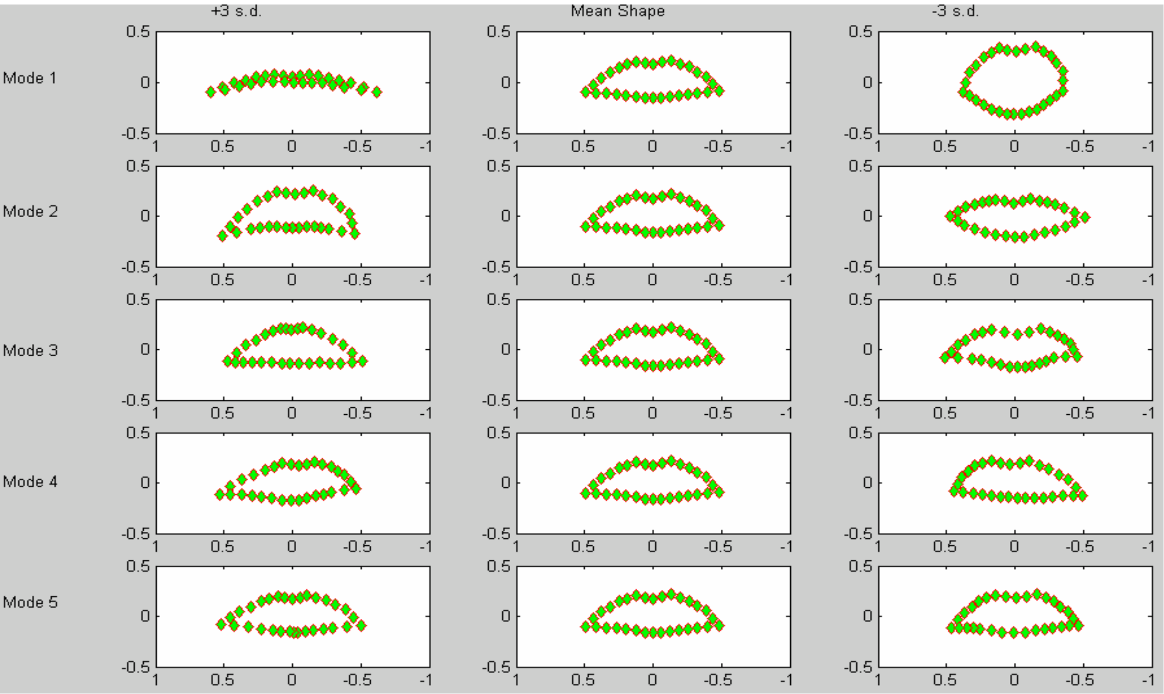

Figure 23: First Five ASM Lip Modes; in the Middle Column the Mean Mouth is Presented and the right and left Columns represent the most significant Shape Variation Modes for ± 3 Standard Deviation respectively.

In an offline training process, the mean value of the mouth and the matrix Φ were obtained, now the process how to analyze an image in order to extract the visual characteristics is explained for an embedded implementation. Matching Process is the way the positions of the contour points are found when an image is provided. First of all, a preprocessing is going to be applied on the ROI containing the mouth. A directional filter (horizontal) is applied to the image, illumination changes from bright to dark for the upper lip, and from dark to bright for the lower lip are used to find the lip contours, as it can be seen in Figure 25.a. Histogram analysis will be applied to obtain an adaptive threshold; a percent of the pixels will be below this threshold. The largest regions (for the upper and lower lip) are assumed to be the upper and lower lip boundaries. Two pixels belong to same region when their value is larger than the threshold and when they have a neighborhood relationship. Over this image the mean mouth is placed using the information provided by Lip Finding and Tracking algorithm. The new points are searched in the perpendicular of the mean mouth, see Figure 25.b, and in these directions the end of the upper and lower lip regions are found. In this way, the points in Figure 25.c are located. The found points are translated into the normalized space by making an alignment, using equations (63)-(65). In the normalized space, PCA is applied and the b coefficients are extracted using equation (62), these will be the visual features for the recognition. The inverse of the rigid transformation parameters α and β are used to place the normalized, synthesized mouth. This mouth is taken as initialization mouth for the next image.

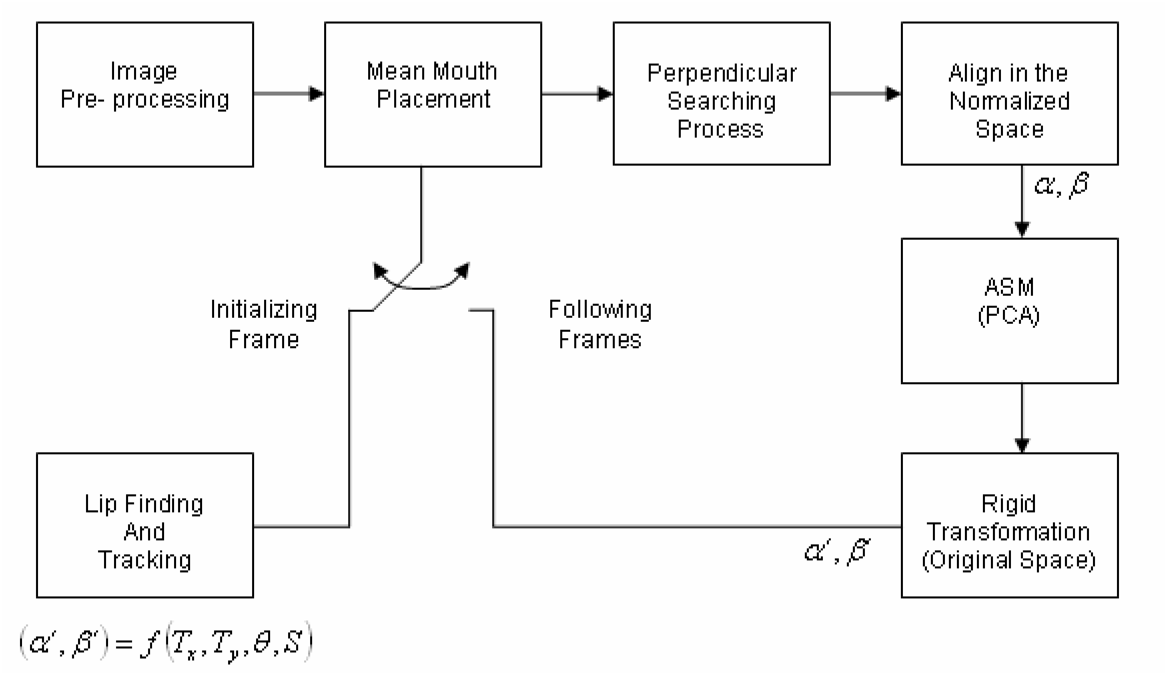
Image Pre- processing
Mean Mouth Placement
Perpendicular Searching Process
Align in the Normalized Space
α, β
ASM (PCA)
Initializing Frame
Following Frames
Lip Finding And Tracking
Rigid Transformation (Original Space)
α', β'
$(\alpha', \beta') = f(T_x, T_y, \theta, S)$

Figure 24: Matching Process, general schema.

In the general schema in Figure 24, it can be seen that for the first frame the rigid transformation information is obtained from the Lip Finding and Tracking algorithm [Guitarte et al., 2003]. This algorithm provides an approximation of the translation, scaling and orientation of the mouth. These parameters are used to obtain the transformation coefficients α and β, see Figure 24. The inverse values of these parameters are used to place the normalized mean mouth on the original pre-processed image.

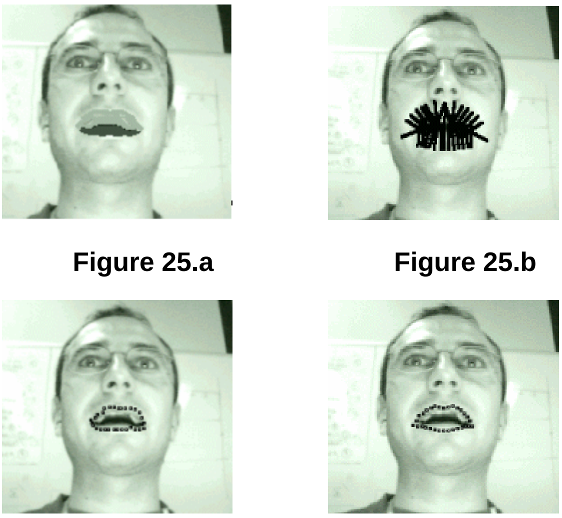

Figure 25.a

Figure 25.b

Figure 25.c

Figure 25.d

Figure 25: ASM Implementation Stages. a) Upper and Lower Lip Region Detection, b) Search Lines, c) , Found Points before PCA, and d) Final Points after PCA

In Figure 26, the temporal evolution of the first five coefficients, obtained with our ASM implementation, is shown. It can be seen how the first coefficient has the largest energy and it represents an important part of the total variability. Figure 23 shows that this coefficient has information of a very important characteristic to discriminate between different visemes: mouth opening.

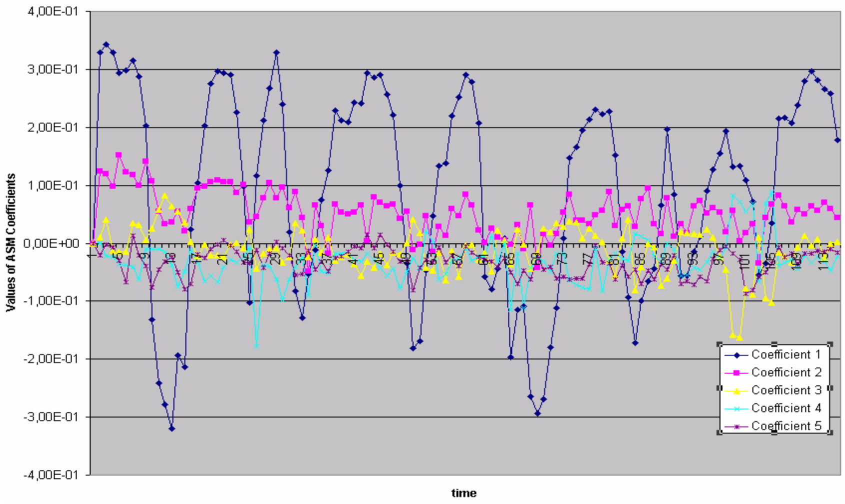

Figure 26: **First 5 Coefficients of our ASM Implementation**

In Figure 27, the first coefficient is represented for the sentence of German digits: “eins zwei drei eins zwei drei eins zwei drei”, we have taken this example because the three digits are phonetically similar as they have the same diphthong /ai/ but they are visually quite different. There is an own characteristic for the utterance “zwei” which allows its recognition just only by inspecting the temporal evolution of this coefficient. The visual characteristic of a word can be distinguished from the others.

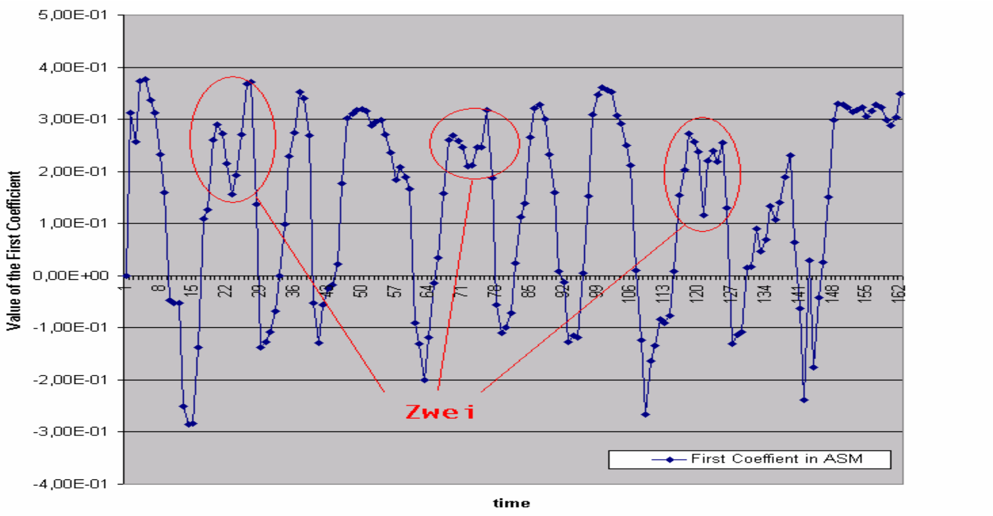

Figure 27: Temporal evolution of the first Coefficient of our ASM Implementation when the Sequence “eins zwei drei eins zwei drei eins zwei drei” is said.

4.2.3 Discrete Cosine Transformation

Appearance-based methods provide visual features by using a transformation of the grey scale intensity lip image. These features contain information about the lip structure but also about the teeth and tongue visibility. This information was not derived from the shape-based systems. As appearance-based methods, the Discrete Cosine Transformation (DCT) is selected, because it has been successfully used in different Lip Reading systems. Moreover, it has been used for image codification and there are efficient implementations of this algorithm suitable for embedded devices [Lee and Huang, 1994]. The bidimensional DCT transformation compresses the image information in a set of coefficients. This transformation is used in many video codification systems, as it is very suitable to compress in a small set of coefficients the most important visual information.

Before DCT is applied, a preprocessing must be performed. Different structures unimportant for the recognition can appear in the original image. Moreover, the dimension of the information would be very high. First of all, Lip Finding and Tracking followed by the stabilization process explained in chapter 3 is applied. Before the DCT is computed, a normalization of the mouth region and a bidimensional windowing must be performed.

Image Normalization and Windowing

Image normalization must be applied to avoid problems of scaling and rotation. The normalization transforms the mouth region into a normalized space where both mouth corners should be aligned with the horizontal line (rotation = 0). The length of the distance between the two mouth corners will also be normalized. In order to find this image, the scaling and rotation (s, θ) are going to be used to define the transformation matrix that match the position of each point from the original image in the normalized image.

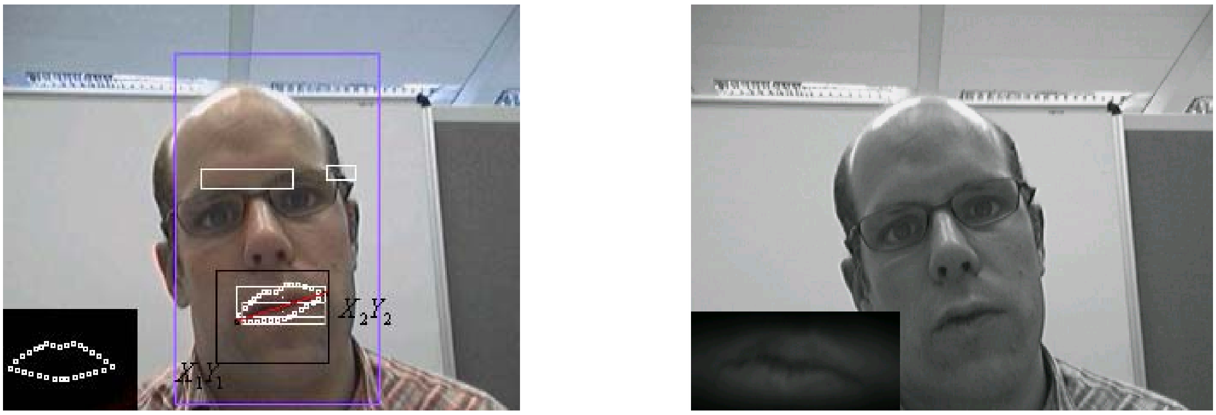

Figure 28.a

Figure 28.b

Figure 28: Normalized Mouth Image, Rotation, Scaling, Translation and Elliptical Windowing

Each pixel position in our normalized image is fulfilled with a pixel from the original image. When one point of the normalized image is associated with a non integer point from the original image, the distance to each pixel will be used as weighting factor and a mean weighted value will be provided.

The scaling and rotation parameters can be obtained from the corners of the mouth, the distance between then will define the scaling factor and the angle between the line that joins both corners and the horizontal will provide the rotation.

The position of the corners can be found by applying the ASM [Guitarte and Lukas, 2004], this implementation will provide the coordinates of the two corners. Another easier solution has been implemented using the values of the lips regions, see Figure 10.

After the normalization, an elliptical windowing is applied in order to avoid the presence of the nose (upper) or the chin (lower). These are non desired structures that can appear in the ROI due to a not very fine Lip Finding and Tracking.

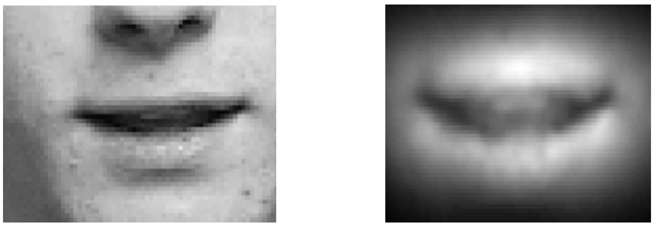

Figure 29.a Figure 29.b

Figure 29: Mouth Region a) before, and b) after the Normalization and Windowing

Before making the DCT an image Low Pass Filter was implemented to avoid high frequency noise (snow noise) that appears in the image especially for weak illuminations. In Figure 29.a, a mouth image after normalization is presented.

Discrete Cosine Transformation

Let us consider the normalized and windowed image of Figure 29.b $I(x,y)$ the DCT will provide a transformed image $O(m,n)$ with the same dimension. In the present work normalized images of 128x64 pixels are used.

$$O(n,m) = \delta_n \delta_m \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} I(x,y) \cos \frac{\pi(2y+1)n}{2M} \cos \frac{\pi(2x+1)m}{2N} \tag{66}$$

where:

$$\delta_n = \begin{cases} 1/\sqrt{M}, & n = 0 \\ \sqrt{2/M}, & 1 \le n \le N-1 \end{cases}$$

$$\delta_m = \begin{cases} 1/\sqrt{N}, & m = 0 \\ \sqrt{2/N}, & 1 \le m \le M-1 \end{cases}$$

An important property of the 2-D DCT (two dimensional DCT) is the separability, this means that the bidimensional transformation can be obtained computing 1-D DCT (one dimensional DCT) on the rows of the image and then one 1-D DCT on the columns of the previous result. This will reduce the complexity and CPU consuming time, there are more efficient implementations for the DCT like the one proposed in [Lee and Huang, 1994].

The normalized image has a size of 128x64 pixels, which implies that the complete number of DCT coefficients is also 128x64. The coefficient (0,0) has information about the offset of the luminance and it can be rejected as it is not an interesting coefficient for the recognition. In order to find a right set of DCT coefficients inverse DCT of different sets of coefficients were computed. The DCT coefficients that are used to reconstruct the images in Figure 31 are shown in Figure 30 (red coloured). In Figure 31.a, the reconstructed image is shown, where all coefficients according to Figure 30.a were used. In the other representations of Figure 30 different configurations for using 48 coefficients are given. The best results for yielding relevant low frequencies are obtained by using the configuration shown in Figure 30.d. This selection can be used as a first approximation for finding the right set of parameters.

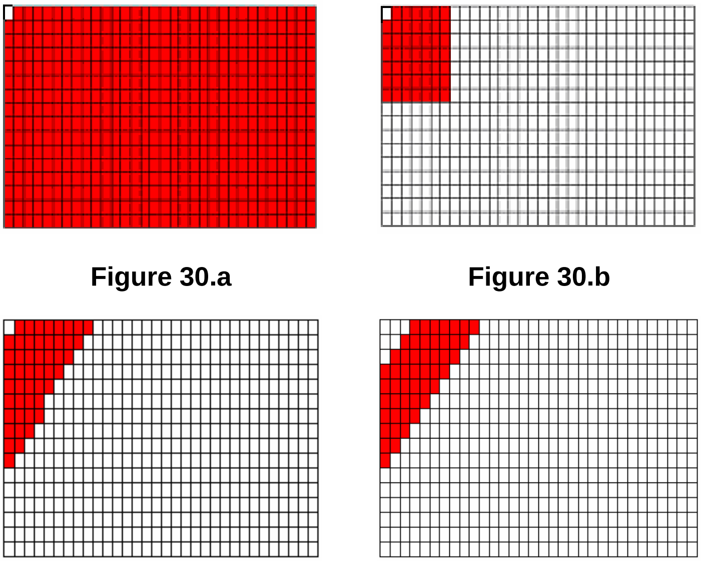

Figure 30.a

Figure 30.b

Figure 30.c

Figure 30.d

Figure 30: Four Different set of 48 Coefficients taken from the DCT Matrix Transformation. In order to know which Information they contain, the inverse DCT will be performed.

Taking into account the dimensions of the normalized image 128x64, the fact that 15 images per second are processed, that only 48 coefficients of the DCT are necessary, assuming two cycles per operation and using the recursive DCT algorithm proposed in [Lee and Huang, 1994] the DCT feature extraction can be performed using less than 4 MHz.

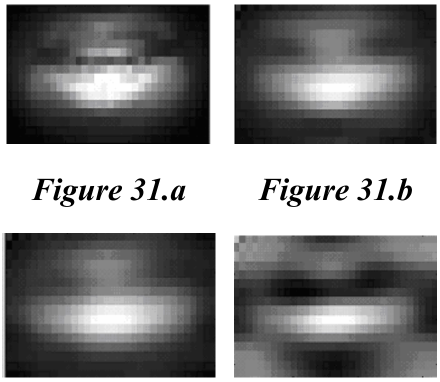

Figure 31.a *Figure 31.b*

Figure 31.c *Figure 31.d*

Figure 31: Four inverse DCT with the Coefficients Marked in red on Figure 19, in such a way that Figure 20.a was obtained with the Coefficients of figure 19.a, and analogue for figures b, c and d

In order to know whether these coefficients contain speech information, we are representing their values over the time for two different utterances of the same word. The utterances "one" (same speaker and light conditions) are shown in Figure 32 and the utterances "two" in Figure 33. As it can be seen, there is a repetition of a dedicated muster for each digit. The features can be distinguished between "one" and "two" and these musters are repeatable in different utterances. Thus, the features we are selecting have the right properties: repeatable and discriminative. Taking this into account good recognition results can be expected by the use of these features.

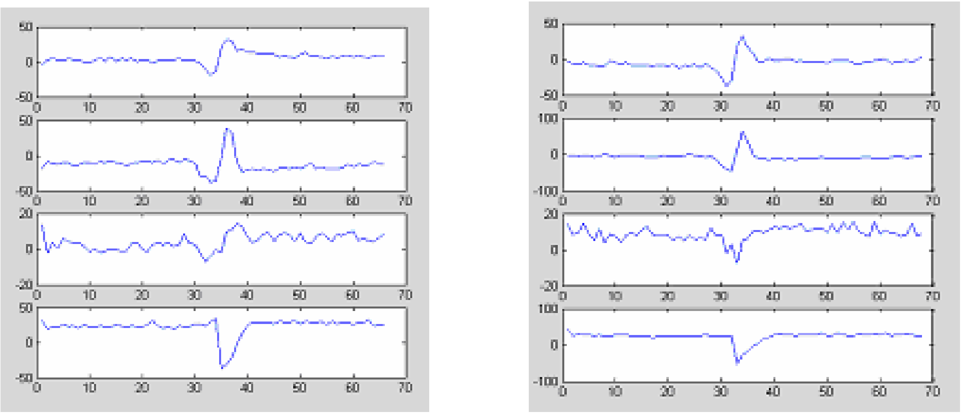

Figure 32: First four DCT Coefficients for two Different Utterances of the Word "one" Spoken by the same Speaker

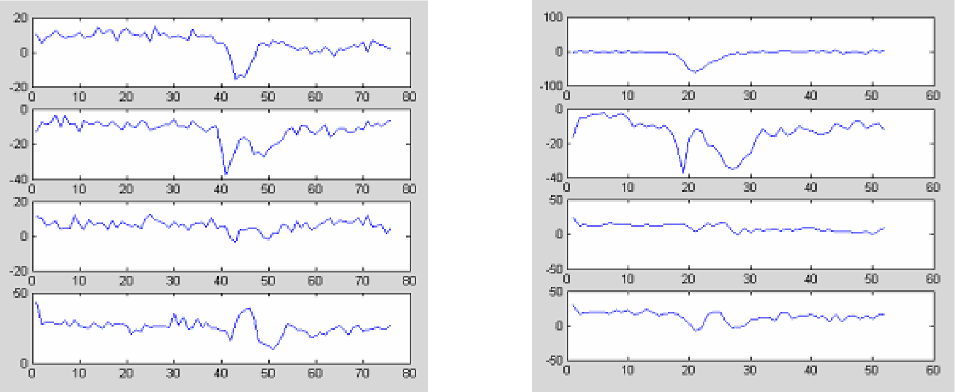

Figure 33: First four DCT Coefficients for two Different Utterances of the Word "two" Spoken by the same Speaker

These results will be confirmed when the system will be trained with these DCT features. First recognitions results using the DCT and also the ASM as feature extraction will be shown in section 4.2.5.

4.2.4 Visual Feature Extraction Post-Processing

In this section we are dealing with the visual features post-processing. As we know from conventional speech recognition, having a suitable feature post-processing is as important as having an appropriate feature extraction algorithm, actually this post-processing can be assumed as a part of the feature extraction. Some additional information can be extracted from the dynamics of the visual coefficients and of course from their energy. Furthermore, a dimension reduction will be achieved by using a Linear Discriminative Analysis. In Figure 34 the cascade of processing we have applied to our visual feature vector is shown. The input of this processing consists of the weighting coefficients of the five first modes; see Figure 26 for the ASM feature extraction implementation. And for the DCT the 48 coefficients showed in red in Figure 30.d. The output will be the final feature vector given to the recognizer; it will consist for both cases in the first 10 coefficients.

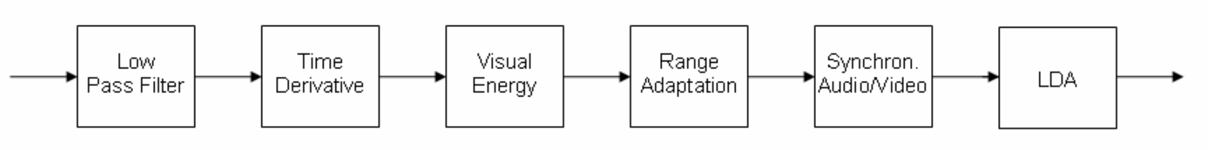

Figure 34 : Cascade of operations applied on the visual features obtained either from ASM or DCT.

First of all, a Low Pass Filter (LPF) is applied to the visual features to avoid high frequency noises, the visual signal is a low frequency signal, so all high frequency components can be rejected because they will be mainly noise, for example generated by errors in the Lip Finding and Tracking.

It is known that dynamic features are very important for Lip Reading [Rosemblum and Saldaña, 1998]. For this reason time derivatives are generated. Derivatives will be evaluated simply by the subtraction of the actual coefficients value and its value 6 frames before. At this point our feature vector will be composed for the ASM implementation by the 5 first coefficients of the highest ASM modes + the first derivatives of the 5 coefficients+ the second derivatives of the 5 coefficients, the dimension of the feature vector at the output of the derivative block will be 15 coefficients for the ASM implementation. Regarding DCT implementation the output will consist in the 48 original DCT coefficients + first derivatives of the 48 coefficients + the second derivatives of the 48 coefficients, we have a total of 144 visual coefficients in every frame. In order to reduce the insertions of errors we have defined a new feature called visual energy, this feature conveys information whether there is or not speech according to the visual information. The visual energy is derived from the visual features and helps to discriminate the word boundaries. Our visual energy is given by the quantity of lip movement. As visual energy the mean energy of the first derivatives of the coefficients will be used. It will be an additional coefficient for every frame. A total of 16 coefficients for the ASM and 145 for the DCT is provided at the output of the visual energy block.

As the number of available DCT coefficients is quite large, a feature dimension reduction is achieved by a Linear Discriminate Analysis (LDA). This operation is quite simple since it comes down to a simple matrix multiplication. Moreover, LDA is an already implemented algorithm in conventional speaker independent speech recognition systems for embedded devices, as it has been seen in the audio feature extraction. LDA delivers a feature vector with a lower dimension. Furthermore, LDA allows the system to model the emission probabilities by Gaussian mean vectors with diagonal covariance matrices, where all variances are unified and to code the mean vector of each Gaussian by less than a byte per component [Varga et al., 2002].

