Multimodal Feature Extraction and Fusion for Audio-Visual Speech Recognition

THÈSE N^{O} 4292 (2009)

PRÉSENTÉE LE 16 JANVIER 2009

À LA FACULTE SCIENCES ET TECHNIQUES DE L'INGÉNIEUR

LABORATOIRE DE TRAITEMENT DES SIGNAUX 5

PROGRAMME DOCTORAL EN INFORMATIQUE, COMMUNICATIONS ET INFORMATION

ÉCOLE POLYTECHNIQUE FÉDÉRALE DE LAUSANNE

POUR L'OBTENTION DU GRADE DE DOCTEUR ÈS SCIENCES

PAR

Mihai GURBAN

acceptée sur proposition du jury:

Prof. S. Süsstrunk, présidente du jury
Prof. J.-Ph. Thiran, directeur de thèse
Dr S. Bengio, rapporteur
Prof. H. Hermansky, rapporteur
Prof. R. Reilly, rapporteur

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Suisse
2009

Don't be afraid to attempt the impossible. Simply knowing what is impossible is useful knowledge—and you may well find, in the wake of some unexpected success, that not half of the things we call impossible have any right at all to wear the label.

Michael Abrash

Contents

List of Figures

List of Tables

Abstract

Multimodal signal processing analyzes a physical phenomenon through several types of measures, or modalities. This leads to the extraction of higher-quality and more reliable information than that obtained from single-modality signals. The advantage is two-fold. First, as the modalities are usually complementary, the end-result of multimodal processing is more informative than for each of the modalities individually, which represents the first advantage. This is true in all application domains: human-machine interaction, multimodal identification or multimodal image processing. The second advantage is that, as modalities are not always reliable, it is possible, when one modality becomes corrupted, to extract the missing information from the other one.

There are two essential challenges in multimodal signal processing. First, the features used from each modality need to be as relevant and as few as possible. The fact that multimodal systems have to process more than just one modality means that they can run into errors caused by the curse of dimensionality much more easily than mono-modal ones. The *curse of dimensionality* is a term used essentially to say that the number of equally-distributed samples required to cover a region of space grows exponentially with the dimensionality of the space. This has important implications in the classification domain, since accurate models can only be obtained if an adequate number of samples is available, and obviously this required number of samples grows with the dimensionality of the features. Dimensionality reduction is thus a necessary step in any application dealing with complex signals, and this is achieved through selection, transforms or the combination of the two.

The second essential challenge is multimodal integration. Since the signals involved do not necessarily have the same data rate, range or even dimensionality, combining information coming from such different sources is not straightforward. This can be done at different levels, starting from the basic signal level by combining the signals themselves, if they are compatible, up to the highest decision level, where only the individual decisions taken based on the signals are combined. Ideally, the fusion method should allow temporal variations in the relative importance of the two streams, to account for possible changes in their quality. However, this can only be done with methods operating at a high decision level.

The aim of this thesis is to offer solutions to both these challenges, in the context of audio-visual speech recognition and speaker localization. Both these applications are from

the field of human-machine interaction. Audio-visual speech recognition aims to improve the accuracy of speech recognizers by augmenting the audio with information extracted from the video, more particularly, the movement of the speaker's lips. This works well especially when the audio is corrupted, leading in this case to significant gains in accuracy. Speaker localization means detecting who is the active speaker in a audio-video sequence containing several persons, something that is useful for videoconferencing and the automated annotation of meetings. These two applications are the context in which we present our solutions to both feature selection and multimodal integration.

First, we show how informative features can be extracted from the visual modality, using an information-theoretic framework which gives us a quantitative measure of the relevance of individual features. We also prove that reducing redundancy between these features is important for avoiding the curse of dimensionality and improving recognition results. The methods that we present are novel in the field of audio-visual speech recognition and we found that their use leads to significant improvements compared to the state of the art.

Second, we present a method of multimodal fusion at the level of intermediate decisions using a weight for each of the streams. The weights are adaptive, changing according to the estimated reliability of each stream. This makes the system tolerant to changes in the quality of either stream, and even to the temporary interruption of one of the streams. The reliability estimate is based on the entropy of the posterior probability distributions of each stream at the intermediate decision level. Our results are superior to those obtained with a state of the art method based on maximizing the same posteriors. Moreover, we analyze the effect of a constraint typically imposed on stream weights in the literature, the constraint that they should sum to one. Our results show that removing this constraint can lead to improvements in recognition accuracy.

Finally, we develop a method for audio-visual speaker localization, based on the correlation between audio energy and the movement of the speaker's lips. Our method is based on a joint probability model of the audio and video which is used to build a likelihood map showing the likely positions of the speaker's mouth. We show that our novel method performs better than a similar method from the literature.

In conclusion, we analyze two different challenges of multimodal signal processing for two audio-visual problems, and offer innovative approaches for solving them.

Keywords: multimodal signal processing, audio-visual speech recognition, feature selection, multimodal integration.