The LDA will be performed using context information; this means that the coefficients of the previous frame will be concatenated with the coefficients of the current frame. To calculate the LDA a segmentation of the features must be provided. As it is explained in the next chapter, the audio HMM with the phoneme segments has been selected as basis of the visual modeling. This is the reason why a synchronization of the audio and video signal must be performed before the visual LDA multiplication. The LDA will use inter-frame information and it has been defined in the sample rate of the audio information one feature frame every 15 ms. before the LDA an interpolation is performed. A detailed description of the synchronization is provided in the next chapter. Our LDA evaluation is an integer implementation and not a floating point. The features must be represented in the dynamical range used for our implementation, we are working with 1 byte, that means that all features must be described by integers in the range [-127, 128]. In our implementation after the energy evaluation the range adaptation takes place followed by the visual interpolation (synchronization with the audio signal). As it has been said, the main idea is to describe the features exploiting as much as possible the range where they can be written. In order to obtain this, a mean subtraction is performed followed by a factor multiplication, the mean value and the factor are different for each coefficient.

After the range adaptation and the synchronization the LDA is applied. In Figure 35, the 5 first DCT coefficients are represented without offset for the utterance “one one one one one”. As we can see there is a repetition of a muster that correspond with the five “one”. It is also outstanding that the two first “ones” are easily to be detected on the first, third and fifth coefficients, meanwhile the last three “ones” are clearly distinguished in the second and fourth coefficient. The effect of the LDA can be easily seen on Figure 36, where the first 5 coefficients after LDA are shown, in this case and for all the five coefficients a clear muster of the five ones is to be shown, so we can see how these features are much more discriminative than those ones before LDA. The LDA works with a supervector comprising two frames of visual coefficients for input, the actual one and the last one. The supervector has a total of 145 + 145 = 290 coefficients. Output of this linear transformation are coefficients ordered according to their discriminative power. We have found that it is optimal to use for each frame only the first 10 coefficients.

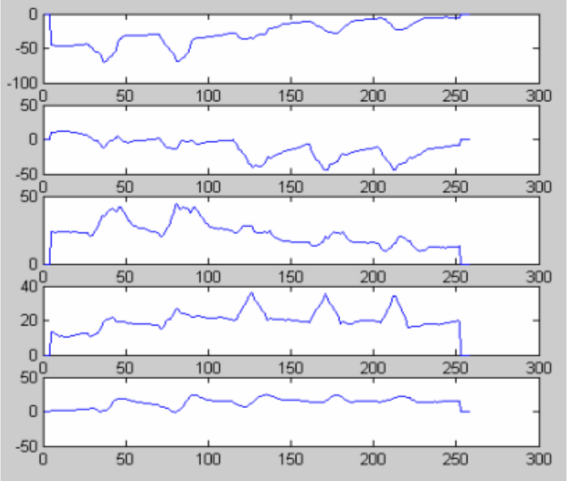

Figure 35: DCT Coefficients for the Sequence „one one one one one“ before the LDA

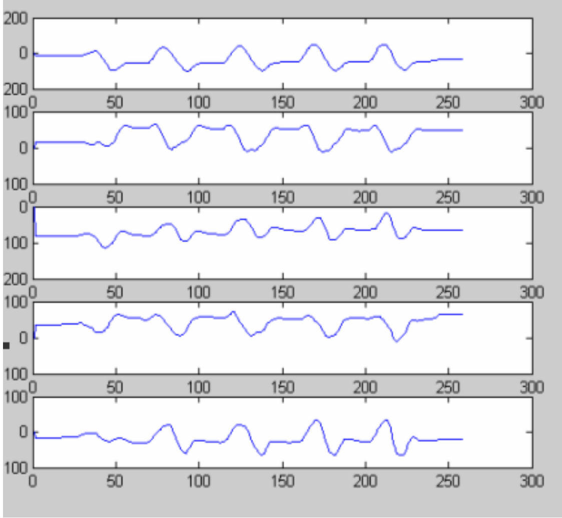

Figure 36: DCT Coefficients for the Sequence „one one one one one" after the LDA

4.2.5 *Evaluation of ASM vs. DCT*

In this chapter two different approaches have been presented for the visual feature extraction. As it was shown in Figure 27 for the ASM and in Figure 36 for the DCT, both set of coefficients convey information about the speech. The objective is to find which of them is the most suitable for our implementation. We have proposed two algorithms that because of their implementation requirements are able to work on an embeddable device, so the main aspect is to decide which provides the highest recognition rate (lower *WER*). Two visual HMM, one using the DCT the other using ASM features have been trained. HMM technology and training will be explained in the next chapter, now it is only interesting to know which of the features provides the best results. In Table 5 the Word Error Rate (*WER*) for speaker independent, continuous digit recognition in American English, spoken by native speakers is presented. The recognition process was performed by 16 speakers of the CUAVE database. In Figure 37 the *WER* is shown for only audio (conventional speech recognition), only video with ASM and DCT feature extraction, and the combination of both of them with the audio features. We have evaluated our system for different kinds of signal to noise ratios in the audio channel. Lip Reading is considered as a new way to deal with audio noise corrupted scenarios. On the Y-axis the *WER* is presented and on the X-axis the Signal to Noise Ratio of the audio channel is drawn. It can be seen that the results using only the audio features are getting worse as the noise is incrementing (lower SNR). Especially for lower SNR our Lip Reading system outperforms the conventional recognition using only the audio information. For noise free signal the results under 5% *WER* are quite good only with audio and in any case due to the lower information of the visual channel in comparison with the audio channel there is not an improvement. Using only the visual information a *WER* of 64.8% was achieved by using the ASM approach and 53% using the DCT, as it can be seen in Table 5 Word Error Rate using the DCT is 11.8% lower as using the ASM. For the ASM implementation, the detailed lip contours must be found, while for the DCT implementation only an approximation of the mouth region is required, which is much easier and works better in our implementation. ASM is often not able to find the lip contours properly, which explains the better results obtained by DCT. The results obtained with our DCT implementation improved the results obtained at Clemson Univertity [Amarnag et al., 2003] where *WER* of 63.2% was achieved also with DCT but with a lower resolution and without LDA.

	ASM	*Our DCT*	*CU DCT* *[Amarnag et al., 2003]*
Word Error Rate	*64.8 %*	*53.0%*	*63.2%*

Table 5: Visual Feature Extraction Comparison for Continuous Speaker Independent digit recognition using only visual information.

By using the combination of DCT visual information and audio information we obtained an improvement of 30% of the recognition rate for bad SNR situations, see Figure 37. This improvement is decreasing when the SNR is getting better. It is also important to point out that for the combination of audio and ASM features there is an improvement of the performance for SNR lower than 5 dB but for audio and DCT this improvement can be seen until SNR lower than 15 dB. This means that an improvement in the video results (horizontal line goes down) implies a translation of the cross point Audio and Audio-Visual results to the right. If we were able to obtain another feature extraction providing better video results more information could be used to improve the results also in clean speech environment.

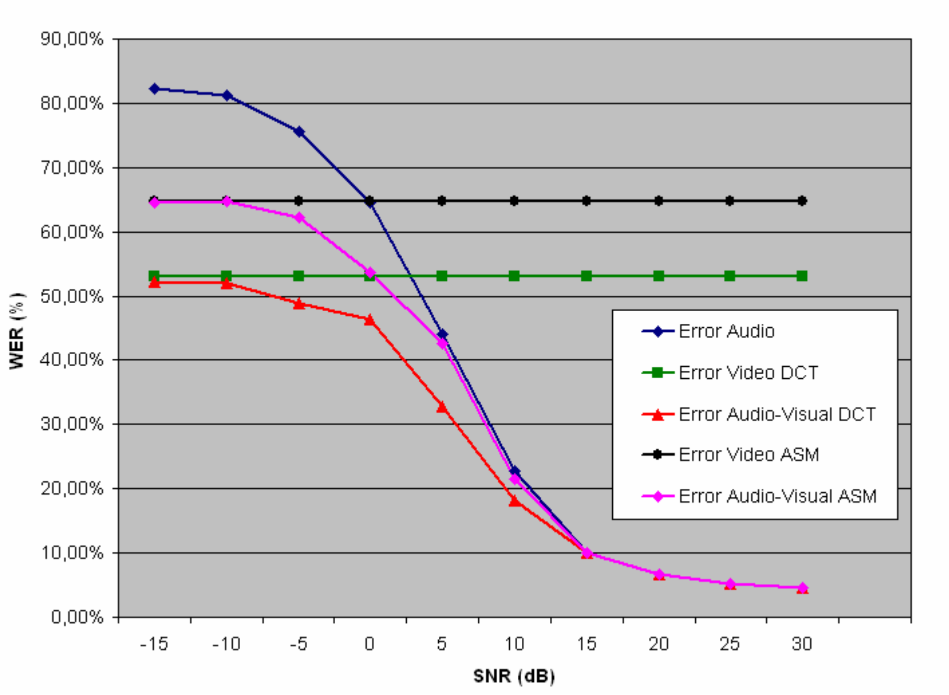

Figure 37: Comparison between ASM and DCT Recognition Results for Continuous Digit Task with Speaker Independent Recognition. Audio-Visual results obtained using multi-stream integration and optimal weightings, see Chapter 5.

Chapter 5
Audio-Visual Recognition

In this chapter the recognition process using audio and video information is going to be studied. In the last chapter a set of audio and visual features has been obtained, now we are going to show how to process these features in order to provide a good recognition result. Several problems must be solved; the synchronization between audio and visual features, the mapping of these features with the words in our vocabulary, and the integration of the information from the two modalities providing a unique result. All these issues are going to be dealt in this chapter.

5.1 Synchronization of Audio and Video Features

In audio-visual recognition two different levels of synchronization between audio and video information can be found: in a signal and semantic level. First of all, the signal level, audio and visual signal not always begin to be captured at the same time and the sampling rate from both signal is not necessarily always the same. Audio and visual signals have different bandwidth and therefore different sampling frequencies are required. Secondly, the asynchrony in the semantic level, assuming synchronization in the signal level the information that both signals are carrying is not always synchronized. In order to pronounce some phonemes, lips will begin to prepare their position before any sound is produced. Therefore, visual information will appear for certain visemes before the sound is pronounced and the phoneme performed. This implies that phonemes and visemes are not always generated on time. In this section we are going to solve the signal level synchronization and we will deal with the semantic level synchronization in the modeling of the audio-visual fusion.

In the last chapter two sets of features were obtained for the audio and visual channels, these features represent the audio and visual signal in certain instants:

$$O_n^A = O^A\left(t_0^A + n \cdot \Delta t^A\right)$$
$$O_n^V = O^V\left(t_0^V + n \cdot \Delta t^V\right) \quad (67)$$

First of all, a delay between audio and video can be caused by the recording process and also by the audio and visual feature extraction algorithms. For synchronization the important point is the relative delay between audio and video:

$$delay = t_0^A - t_0^V \quad (68)$$

If this delay is fixed and known, the problem is easy to be solved because an additional delay can be entered in order to adjust the beginning of both signals. Difficulties arise when the delay is variable and no accurate prediction of its value can be made. This situation occurs when the startup of the camera has not a defined duration or it can be delayed due to unexpected higher priority processes are taking use of the processor unit. This situation is difficult to be solved in signal level. However, there exist implementations of the HMM as the bimodal Coupled HMM (CHMM) structure that models a certain delay between both streams as it will be seen in 5.3.3. In the CUAVE database used for our experiments there is not a significant delay between the audio and video stream. For an implementation in an embedded device we should have a correctly synchronization between both signals as in our case with our test database, or at least a fixed delay. If the delay could not be estimated complex HMM structures like coupled HMM should be used.

Other important aspect related to the feature fusion is the difference between the sampling rates of both sources, $1/\Delta t^A, 1/\Delta t^V$. Whereas the acoustic sampling rate in our case is 8 KHz, the visual data are sampled at 15 Hz for many embedded applications and at 29.97 Hz for the database used in our experiments. Nevertheless, we are making the integration of the features obtained from the visual and acoustic feature extraction and these must be synchronized (and not necessarily the input signal). Therefore, the sampling rate of the audio and visual feature vector must be the same. In order to avoid a loss of information, the highest sampling rate must be taken as reference and the signal with a lowest sampling rate should be interpolated to obtain the same sampling rate. The original audio signal conveys almost all important information until 4Khz and so that a sampling rate of 8 Khz is appropriate according to the Nyquist theorem [Bruce 1986]. As it was presented in chapter 4, a long-term frequency analysis is performed in the audio signal using a window with a shift of 15 ms. This is the sampling frequency of our final audio feature vector, and this is the sampling frequency that must have the visual feature vector. Visual features are representing the movements of the lips; due to its nature, the bandwidth of this signal is quite smaller than the one of the audio signal. Some studies assure that the limit for human speech reading is 5 frames per second [Willian et al., 1997], with a lowest sampling frequency humans cannot read the lips. Potamianos showed in [Potamianos et al., 1998] that in automatic Lip Reading systems the recognition rate deteriorates dramatically below 10 frames per second. Thus the sampling rate of 15 frames per second (and 29.97 for the CUAVE DB) used for this work allows an automatic Lip Reading.

As it can be seen in Figure 38, the video features must be interpolated in order to obtain the same sampling frequency as for the audio features. This synchronization is necessarily for integration strategies based on the features “early integration” or as we will see in section 5.4.2 for “hybrid integration” strategies (Multi-stream). Regarding so called “late integration” no synchronization between both channels would be required as the integration would be in a decision level, at least for isolated speech recognition.

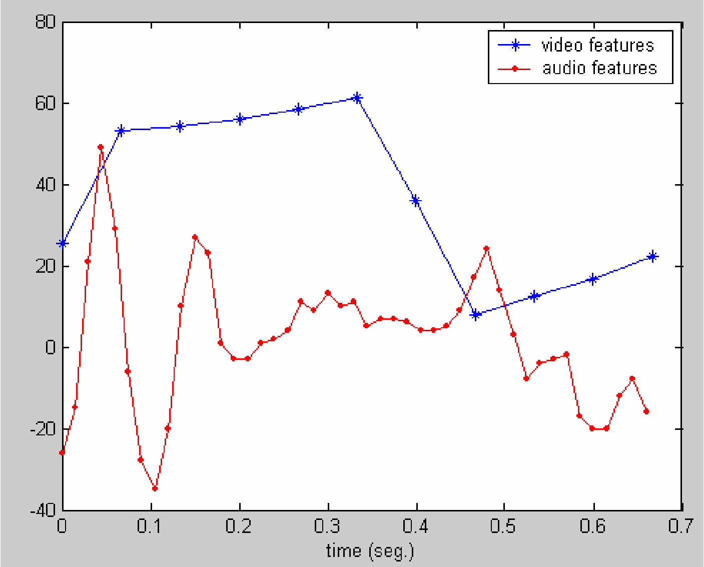

Figure 38: Audio and visual features represented in the time instant they have been generated

In Figure 39 the height and width of the outside and inside lip contour in the middle of the lips is represented. We would like to point out different aspects:

- As we have said the right solution to adapt both sampling rates is to interpolate the visual signal. A down sampling on the audio information would convey a loss of information.
- It can be observed that with the resolution of our camera (320x420) and for a distance of approximately 20-50 cm between the user and the camera the movements of the lips can be appreciated as a pixel variation.

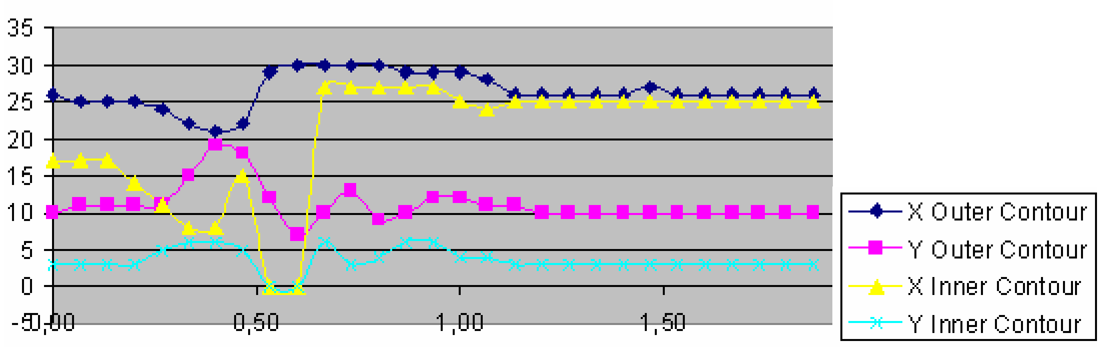

Figure 39: Visual Information; Coordinates of four Points positioned on the lips Contours

Consequently, we have enough time and pixel resolution to describe the lip movements and in order to adjust audio and video sampling rates, an interpolation in the visual feature vectors is applied.

5.2 Acoustical and Visual Modeling

Once audio and visual features have been obtained they are used to classify every utterance as one word from our lexicon. The input of the Automatic Speech Recognition (ASR) is composed by a vector of features O as it was explained in chapter 0. The output of the system will be a word or a sequence of words in a continuous recognition task, W_i of the lexicon $\{W_1, W_2, \ldots, W_N\}$ as it can be seen in Figure 40.

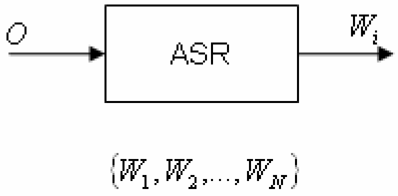

Figure 40: Automatic Speech Recognition Process (ASR)

According to the Bayes decision rule the optimal classification provides the word with the highest a posteriori probability. The systems output will be the word that maximizes the probability of being said, knowing the observation vector:

$$W_i = \underset{j=1,\dots,N}{\arg\max} P(W_j \mid O) \tag{69}$$

For an optimal classification a description of the probabilistic function $P(W_j \mid O)$ must be known**.** This description is not possible to be obtained in practical situations; this is the reason why a new formulation is used taking advantage of the Bayes theorem:

$$W_i = \underset{j=1,\dots,N}{\arg\max} \frac{P(O \mid W_j) \cdot P(W_j)}{P(O)} \tag{70}$$

In equation (70) the result is independent of $P(O)$, therefore the decision rule can be simplified as:

$$W_i = \underset{j=1,\dots,N}{\arg\max} P(O \mid W_j) \cdot P(W_j) \tag{71}$$

$P(O \mid W_j)$ and $P(W_j)$ must be estimated. These two probabilities are much easier to be found in an apriori process than the previous $P(W_j \mid O)$. $P(O \mid W_j)$ is called acoustic and visual model depending on the kind of characteristics it models. Finally $P(W_j)$ is called language model, and it describes the probability of one word to be uttered taking into account the language rules, it can be also called grammar. The language model is independent on the kind of information (audio or visual). It will not be covered in this work where we are going to concentrate on the audio and visual models. In the recognition task that will be used for our experiments “continuous digits” all words have the same probability, so the language model is a uniform distribution and can be ignored for the decision rule of equation (71). The acoustic and visual models are going to be obtained by using the Hidden Markov Models [Rabiner, 1989].

5.2.1 Hidden Markov Models

Hidden Markov Models (HMM) [Rabiner, 1989] describe stochastic processes for sequences of samples. HMM provide an approximation of $P(O|W_j)$ to describe the probability that a set of features has been produced knowing that a word was uttered:

$$P(O|W_j) \approx \hat{P}(O|W_j) \tag{72}$$

Up to now we have assumed that we are obtaining the models of the words we want to be recognized. In the praxis, models are going to be obtained not for words but for the phonemes of a language. In this way once the models of all phonemes of a language are known, every word in this language is possible to be modeled. Just a phonetic transcription of the word will be needed. The word will be modeled as the concatenation of the different phoneme models. For difficult recognition tasks like speaker independent continuous digits, where the vocabulary is always fixed and also enough training material is available, specific models can be trained. In these models pseudo-phonemes [Bauer, 2001] are going to be used. These are special divisions of words that optimize the recognition.

HMM are stochastic automatons set up by a several states. Each state is connected with other states with certain transition probabilities and in every state each feature vector has a probability of being produced. In this way a HMM can be characterized by a set of parameters:

$$\hat{P}(O|W_j) = f\{a_{w_j}, b_{w_j}, \pi_{w_j}\} \tag{73}$$

where:

- $a_{w_j,i->e} = P(q_t = e | q_{t-1} = i, W_j)$ is the transition probability: the probability of being in the state e at time t knowing that at the previous instant the system was in state i for the HMM model W_j.

- $b_{w_j,i}(O) = P(O \mid q_t = i, W_j)$ is the emission probability: the probability of producing a vector of observations O on time t knowing that the system is in state i for the HMM model of the phoneme W_j.

- $\pi_{w_j,i->e} = P(q_{t=0} = i \mid W_j)$ is the initialization probability: the probability of being in state i at time $t=0$ for the HMM model of the phoneme W_j.

In a training process this set of parameters is going to be estimated for every phoneme or pseudo-phoneme. The task of the training process is therefore to obtain the audio and visual models. All these parameters are going to be used in the recognition process to obtain the $\hat{P}(O \mid W_j)$ by using the so called Viterbi-Decoding [Rabiner, 1989]:

$$\hat{P}(O \mid W_j) = F\{O, a_{W_j}, b_{W_j}, \pi_{W_j}\} = \sum_q \pi_q \prod_{t=1}^{T} a_{q_{t-1}->q_t} b_{q_t}(O_t) \tag{74}$$

5.2.2 Hidden Markov Models for Embedded Devices

A brief description of the HMM parameters has been provided in the last paragraph. Now we are going to introduce some specific characteristics of our HMM which simplifies its implementation in embedded devices [Bauer, 2001], [Varga et al., 2002]. These Characteristics are implemented in our VSR Very Smart Recognizer $^{®}$ (VSR) [VSR, 2002] and are used for the acoustic and also for the visual modeling.

There are different kinds of HMM topologies. In VSR the Bakis-Topology has been used because of its simplicity and correctly time modeling. In this structure the temporal development is quite related with the state evolution. One state is followed by the same one; the next one or a spring of one state as it can be seen in Figure 41.

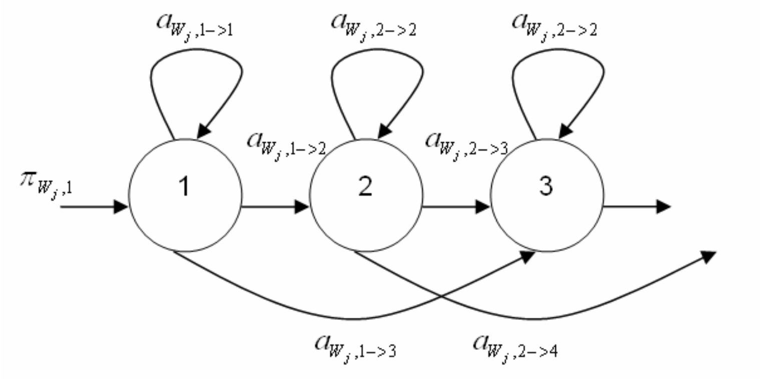

Figure 41: HMM with Bakis Topology

These transition probabilities will be reduced in our embedded system to the next one:

$$a_{W_j,s->s'} = \begin{cases} a_0 & : s' = s \\ a_0 & : s' = s+2 \\ 1-2a_0 & : s' = s+1 \end{cases} \tag{75}$$

As it can be seen in equation (75) the self-transition and the one state spring have the same probability. In the praxis the transition parameter a_0 will not be obtained in the training process but it will be set as a constant value.

Now the simplifications assumed for the estimation of the emissions probabilities are presented. Emission probabilities convey the main information about the feature vector, as we have seen they provide the probability of obtaining a vector of features in a certain state:

$$b_{w_j,i}(O) = P(O \mid q_t = i, W_j) \tag{76}$$

In general, these emission probabilities will be represented as a mixture of several distributions. Indeed, they will be represented as a lineal combination from different basic functions:

$$b_{w_j,i}(O) = \sum_{m=1}^{M_i} c_{i,m} P_{i,m}(O) \tag{77}$$

Each of these kernel density functions can be called Mode. M_i is the number of modes for the state i. Each mode is weighted by a coefficient $c_{i,m}$ for which it is assure that:

$$\sum_{m=1}^{M_i} c_{i,m} = 1 \tag{78}$$

For high dimension feature vectors and for a finite number of modes, the weighted sum of different kernel density functions can be approximated by the mode with the highest probability and the correspondent coefficient:

$$b_{W_j,i} \approx \max\{c_{i,m} \cdot P_{i,m}(O)\} \tag{79}$$

In the previous approximation it is assumed that there is a dominant term in the sum of equation (77). This assumption can be supported when the dimension of the feature space is high and the number of modes is limited, in such a way that the feature domain is not very densely occupied by the different modes.
As kernel density functions for ASR the Gaussian distributions will be used:

$$P_{i,m}(O) = N(O, \mu_{i,m}, \Sigma_{i,m}) \tag{80}$$

where $N(O, \mu_{i,m}, \Sigma_{i,m})$ is a Normal distribution with a mean vector $\mu_{i,m}$ and a covariance matrix $\Sigma_{i,m}$

$$N(O, \mu_{i,m}, \Sigma_{i,m}) = \frac{1}{\sqrt{(2\pi)|\Sigma|}} e^{-\frac{1}{2}(O-\mu)^T \Sigma^{-1} (O-\mu)} \tag{81}$$

The covariance matrix will be a diagonal matrix:

$$\Sigma_{i,m} = diag(\sigma_{i,m}^2) \tag{82}$$

The use of diagonal covariance matrix is justified by the fact that the different components of our feature vector are not correlated. This characteristic is achieved at least in part by the use of the LDA. The use of diagonal covariance matrix is generalized in the Speech Recognition and it has the advantage of reducing the quantity of parameters to describe the models. In our system just one global variance will be used:

$$\sigma_{i,m}^{2} = \sigma_{0}^{2} \tag{83}$$

Experimental investigations have shown that the use of this kind of easy variance description in combination with LDA provides optimal recognition results [Bauer, 2001].

There are other possibilities to reduce the number of parameters that describe the emission probabilities for example semi-continuous HMM where there are not state specific basis functions but a set of global basis functions used for all states (Codebook) this approach is not used in our work.

In the use of HMM the use of logarithmic probabilities has been quite extended. In our embedded implementation a log-likelihood has been defined as:

$$B_i(O) = -2 \cdot \sigma_0^2 \cdot \log b_{W_j,i}(O) \tag{84}$$

This is the so called neg-log transformation from the emission probability (emission score) and will be the description of the emission probabilities in our embedded implementation.

Taking into account the Gauss distributions in equation (81), the use of a unique global variance of (83) and the approximation of equation (79), the neg-log emission probability defined in equation (84) can be simplified as:

$$B_i(O) = \min_m \left\{ C_{i,m} + |\, O - \mu \,|^2 \right\} + const \tag{85}$$

where:

$$C_{i,m} = -2 \cdot \sigma_0^2 \cdot \log c_{i,m} \tag{86}$$

As it can be seen, the transformed emission probabilities have been reduced to a Euclidean distance between the feature vector and the mean vector of the Gaussian.

5.3 Integration Strategies

A very challenging aspect in audio-visual speech recognition is the integration of both information sources: audio and video. Several approaches can be found in the literature to proceed with the integration problem [Potamianos et al., 2003], [Nefian et al., 2002], [Dupont et al., 2000], [Teissier et al., 1999]. The terminology to classify the different approaches is not universal; in this work we have tried to summarize all different approaches. They have been classified according to the position where the integration takes place using the same nomenclature as in the psychology studies of the speech human audio-visual integration [Summerfield, 1979]: early integration or feature fusion, late integration or decision fusion and hybrid integration or model fusion. With this double terminology all different terms found in the literature have been covered.

5.3.1 Early Integration or Feature Fusion

In the Early Integration, also called Feature Fusion [Potamianos et al., 2003], the audio and visual features are combined before the state emission probabilities estimation and the Viterbi decoding (search). Implementations of this fusion strategy can be found in the literature [Potamianos et al., 2001]. Just one single classifier is trained on a concatenated vector of audio and visual features as we can see in Figure 42.

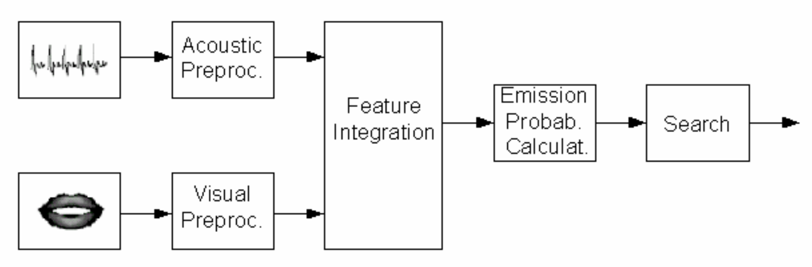

Figure 42: General Diagram of Early Integration

The recognition process will provide the word from the vocabulary that maximizes the probability of being uttered knowing a certain set of audio and visual features. These features will be concatenated and a HMM will be trained for this set of features.

$$O^{AV} = [O^{A}, O^{V}]^{T} \tag{87}$$

using equation (69) with the audio-visual set of features of equation (87) the selected word should maximize:

$$\arg\max_{W_i} \left\{ P(W_i \mid O^{AV}) \right\} \tag{88}$$

Taking into account the Bayes theorem of equation (70) an audio-visual model will be defined. For this modeling HMM theory will be used and the parameters defining this model will be estimated using the audio-visual vector of equation (87). The new audio-visual HMM will combine in the same state the audio and visual features; there will be therefore a compulsory state synchronization between audio and video features. The parameters that will define this new HMM will be the same as in equation (73); the emission probabilities will have now the influence of the audio visual feature vector:

$$b_{W_j,i}\left(O^{AV}\right) = P\left(O^{AV} \mid q_t = i, W_j\right) \tag{89}$$

On the one hand, an important advantage of this integration strategy is the modeling in the same state of the audio and video information, which allows it to take into account the conditional dependencies between both modalities. On the other hand, it is not possible to dynamically give different importance to each channel according to their reliability [Potamianos et al., 1998]. The emission probabilities of audio and video are modeled together in equation (89), and it is not possible to give to one of the two modalities more importance. It is well known that in noise clean environments the quantity of discriminate information of audio modality is higher than the one of video, in such a situation will be desirable to give more importance to the audio channel. But when the audio signal is corrupted the importance of the video should be higher to obtain the optimal results. This modality weighting is not possible by using the early integration.

We have implemented an early integration strategy using 24 audio features and 10 video features obtained after two independent LDA one for each modality. These after LDA features are combined in a unique feature vector of dimension 34 as in formula (87). With these features a HMM was trained using the 20 training set speakers of our CUAVE Database. 181 different states are modeling the word-base digits pseudo-phonemes. A total of 989 Gaussians constitute the audio-visual models, these Gaussians are described for our embedded implementation using only their means as it was seen in formula (84).

In Figure 43 we show the results of our implementation for continuous digit speaker independent recognition. The different conditions on the audio channel have been obtained just by adding to the original signal tone variable noise with a certain level of energy for the different SNR ranges The audio information has been used using an only audio HMM, the video information has also been presented using the only visual HMM and finally the combination of audio and video information using the early integration strategy.

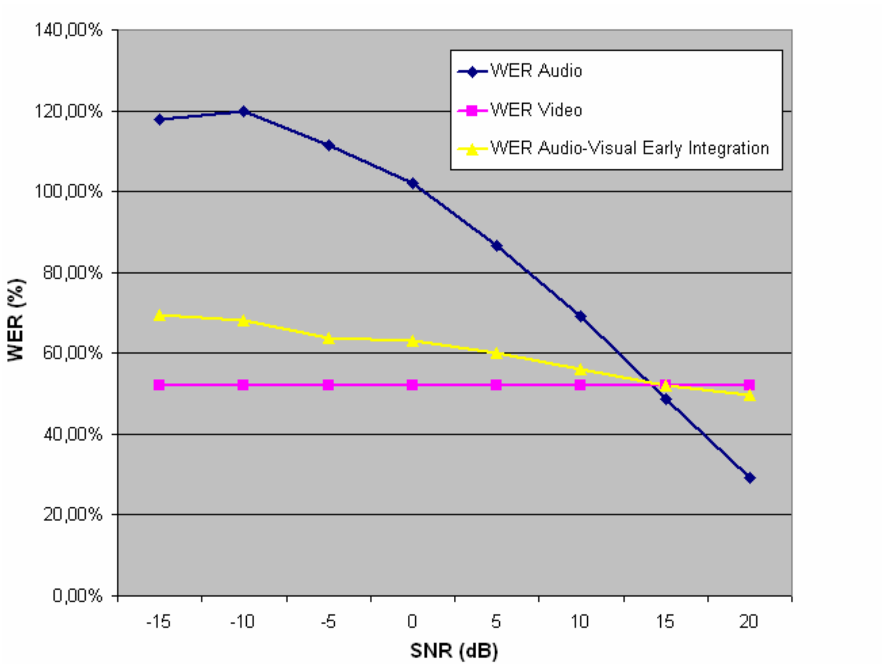

Figure 43: Early Integration in comparison with Audio and Video Results

As it can be seen the results obtained with early integration are always between the results obtained using the two pure modalities. For bad SNR the results are better than using only audio but worse than using only video. For good SNR the inverse effect can be observed. This is due to the fact that in early integration there is not possibility to give different importance to each modality depending on the reliability of each channel. This point reduces the interest of this solution.

5.3.2 Late Integration or Decision Fusion

In Late Integration [Matthews et al., 2002] there are two whole recognizers working independently for the audio and the video channel respectively, as it can be seen in Figure 44.

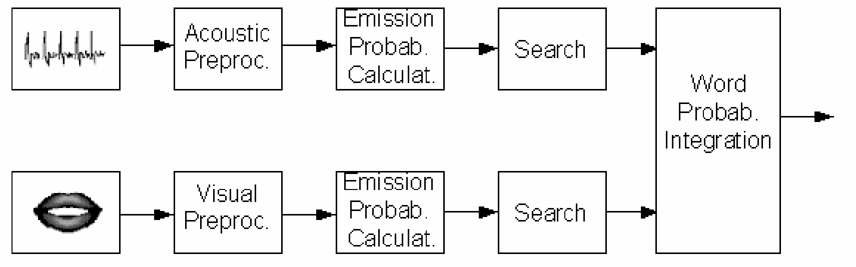

Figure 44: General Diagram of Late Integration

The combination is performed at the output of the recognition, at the end of the Viterbi decoding (search) process. Therefore for each word in our lexicon two different probabilities will be provided one for each modality $P(W_j | O^A)$ and $P(W_j | O^V)$. In this integration strategy we are not combining the features but the decisions of each classifier:

$$P_{Late}(W_j | O^{AV}) = P(W_j | O^A) \cdot \alpha_A + P(W_j | O^V) \cdot \alpha_V \tag{90}$$

This is the reason why it is also called decision fusion. The final solution will be the word in equation (90) that maximizes the combined probability of equation (91).

$$\arg\max_{W_i} \left\{ P_{Late}(W_j | O^{AV}) \right\} \tag{91}$$

In order to obtain the two probabilities $P(W_j | O^A)$ and $P(W_j | O^V)$ an acoustic model and a visual model must be found. Each of these independent models will be defined by a set of parameters that will be obtained in two independent training processes one for the audio and another one for video:

$$P\left(O^A | \lambda_{W_j}\right) = f\left\{a^A_{W_j}, b^A_{W_j}, \pi^A_{W_j}\right\} \tag{92}$$

$$P\left(O^V | \lambda_{W_j}\right) = f\left\{a^V_{W_j}, b^V_{W_j}, \pi^V_{W_j}\right\} \tag{93}$$

Late integration presents several advantages in comparison with early integration. First of all, a reliability measure can be introduced to weight the audio and the visual information according to the current channel conditions as it can be seen in equation (90). Secondly, the fact that the synchronization is carried out at a word level and not at a state level as in the early integration allows a possible asynchrony between the audio and visual signals. It is shown that the visual activity can precede the audio signal by as much as $120\ ms$ [Grant et al., 2001]. This kind of solution would be the optimal to be implemented in systems where a signal level synchronization (see section 5.1) is not possible. Continuous recognition will require also signal level synchronization as it will be shown now in our late integration implementation.

Our late integration implementation will follow the philosophy of equation (90) and (91) but we are going to meet some limitations due to the characteristics of our embedded recognizer:

- Our experiments are made on continuous digit recognition, for this task our recognition engine provides not all possible words with its scores, but only the two more likely words.
- Our recognition is providing relative scores from the search process. Due to the fact that the Viterbi decoding is performed twice and independently for audio and video the relatives scores obtained for both modalities cannot be matched together as they are not containing the same information.
- In a continuous mode the recognizer will provide a word when one hypothesis has been the winner for a period of time larger than a minimum stable time. At this point the second best word will be provided with its score at this time. It is highly probable that the second word achieves its best score in a different instant. To make a correctly continuous late integration we should wait to the end of the utterance and obtain the value of the best hypothesis (each one composed by chain of digits). The whole 2nd path should be saved, which would consume more resources in an embedded implementation. Moreover the result would not be ready until the end of the whole utterance.

Taking into account these limitations we have implemented our late integration strategy. Two different HMMs have been trained: one just for video (with 856 Gaussians) and one only for audio (with 1192 Gaussians) using Maximum Likelihood training proceeding. Each recognition system has provided two words: the most likely (Word1) and the second one (Word2), see Figure 45. For each time instant we have a maximum of four different words (probably some of them will be the same). Our late integration will provide a result when the audio or the video outputs changes. In order to avoid small insertions due to synchronization problems (in the signal level or in semantic level) only audiovisual outputs which are stable longer than a minimum stable time (15 frames) will be considered as it can be seen in Figure

45. In order to find the properly output of the system each word of each modality is going to receive a score depending on the likelihood it has been recognized β and on the reliability of its channel α :

$$
\begin{aligned}
S(W_A^1) &= \beta_1 \cdot \alpha_A \\
S(W_V^1) &= \beta_1 \cdot \alpha_V \\
\ldots \quad &\quad \ldots \\
\ldots \quad &\quad \ldots \\
\ldots \quad &\quad \ldots \\
S(W_A^n) &= \beta_n \cdot \alpha_A \\
S(W_V^n) &= \beta_n \cdot \alpha_V
\end{aligned}
\tag{94}
$$

In our implementation only the 2 best words will be provided therefore *n=2*. For each word in each modality an accumulate score will be calculated, in such a way that if the same word is recognized in both modalities its scores will be added:

$$\underset{M=A,V;I=1,2\ldots n}{AS(W_M^I)} = \sum_{c=A,V} \sum_{j=1,2,\ldots,n} S(W_c^j) \cdot \delta(W_c^j - W_M^I) \tag{95}$$

Finally, as result of the late integration the word with a highest score will be selected:

$$W = \arg\max_{W_M^I} \left\{ AS(W_M^I) \right\} \tag{96}$$

In our implementation $\beta_1 = 2; \beta_2 = 1$ and the channel dependent coefficients α are optimized for the different SNR. When two different words have the same score the audio word modality will be preferred.

In Figure 45 and Figure 46 two examples of our late integration are provided; the first one with 20 dB of noise in the audio channel and the second one with 0 dB. The uttered digits chain was: “one two three four”. In the first line the two most likely recognized words using the visual information are provided. In the second line the results using only audio information can be found. In the third line the four results for audio and video are shown and finally in the fourth only the finally results are drawn. For 20 dB it can be shown that the audio-visual results are worse than the results obtained only using the audio information, two numbers have been inserted in periods of time where in audio channel a silence was detected. When a hypothesis remains active a period of time shorter than a threshold, it will not be considered as final result, this situation is shown by red lines.

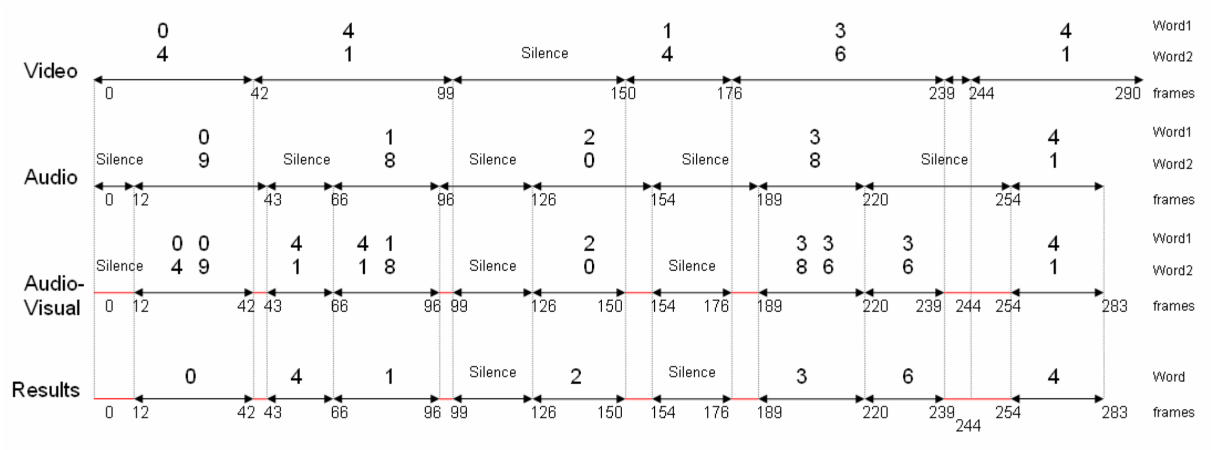
Video
0 4
4 1
Silence
1 4
3 6
4 1
Word1
Word2
0 42 99 150 176 239 244 290 frames
Audio
Silence 0 9
Silence 1 8
Silence 2 0
Silence 3 8
Silence 4 1
0 12 43 66 96 126 154 189 220 254 frames
Audio-Visual
Silence 0 4 0 9
4 1
4 1 1 8
Silence 2 0
Silence 3 8 3 6
3 6
4 1
0 12 42 43 66 96 99 126 150 154 176 189 220 239 244 254 283 frames
Results
0 4 1 Silence 2 Silence 3 6 4
Word
0 12 42 43 66 96 99 126 150 154 176 189 220 239 244 254 283 frames

Figure 45: Time chronogram for Late Integration for 20 dB Car Noise.

For 0 dB of noise, Figure 46, it can be seen that in audio results there are much more insertions than for 20 dB. Some of these insertions will be rejected with our late integration strategy. Audio-Visual integration will reduce some substitutions, e.g. the audio channel has found in frame 178-222 “zero” where it should have found “three” the fact that the second most likely word from audio and the first likely word from video is “three” has implied that the audio-visual result is also “three”. We can see how for bad environments late integration has improved the results of the audio recognition not only reducing the insertions but also reducing the substitutions.

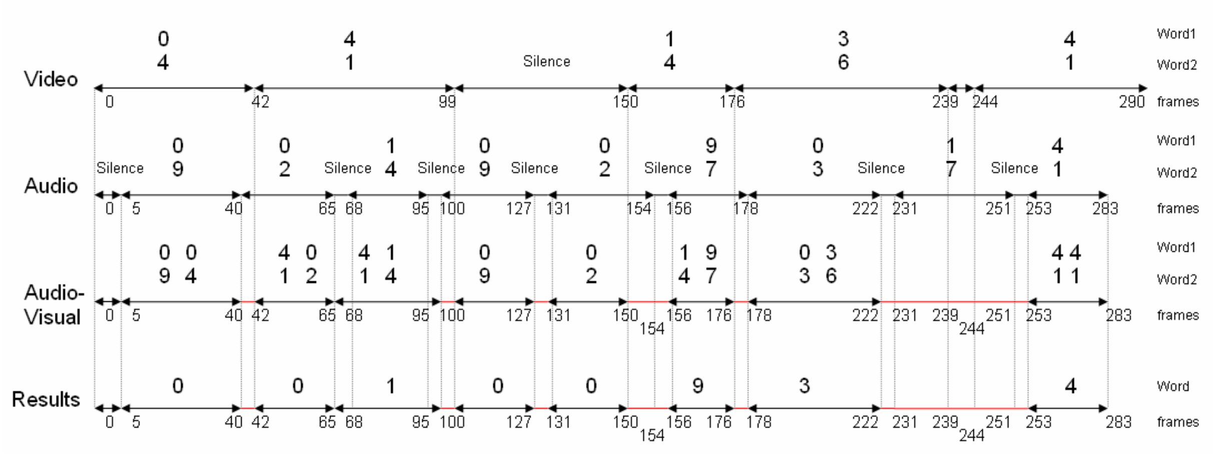

Figure 46: Time chronogram for Late Integration for 0 dB Car Noise.

In Figure 47 the results of Audio, Video and Late Integration are shown for different SNR. We can see how Late Integration is improving the results of audio for bad SNR without any improvement for good SNR. It is also important to point out that for SNR lower than 10 dB results obtained using only the video information outperform the results of Late Integration, which implies that this integration strategy is not the optimal because pure modalities are providing better results.

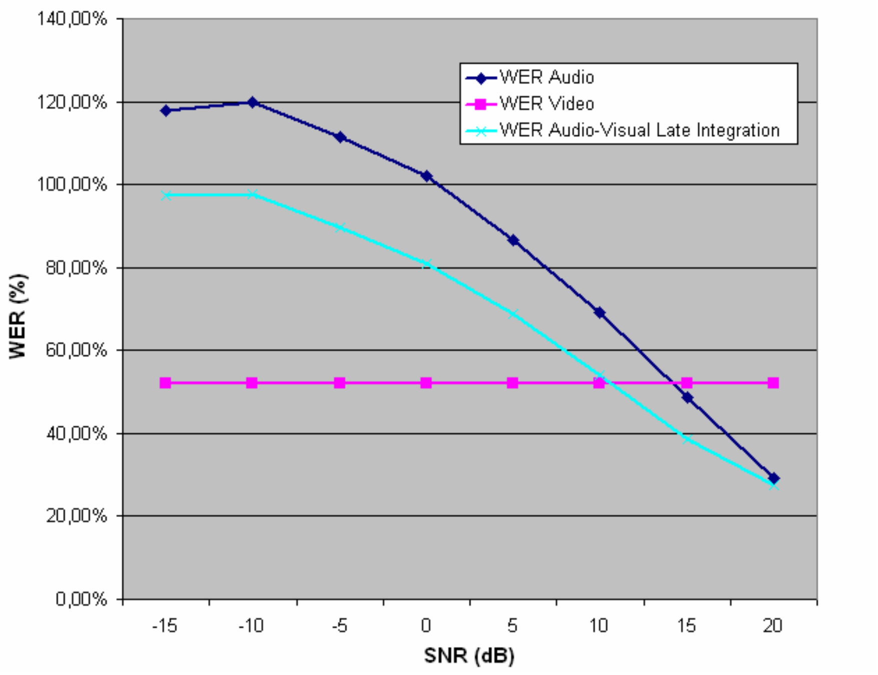

Figure 47: Word Error Rate for Audio, Video and Late Integration for different SNR in Audio Channel.

5.3.3 Hybrid Integration or Model Fusion

Hybrid integration is a solution between the two previous ones. In this solution the integration is going to take place inside the acoustic and visual models. There are different algorithms using this kind of hybrid integration. The most important ones are multi-stream and coupled HMM. Both of them are going to be discussed.

Multi-stream HMM

Multi-stream has been considered in audio-only ASR. Different streams are used for the energy, audio features, MFCC static features, as well as their first and possibly second derivatives as described in [Hernando, 1997]. In Lip Reading our two streams are audio and visual information [Potamianos et al., 2003]. In the multi-stream solution the integration of audio and video information takes place as an integration of the emission probabilities of both modalities, as it can be seen in Figure 48.

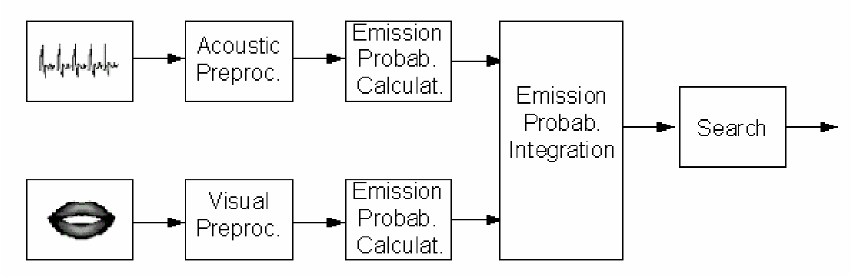
Acoustic Preproc.
Emission Probab. Calculat.
Visual Preproc.
Emission Probab. Calculat.
Emission Probab. Integration
Search

Figure 48: General Diagram for Hybrid Integration

In its general form, the class conditional observation likelihood of the Multi-stream HMM is the product of the observation likelihoods of its single-streams, raised to appropriate stream exponents that capture the reliability of each modality, or equivalently, the confidence of each single-stream classifier.

$$b_{W_j,i}\left(O^{AV}\right) = P(O^{AV} \mid q_t = i, W_j) = \prod_{S\in(A,V)} \left[b^S_{W_j,i}\left(O^S\right)\right]^{\alpha_S} = \prod_{S\in(A,V)} \left[P(O^S \mid q_t = i, W_j)\right]^{\alpha_S} \tag{97}$$

By defining the weighting as exponential it will imply a linear combination in the log-likelihood domain. The exponents (weights) are non-negative and in general are a function of the modality S, the state i and the time instant t. The other parameters of the HMM are defined in the same way as in equation (73).

Multi-stream provides several advantages. Firstly, it allows a different weighting for both information modalities. The audio and the visual channel will not always convey the same quantity of information. This is an advantage in comparison with the early integration where this weighting was not possible. Secondly, the architecture of this solution, as it can be seen in Figure 48, where only one Viterbi decoding process (search) must be performed, will save resources for an embedded implementation. As disadvantage, the synchronization between the audio and the visual information is performed at the state level and as it was seen in [Grant et al., 2001] audio and video information are not always synchronized in the nature (semantic level asynchrony). Sometimes, the visual information will appear before the audio information. The Multi-stream integration will not model this asynchrony and of course it will not model asynchrony between the modalities in a signal level, which actually was already solved in paragraph 5.1. Early integration assumes dependence between both streams of features. If we assume independency, the expression of Multi-stream (97) with both exponents set to one, can be derived from the expression of early integration (89).

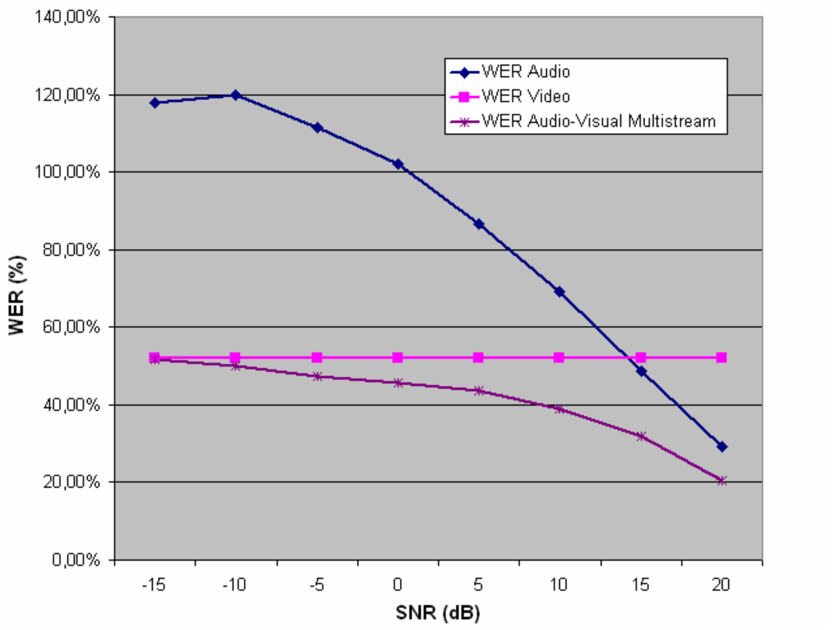

Figure 49: Word Error Rate for Audio and Multi-stream audio-visual integration for different SNR in Audio Channel.

Coupled HMM

In this integration technology, also called product HMM, two parallel HMMs are working at the same time [Amarnag et al., 2003], [Nefian et al., 2002].

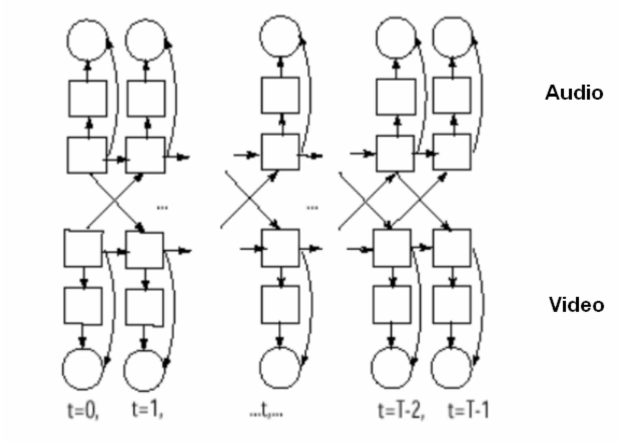

Figure 50: Coupled HMM Model [Nefian et al., 2002]

Both HMM are related by the transition probabilities that will connect both of them as it can be see Figure 50.

$$\alpha^{A}_{i|j,k} = P(q^{A}_{t} = i \mid q^{A}_{t-1} = j, q^{V}_{t-1} = k) \quad (98)$$

$$\alpha^{V}_{i|j,k} = P(q^{V}_{t} = i \mid q^{V}_{t-1} = j, q^{A}_{t-1} = k) \quad (99)$$

In equation (98) and (99) the transition probabilities that connect both HMM are shown. As it can be seen for this implementation transition probabilities play an important role. In our HMM embedded implementation the transition probabilities were not estimated in the training process but they have a constant value because they were not very relevant and with this approximation the model parameters were significantly simplified. If we would decide to use the couple HMM the transition probabilities should be estimated and they should not have a constant value incrementing the complexity of the system.

This integration structure is performed at a phoneme level allowing an asynchrony between the different channels on the states of a phoneme. This would be useful when the audio and visual signals are not synchronized. When we are working with audio and video signals correctly synchronized, this kind of algorithm does not provide any improvement in comparison with the conventional Multi-stream, it just models the inherent asynchrony of both streams. Furthermore, it complicates the structure of the HMM that will not be any longer a Bakis topology but a bidimensional HMM. Supposing a modeling of each phoneme with 3 states for the audio and for the visual features the coupled HMM of this phoneme would have a total of 3x3 states (matrix of states) where the diagonal states would be the same as in the Multi-stream solution. The states out of diagonal would model the emission with a delay of 1, 2 or 3 states between audio and video. Assuming synchrony between audio and video information at the signal level, the only advantage of the coupled HMM is that it can be used to model the semantic asynchrony between audio and video. The structure of a Couple HMM complicates too much the implementation for an embedded device and the improvement is not clear for signal synchronized channels. This is the reason why we have decided not to implement this integration strategy.

5.4 Audio-Visual Integration for Embedded Devices

The most important integration strategies that can be found in the literature of audio-visual speech recognition have been summarized in paragraph 5.3. Now we are going to compare all these integration possibilities and find the most appropriate solution for an embedded implementation. After this, a description of our implementation, audio-visual HMM structure and training procedure will be provided.

5.4.1 Integration Strategy Discussion

In Table 6 a summary with the main advantages and disadvantages of the different integration strategies is presented.