Version Abrégée

Le traitement des signaux multimodaux analyse un phénomène physique par plusieurs genres de mesures, ou modalités. Cela conduit à l'extraction d'information de meilleure qualité et plus fiable que celle obtenu à partir de signaux monomodaux. L'avantage est double. Tout d'abord, comme les modalités sont généralement complémentaires, le résultat final du traitement multimodal est plus informatif que pour chacune des modalités individuellement, ce qui constitue le premier avantage. Ceci est vrai dans tous les domaines d'application: l'interaction homme-machine, l'identification multimodale ou le traitement d'images multimodales. Le deuxième avantage est qu'il est possible, quand une modalité devient corrompue, d'extraire l'information manquante à partir de l'autre modalité.

Il y a deux défis essentiels dans le traitement des signaux multimodaux. D'abord, les attributs utilisés pour chaque modalité doivent être plus pertinents et aussi les moins nombreux possible. Le fait que les systèmes multimodaux doivent traiter plus qu'une seule modalité signifie qu'ils sont sujets aux erreurs causées par le phénomène communément appelé *malédiction de la dimensionnalité*, et ce de manière beaucoup plus critique que pour les systèmes monomodaux. La malédiction de la dimensionnalité est un terme traduisant le fait que le nombre d'échantillons également distribués nécessaires pour couvrir une région de l'espace croît de façon exponentielle avec la dimension de l'espace. Ceci a des implications importantes dans le domaine de la classification, car des modèles précis ne peuvent être obtenus que si un nombre suffisant d'échantillons est disponible, nombre croissant fortement avec la dimension des attributs. La réduction de dimensionnalité est donc une étape indispensable dans le développement de toute application traitant des signaux complexes. Cette phase est généralement réalisée par sélection, par transformation ou par la combinaison des deux.

Le second défi essentiel concerne l'intégration multimodale. Comme les signaux impliqués n'ont pas nécessairement la même fréquence, gamme de variation ou dimensionnalité, combiner les informations provenant de telles sources n'est pas trivial. Cela peut s'opérer à différents niveaux, à partir du niveau de signal de base en combinant les signaux eux-mêmes, s'ils sont compatibles, jusqu'au plus haut niveau de décision, où seules les décisions individuelles prises sur base des signaux sont combinées. Idéalement, la méthode de fusion devrait permettre des variations temporelles dans l'importance relative de ces deux flux, tenant ainsi compte des éventuelles modifications dans leur fiabilité. Mais ceci

ne peut se réaliser qu'avec des méthodes agissant à un niveau supérieur de décision.

L'objectif de cette thèse est de proposer des solutions à ces deux défis, dans le cadre de la reconnaissance audio-visuelle de la parole et la localisation du locuteur. Ces deux applications appartiennent au domaine de l'interaction homme-machine. La reconnaissance audio-visuelle de la parole vise à améliorer la performance des reconnaisseurs traditionnels de parole en ajoutant à l'audio des informations extraites de la vidéo, et plus précisément du mouvement des lèvres du locuteur. Ceci est particulièrement bénéfique lorsque le signal audio est corrompu, conduisant dans ce cas à des gains considérables de reconnaissance. La localisation du locuteur, quant à elle, vise à détecter le locuteur actif dans une séquence audio-vidéo contenant plusieurs personnes, ce que est utile en vidéoconférence et pour l'annotation automatique de réunions. Ces deux applications représentent le contexte dans lequel nous présentons nos solutions pour la sélection d'attributs et l'intégration multimodale.

Premièrement, nous montrons comment des attributs informatifs peuvent être extraits de la modalité visuelle, en utilisant le cadre de la Théorie de l'Information qui nous donne une mesure quantitative de la pertinence des attributs individuels. Nous prouvons aussi que la réduction de la redondance entre ces attributs est vitale pour éviter la malédiction de la dimensionnalité et améliorer les résultats de reconnaissance. Les méthodes que nous présentons sont originales dans le domaine de la reconnaissance audio-visuelle de la parole et nous avons obtenu des améliorations significatives par rapport à l'état de l'art.

Deuxièmement, nous présentons une méthode d'intégration multimodale au niveau des décisions intermédiaires en utilisant un poids pour chacun des flux. Les poids sont adaptatifs, évoluant en fonction de l'estimation de fiabilité de chaque flux. Cela rend le système tolérant aux changements de qualité des flux, voire à l'interruption temporaire de l'un des flux. L'estimation de la fiabilité se base sur l'entropie de la distribution des probabilités postérieures de chaque flux au niveau des décisions intermédiaires. Nos résultats sont supérieurs à ceux obtenus par une méthode de l'état de l'art basée sur la maximisation de ces mêmes postérieurs. De plus, nous analysons l'effet d'une contrainte typique imposée dans la littérature sur les poids des flux, selon laquelle ces poids doivent se sommer à l'unité. Nos résultats montrent que l'élimination de cette contrainte peut mener à des améliorations en terme de précision de reconnaissance.

Enfin, nous développons une méthode pour la localisation audio-visuelle du locuteur, basée sur la corrélation entre l'énergie audio et le mouvement des lèvres du locuteur. Notre méthode se fonde sur un modèle de probabilité jointe entre l'audio et la vidéo qui est utilisé pour construire une carte de probabilité indiquant les positions probables de la région buccale du locuteur. Nous montrons que notre méthode originale donne de meilleurs résultats qu'une méthode similaire de la littérature.