	Early Integration	**Late Integration**	**Hybrid Integration: Multi-stream**	**Hybrid Integration: coupled HMM**
Advantages	Audio-Visual Dependence.	Audio and Visual Weightings.	Audio and Visual Weightings.	Asynchrony Modeling
Disadvantages	No Weighting for Audio and Video	No adequate for Continuous ASR	Require Channel Synchronization	Bi-dimensional HMM Structure
Implementation Remarks	No use of only Audio DB	Two Decoding Processes	Only 1 Decoding Use only Audio DB	No use of only Audio DB

Table 6: Comparison of the Different Audio-Visual Integration Strategies

In the first two rows of Table 6 the advantages and disadvantages of the different integration technologies are summarized without taking into account embedded implementation issues.

Early integration does not allow the system to provide channel weighting. This is very important because it is well known that in noise free situations the audio channel carries much more information than the video stream. In such a situation it would be desirable to provide to the audio channel more importance than to the video channel. Moreover, early integration is the only solution that integrates in the same state audio and video features, modeling its dependences. Late integration will allow the system to use different weightings for each channel. But as we have seen in 5.3.2, it is not the best solution for continuous speech recognition as the temporal boundaries of each word will not be synchronized in each modality producing insertions. Multi-stream allows different weightings for the channels but it requires a signal level synchronization between modalities, due to the fact that each model will be trained separately but the recognition will be performed at the same time. This synchrony is not a requirement for Couple HMMs. They are able to model the asynchrony between both channels, but at expense of increasing the complexity of the HMM using a bidimensional structure.

In order to choose one integration strategy for our embedded implementation several additional aspects must be taken into account:

- The Lip Reading system should outperform the results of our conventional speech recognition system and it can never generate worse results than our acoustic speech recognition.
- A system must be built in such a way that it allows independently training of audio and video models in order to reuse the existing well-trained only audio databases.
- Computing complexity and memory consumption are still an issue for an embedded implementation
- In our implementation we are assuming synchronization between the audio and the video streaming.

Taking these considerations into account the third row of Table 6 has been filled out with implementation remarks for every solution:

- With Early integration the only audio databases can not be used because the models must be trained together with audio and visual information.
- Late Integration needs the use of two different and independent recognition processes. This implies that the Viterbi decoding must be performed twice, which is a problem for embedded implementation especially for large vocabulary tasks.
- Couple HMM, as well as Early Integration, must be trained with audio-visual databases and the only audio databases cannot be used. Additionally, another disadvantage emerge as the bidimensional HMM structure is much more complicate

as the Bakis HMM structure used for our embedded implementation. These are the reasons why this solution was not implemented.

- For multi-stream only audio databases can be used to train the audio part of the HMMs, and the audio-visual databases will be used to train the visual part of the HMMs, as it will show in 5.4.3. Moreover, as an important issue for the implementation in an embedded device multi-stream needs only one Viterbi decoding, which saves CPU resources.

One of the main points to take a decision is the recognition performance of each integration strategy. In Figure 51 a summary of the three implemented integration strategies (Early, Late and Multi-stream) is presented for different SNR in the audio channel. As we have seen Early integration performs worse results than each of the pure modalities, Late integration outperforms the results of Early Integration for good SNR but not for bad SNR. Multi-stream outperforms the results of the other integration strategies and even the results of pure modality for very good or very bad SNR. In terms of *WER* multi-stream is clearly the best integration strategy.

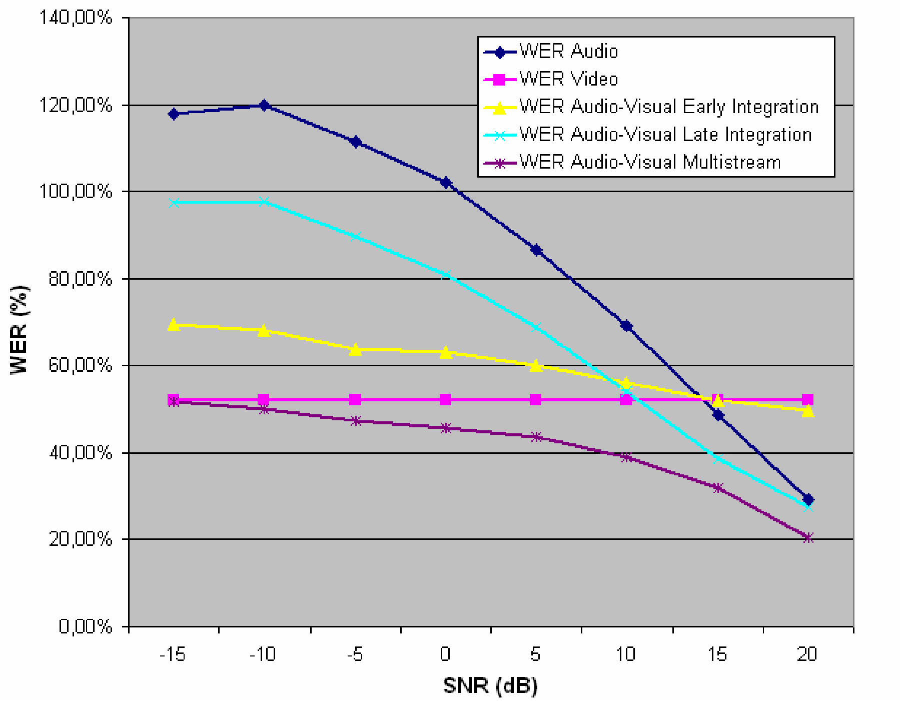

Figure 51: Word Error Rate for Audio, Video Early, Late Integration and Multi-stream audio-visual integration for different SNR in Audio Channel.

Taking into account the advantages and disadvantages of each solution, the implementation consideration as well as the *WER* results we have decided to use multi-stream as the integration strategy of our system. In the next chapter a profusely description of its implementation will be provided.

5.4.2 Multi-stream Audio-Visual HMM Structure

As we have seen in the general formulation of the multi-stream in equation (97), the emission probabilities of audio and video channels are integrated by using products and powers for the weightings. In our embedded implementation we are not longer working with probabilities, we are working in the log-likelihood domain. Assuming the approximations of section 5.2.2 we are going to integrate the neg-log emission probabilities (84) of the audio and the video channel. As it can be seen in equation (85) this value is basically constituted by the Euclidean distance. In our multi-stream implementation the values of the neg-log emission scores are going to be linearly integrated by using the equation:

$$B_i^{AV}(O) = \alpha_A \cdot B_i^A(O) + \alpha_V \cdot B_i^V(O) \tag{100}$$

where the weightings for both channels fulfill the condition:

$$\alpha_A = 1 - \alpha_V \tag{101}$$

In our implementation the modes that describe the audio and visual HMM are trained independently in such a way that the audio modes will be obtained using the audio databases and the video modes will be calculated with the audio-visual databases. The recognition task in this work is “continuous digits”. Word specific HMMs will be used, this means that we are not going to use phonemes but pseudo-phonemes. The number of pseudo-phonemes for each word is optimized for the audio recognition. Our extended visual HMM will have the same structure as the audio one. All parameters that play a role in both streams (transitions probabilities and HMM structure) have been optimized for the audio stream as we want to provide the same results as with the conventional recognition when the visual information is not available. The number of states used is 181. The audio emission scores have been modeled with 1192 Gaussians and for the video a total of 857 Gaussians have been employed.

5.4.3 Audio-Visual HMM Training

The training process of a HMM consists in the estimation of the HMM parameters $\{a_i, b_i, \pi_i\}$. In our implementation, as it was seen in section 5.2.2, only the emission scores $\{B_i^A(O), B_i^V(O)\}$ must be estimated. In the audio-visual multi-stream two kinds of emission probabilities are associated to every state, one for the visual and one for the acoustic channel. In this work all HMM parameters have been obtained using the Maximum-Likelihood-Training algorithm [Bauer, 2001] which optimize the HMM parameters in order to maximize the probability generated by a known training sequence.

We are designing an audio-visual recognition system and we assume that the main stream of information comes from the acoustic channel; for this reason the system is called auxiliary Lip Reading, and the audio-visual HMM will be based on a previously computed acoustic HMM. As it was seen in section 5.4.1 the integration of audio and video features is going to be performed by using the Multi-stream. The HMM will have the structure of the audio HMM, in fact the audio part of the HMM is not going to be retrained, in the final system we are going to use the acoustic emission probabilities obtained from the acoustic databases, in such a way that for the acoustic part we are going to use knowledge obtained from the conventional training (audio HMM), with the advantage that the conventional audio databases are much larger than the audio-visual databases. The quality of the audio part of the HMM is maintained with this approach, although a small audio-visual database is used to train the visual part of the HMM. In fact each audio HMM is going to be filled with visual information related to the corresponding phoneme. For each state of each phoneme (audio) we are going to get the visual features and with them the emission probabilities of the visual HMM are obtained. The training process can be summarized in the next way:

1. - The acoustic channel (clean audio) is going to be used to obtain a transcription of what is said and to label the visual frames according to the audio information. The acoustic recognition system is going to run giving timing information for every pseudo phoneme, that is to say in a label file the sequence of decoded states is going to be saved together with the exactly time information. For this time labeling the Forced Viterbi decoding is applied on the audio part of the audio-visual database and the previous audio HMM is used for decoding.

2. - In a further step the visual features for every frame are going to be extracted, with this information and knowing the corresponding audio HMM state from the previous label file, the emission probabilities functions of the visual HMM states associated to each acoustic state is going to be constructed. For this estimation the Maximum Likelihood Training [Bauer, 2001] is used. In this way we are going to have two different emission probabilities for every state,

one from the visual channel and one from the acoustic. These steps can be shown in Figure 52

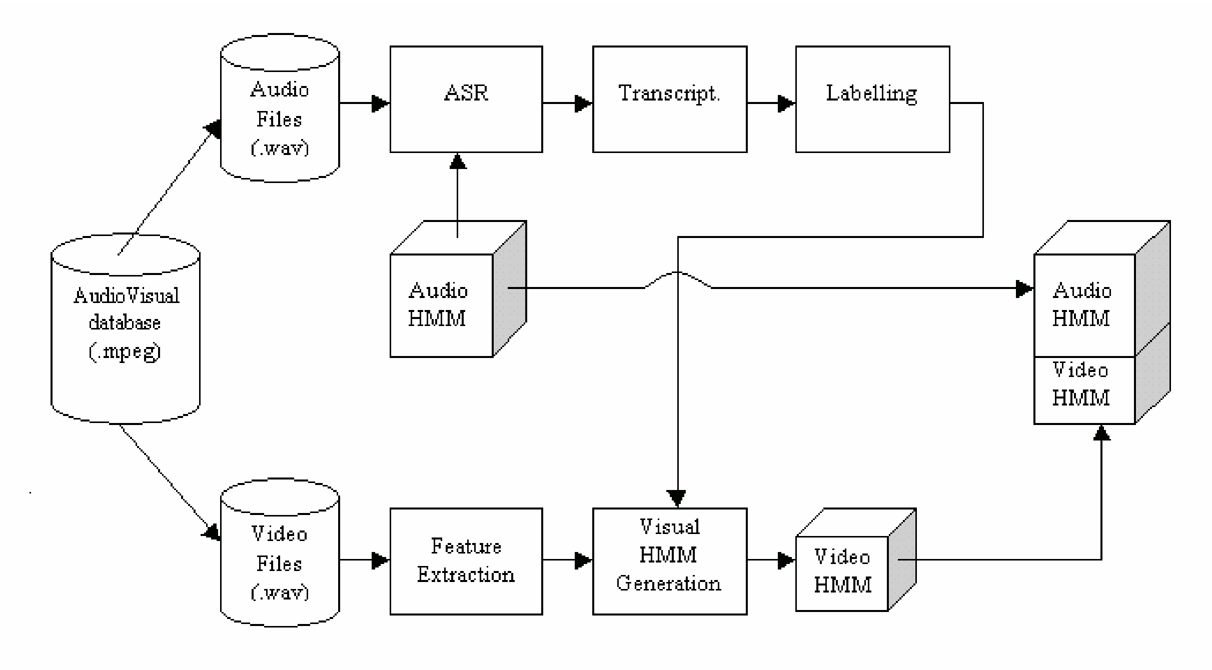

Figure 52: Diagram of the Visual Training using existing audio HMM

This solution solves the problem of the lack of large enough audio-visual transcribed and labeled databases. With this solution any clean speech audio-visual files (without labeling) could be used to perform the training. Moreover, the audio HMM that will be used at the end will be the audio HMM that is already used for the conventional speech recognition.

The acquisition of suitable databases is an important problem, because they must be generated, transcribed and labeled. For this reason databases are quite expensive and as we have said up to now there is not an available large enough audio-visual database. And this would be necessary for the training of a general purpose speaker independent Lip Reading system. This kind of training procedure will be very interesting, having in mind that the problem of database arise for every new language where the speech recognition must be used. If the audio-visual speech recognition systems come to commercial solutions for every new language a database should be collected. The huge effort undertaken by U.S. government agencies or by the European Languages Resource Association (ELRA) [ELRA, 2006] to collect only audio database has taken many years and it was quite expensive. This solution will make the acquisition of new training material for the new promising audio-visual speech recognition systems easy.

5.4.4 Channel Weightings

In equation (100) the visual and acoustic information is being weighted by the coefficients α^V, α^A. These coefficients are dependent on the channel but they can be also dependent on the state where they are applied α_i^V, α_i^A. In this way two different kind of information can be introduced:

1. Channel reliability information: when the acoustic signal is corrupted more importance will be given to the visual information by making α^Avery small, and vice versa. When the audio channel is not corrupted it is demonstrated that it conveys more discriminative information than the visual one, so it is expected that in such situations α^A will be higher.
2. Phoneme discriminative information: it is a fact that the recognition of some phonemes relies more on the audio than on the visual information, and vice versa. It is well know that some phonemes are acoustically very similar but can be visually very good distinguished (e.g. /m/ and /n/) and other phonemes are better discriminated by the acoustic information. This fact will be used to give more importance to the source information that is more critical on distinguishing every single phoneme.

In our research the channel reliability is going to be used to find the optimal weighting. State and phoneme discrimination will not be beyond the scope of this work. We are going to find empirically the optimal weighting for the audio and visual channels for different kind of noises and for different noise levels.

$$\alpha_A = f(SNR) = \arg\min_{\alpha_A}\{WER(\alpha_A, SNR)\} \tag{102}$$

The optimal audio and video weightings are not only dependent on the level of noise but also on the nature of this noise. We found empirically the optimal weightings for three different types of noises:

- One variable tone in the lower frequencies where speech is located 40 – 4000 Hz
- Interfering talker, a different speaker is talking in background at the same time as the user wants to perform the recognition.
- Non-stationary car noise, this is not only the monotone motor noise but accelerations are performed, clicks of the indicators etc.

For a wide range of SNR and for three different kind of noises recognition experiments have been carried out for the whole range of weights (0%, 10%, …, 90%, 100%). The results of these experiments are showed in Figure 53, Figure 54 and Figure 55. According to these results Table 7 is written where all the optimal weightings are summarized.

In these three different scenarios the visual information has not been changed, but depending on the quantity of noise and on the nature of these noises the use of visual information is more or less helpful. First of all we can see how the optimal weightings when there is a lot of audio noise give always more importance to the video channel than to the audio one, but of course when there is almost no audio noise the audio channel conveys more information as the video channel and therefore audio weightings are higher. Other differences between the behaviour for different noises can be pointed out, for example for the interfering talker in the range of $5 < \text{SNR} \le 10$ dB we can see how the optimal video weighting for video is 30% meanwhile for the other two types of noises this weighting is reduced to 10%. We can observe how the voice noise is more negative for the recognition and in this situation an improvement can be obtained by giving more importance to video information.

For an implementation in a mobile phone or in a car environment the kind of usual noise must be studied. Different profiles can be defined for different noise situations as it nowadays occurs for the coders/decoders in mobile phones. Each of these profiles would correspond to a part of Table 7. The user would select manually the environment where he or she is using the device. More complex solutions could take advantage of SNR estimators and find automatically the optimal weighting.

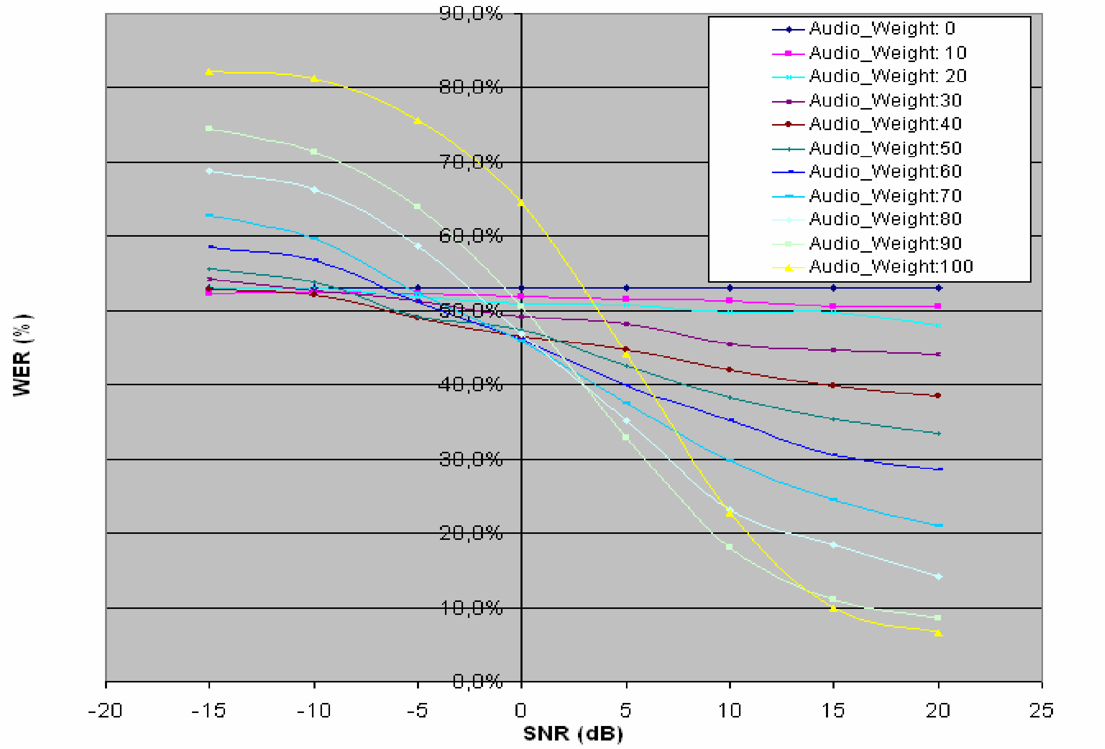

Figure 53: WER for different Audio and Visual Weightings in different Acoustic SNR and for a variable Tone Noise

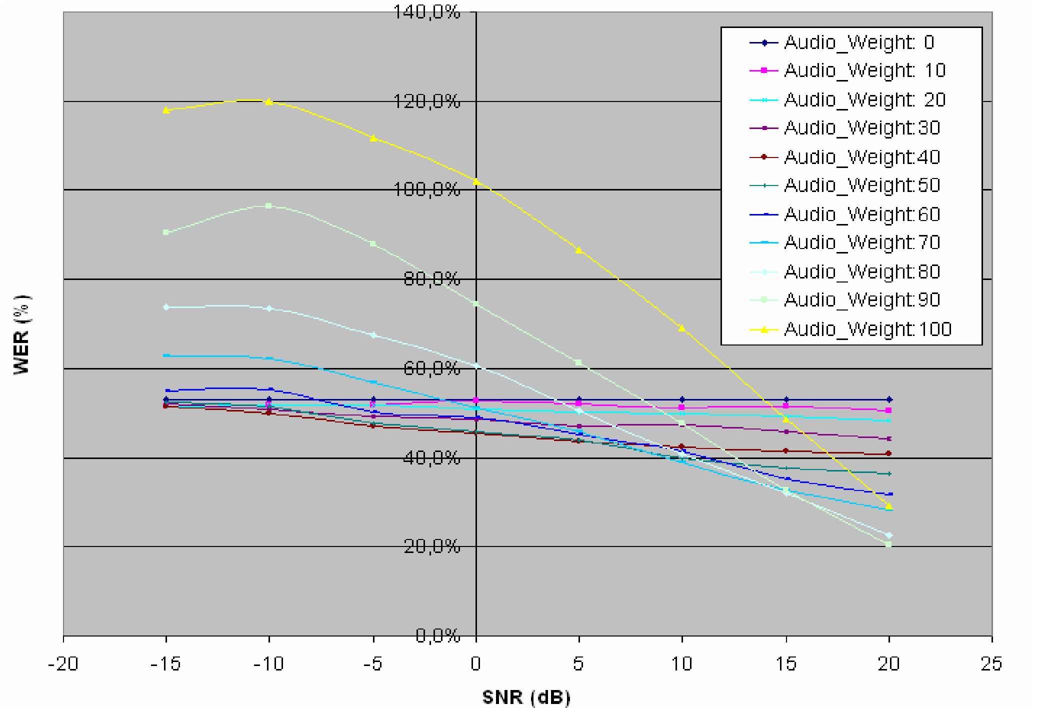

Figure 54: WER for different Audio and Visual Weightings in different Acoustic SNR and for Interfering Talker

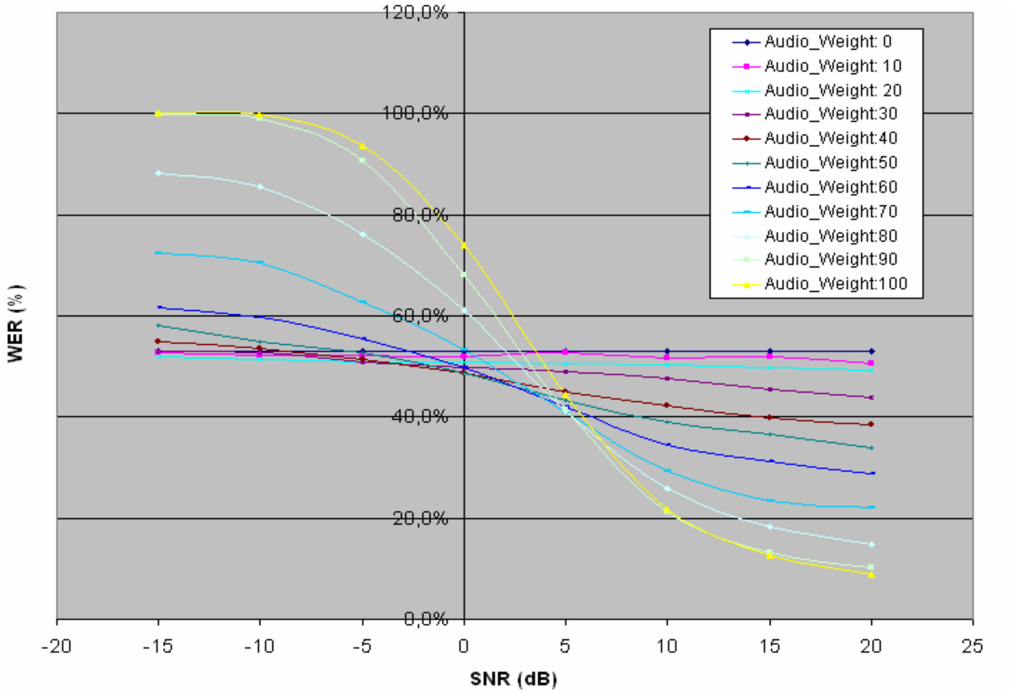

Figure 55: WER for different Audio and Visual Weightings in different Acoustic SNR and for a Non-Stationary Car Noise

	SNR ≤ -15 (dB)	*-15 < SNR ≤ -5 (dB)*	*-5 < SNR ≤ 0 (dB)*	*0 < SNR ≤ 5 (dB)*	*5 < SNR ≤ 10 (dB)*	*10 < SNR ≤ 15 (dB)*	*15 < SNR (dB)*
Optimal Weightings Tone Noise	*V = 90%* *A = 10%*	*V = 60%* *A = 40%*	*V = 60%* *A = 40%*	*V = 10%* *A = 90%*	*V = 10%* *A = 90%*	*V = 0%* *A =100%*	*V = 0%* *A =100%*
Optimal Weightings Auto Noise	*V = 80%* *A = 20%*	*V = 80%* *A = 20%*	*V = 60%* *A = 40%*	*V = 30%* *A = 70%*	*V = 10%* *A = 90%*	*V = 0%* *A =100%*	*V = 0%* *A =100%*
Optimal Weightings Voice Noise	*V = 60%* *A = 40%*	*V = 60%* *A = 40%*	*V = 60%* *A = 40%*	*V = 60%* *A = 40%*	*V = 30%* *A = 70%*	*V = 20%* *A = 80%*	*V = 10%* *A = 90%*

Table 7: Optimal Weightings for Audio and Visual Channel depending on the Noise Kind and Level

Chapter 6

System Evaluation

In this chapter the performance of our audio-visual speech recognition system is evaluated. Recognition results are going to be profusely analyzed. Our system is evaluated using only the visual information and in combination with the audio information. Different types of acoustic noise and noise levels were used in our experiments. A comparison between Lip Reading and other kind of conventional Noise Reduction systems is provided; furthermore, a combination of Lip Reading with spectral subtraction and Wiener filtering is evaluated. The robustness of our system against visual noise was also checked. Due to the focus of this work is the embedded systems the requirements of our algorithm constitute an important aspect to evaluate the feasibility of the system, this is the reason why they are estimated.

For all the experiments summarized in this chapter the system explained in the last chapters has been used:

- Our Lip Finding and Tracking algorithm shown in 3.1.2 has been applied. It is a contour-based system that is able to work properly in embedded devices.
- The audio feature extraction has been implemented using the algorithms explained in 4.1. As visual feature extraction the DCT method shown in 4.2.3 has been used as it outperforms the ASM method, see Figure 37.
- Finally for audio-visual integration the multi-stream strategy presented in 5.4 has been applied using the optimal coefficients of Table 7.

All results, with the exception of the visual speaker dependent results of Table 9, have been obtained with the CUAVE Database. Our recognition task is continuous digits speaker independent. For speaker dependent test experiments a database recorded at Siemens

Facilities has been used. The evaluation of the recognition rates have been made according to the definitions given in section 2.5.

6.1 Visual Recognition Results

In this section we are going to study the recognition results obtained using only the visual information. First of all, as it was seen in chapter 4, ten coefficients from the LDA were used to train the visual models and perform the recognition. The selection of these first ten coefficients was not casual. First of all, as it was seen in 4.1.5, the LDA provides a set of coefficients ordered by its discriminative power, this means that the first coefficient will be more important than the second one. This simplifies the process to select the set of coefficients that should be used for the recognition. We have to find the number of coefficients that minimizes the *WER.* For this purpose different HMMs have been trained with different number of coefficients. For each HMM a test has been performed, the results are summarized in Figure 56. We can see that the optimal number of coefficients for the visual modeling will be 10. The audio modeling has been trained using 24 coefficients.

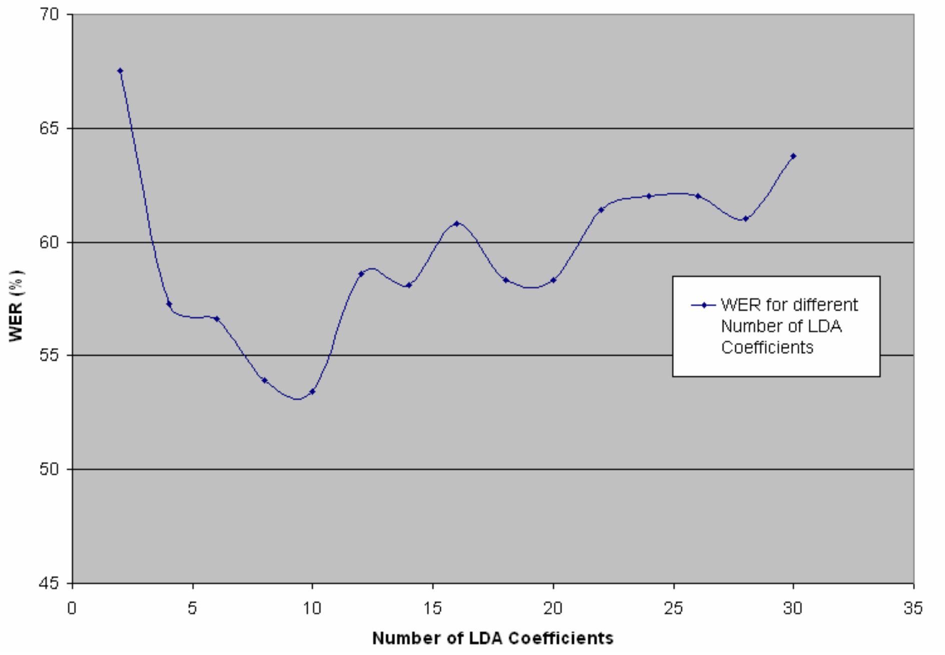

Figure 56: WER for different Number of LDA Coefficients

Visual HMM has been trained using a digits specific word model with 181 states, for speaker independent recognition. Maximum Likelihood training algorithm has been used to extract the HMM parameters. The visual model is described by a total 856 Gaussians.

Using only the visual information we have obtained a total *WER* of 53.0%, where 12.0% are insertions, 14.8% are deletions and 26.1% are substitutions. In Table 8 we can see how substitutions are distributed; a confusion matrix is shown where in the vertical columns the pronounced word is presented and in the horizontal row the recognized word appears. As given value the percentage of the substitutions in relation with the whole substitutions is shown. We can see that the most confused pair using the visual information are “six-nine”, “three-two”, “three-five” and “four-one” these pairs of digits are visually quite similar. For the last one the matrix is quite symmetric with respect to its diagonal, this means that not only when “four” is said “one” is recognized but also when “one” is said quite often “four” will be recognized. This characteristic can not be generalized, as for example the substitution pair “nine-six” is much more common than “six-nine”.

		HYPOTHESE									
		Zero	**One**	**Two**	**Three**	**Four**	**Five**	**Six**	**Seven**	**Eight**	**Nine**
REFERENCE	**Zero**	X	0.66	2.66	0.66	3, 33	0.66	2	0	0	1.33
	One	1.33	X	0.66	1.33	2.66	2	1.33	0	0	0
	Two	1.33	3, 33	X	0	0.66	2	0	1.33	0.66	2.66
	Three	0	1.33	4	X	0.66	4	2.66	0	0	2
	Four	1.33	3, 33	1.33	0.66	X	0	0.66	0.66	1.33	1.33
	Five	0	0.66	1.33	0	0	X	0	0	1.33	0.66
	Six	2	3, 33	0.66	2	0	0	X	3, 33	2	5.33
	Seven	2	0.66	0	1.33	0	0	0	X	0.66	0.66
	Eight	0	0.66	0.66	0.66	1.33	3, 33	1.33	1.33	X	0.66
	Nine	0	0	0	0	0	1.33	1.33	0	1.33	X

Table 8: Confusion Matrix for Visual Recognition (% of all substitutions)

We want to know the performance of our visual recognition system in a speaker dependent task. In this case, the HMM will be trained using information of only one speaker, the same speaker will be used for test. Different utterances for training and test have been used. For this experiment we have recorded a database with one speaker because the CUAVE Database has not enough material from one speaker to train and test a model. With this database we have trained a HMM digit model using 181 states and 437 Gaussians. The results are summarized in Table 9 where they can be compared with the Speaker Independent results. In these results we have used only the visual information, no integration of audio and video has been performed. We can see how the *WER* has been reduced by 16% absolute, being this improvement mainly due to a reduction of the deletions, although insertions and substitutions have also been improved [Guitarte et al., 2005b]. The improvement of the results by using speaker dependent systems is also observed in conventional speech recognition; there are discriminative speaker dependent features that do not appear in the speaker independent set. These results encourage the investigations on Lip Reading as it seems that we are still not using all the visual information available in the image. A different set of visual features could be investigated in order to extract more speaker independent information that is still not in our set of features.

	Speaker Dependent	**Speaker Independent**
Errors	37.00%	53.0%
Insertions	9.00%	12.0%
Deletions	4.00%	14.8%
Substitutions	24.00%	26.1%

Table 9: Recognition results for Speaker Dependent and Speaker Independent Visual Recognition

In the speaker dependent system the half of the resources have been used to save the information of the HMM (437 vs. 856 Gaussians) but a better description has been obtained as it must model only one speaker, the same as used for test.

6.2 Recognition with Different Levels and Types of Audio Noise

One of the motivations for using Lip Reading is to provide more information in order to improve the results of the recognition in noise degraded environments. This is the reason why we are going to study the performance of our Lip Reading system with different types and levels of noises.

We are going to study three different types of noises that can be found in different application environments: variable tone, car noise and speech interference (interfering talker). For each noise we are going to show the performance of the audio, video and audio-visual system for different levels of noise energy using the Signal to Noise Ratio (SNR). Continuous digit speaker independent recognition is the common task for all experiments. The audio noise has been artificially added to the original signal with a certain level of energy for the different SNR ranges. For this work the noise is considered as additive; the impulse response of the recording room has not been taken into account and of course with this additive noise the Lombard Effect [Junqua, 1993] can not be modeled. For each environment we are going to study the characteristics of the noise by taking use of its spectrogram. In Figure 57 we can see the spectrogram of our speech signal.

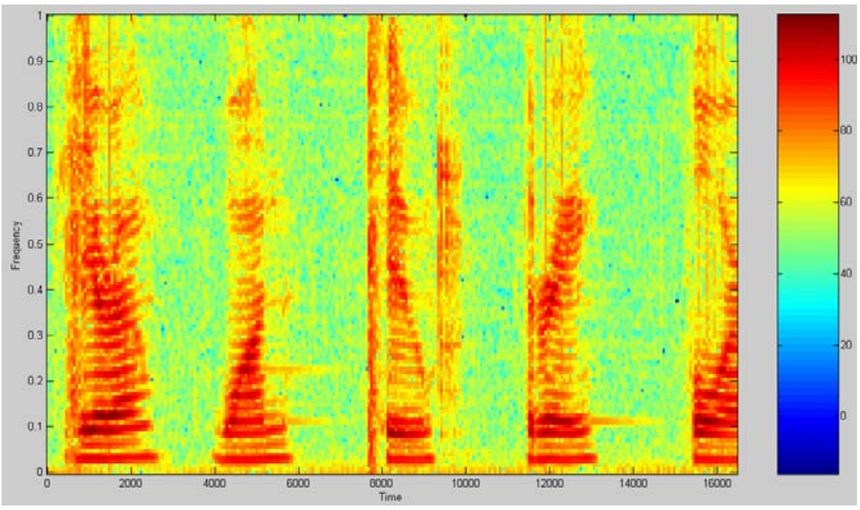

Figure 57: Spectrogram of a Voice Signal

In a spectrogram the evolution in time of the different frequencies of the signal can be observed, in the x-axis the time evolution is given and in the y-axis the frequency components of the signal are drawn (they are normalized to the half of the sampling frequency: 1 means 4000 Hz). The notation showing the energy values of each frequency can be read in the scale plotted on the right; red colours mean high energy values and low energy values are plotted with blue tonalities. All spectrograms shown in this chapter have been obtained using a hamming window of 256 samples with a shift of 128 samples.

6.2.1 Variable Tone Noise

First of all, we are going to study the effects on the recognition of a variable tone noise. This noise can be found in industrial environments. As it can be seen by observing Figure 57 and Figure 58 the frequency characteristics of the variable tone are similar to the speech. The presence of this noise disturbs significantly the recognition performance, as it can be seen in Table 10, Table 11 and Figure 59.

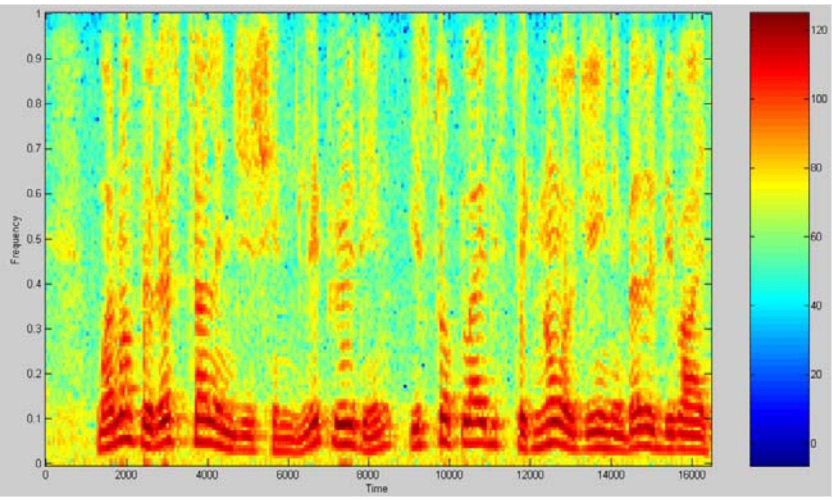

Figure 58: Spectrogram of Variable Tone Noise

The influence of this noise in the audio results can be seen as an increment of the substitutions and deletions.

In Variable Tone Noise and for SNR of -15 dB the Word Error Rate raises up to 82.3% with the use of the visual signal in combination with the audio signal and for this degraded environment the improvement is from 30% absolute. We can see how the substitutions and deletions have been almost reduced to the half.

		-15 dB	-10 dB	-5 dB	0 dB	5 dB
Audio	**Errors**	82.30%	81.30%	75.60%	64.50%	44.10%
	Insertions	8.80%	8.40%	10.50%	12.00%	12.30%
	Deletions	28.00%	28.60%	25.80%	19.40%	10.80%
	Substitutions	45.60%	44.20%	39.40%	33.10%	20.90%
Video	**Errors**	53.0%	53.0%	53.0%	53.0%	53.0%
	Insertions	12.0%	12.0%	12.0%	12.0%	12.0%
	Deletions	14.8%	14.8%	14.8%	14.8%	14.8%
	Substitutions	26.1%	26.1%	26.1%	26.1%	26.1%
Audio-Visual	**Errors**	52.30%	52.00%	48.90%	46.40%	32.80%
	Insertions	11.10%	9.40%	8.80%	8.40%	2.20%
	Deletions	15.20%	19.50%	17.80%	15.90%	18.00%
	Substitutions	26.10%	23.10%	22.30%	22.00%	12.70%

Table 10: Recognition Results for Speaker Independent Recognition with Variable Tone Audio Noise for different SNR (-15 dB – 5 dB)

		10 dB	15 dB	20 dB	25 dB	30 dB
Audio	**Errors**	22.80%	10.00%	6.60%	5.20%	4.50%
	Insertions	4.50%	0.30%	0.20%	0.20%	0.20%
	Deletions	5.80%	3.90%	2.50%	1.70%	1.70%
	Substitutions	12.50%	5.80%	3.90%	3.30%	2.70%
Video	**Errors**	53.0%	53.0%	53.0%	53.0%	53.0%
	Insertions	12.0%	12.0%	12.0%	12.0%	12.0%
	Deletions	14.8%	14.8%	14.8%	14.8%	14.8%
	Substitutions	26.1%	26.1%	26.1%	26.1%	26.1%
Audio-Visual	**Errors**	18.10%	10.00%	6.60%	5.20%	4.50%
	Insertions	1.40%	0.30%	0.20%	0.20%	0.20%
	Deletions	8.60%	3.90%	2.50%	1.70%	1.70%
	Substitutions	8.10%	5.80%	3.90%	3.30%	2.70%

Table 11: Recognition Results for Speaker Independent Recognition with Variable Tone Audio Noise for different SNR (10 dB – 30 dB)

Improvements by using the visual information in combination with audio can be seen for SNR lower than 15 dB, up to this value the visual information does not increase the performance of the system. For higher SNR and with this variable noise the best results were obtained by using only the audio information, as it can be seen in Figure 53, the optimal weightings for this SNR range give the total decision criterion to the audio signal.

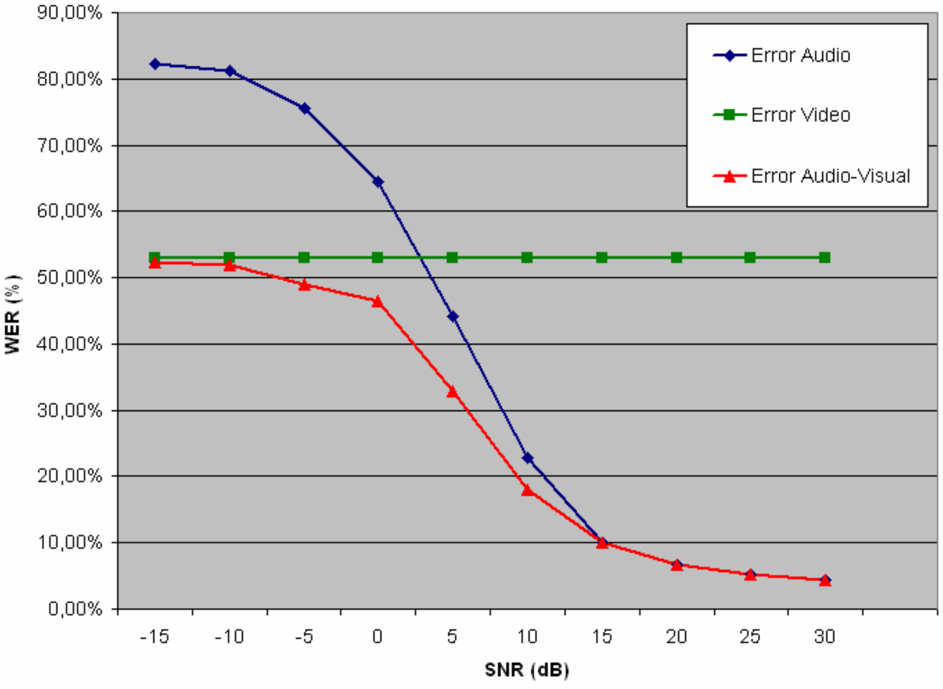

Figure 59: WER for different SNR and Modalities with Variable Tone Audio Noise

6.2.2 Car Noise

A very important scenario for speech recognition systems is the car environment. We are going to study a non-stationary car noise. This car noise has been recorded in a real driving situation and it includes accelerations, clicks from indicators etc.

As it can be seen in Figure 60 there is a broad spread of the energy over all frequencies with a high concentration in frequencies lower than 1000 Hz. The frequency muster of this noise differs more from the speech spectrum than the spectrum of the previous variable tone noise.

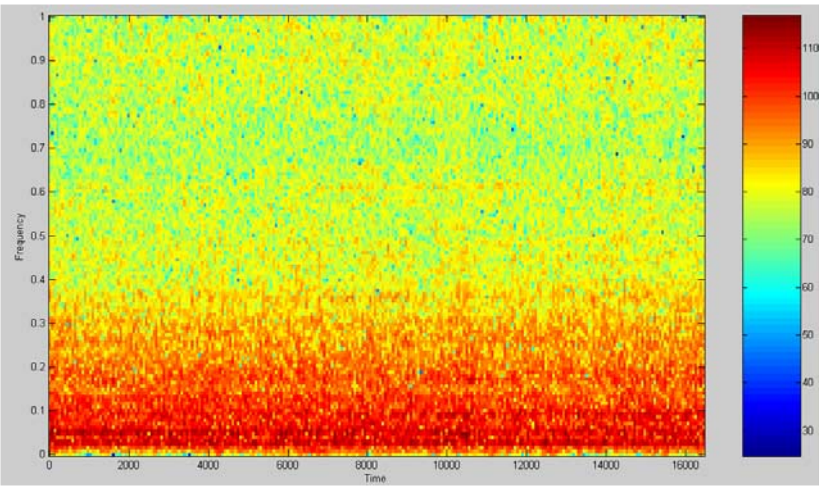

Figure 60: Spectrogram of Non-Stationary Car Noise

When this noise is applied on our system with very bad SNR the deletions increment extremely (for -15 dB 100% deletions are obtained), see Table 12 and Table 13, this is due to the fact that the spectrum distribution of this noise is very different to the one of the speech and the classifier can not match any word. We can observe how for this situation the audio-visual recognition is providing similar results as the audio only for SNR of 5 dB, this implies an improvement of almost 20 dB. The improvement can only be appreciated for degraded SNR, but when the audio signal has a SNR higher than 10 dB no improvement is achieved by using the visual information.

		-15 dB	-10 dB	-5 dB	0 dB	5 dB
Audio	Errors	100.00%	99.80%	93.60%	73.90%	44.40%
	Insertions	0.00%	0.00%	0.00%	0.00%	0.30%
	Deletions	100.00%	99.80%	93.40%	70.80%	35.20%
	Substitutions	0.00%	0.00%	0.20%	3.10%	8.90%
Video	Errors	53.0%	53.0%	53.0%	53.0%	53.0%
	Insertions	12.0%	12.0%	12.0%	12.0%	12.0%
	Deletions	14.8%	14.8%	14.8%	14.8%	14.8%
	Substitutions	26.1%	26.1%	26.1%	26.1%	26.1%
Audio-Visual	Errors	51.90%	51.40%	50.80%	48.60%	40.90%
	Insertions	9.70%	9.20%	9.70%	7.00%	1.10%
	Deletions	18.30%	17.00%	16.70%	19.20%	26.70%
	Substitutions	23.90%	25.20%	24.40%	22.30%	13.10%

Table 12: Recognition Results for Speaker Independent Recognition with Non-Stationary Car Environment Audio Noise for different SNR (-15 dB – 5 dB)

		10 dB	15 dB	20 dB	25 dB	30 dB
Audio	Errors	21.90%	12.70%	8.80%	6.40%	5.20%
	Insertions	0.30%	0.30%	0.20%	0.20%	0.20%
	Deletions	9.50%	4.50%	2.70%	1.70%	1.70%
	Substitutions	12.00%	7.80%	5.90%	4.50%	3.30%
Video	Errors	53.0%	53.0%	53.0%	53.0%	53.0%
	Insertions	12.0%	12.0%	12.0%	12.0%	12.0%
	Deletions	14.8%	14.8%	14.8%	14.8%	14.8%
	Substitutions	26.1%	26.1%	26.1%	26.1%	26.1%
Audio-Visual	Errors	21.30%	12.70%	8.80%	6.40%	5.20%
	Insertions	0.30%	0.30%	0.20%	0.20%	0.20%
	Deletions	13.30%	4.50%	2.70%	1.70%	1.70%
	Substitutions	7.70%	7.80%	5.90%	4.50%	3.30%

Table 13: Recognition Results for Speaker Independent Recognition with Non-Stationary Car Audio Noise for different SNR (10 dB – 30 dB)

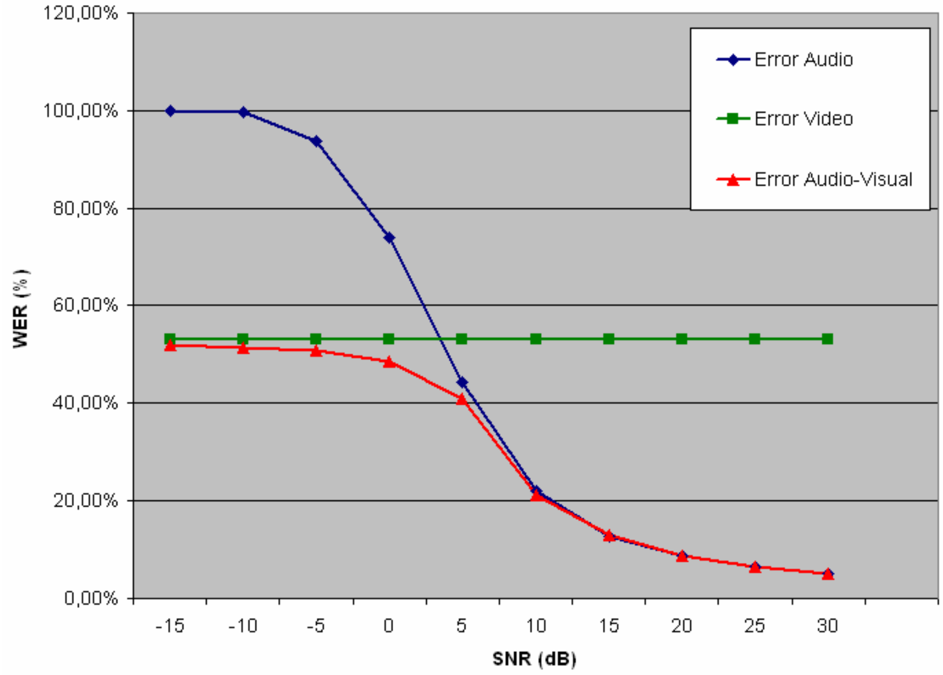

Figure 61: WER for different SNR and Modalities with Non-Stationary Car Noise

6.2.3 Interfering talker

This type of noise appears when one speaker has been spoken at the same time as when the user wants to use the speech recognition. It is a very common source of noise. Under this type of noise our conventional speech recognition system provides very bad results as it is not possible to distinguish between the part of the signal coming from the user and the part of the signal that is interference. For this speech noise only “out of vocabulary words” have been used, this implies that in the interference noise no digits have been said. Both signals are of the same nature and taking into account that we are developing speaker independent systems the speaker identification cannot be used. The spectrums shown in Figure 57 and Figure 62 are not possible to be distinguished.

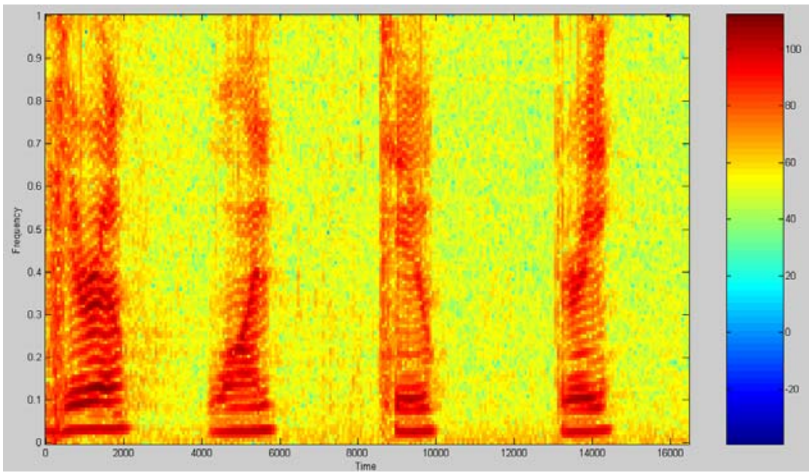

Figure 62: Spectrogram of Interfering Talker

We can see how in this case and for very bad SNR the main part of the errors is due to insertions. The noise is very similar to the trained speech (actually words said are different but the noise is also speech signal). If we compare the Table 12 with Table 14 we can see that for -15 dB of car noise in the conventional speech recognition almost all errors are due to the deletions, while under interfering talker almost all errors are derived by insertions. For this type of noise the use of visual information seems to be very beneficial as for example for -15 dB an improvement of more than 50% is obtained and also for better SNR (15 dB) an absolute improvement of 16.7% is achieved (relative 34%), as it can be seen in Figure 63. But the most interesting point is also that for this environment and for SNR of 25 dB the *WER* is reduced from 14.7% to 9.1% with a very important decrease on the number of insertions. As it can be seen in Table 14 and Table 15 the main improvements are due to the reduction of the insertions for the whole SNR range.

		-15 dB	-10 dB	-5 dB	0 dB	5 dB
Audio	**Errors**	118.00%	119.80%	111.60%	101.90%	86.60%
	Insertions	70.90%	77.30%	73.10%	69.20%	59.70%
	Deletions	6.60%	6.90%	5.00%	4.70%	2.70%
	Substitutions	40.50%	35.60%	33.40%	28.00%	24.20%
Video	**Errors**	53.0%	53.0%	53.0%	53.0%	53.0%
	Insertions	12.0%	12.0%	12.0%	12.0%	12.0%
	Deletions	14.8%	14.8%	14.8%	14.8%	14.8%
	Substitutions	26.1%	26.1%	26.1%	26.1%	26.1%
Audio-Visual	**Errors**	51.60%	50.00%	47.20%	45.50%	43.60%
	Insertions	12.70%	12.20%	11.60%	11.30%	10.80%
	Deletions	14.10%	13.90%	13.00%	11.70%	11.10%
	Substitutions	24.80%	23.90%	22.70%	22.50%	21.70%

Table 14: Recognition Results for Speaker Independent Recognition with Interfering Talker for different SNR (-15 dB – 5 dB)

		10 dB	15 dB	20 dB	25 dB	30 dB
Audio	**Errors**	69.10%	48.60%	29.10%	14.70%	5.60%
	Insertions	49.70%	35.50%	19.50%	7.70%	0.80%
	Deletions	2.20%	2.00%	1.70%	1.70%	1.60%
	Substitutions	17.20%	11.10%	7.80%	5.30%	3.30%
Video	**Errors**	53.0%	53.0%	53.0%	53.0%	53.0%
	Insertions	12.0%	12.0%	12.0%	12.0%	12.0%
	Deletions	14.8%	14.8%	14.8%	14.8%	14.8%
	Substitutions	26.1%	26.1%	26.1%	26.1%	26.1%
Audio-Visual	**Errors**	38.80%	31.90%	20.50%	9.10%	5.60%
	Insertions	17.20%	15.90%	11.60%	2.80%	0.80%
	Deletions	5.60%	4.10%	3.30%	2.80%	1.60%
	Substitutions	15.90%	11.90%	5.60%	3.40%	3.30%

Table 15: Recognition Results for Speaker Independent Recognition with Interfering Talker for different SNR (10 dB – 30 dB)

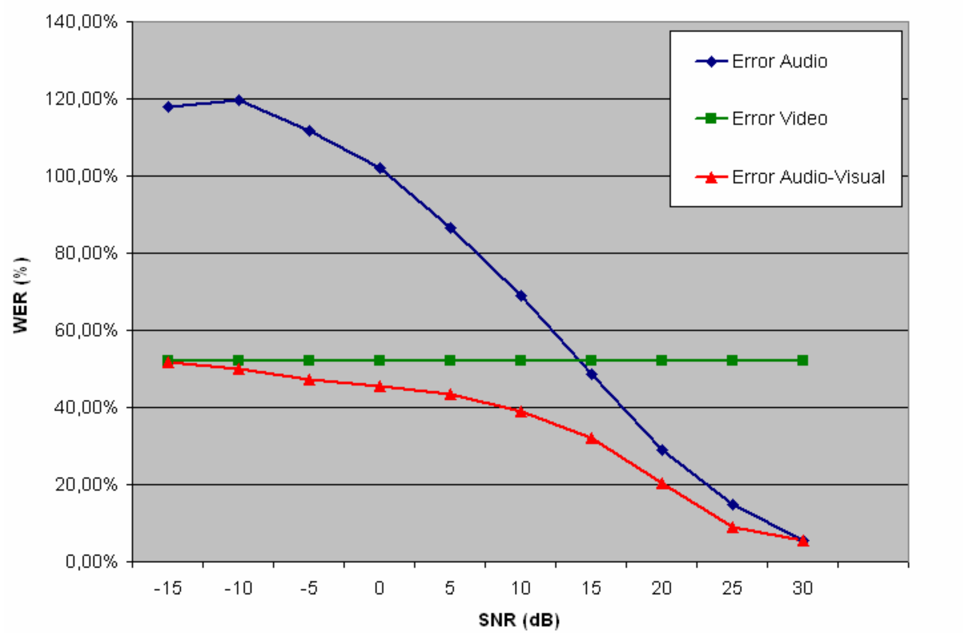

Figure 63: WER for different SNR and Modalities with Interfering Talker

6.3 Lip Reading Recognition Results in Comparison and in Combination with Noise Reduction

In section 4.1.2 two conventional techniques to reduce the bad influence of noise have been explained: spectral subtraction and Wiener filtering, both techniques work in the frequency domain trying to separate the noisy frequency components from the speech. As it was shown in Figure 59, Figure 61 and Figure 63 Lip Reading has provided improvements especially for bad audio SNR. Visual information is useful when the acoustic information is degraded. This is the reason why Lip Reading can be considered as another type of Noise Reduction [Guitarte et al., 2005b]. We are going to compare Lip Reading with spectral subtraction and Wiener filtering for the different types of noises explained in the last section.

In Figure 64 the results for a variable tone are shown. As it can be seen, the use of either spectral subtraction or Wiener filtering is not advantageous for this type of noise. If we compare the spectral distribution of the noise (Figure 58) and of the signal (Figure 57) it can be pointed out that they are very similar and therefore techniques that try to separate both signals in the frequency domain are not going to success. Actually, the use of Wiener filtering or spectral subtraction in this noise scenario is increasing drastically the insertions. It is better not to use a Noise Reduction than using the spectral subtraction or Wiener filtering. For this noise Lip Reading outperforms the results of the others for all SNR.

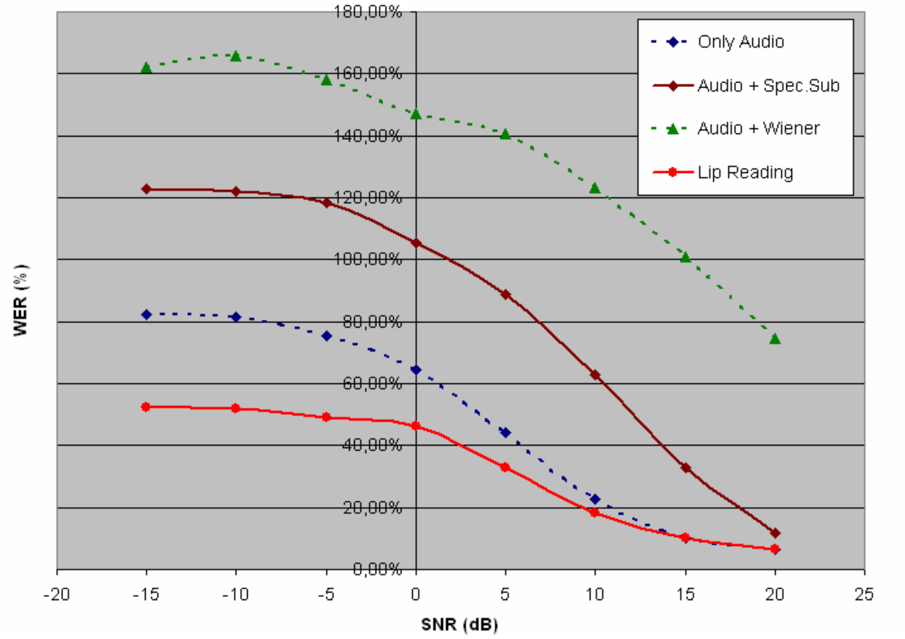

Figure 64: WER for different Noise Reduction Techniques with Variable Tone Noise

When a car noise is taken into consideration important benefits of using spectral subtraction or Wiener filtering are provided. As we can see in Figure 65, for 0 dB of SNR an improvement of 40% absolute in the *WER* is obtained using spectral subtraction. Wiener filtering outperforms the results obtained by spectral subtraction for all SNR. If we analyse the spectral description of the noise and the signal in Figure 57 and Figure 60, we can appreciate that both spectrograms are quite different and therefore techniques that try to discriminate noise and signal in a frequency domain for this type of noise will be successful, as it has been demonstrated in the results of Figure 64.

Improvements obtained by Wiener filtering and spectral subtraction are reduced when the SNR is being lower than -10 dB. For these bad SNR Lip Reading outperforms the results of both conventional Noise reduction Algorithms. For SNR higher than -5 dB spectral subtraction and Wiener filtering are offering better results than Lip Reading. It is interesting to point out that for this noise where conventional Noise Reduction systems performs good results there is also a SNR range lower than -5 dB where the use of visual information could improve the results obtained by conventional Noise Reduction systems.

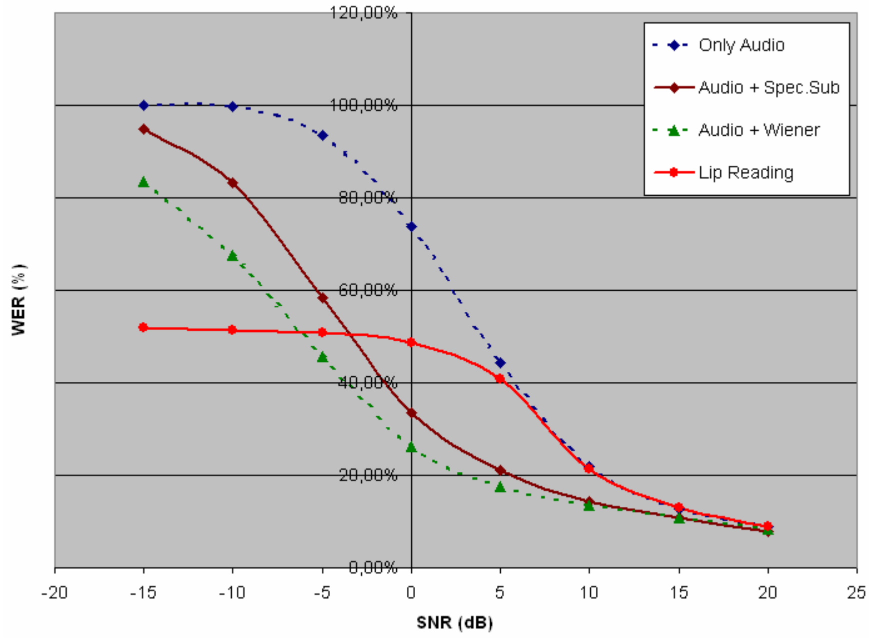

Figure 65: WER for different Noise Reduction Techniques with Non-Stationary Car noise

Now we are going to show the results of using conventional Noise Reduction techniques in an interfering talker scenario. As it can be seen in Figure 66, as well as it has occurred with variable tone, the use of spectral subtraction and Wiener filtering increases the quantity of errors. These both technologies work in the frequency domain, using only this information it is impossible to distinguish between the speech of the speaker and the speech of the interference noise. In this scenario Lip Reading is providing better results.

Lip Reading can be applied as a new Noise Reduction technique to fight against noises like interference noise or variable tone where conventional Noise Reduction techniques are providing bad results. Moreover, for a car noise where spectral subtraction and Wiener filtering are giving good results we have found out that for very bad SNR the visual information outperforms the results of conventional Noise Reduction.

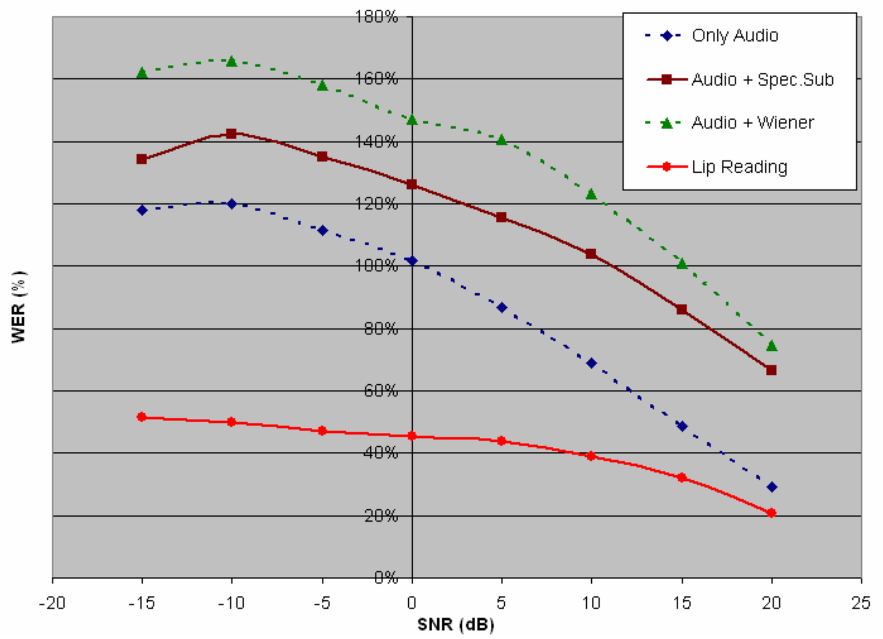

Figure 66: WER for different Noise Reduction Techniques with Interfering Talker

We have seen that due to the different nature of the audio and visual information its use can be complementary for different type of noises and levels of degradation. This point has encouraged us investigating the combination of both conventional Noise Reduction with the visual information. The schema of this implementation is quite simple, if we see the general description of the Lip Reading system of Figure 1. In the audio pre-processing the Noise Reduction techniques are going to be applied and the audio features obtained after this pre-processing are going to be combined with the visual features using the multi-stream integration strategy. For this experiment we have chosen the car noise, because this is the noise where both modalities of Noise Reduction provide good results.

In Figure 67 the *WER* for the different Noise Reduction and combinations is shown for different SNR. As we can see for very bad SNR (-15 dB), Lip Reading reduces the *WER* obtained from Wiener filtering from 81% to 51%. But also for better SNR (5 dB) the combination of Lip Reading with Wiener filtering gives the lowest *WER*, improving the results of Wiener filtering in 24% relative.

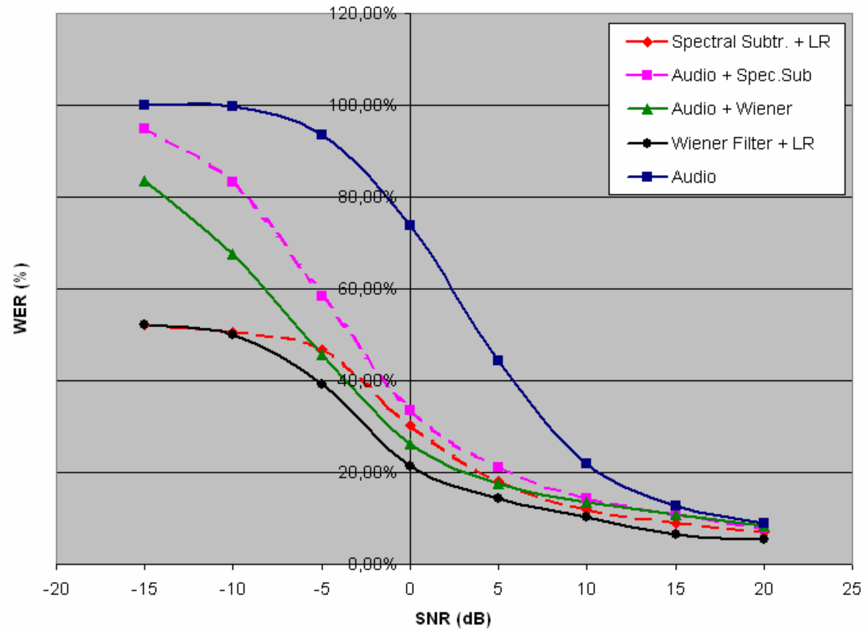

Figure 67: WER for Lip Reading combined with different Noise Reduction Techniques for Non-Stationary Car Noise

6.4 Recognition Results with Different Levels of Visual Noise

In the previous section the acoustic noise has been studied. We have supposed that the visual information was not corrupted but in a real world the video channel will also be affected by different types of degradations. In this section we want to study the influence of the visual noise in the audio-visual recognition. We are going to corrupt the visual information in two different ways. First of all a general random white noise will be added in the visual features, this general noise models different artifacts that can corrupt the visual information. Secondly a horizontal and vertical shift in the ROI will be simulated and features will be obtained within the shifted region. These two different types of noise will be evaluated for different levels of degradation; in the case of the first one the percentage of corrupted frames will be the variable of our experiments. Regarding second one the number of shifted pixels will be used as a degradation variable. For all test the audio, visual and multi-stream audio-visual HMMs shown in the last chapter will be used. Test are performed using our test CUAVE set in a speaker independent way.

6.4.1 Random Visual Noise

We have analyzed the system when the visual information is corrupted by random noise. We have changed the visual features introducing a percentage of false frames. The selection of false frames was made with a uniform random variable and the percentage of false frames is a variable given in the experiments. When a frame was classified as false all its visual features were substituted with random white noise in the range of the visual features [-127, 128].

We have made this experiment using two different types of audio noises “Variable Tone” and “Interfering talker”, in both cases the SNR of the acoustic signal is fixed. For each experiment the percentage of false visual frames has been varied between 0% and 100%. “Interfering talker” is especially interesting for Lip Reading because this technique is capable to improve the results where conventional Noise Reduction solutions fail as it was previously shown.

Variable Tone Audio Noise

As it can be seen in Figure 68 for the visual HMM when the percentage of false visual frames grows up, the *WER* increases. Insertions and substitutions grow up significantly. However, deletions decrease as it can be seen in Figure 68.d. The visual HMM introduces more insertions than deletions when the features contain noise.

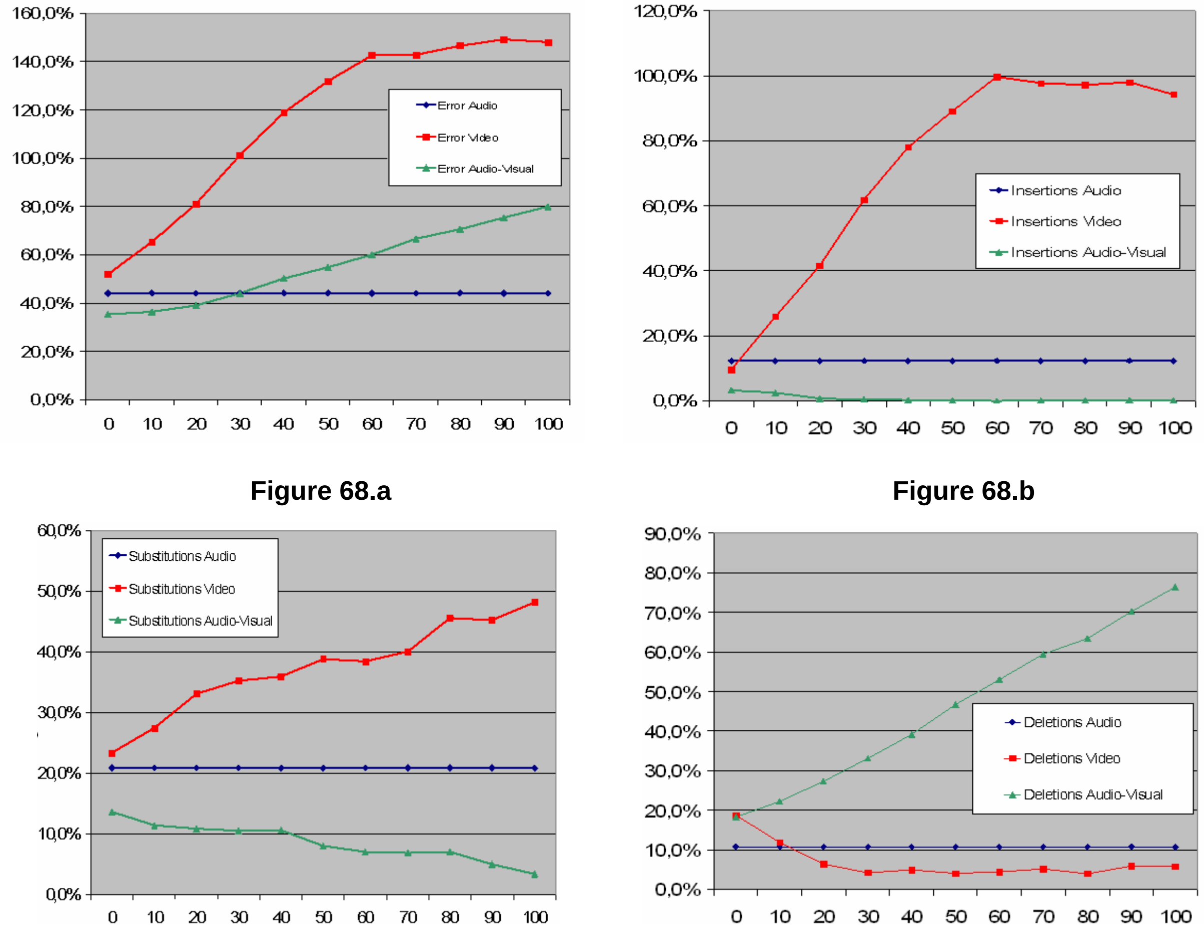

Figure 68.a

Figure 68.b

Figure 68.c

Figure 68.d

Figure 68: a) Word Error Rate b) Percentage of Insertions c) Percentage of Substitutions d) Percentage Deletions. The Horizontal Axis shows the Percentage of False Visual Frames. Audio Channel was corrupted with a Variable Tone Noise with 5 dB SNR.

The acoustic channel remains constant. Therefore, the recognition rate of the audio HMM remains constant as well.

The audio-visual HMM has the same internal structure as the audio HMM and they have the same transition probabilities. When number of false frames increases, the audio-visual HMM generates more deletions. The number of results tends to be very small producing almost no insertions but many deletions. It can be said that the audio-visual HMM is much more conservative than the video HMM.

The different behavior of the visual and audio-visual HMM against noise can be explained as follows; first of all, both HMM have a different internal configuration, the transitions probabilities have been optimized for the visual HMM with visual information and for the audio-visual using only the audio Information. The higher variance of the visual data makes that the Visual HMM tends to give as valid more results. It is not so much restrictive as the audio HMM where much more data were available and smaller variances were obtained.

If we examine the *WER* we can conclude that up to 10% of false visual frames there is not an important increment in the percentage of the word error rate. It can be said that our system can tolerate up to 10% of errors in the visual features without degrading significantly the final recognition rate. In any case, the combined result of audio and video information remains better than the one obtained using only the audio channel up to 30% of false video frames. As it will be seen in the next paragraph the tolerance against the video errors of the audio-visual recognizer is dependent on the type of audio noise we are dealing with.

Interfering talker Audio Noise

As it can be see in Figure 69 the audio-visual HMM is more tolerant to errors in the visual channel for this type of Interfering talker than for a variable tone noise.

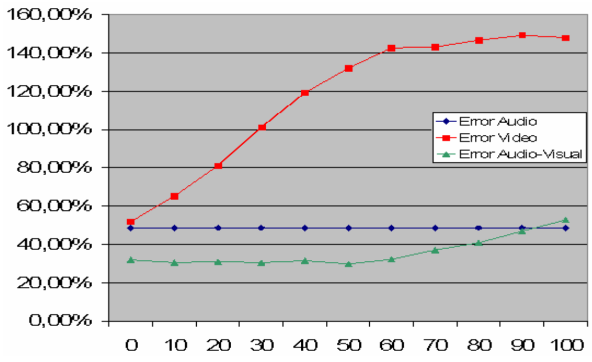

Figure 69.a

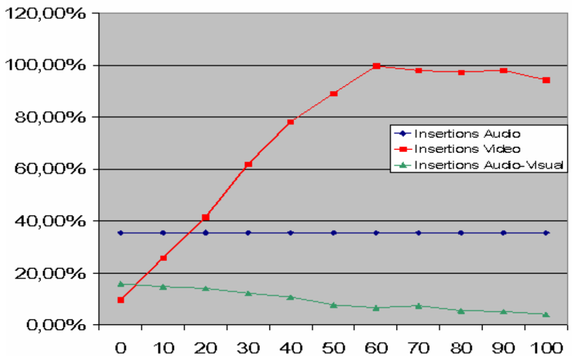

Figure 69.b

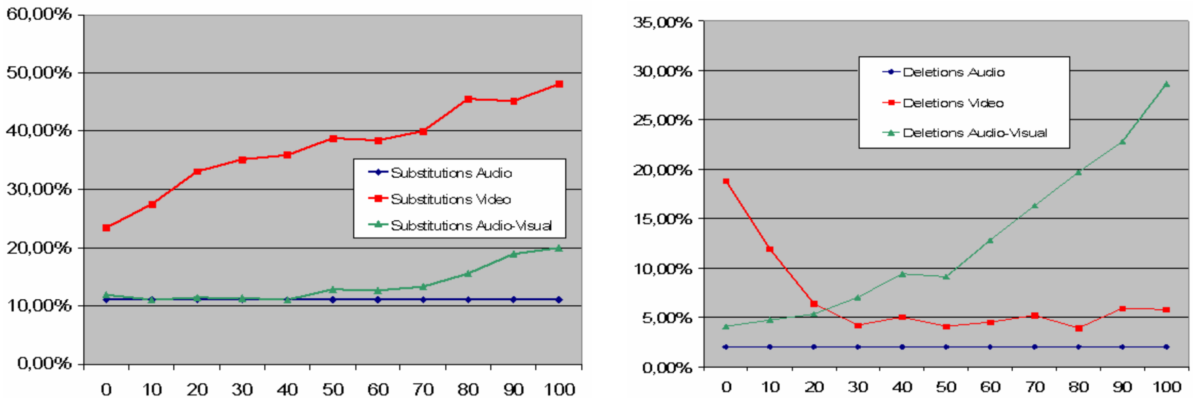

Figure 69.c **Figure 69.d**

Figure 69: a) Word Error Rate b) Percentage of Insertions c) Percentage of Substitutions d) Percentage Deletions. The Horizontal Axis shows the Percentage of False Visual Frames. Audio Channel was corrupted with “Interfering talker” with 15 dB SNR.

In the experiment shown in the last paragraph the audio channel was contaminated with a variable tone noise and the word error rate was incrementing from the 10 % of false visual frames. In this experiment an interfering talker is added to the acoustic channel. Using this acoustic noise the word error rate does not suffer degradation up to 50% of false video frames as it can be seen in Figure 69.a. It is an interesting coupling effect between both information channels; when the audio information is degraded by interfering talker, the combination of video and audio information remains more tolerant to errors in the video channel than when the audio noise was the variable tone. This result can be explained as follows: If we look at Figure 69.b the audio-visual HMM provides approximately 17% of insertions for 0% of false visual frames. This quantity of insertions is quite higher than the 2% obtained with a variable tone audio noise, see Figure 68.b. This is due to the fact that the Interfering talker introduces much more insertions than the variable tone noise. Looking at Figure 69.b we can see that when the percentage of visual noise increases the percentage of insertions in the audio-visual HMM decreases. This behavior was explained in the last section because of the nature of the audio-visual HMM: an increment of the errors in the feature vectors implies an important increment in the deletions but the insertions tend to decrease. Up to 50% of false video frames the increment of the deletions is compensated with a drop of the insertions in such away that the total error rate remains constant. This cannot happen with the variable tone noise because the nature of this noise introduces fewer insertions and there was not room to compensate the natural increment of deletions with a reduction of the insertions.

The main conclusion of this experiment can be summarized in this way: dealing with interfering talker makes the system more robust against visual noise than when the audio channel is corrupted by variable tone noise. This leaves good news that confirms the application field of Lip Reading. As it was shown in the previous sections it is especially for this type of non-stationary noises where the application of Lip Reading leads to an advantage against other conventional Noise Reduction solutions and as we have shown now just with these noises the audio-visual system is less sensitive to video errors.

6.4.2 Vertical and Horizontal Shift

With this experiment we want to simulate a typical type of visual noise due to the inaccuracy of the Lip Finding and Tracking algorithm. In the next experiment the ROI given by the Lip Finding and Tracking algorithm will be shifted on purpose in the horizontal and vertical axes to simulate the inaccuracies that can be found in the position of the mouth. The independent variable in these experiments will be the amount of pixels in the horizontal and vertical direction that the giving position of the mouth is shifted from its right position. It is a usual effect that the position of the mouth given by the localization algorithm has some errors, for example the shadows in the region of the lips can generate a translation of the lip region. Due to the mouth is symmetrical to the vertical axis we have studied the shift in the horizontal direction only in one direction. The mouth is not symmetrical to the horizontal axis; this is the reason why we have decided to make the study in both directions (positive and negative shift in the vertical axis). In this experiment we assume that all frames are affected by this noise, all frames in an utterance have the same shift. In Figure 70 on the left bottom the ROI provided by the Lip Finding and Tracking that is used to extract the visual features (DCT) is shown. In Figure 70.a this ROI has no shift. We have added artificially a controlled shift; in Figure 70.b we can see the mouth with a positive shift in the vertical axis and in Figure 70.c with a negative shift. A shift in the horizontal axis (only positive due to the symmetry of the mouth) can be seen in Figure 70.d.

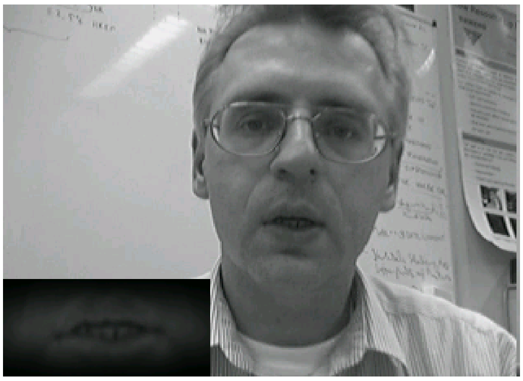

Figure 70.a

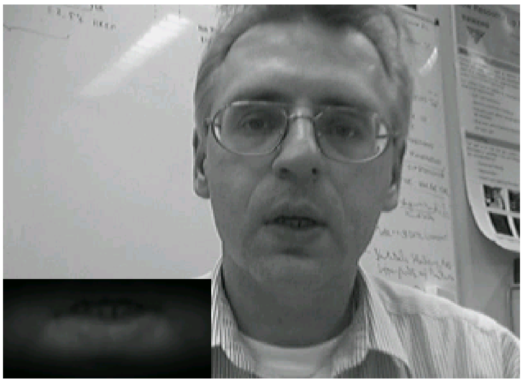

Figure 70.b

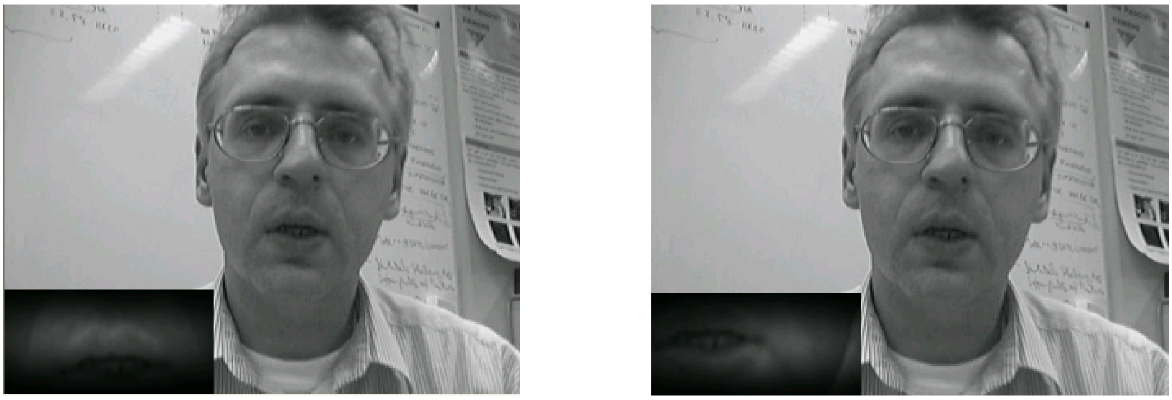

Figure 70.c

Figure 70.d

Figure 70: Region of interest shifting a) Original Image without Shift b) c) Shift in the Vertical Direction and d) Shift in the Horizontal Direction.

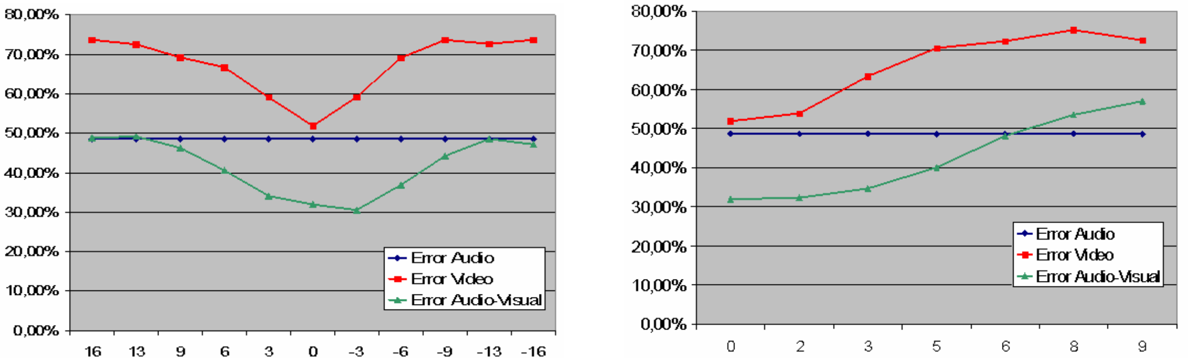

Figure 71.a

Figure 71.b

Figure 71: Word Error Rate for different Pixel Shifts: a) Vertical Shifts Percentage with respect to the Height of the Normalized Image (64 Pixels) b) Horizontal Shifts Percentage with respect to the Width of the Normalized Image (128 Pixels).

In Figure 71.a the Word Error Rate obtained with audio, video and audio-visual HMM for different vertical shifts is shown. In the horizontal axis the shift pixels in percentage with regard to the normalized height mouth (64 pixels) are provided. Figure 71.b is the same but for the horizontal shifts, in this case the horizontal axis shows the percentage of shift with respect to the normalized width (128 pixels). The results of Figure 71 correspond to images like from Figure 70.b and Figure 70.c. it can be seen that the visual HMM is very sensitive to shifts in the mouth position, this influence is lower when we are dealing with the audio-visual HMM. The behavior of the *WER* regarding the shift in both directions is more or less symmetrical. We can see how the audio visual HMM outperforms the results of the audio HMM for horizontal shifts smaller than 13% of the ROI height.

In Figure 71.b the Word Error Rate is shown for horizontal shift, as it was said due to the symmetry of the mouth in the horizontal direction only positive shifts have been studied, as it is shown in Figure 70.d. In this case a shift up to 6% of ROI width will provide better results than the audio only HMM.

Inaccuracies of the Lip Finding and Tracking algorithm generate shifts in the horizontal and vertical axis and they have an influence on the *WER*. A tolerance of 3% in both axis and directions can be assumed without an important degradation on the recognition rate.

6.5 System Requirement Estimations

An important issue for an embedded device implementation is the requirements of the system. How much memory and CPU needs our system to work? [Guitarte and Lukas, 2002]. These measurements are normally dependent on the characteristics of the platform where it must be implemented. The objective of this study is to show that our algorithms can be run in an embedded device. We have taken as reference the ARM920T, an exemplary microprocessor suitable for 3G of Mobile Devices (i.e. UMTS Services). The ARM920T used for testing was a 150 MHz 32-bit RISC CPU processor with 16 Kbyte bi-directional cache. The external memory access speed is 150 nsec. for non sequential and 10 nsec. for sequential access.

In Table 16 the requirements in terms of CPU, code and data memory of our complete Lip Reading System have been summarized. We have shown the requirements of each subsystem. These results are estimations, for this evaluation we have based on measurements of our conventional audio recognition system [VSR, 2002] and of our Finding and Tracking algorithm shown in section 3.4. Memory requirements have been divided into code memory for the algorithm implementations, and into data processing memory for intermediate memory allocation.

For audio feature extraction the algorithms described in 4.1 were used, approximately 10 MHz CPU will be needed. For visual feature extraction the DCT was selected. As it was seen in 4.2.4, 4 MHz are necessary to perform the optimized DCT. The whole preprocessing (LDA, derivatives…) is similar to the preprocessing used for audio, this is why an estimation of 15 MHz for the whole video feature extraction is reasonable. As integration strategy multi-stream was selected as it was seen in 5.4, for this kind of integration the emission probabilities must be evaluated for audio and video, taking into account that the size of the visual features is smaller than the size of the audio features the resources needed for the visual emission calculation will not be higher than the resources needed for audio. Finally using multi-stream the search must be done only once. The total estimations for a Lip Reading system on an embedded device are given below.

	CPU (MHz)	*Code (KByte)*	*Data (KByte)*
Audio Feature Extraction	10.7 MHz	24.0 KB	6.5 KB
LipFinding and Tracking	4.0 MHz	9.0 KB	100.0 KB
Visual Feature Extraction	15.0 MHz	20.0 KB	8.0 KB
Audio Emission Calculation	8.5 MHz	} 5.0 KB	32.0 KB
Visual Emission Calculation	6.0 MHz		32.0 KB
Search (Viterbi Decoding)	4.5 MHz	30.0 KB	1.0 KB
Total	**ca. 50 MHz**	**ca. 88 KB**	**ca. 180 KB**

Table 16: Overall resource estimations for Embedded Lip Reading for 10 words vocabulary.

The overall resource estimation summarizes the above mentioned resource estimations of the particular sub-systems. It has to be seen as a first guess that can vary to a certain extent depending on the realization, the kind of integration and device specifics, which can not be foreseen at the current state.

In any case as estimation the use of 50 MHz of CPU from a total of 150 MHz that are available in the exemplary ARM920T microprocessor show that the State of the art CPUs are prepared to work with this kind of technology that improves the recognition rate significantly. On the other hand the memory requirements (88 KB for code and 180 KB for data) are nowadays not a problem for the implementation. Furthermore the microprocessor technologies are being developed in such a fast way that throughout the writing of this thesis several updates to the maximal available resources have been needed. For sure this will not be an issue for the implementation of Lip Reading technology in an embedded device.

Chapter 7
Conclusions, Discussion and Future Work

In this chapter the main conclusions of our work will be presented. Furthermore, a comprehensive discussion of the most important results will be carried out. Finally, new ideas for the use of visual information that were obtained along this work will be raised for future possible research. We think that this is a good place to summarize new audio-visual research topics and encourage new studies.

7.1 Conclusions and Results Discussion

First of all we will enumerate the main conclusions of this work; afterwards we will discuss each of them together with their associated results. We have not ordered the conclusions by order or relevance but beginning with the most general ones and finishing with the aspects which affect only some part of the system:

- Lip Reading can be implemented in embedded devices improving the recognition performance.
- Visual information alone provides poor recognition results. The challenge is to combine it properly with audio information.
- Lip Reading can be considered as a new Noise Reduction (NR) technique.
- Lip Reading clearly outperforms the results of conventional NR for interfering talker noise, where conventional NR fails to improve the recognition performance.

- Visual information can be combined with audio signal processed with conventional NR improving the recognition results obtained just only with conventional NR.
- Audio-visual recognition is tolerant enough to errors in the visual channel.
- A deterministic Lip Finding and Tracking based on face geometry can be used for Lip Reading and allows it to be implemented in embedded devices.
- For our embedded solution, and taking into account inaccuracies due to light conditions, pixel-based feature extraction algorithms provide better results than shape-based algorithms.
- Multi-stream fusion seems to be the best solution for fusion of audio-visual information in terms of recognition rate but also the most suitable one for an embedded implementation.

Now each of the previous conclusions will be explained.

Lip Reading can be implemented in embedded devices improving the recognition performance. Showing the feasibility of Lip Reading for embedded devices is the main objective of this doctoral thesis. The implementation of Lip Reading algorithms in embedded devices involves two different aspects: firstly, the resource consumption of our algorithms should not surpass the available resources of our embedded devices and secondly the type of algorithms should be adequate to be used in commercial devices. In our implementation both conditions were met. The first one can be seen in our resource estimations ca. 50 MHz of the 150 MHz available in the processor were used for Lip Reading, this implies 33% of the total CPU, that is not critical at all considering that many mobile phones normally have one microprocessor and one DSP, one for the main communication tasks and another for other applications running on the device. For other environments, like car infotainment systems, the actual DSPs used for car navigation applications provide ca. 400 MHz which implies a large margin for the implementation of Lip Reading. Furthermore, DSPs technology is evolving constantly and we can assume that when Lip Reading will be applied in products these margins will even be larger. Another important point for embedded devices is the usability of the algorithms; the use of non-invasive techniques (without any kind of special reflection points or sensors on speakers head) has been taken into account in our algorithms. A condition for our Lip Finding and Tracking algorithm is a frontal view of the speaker. This will not be a problem for example for car appliances where the camera can be placed looking at the driver, who is not expected to change his position. For other scenarios like mobile phones a cooperative speaker should be looking at the device while recognition system is on, a cooperative speaker can be assumed. Significant improvements of recognition results using our Lip Reading system were shown for different types of noises. We have presented a Lip Reading system that improves the recognition performance and that can be implemented in an embedded device.

Visual information alone provides poor recognition results. The challenge is to combine it properly with audio information. For speaker independent recognition using only the visual information an error rate of 53% was obtained, this result is not good enough for being used in any application. The quantity of speech information included in the visual cues is much smaller than the information included in the audio cues, or at least the actual state of the art of Lip Reading is not able to extract more information from the visual signal. The challenge in the current system is to combine the audio and visual information. It was shown how this combination leads to important improvements. For example for interfering talker condition in a highly degraded environment (-15 dB SNR) the use of visual cues combined with audio information has resulted in a relative reduction from 56% of the error rate. This improvement is not very significant if we take into account the absolute error results. As, although the error rate has been halved using Lip Reading there is still an absolute word error rate of ca. 50% which is much too high for any application. Nevertheless, if we consider better SNR it can also be seen how for 25 dB the error rate has been reduced from 14.7% with audio only to 9.1% using Lip Reading, this implies a relative reduction of 38%. Although this improvement is smaller than the previous 56% it is much more important because this word error rate reduction for continuous digit recognition with background voice interference noise could make the application acceptable and is therefore very useful.

Lip Reading can be considered as a new Noise Reduction (NR) technique. We have shown that Lip Reading provides significant improvements in the recognition rate in situations where the acoustic signal is highly degraded. These improvements will be smaller if we take into account clean audio signal. Lip Reading can be defined as a new way to combat audio noise. Not only machines use the visual information to provide better recognition results in noisy environments, in [Sumby, 1954] it was shown that also for humans especially in noise degraded situations the visual cues are very useful for understanding. This is the current state of the art of audio-visual recognition, if in the future we are able to reduce the word error rate using only visual information due for example to new visual feature extraction algorithms, then probably it would be possible to use Lip Reading not only as a NR technique but also to improve the recognition performance with clean speech. But nowadays the main contribution of Lip Reading is being a new NR Technique.

Lip Reading can be used as NR for interfering talker where conventional NR does not work. Assuming that the most important achievement of Lip Reading is to improve the recognition rate in degraded environments we have decided to compare its results with other conventional NR techniques like spectral subtraction and Wiener filtering. We have seen that when we are dealing with a noise which is spectrally quite different to the spoken signal, as for example car noise, Lip Reading outperforms the results of conventional NR solutions only for bad SNR (lower than -10 dB). Due to the spectral differences between both signals the conventional NR techniques can separate very well the noise and the signal achieving very good results. Nevertheless, when the noise signal is spectrally similar to the speech, which happens for example with an interfering talker type of noise or with a variable tone noise, the conventional NR solutions are not able to distinguish between both signals and they cannot improve the recognition results. Especially in these situations the power of Lip Reading can be seen as an alternative to the conventional NR. For these noises and for all SNR ranges Lip Reading is providing better results than the conventional NR techniques.

Visual information can be combined with audio signal processed with Wiener filtering improving the recognition results obtained with conventional NR. For the noise scenario where the conventional NR techniques provide better results than Lip Reading we have tried to combine the improved audio signal using the conventional NR with the visual features (Lip Reading). Better results have been obtained for all SNR using this combination than using the conventional NR only. Therefore, for scenarios where the conventional NR fails Lip Reading seems to be a good tool to fight against noise and in the scenarios where the conventional NR already achieves good results Lip Reading in combination with conventional NR delivers even better results. Taking into account all the mentioned results Lip Reading should be taken into account as a new modality of NR. Lip Reading uses additional information to distinguish between the noise and the signal, it uses the visual information. Something similar occurs with other NR Techniques that are not considered in this work like NR using microphone array where the location information is used to distinguish between signal and noise depending on the source direction. Therefore, these results encourage us to consider Lip Reading as a way to reduce the effect of noise in the same way as a microphone array is used. A comparison between both technologies was not beyond the scope of this work. From the implementation requirements both technologies need additional hardware: a camera or an array of microphones. A decisive factor could be if this additional hardware is intended to be integrated for other purposes, for example in mobile phones microphone array implies new hardware but not Lip Reading as in most of the mobile phones cameras are available for other applications.

Audio-visual recognition is tolerant enough to errors in the visual channel.

As the objective of this work was to provide solutions that must work in normal conditions we had to know how our system reacts when the visual information is degraded. We have shown that the visual cues help the system when the acoustical channel is corrupted but of course the visual channel can be also degraded due to bad light conditions, occlusions, failure of the tracking system etc. We have evaluated the tolerance of our system to the percentage of false visual frames. In the worst case, a percentage of bad visual frames higher than 30% would produce a degradation of the WER of our audio-visual system. Therefore a tolerance of 30% of the visual frames can be assumed for the audio-visual system. When Lip Finding and Tracking is used in office light conditions a frame error rate of 5.8% is obtained, therefore in these normal office conditions this error rate of our visual system would not affect the recognition results. Other problems could appear if we test our system in bad light scenarios for example in car garage, tunnels or at night. In such scenarios the current system would surely not provide good results. For such complicated scenarios the use of infrared cameras providing constant illumination could be the right solution.

In addition, the influence of the inaccuracies on the detection of the ROI has been investigated. The lips detection accuracy has also influence on the recognition rate. A tolerance of 3% of the normalize ROI size can be assumed without degradations of the recognition rates.

A deterministic Lip Finding and Tracking based on face geometry can be used for Lip Reading and allows it to be implemented in embedded devices. In chapter 0 a novel implementation of a Lip Finding and Tracking algorithm has been proposed. The algorithm is based on plain gradient filters and on the face geometry. The search of the mouth region performed on segmented regions has speeded up the algorithm in such a way that it is adequate to be implemented in embedded devices. Our Lip Finding and Tracking algorithm uses in average just only 4MHz of the CPU. All results provided in this work have been obtained using this Lip Finding and Tracking algorithm which works properly for frontal view and in normal office light conditions. For special and complicated scenarios like tunnel or night driving as well as view against the light more complicated solutions like infrared cameras should be investigated. For our application scenario in office environment our Lip Finding and Tracking algorithm has provided a frame error rate of 5.8% which, as it was commented in the last paragraph, does not degrade audio-visual recognition. Our Lip Finding and Tracking algorithm has been implemented in a Mobile Phone (Siemens SX1) using a Symbian operative system for other spin off applications that have been generated due to the success of the implementation. This was a non expected collateral effect of our investigations as just when the first component of our system was working and evaluated, its implementation in a Mobile Phone prototype proposed. The number of image processing applications available in Mobile Phones has incremented in the last years. Our Lip Finding and Tracking algorithm allows Siemens to develop different image applications that take use of the position of the speaker in the image. Some of these applications are nowadays prototypes and we hope that some of them can be implemented into products in the near future. The main Spin off applications obtained from our Lip Finding and Tracking are going to be summarized in the annex 1.

For our embedded solution, and taking into account inaccuracies due to light conditions, pixel-based feature extraction algorithms provide better results than shape-based approaches. In chapter 4 we have shown different attempts of visual feature extraction. There are two different types of approaches, on the one hand we can try to extract the contours of the lips and make a geometrical model of these contours, the parameters describing the model are going to be used as features, and this is called shape-based approach. This kind of solution extracts the most important information for the recognition but requires an accurate extraction of the lips contours which is not always an easy task and can be very affected by light conditions e.g. shadows. On the other hand assuming that the position of the mouth is known a mathematical transformation of this region can be obtained and the values of the transformation used as features. This solution does not provide the most important information for the recognition as it conveys more information of the image that is not relevant. But it only requires an estimation of the mouth position and not the extraction of the detailed lip contours. For our embedded implementation the shape-based approach provides better recognition results because the contours extraction of the lips generates many errors and degrades more the result as the benefit obtained by using only the most relevant information. Furthermore the pixel-based approach provides information about the tongue and teeth visibility, this information can be important to visually distinguish different visemes and was not enclosed in the geometrical approach. The selection of the DCT as our pixel-based feature extraction solution is appropriate for embedded devices as there exists efficient implementations of this transformation used in video compression.

Multi-stream fusion seems to be the best solution for fusion of audio-visual information in terms of recognition rate but also the most suitable one for an embedded implementation. In chapter 5 different integration strategies for audio and visual information have been investigated and implemented: early integration, Multi-stream and late integration. The results in terms of recognition rate show clearly that the Multi-stream solution provides the best performance. This is not the unique reason why we decided to use Multi-stream. There are also some very important implementation considerations. First of all in Multi-stream in comparison with late integration the use of only one decoding process will save resources for an embedded implementation. Secondly, the possibility of using different weightings for the audio and the video channel depending on both channel conditions. This is not possible in early integration. Another reason is that the Multi-stream approach allows the possibility of re-using all available audio-only databases for training, which does not happen with Couple HMM and early integration. Finally, the simplicity of the Bakis topology in comparison with the bidimensional topology of a Couple HMM is an advantage for embedded implementations. For all these reasons we have considered to use the Multi-stream approach to combine audio and visual information.

7.2 Future Work

In this work promising results have been obtained which are intended to conduct to further investigations in order to achieve even better use of the visual information. Conventional speech recognition is a field that has been profusely developed in the last thirty years. In comparison, in audio-visual speech recognition still much effort can be invested and results as the ones presented in this work should encourage researchers to new investigations. We are going to propose some main areas where we see potential for further development. Audio-visual recognition is a new research field full of challenges and where many new ideas can be applied. We want to mention different proposals and ideas that were generated in the course of this doctoral thesis and that were not beyond the scope of our investigations but can field new research.

We summarize the points where we see potential for improvements and optimizations of auxiliary Lip Reading:

- ***New Visual Feature Extraction***. The current visual feature extraction is still dependent on the light conditions. It would be desirable to obtain an extraction algorithm that would be independent of these conditions in such a way that the training and test could be mismatched without result degradations. As well another visual extraction algorithm could extract more information about the visemes and therefore reduce the word error rate of visual only system improving the performance of the whole system.

- ***Use of Infrared Cameras*** which provide constant illumination in light changing environments like cars for example for tunnels, night driving or garage conditions. An audio-visual database using infrared camera has been recorded in the last years at the University of Zaragoza [Ortega et al., 2004].
- ***Visual Voice Activity Detector (V-VAD).*** In our work we have tried to use the visual information to improve the recognition rate directly by using the visual features for the recognition. Nevertheless, there exists other different functionality blocks in a speech recognition system that could also make use of the visual information. Many speech recognition systems use a Voice Activity Detector to distinguish the speech from noise and process the recognition only for frames containing speech, in this way the number of insertions is reduced. Conventional Voice Activity Detector (VAD) systems are based on the audio signal. These systems cannot differentiate between the user voice (signal) and another person speaking in background (noise). Furthermore, their performance in highly degraded environments is not always good enough. We could think on a Visual Voice Activity Detector (V-VAD) system that recognizes when the user is speaking based on its lips movements. VAD is an important part of the SNR estimators that are necessary in our system in order to estimate the optimal weighting between the audio and the video channel.
- ***Speaker directive Microphone Arrays.*** Another application of the visual information could be the use of the speaker's position obtained from the Lip Finding and Tracking algorithm to provide the microphone array system information about the direction of the main signal and adapt the beam-forming in order to separate properly the speech signal and the noise. This information would improve the results of microphone array in interfering talker scenarios.
- ***Use of audiovisual information to increment the security of speaker verification.*** The field of speaker verification has provided promising results in the last years. Some improvements must be done in order to use this biometric information as an access code. Audio information is very easily to be recorded and it could be used to hick jack a system. In order to make verification systems more robust against these kinds of attacks, it can be checked whether the visual information could be used not only to recognize the users face but also to detect whether there is synchrony between the audio and the lips movements.
- ***Mapping the audio signal from a noise corrupted space into a clean space by using additional non corrupted information from the lips movements.*** A way to

reduce the influence of the noise in the speech recognition rate is to apply a linear transformation to the audio features in order to transform them into a feature space where the noise has less influence. This transformation should be learned in a training process. In order to obtain better results an additional signal could be used. Visual signal is correlated with audio signal in the clean space and it could be used to adapt the transformation.

Appendix 1:

Spin off Applications of Embedded Lip Finding and Tracking Algorithm

While this thesis has being written there has been a high market penetration of cameras in mobile phones and other consumer electronics. In this work we have presented an efficient algorithm to find automatically the position of the mouth, this algorithm due to its low complexity requirements is suitable to run in embedded devices. We have implemented our Lip Finding and Tracking algorithm in SX1, a Siemens Mobile Phone based on Symbian Operative System. Several applications, which take use of our algorithm, have been developed. We are going to present here four different applications that were implemented as demonstration and prototypes.

Video Conferencing: Automatic Zoom for Video Telephony

With 3G mobile phones video telephony is possible, also from small devices. The size of the terminals has in the last years considerably reduced but an important limitation is still always the size of the display that must be large enough to properly show the speakers face for video telephony. Cameras in mobile phones are able to capture high resolution pictures. The limitation on the resolution of the pictures for video telephony is due beside other considerations like bandwidth to the size of the displays. High level resolution picture could only be partially shown in a small mobile phone display. The problem can be solved by showing on the display only the main information for a video call: the speakers face.

Our Lip Finding and Tracking algorithm will automatically find the position of the face on the original image taken from the camera with a VGA (640 x 480) or CIF (352 x 288) resolution, the region covering the face is selected and only this region is transmitted with the highest resolution in the QCIF format (174 x 144). In Figure 72 we can see an example where the conventional solution is used: by down sampling the original image is transformed in a smaller one with less resolution. In our proposal the resolution of the speaker image will not be reduced but the size of the picture to include only the speakers face.

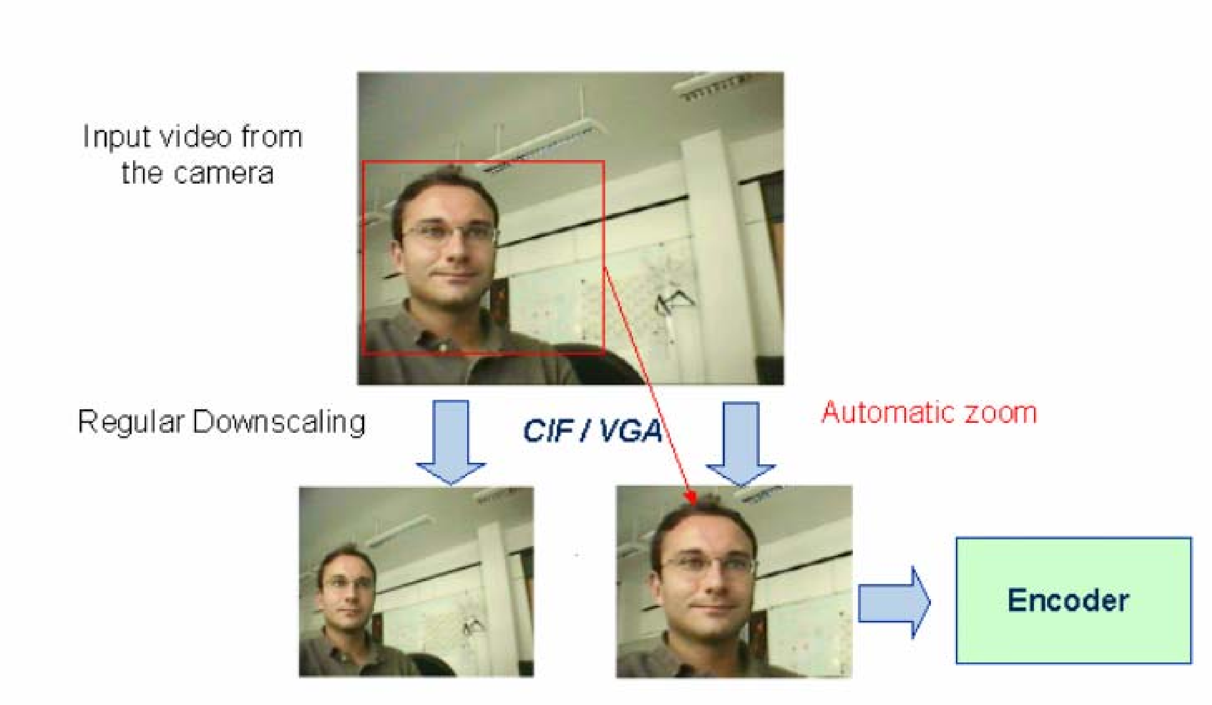

Figure 72: Automatic Zoom for Video Telephony

Insertion of Context depending Visual Information

The use of Multimedia Messaging Service (MMS) has been increased in the last years. This kind of multimedia message extends the Short Messaging Service (SMS) allowing us to send a picture along with the message. In conventional MMS the text of the message is independent of the image; some mobile phones offer the possibility of integrating the text in the image by using babbles as in a comic. The location of the babble in the conventional solution is manually defined by the user, which requires an editing effort. We have proposed an application that taking use of our Lip Finding and Tracking algorithm locates automatically the babble in such a way that the speakers face will not be covered by the babble and this one will arise from the speaker's mouth.

Other interesting application which can take advantage of our algorithm is the locating of some advertising or additional information in a video phone call. For example some telephony service providers will be interested in offering some price reduction for video telephony if the logo of the provider or some additional advertising is integrated in the video. In order to integrate this information properly in such a way that the communication is not degraded (for example with the provider's logo covering the face of the user) our Lip Finding and Tracking algorithm can be applied as it can be seen in Figure 73.

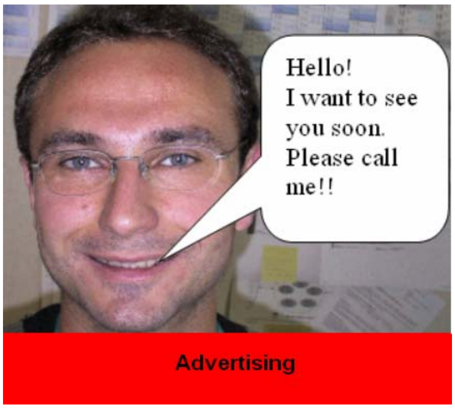

Figure 73: Automatic Insertion of Additional Information in MMS or Video Telephony

SMS Reading

SMS Reading is an application based on Text-To-Speech that will generate an audio stream reading a received SMS. In order to give an added-value to this service we have developed an application where an animated face is shown on the display "reading" the SMS while the audio file is being played. There are Mobile Phones in which a picture can be associated to each agenda directory entry. These systems allow the so called calling faces applications, where the picture of the calling person will be shown on the display while ringing.

In our application when a new short message is received and the calling person is saved in the agenda with a picture this one will be taken. Using our algorithm a set of reference points will be placed on the face (lips, eyebrows and other points that will be interpolated from the previous ones). This reference points will provide a meshing of the face which will used to animate it while the short message is played. First of all a Grapheme to Phoneme, as it is also a key component of the Text-To-Speech will be needed and then each phoneme will be converted into speech using a Text-to-Speech system, the phoneme will be directly converted into viseme with a table and this will be used to show the main movements.

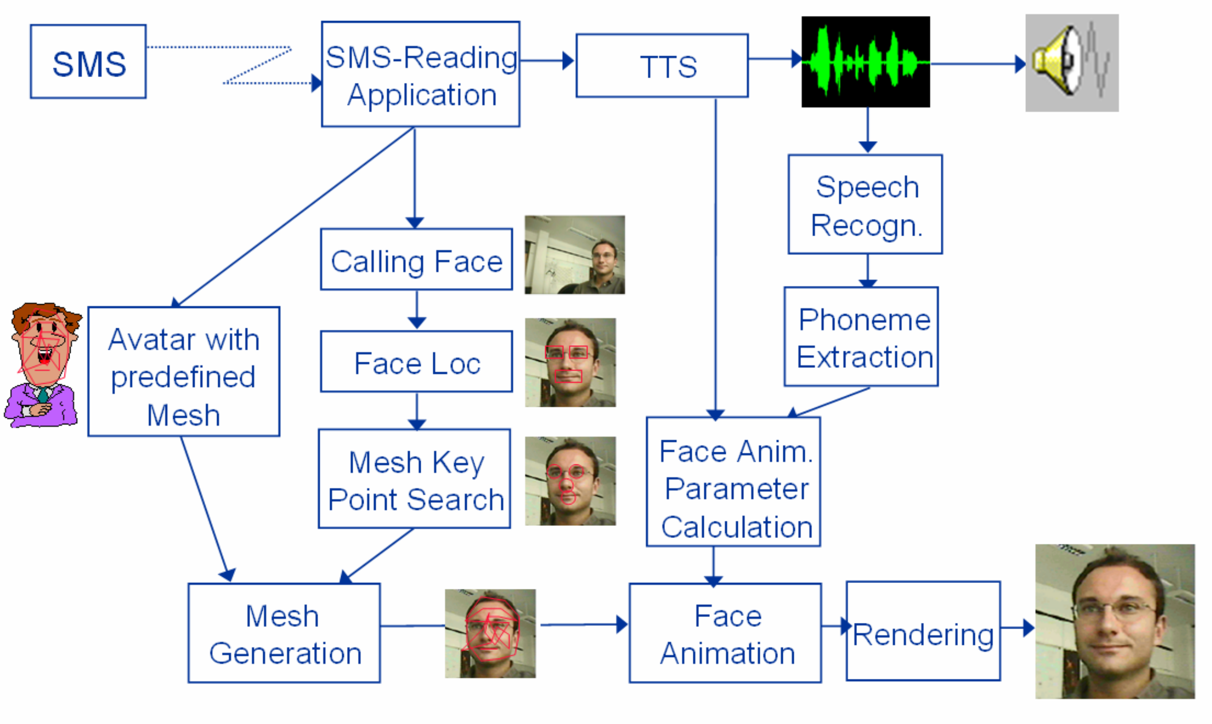

Figure 74: Short Instant Messaging

Visual Voice Activity Detection

There are many scenarios where speech recognition could be applied but because of the simplicity of the interaction and the necessity of a “Push-to-Talk” system its use is not convenient. For example, if in a coffee machine the user has to press a button in order to activate the recognition (Push-to-Talk to avoid high false alarm rates) he would directly press the button to get the coffee. In such situations a system that automatically could decide whether there is a user will solve the problem because the Push-to-Talk will be performed automatically just when a user is looking at the device. For this reason we have called this application “Look-to-Talk”. We have developed an application that will only be active when the mouth of the user is in front of the machine and therefore it is assumed that he wants to use it. This system has the advantage against other activity detectors like movement sensors that it will only be activated when a user really is placed in front of the device which means that he wants to use it.

A further level of taking advantage of the visual information is the activation of the speech recognition not just when the speaker is in front of the device but also when the system detects a visual speech activity. This means that the speech recognition system will only listen when the lips of the speaker are moving. This would be a Visual-Voice-Activity-Detector (V-VAD). It is proved that the use of voice activity detectors can improve the recognition rates reducing significantly the number of insertions, up to now all voce activity detectors were based on the audio signal. They do not usually work properly with background noise. We propose to use the visual information to decide whether there is speech or not. For the design of this system the coefficients of the ASM have been used to decide if there is a speech mouth movement. These coefficients are very appropriated as they just have information of the of the mouth and they are independent on the scaling, rotation and translation, this implies that when the speaker is moving but he does not say anything (there is not lip movements) the system will not produce false alarms, which would be generated probably if just a movement sensor is used in the mouth region.

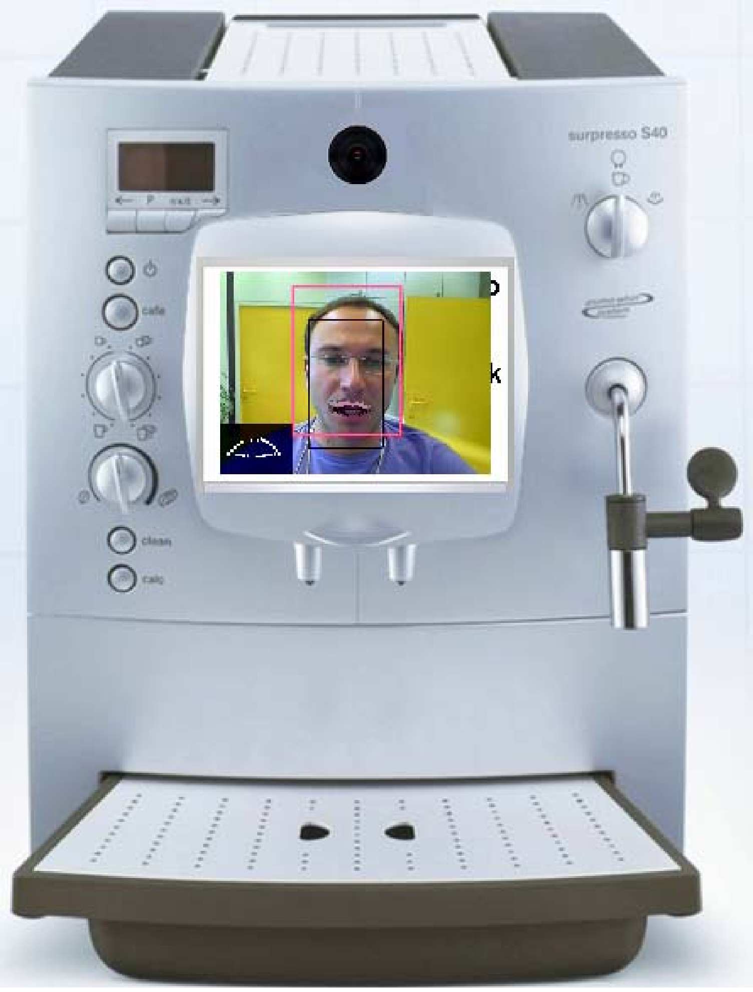

Figure 75: Visual Voice Activity Detector

Appendix 2:

Application Patents

Here we offer a summary of all application patents that has been registered by the author of this thesis as co-author and that are related with the algorithm proposed in this work or with some of the applications shown in the first appendix. All of them can be found in the official registry of the European Patent Office (http://ep.espacenet.com).

Method for analysing a scene and corresponding data processing device and program product.

Patent number:	EP1504407
Publication date:	2005-02-09
Inventor:	CLEMENS GERALD (DE); LUCAS VERDOY CARLOS (DE); MARKE MATTHIAS (DE); LUKAS KLAUS (DE); GUITARTE PEREZ JESUS FERNANDO (ES)
Applicant:	SIEMENS AG (DE)

Mobile phone image data transmission system determines face image position and extracts it for higher rate transmission than background.

Patent number:	DE10321498
Publication date:	2004-12-02
Inventor:	GUITARTE PEREZ JESUS FERNANDO (ES); VERDOY CARLOS LUCAS (DE); LUKAS KLAUS (DE)
Applicant:	SIEMENS AG (DE)

Insertion of information fields into an image, e.g. a videophone image, whereby a face position is placed in a rectangular frame object so that information fields can be added in a non-overlapping manner.

Patent number:	DE10321501
Publication date:	2004-12-02
Inventor:	GUITARTE PEREZ JESUS FERNANDO (ES); VERDOY CARLOS LUCAS (DE); LUKAS KLAUS (DE)
Applicant:	SIEMENS AG (DE)

Single handed operation of a mobile terminal by gesture recognition, whereby movement of the terminal relative to the user is measured and used to control the positioning of input selection means.

Patent number:	DE10313019
Publication date:	2004-10-28
Inventor:	PEREZ JESUS FERNANDO GUITARTE (ES); LUKAS KLAUS (DE); ROETTGER HANS (DE)
Applicant:	SIEMENS AG (DE)

Communications device e.g. for mobile telephone, has memory unit for storing image data and telephone book entries and processor with processor configured to favor image data containing information that describes faces.

Patent number:	WO2006005666
Publication date:	2006-01-19
Inventor:	GUITARTE PEREZ JESUS FERNANDO (DE); LUCAS CARLOS (DE); LUKAS KLAUS (DE)
Applicant:	SIEMENS AG (DE); GUITARTE PEREZ JESUS FERNANDO (DE); LUCAS CARLOS (DE); LUKAS KLAUS (DE)

User navigating method, involves processing sensor information to realize motion in area of head or line of user`s vision, evaluating motion information to derive navigation action and effecting modification of document section by action.

Patent number:	DE102004027289
Publication date:	2005-12-29
Inventor:	GUITARTE PEREZ JESUS FERNANDO (ES); KUTKA ROBERT (DE); VERDOY CARLOS LUCAS (DE); LUKAS KLAUS (DE)
Applicant:	SIEMENS AG (DE)

List of Abbreviations

AAM	Active Appearance Models
ASM	Active Shape Models
ASR	Automatic Speech Recognition
CHMM	Coupled Hidden Markov Models
DCT	Discrete Cosine Transformation
ETSI	European Telecommunication Standards Institute
ELDA	Evaluations and Language resources Distribution Agency
EURASIP	European Association for Signal Processing
FCC	Face Colour Classifier
HMM	Hidden Markov Models
IDIAP	Institut Della Molled'Intelligence Artificielle Perceptive
ICP	Institut de la Communication Parlée
ICASSP	International Conference on Acoustics, Speech, and Signal Processing
IEEE	nstitute of Electrical and Electronics Engineers
MAP	Maximum a Posteriori
MFCC	Mel Frequency Cepstral Coefficients
NN	Neural Networks
MSR	Movement Size Ratio
NR	Noise Reduction
PDF	Probability Density Function
RLC	Run-Length Coding
RLS	Recursive Least Square
ROI	Region of Interest
SNR	Signal Noise Ratio
VSR	Very Smart Recognizer ®
V-VAD	Visual Voice Activity Detection
WER	Word Error Rate
e.g.	example gratia
et al.	et alli

List of Figures

List of Tables

Bibliography

[Adjoudani and Benoit, 1996] A. Adjoudani, and C. Benoit, "On the integration of auditory and visual parameters in an HMM-based ASR" In D. G. Stork and M. E. Hennecke, (Eds.), "Speechreading by Humans and Machines," Springer, pp. 461-471, 1996.

[Amarnag et al., 2003] S. Amarnag, S. Gurbuz, E. Patterson, and J. N. Gowdy, "Audio-Visual Speech Integration using Coupled Hidden Markov Models for Continuous Speech Recognition," Student Forum Paper at ICASSP, 2003. http://students.washington.edu/asubram/Pubs/final.pdf

[ARM, 2006] ARM Internet Page (microprocessors for embedded devices): http://arm.com, 2006.

[Arulampalam et al., 2002] S. Arulampalam, S.Maskell, N. Gordon, and T. Clapp, "A tutorial on Particle Filters for on-line non-linear/non-Gaussian Bayesian tracking," IEEE Trans. Signal Processing, vol. 50, no. 2, pp. 174-189, 2002.

[Bauer, 2001] J. Bauer, "Diskriminative Methoden zur automatischen Spracherkennung für Telefon-Anwendungen," Ph.D. thesis, Technische Universität München, 2001.

[Beaugeant et al., 1998] C. Beaugeant, V. Turbin, P. Scalart, A. Gilloire, "New Optimal Filtering Approaches for Hands-free Telecommunication Terminals," Signal Processing, vol. 64, pp.33-47, 1998.

[Bolic 2004] M. Bolic "Architectures for efficient implementation of Particle Filters," Ph.D. thesis, Stony Brook University, 2004

[Bregler and Konig, 1994] C. Bregler, and Y. Konig, "Eigenlips for robust speech recognition," Proc. Int. Conf. Acoust. Speech Signal Process. pp.669-672,1994.

[Bruce 1986] A. Bruce Carlson "Communication systems. An Introduction to Signals and Noise in Electrical Communications" Mc.Graw-Hill International Editions, 1986.

[Buera et al., 2004] L. Buera, E. Lleida, A. Miguel, and A. Ortega, "Multi-Environment Models Based Linear Normalization for Speech Recognition in Car Conditions," Proc. ICASSP, vol. 1, pp. 1013 1016, 2004.

[Bulwer, 1648] J Bulwer, "Philocopus, or the death and Dumbe Mans Friend," Humphrey and Moseley, London, 1648.

[Cootes and Taylor, 2000] T. F. Cootes and C. J. Taylor, "Statistical Models of Appearance for Computer Vision," Technology Report, University of Manchester, 2000.

[Cootes et al., 2001] T. F. Cootes, G.J. Edwards, and C.J. Taylor, "Active Appearance Models," IEEE trans. on Pattern Analyses and Machine Intelligence, no. 23, vol.6, pp. 681-685, 2001.

[Cox et al., 1986] H. Cox, R. Zeskind, T. Kooij, "Practical Supergain," IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. Assp-34, no. 3, 1986.

[Cox et al., 1987] H. Cox, R. Zeskind, M.M. Owen, "Robust Adaptive Beamforming," IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. Assp-35, no. 10, 1987.

[Charles, 1996] P. Charles, "A Technical Introduction to Digital Video", John Wiley & Sons Inc, 1996.

[Chen, 2001] T. Chen, "Audiovisual speech processing," IEEE Signal Processing Magazine, pp. 9-31, 2001.

[Chiou and Hwang, 1997] G. Chiou, and J. N. Hwang, "Lipreading from color video," IEEE transactions on Image Processing, vol. 6, pp. 1192-1195, 1997.

[Dalton et al., 1996] B. Dalton, R. Kaucic, and A. Blake, "Automatic Speech Reading using dynamic contours," In D. G. Stork and M. E. Hennecke, (Eds.), "Speechreading by Humans and Machines. Berlin Germany," Springer, pp. 373-382, 1996.

[Doclo and Moonen, 2003] S. Doclo, and M. Moonen, "Design of Broadband Beamformers Robust against Microphone Position Errors," Proc. International Workshop on Acoustic Echo and Noise Control, pp. 207-210, 2003.

[Duchnowski et al., 1995] P. Duchnowski, M. Hunke, D. Büsching, U. Meier, and A. Waibel, "Toward Movement-Invariant Automatic Lip-Reading and Speech Recognition," Proc. ICASSP, vol. 1, pp. 109 112, 1995.

[Duda and Hart, 1973] H. Duda, P. Hart, "Pattern Classification and Scene Analysis," John Wiley & Sons, 1973.

[Dupont et al., 2000] S. Dupont, and J. Luettin, "Audio-Visual Speech Modeling for Continuous Speech Recognition," IEEE Transactions on Multimedia, vol. 2, no. 3, 2000.

[ELRA, 2006] European Languages Resource Association: http://www.elra.info

[Gonzalez and Woods, 2001] R. C. Gonzalez, and R. E. Woods, "Digital image processing," Prentice Hall, 2001.

[Gowdy et al., 2004] J. N. Gowdy, A. Subramanya, C. Bartels, and J. Bilmes, "DBN Based Multi-Stream Models for Audio-Visual Speech recognition," Proc. ICASSP, vol. 1, pp. 993-996, 2004.

[Grant et al., 2001] K. W. Grant, and S. Greenberg, "Speech Intelligibility Derived from Asynchronous Processing of Auditory-Visual Information", Proc. International Conferrence on Auditory-Visual Speech Processing, Aalborg, Denmark, pp. 132-137, 2001.

[Guitarte and Lukas, 2002] J. F. Guitarte, K. Lukas, “Feasibility Study on Lip Reading Technologies for Mobile Devices,” Siemens Internal Technical Report, 2000.

[Guitarte et al., 2003] J. F. Guitarte, K. Lukas, A. F. Frangi, "Low Resource Lip Finding and Tracking Algorithm for Embedded Devices," Proc. Eurospeech, vol. 3, pp. 2253-2256, 2003.

[Guitarte and Lukas, 2004] J. F. Guitarte, K. Lukas, “Auxiliary Lip Reading Final Report,” Siemens and BMW Research and Technology GmbH Internal Technical Report, 2004.

[Guitarte et al., 2005a] J. F. Guitarte, A. F. Frangi, E. Lleida, and, K. Lukas, "Lip Reading for Robust Speech Recognition on Embedded Devices," Proc. ICASSP, vol. 1, pp. 473 476, 2005.

[Guitarte et al., 2005b] J. F. Guitarte, K. Lukas, F. Althoff, S. Hoch, and E. Lleida "Embedded Lip Reading for Automotive Environments," Proc. Deutschen Jahrestagung für Akustik, pp.551-552, 2005.

[Gurbuz et al., 2001] S. Gurbuz, Z. Tufekci, E. Patterson, and J. N. Gowdy, "Application of affine-invariant Fourier descriptors to lipreading for audio-visual speech recognition," Proc. ICASSP, pp. 177-180, 2001.

[Höge et al.,2000] H. Höge et al., "The Siemens Feature Extraction Module SFEM for Speech Recognition," Siemens Internal Technical Report, 2000.

[Hamlaoui et al., 2005] S. Hamlaoui, and F. Davoine, "Facial action tracking using Particle Filters and active appearance models," Proc. Joint sOc-EUSAI conference, Grenoble 2005.

[Hernando, 1997] J. Hernando, "Maximum Likelihood Weighting of Dynamic Speech Features for CDHMM Speech Recognition," Proc. ICASSP, pp. 1267-1270, 1997.

[Hojas, 1994] K. Hojas, "Lineare Diskriminanzanalyse für die Spracherkennung," Ph.D. thesis, Technische Universität München, 1994.

[Huang and Visweswariah, 2005] J. Huang, and K Visweswariah, "Improving Lip-Reading with Feature Space Transforms for Multi-Stream Audio-Visual Speech Recognition," Proc. INTERSPEECH, pp. 1221-1224, 2005.

[Jelinek 1997] F. Jelinek "Statistical Methods for Speech Recognition," MIT Press, 1997.

[Junqua, 1993] J. C. Junqua, "The Lombard reflex and its role on human listeners and automatic speech recognizer," J. Acoustic. Soc. Amer., vol. 93, pp. 510-524, 1993.

[Kalman, 1960] R. Kalman, "A new approach to linear filtering and prediction problems," Transactions of the ASME-Journal of Basic Engineering, no. 82, pp. 35-45, 1960.

[Klatt, 1982] D. Klatt, "Prediction of Perceived Phonetic Distance from Critical-Band Spectra: a first step," Proc. ICASSP, vol.1 pp. 1278-1282, 1982.

[Lee and Huang, 1994] P. Lee and, F. Y. Huang, "Restructured Recursive DCT and DST algorithms," IEEE Transactions on Signal Processing, no. 42, vol. 7, 1994.

[Lucey et al., 2004] P. C. Lucey, T. Martin and S. Sridharan, "Confusability of Phonemes Grouped According to their Viseme Classes in Noisy Environments," Proc. Australian International Conference on Speech Science & Technology, pp. 265-270, 2004.

[Luettin et al., 1997] J. Luettin, N. A. Thacker, "Speech reading using probabilistic models," Computer Vision and Image Understanding, vol. 65, pp. 163-178, 1997.

[Marschark et al., 1998] M. Marschark, D. LePoutre, and L. Bement, "Mouth Movement and Sign Communication," In R. Campbell, B. Dodd, D. Burham (Eds.) "Hearing by eye II," United Kingdom: Psychology Press Ltd. Publishers, pp. 245-266.

[Martin, 1994] R. Martin, "spectral subtraction Based on Minimum Statistics," Proc. EURASIP, Signal Processing VII, pp. 1182-1185, 1994.

[Martin, 2001] R. Martin, "Noise Power Spectral Density Estimation Base don Optimal Smoothing and Minimum Statistics", IEEE Transactions on Speech and Audio Processing, no. 9, vol. 5, pp. 504-512.

[Mase and Pentland, 1991] K. Mase and A. Pentland, "Automatic lipreading by optical flow analysis," Systems and Computers in Japan, vol. 22, pp. 67-75, 1991.

[Matthews, 1998] I. Matthews, "Features for Audio-Visual Speech Recognition," Ph.D. thesis, University of East Anglia, 1998.

[Matthews et al., 2002] I. Matthews, T. Cootes, J. Bangham, S. Cox, and R. Harvey, "Extraction of Visual Features for Lipreading," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 2, no. 24, pp.779-789, 2002.

[McGurk, 1976] H. McGurk, and J. MacDonald, "Hearing lips and seeing voices," Nature, vol. 264, pp. 746-748, 1976.

[Meier et al., 2000] U. Meier, R. Stiefelhagen, J. Yang, and A. Waibel, "Toward Unrestricted Lip Reading," International Journal of pattern Recognition and Artificial Intelligence, vol. 14, no. 5, pp. 571-785, 2000.

[Mustafa et al., 2004] N. K. Mustafa, Q. Zhi, A. D. Cheok, K. Sengupta, Z. Jian, K. C. Chung, "Lip Geometric Features for human computer interaction using bimodal speech recognition: comparison and analysis" Speech Communication, no. 43, pp. 1-16, 2004.

[Nefian et al., 2002] A. Nefian, L. H. Liang, L. Xiao, X. X. Liu, X. Pi, and K. Murphy, "Dynamic Bayesian networks for audio-visual speech recognition," EURASIP Journal of Applied Signal Processing, no. 11, pp. 1274-1288, 2002.

[Neti et al., 2000] C. Neti, G. Potamianos, J. Luettin, I. Matthews, and H. Glotin, D. Vergiry, "Audio-Visual Speech Recognition," Workshop Final Report, IBM T.J. Watson Research Center, 2000.

[Ortega et al., 2004] A. Ortega, F. Sukno, E. Lleida, A. Frangi, A. Miguel, L. Buera, and E. Zacur, "AV@CAR: A Spanish Multichannel Multimodal Corpus for In-Vehicle Automatic Audio-Visual Speech Recognition," International Conference on Language Resources and Evaluation, pp.763-766.

[Patterson et al., 2002] E. K. Patterson, S. Gurbuz, Z. Tufekci, and J. N. Gowdy, "CUAVE: A new Audio-Visual Database for multimodal Human Computer Interface Research," Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, 2002.

[Pearce, 2000] D. Pearce, "Enabling new speech driven services for mobile devices: An overview of the ETSI standards activities for distributed speech recognition front-ends", Proc. American Voice I/O Society (AVIOS), 2000.

[Petajan, 1985] E. D. Petajan, "Automatic lipreading to enhance speech recognition," IEEE Computer Vision and Pattern Recognition, pp. 44-47, 1985.

[Peters et al., 1999] D. Peters, P. Stubley, J.M. Valin, "On the limits of speech recognition in noise," Proc. ICASSP, paper 1026, 1999.

[Potamianos et al., 1998] G. Potamianos, H. P. Graf, and E. Cosatto "An Image Transform Approach for HMM Based Automatic Lipreading," International Conference on Image Processing, vol. 3, pp. 173 177, 1998.

[Potamianos et al., 2001] G. Potamianos, C. Neti, G. Iyengar, E. Helmuth, "Large-VocabularyAudio-Visual SpeechRecognition by Machines and Humans," Proc. Eurospeech, pp. 1027-1030, 2001.

[Potamianos et al., 2003] G. Potamianos, C. Neti G. Gravier, and A. Garg, "Recent advances in the Automatic Recognition of Audio-Visual Speech," Proc. of the IEEE, vol. 91, no. 9, 2003.

[Potamianos et al., 2004] G. Potamianos, C. Neti, J. Luettin, and I. Matthews, "Audio-Visual Automatic Speech Recognition: An Overview," In Bailly, E. Vatikiotis-Bateson, and P. Perrier (Eds.), "Issues in Visual and Audio-Visual Speech Processing," GMIT Press, 2004.

[Rabiner, 1989] L. R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in SpeechRecognition," Procedings of IEEE, vol. 77, no. 2 pp. 257-286, 1989.

[Renesas, 2006] Renesas (single-chip solution for car information systems): http://eu.renesas.com, 2006.

[Rosemblum and Saldaña, 1998] L. D. Rosemblum and H. M. Saldaña, "Time-varing information for visual speech perception," In R. Campbell B. Dodd, and D. Burnham, (Eds.), "Hearing by Eye II," Psychology Press Ltd. Publishers, pp. 61-81, 1998.

[Scalart and Filho, 1996] P. Scalart, J. Filho, "Speech Enhancement Based on apriori Signal to Noise Estimantion," Proc. ICASSP, pp. 629-632, 1996.

[Schmidbauer and Höge, 1991] O. Schmidbauer, and H. Höge, "Speaker Adaptation Based on Articulatory Features," Proc. Eurospeech, pp. 1099-1102, 1991.

[Singh and Stern, 2002] R. Singh, R. M. Stern, and B. Raj, "Signal and Feature Compensation Methods for Robust Speech Recognition," In Guillian M. Davis (Eds.), "Noise Reduction in Speech Applications," CRC Press LLC, pp. 219-243, 2002.

[Smith, 1982] A. R. Smith, "Paint," in Tutorial: Computer Graphics, John Beatty and Kellogg Booth, editors, IEEE Computer Society Press, 1982.

[Smith, 1997] S. W. Smith, "The Scientist and Engineer's Guide to Digital Signal Processing," California Technical Publishing, 1997.

[Stiefelhagen and Yang, 1996] R. Stiefelhagen, Jie Yang and A. Waibel, "A Model-Based Gaze Tracking System," Proc. IEEE Intl. Joint Symposia on Intelligence and Systems-Image, Speech and Natural Language Systems, Washington DC, USA, 1996.

[Sumby, 1954] W. H. Sumby, and I. Pollack, "Visual contribution to speech intelligibility in noise," Journal of Acoustic Society of America, vol. 26, no. 2, pp. 212-215, 1954.

[Summerfield, 1979] Q. Summerfield, "Use of visual information in phonetic perception," Phonetica, vol. 36, pp. 314-331, 1979.

[Tatsuno, 2006] K. Tatsuno, "Current Trends in Digital Cameras and Camera-Phones," Technical Report, Science & Technology Trends, Science and Technology Foresight Center, NISTEP, Quarterly Review, no. 18, January 2006.

[Teissier et al., 1999] P. Teissier, J. Robert-Ribes, J. Schwartz, A. Guérin-Dugué, "Comparing Models for Audiovisual Fusion in a Noisy-Vowel Recognition Task," Transactions on Speech and Audio Processing, vol. 7, no. 6, pp.629-642, 1999.

[van Ginneken et al., 2002] B. van Ginneken, A. F. Frangi, J. J. Staal, B. M. ter Haar Romeny, and M. A. Viergever , "Active Shape Models Segmentation with optimal Features," IEEE Transactions on Medical Imaging, vol. 21, no. 8, 2002.

[Varga et al., 2002] I. Varga, S. Aalburg, B. Andrassy, S. Astrov, J. G. Bauer, C. Beaugeant, C. Geissler, and H. Höge, "ASR in mobile phones - an industrial approach," IEEE Transactions on Speech and Audio Processing, vol. 10, pp. 562-569, 2002.

[VSR, 2002] VSR Very Smart Recognizer ®, "User Documentation 3.00," Siemens SpeechCenter, CT IC 5, Munich, 2002.

[Viola and Jones, 2004] P. Viola, and M. J. Jones, "Robust Real-Time Face Detection", International Journal of Computer Vision, no. 57, vol. 2, pp. 137-154, 2004.

[Wan and Van der Merwe, 2000] E. A. Wan, and R. Van der Merwe, "The Unscented Kalman Diler for Non-linear Estimation," Proc. of Symposium 2001 on Adaptive Systems for Signal Processing, Communications and Control, Lake Louise, Alberta, Canada, 2000.

[Wiener, 1949] N. Wiener, "Extrapolation, Interpolatin and Smoothing of Stationary Time Series," MIT Press, Cambridge, 1949.

[Wieghardt, 2001] J. Wieghardt, “Learning the Topology of Views; From Images to Objects,” Ph.D. Thesis, Ruhr University, 2001.

[Willian et al., 1997] J. Willian, J. Rutledge, D. Garstecki, and A. Katsaggelos, “Frame Rate and Viseme Analysis for Multimedia Applications,” Multimedia and Signal Process Conference, Princeton N.J., vol. 20, pp.7-23, 1997.

[Yang et al., 1999] H. Yang, S. V. Vuuren, H. Hermansky, "Relevancy of Time-Frequency Features for Phonetic Classification Measured by Mutual Information," Proc. ICASSP, paper 2454, 1999.