En conclusion, nous analysons deux défis différents du traitement des signaux multimodaux pour deux problèmes audio-visuels, et nous offrons des approches innovatrices afin de les résoudre.

Mots-clés: traitement des signaux multimodaux, reconnaissance audio-visuelle de la parole, sélection d'attributs, intégration multimodale.

Notations

Abbreviations

2D	two-dimensional
3D	three-dimensional
AO	Audio-Only
ASR	Automatic Speech Recognition
AV	Audio-Visual
AVSR	Audio-Visual Speech Recognition
CMI	Conditional Mutual Information
CMN	Cepstral Mean Normalization
DCT	Discrete Cosine Transform
DTW	Dynamic Time Warping
DWT	Discrete Wavelet Transform
EM	Expectation Maximization
FMN	Feature Mean Normalization
fps	frames per second
GMM	Gaussian Mixture Models
HMM	Hidden Markov Models
kbps	kilo bits per second
LDA	Linear Discriminant Analysis
LPC	Linear Predictive Coding
MFCC	Mel Frequency Cepstral Coefficient
MI	Mutual Information
ML	Maximum Likelihood
MSHMM	Multi-Stream Hidden Markov Model
PCA	Principal Components Analysis
pdf	probability density function
PLP	Perceptual Linear Predictive
RASTA	RelAtive SpectTrA
ROI	Region of Interest
TRAPs	TempoRAl Patterns
VO	Visual-Only

Introduction and Preview

1

1.1 Multimodal signals and systems

The word *multimodal* is used by researchers in different fields and often with different meanings. One of its most common uses is in the field of human-computer (or man-machine) interaction. Here, a *modality* is a natural way of interaction: speech, vision, face expressions, handwriting, gestures or even head and body movements. Using several such modalities can lead to *multimodal speaker tracking systems*, *multimodal person identification systems*, *multimodal speech recognizers*, or, more generally, *multimodal interfaces*. Such interfaces aim to facilitate human-computer interaction, augmenting or even replacing the traditional keyboard and mouse. Let us go into more detail about these particular examples.

Multimodal speaker detection, *tracking*, or *localization*, consists of identifying the active speaker in an audio-video sequence which contains several speakers, based on the correlation between the audio and the movement in the video [74]. In the case of a system with several cameras, the speakers might not necessarily be in the same image, but the underlying idea remains the same. The gain from multimodality over using for example only video is that correlations with the audio can help discriminate between the person that is actually speaking and someone which might be only murmuring something inaudible. It is also possible to find the active speaker using audio only, with the help of microphone arrays, but this makes the hardware setup more complex and expensive.

For *multimodal speech recognition*, or *audio-visual speech recognition (AVSR)*, some video information from the speakers' lips is used to augment the audio stream in order to improve the speech recognition accuracy [86]. This is motivated by human speech perception, as it has been proven that humans subconsciously use both audio and visual information when understanding speech [63]. There are speech sounds which are very similar in

the audio modality, but easy to discriminate visually. Using both modalities significantly increases automatic speech recognition results, especially in the case where the audio is corrupted by noise.

A *multimodal biometrics system* [94] establishes the identity of a person from a list of candidates previously enrolled in the system, based on not one, but several modalities, which can be taken from a long list: face images, audio speech, visual speech or lipreading, fingerprints, iris images, retinal images, handwriting, gait and so on. The use of more than one modality makes results more reliable, decreasing the number of errors. It should be noted here how heterogenous these modalities can be.

There are still other examples in the domain of human-computer interaction. *Multimodal annotation systems* are annotating systems typically used on audio-visual sequences, to add metadata describing the speech, gestures, actions and even emotions of the individuals present in these sequences. Indeed, *multimodal emotion recognition* [103] is a very active area of research, using typically the speech, face expressions, poses and gestures of a person to assess his or her emotional state.

However, all these applications are taken from just one limited domain, human-computer interaction. There are many other fields where different *modalities* are used.

For psychologists, *sensory modalities* represent the human senses (sight, hearing, touch and so on). This is not very different from the previous interpretation. However, other researchers can use the word *modality* in a completely different way. For example, linked to the concept of *medium* as a physical device, *modalities* are the ways to use such *media* [9]. The pen device is a medium, while the actions associated to it, like drawing, pointing, writing or gestures are all modalities.

Generally, the word *multimodal* is associated to the input of information. However, there are cases where the output of the computer is considered multimodal, as is the case of *multimodal speech synthesis*, the augmentation of synthesized speech with animated talking heads.

In the context of medical image registration, a *modality* can be any of a large array of imaging techniques, ranging from anatomical, such as X-ray, MRI or ultrasound, to functional, like fMRI or PET [60]. *Multimodal registration* is the process of bringing images from several such modalities into spatial alignment. The same term is used in remote sensing, where the modalities are images with different spectrums [122] (visible, infrared or microwave for example).

For both medical image analysis and remote sensing, the gain from using images from multiple modalities comes from their complementarity. More information will be present in a combined multimodal image that in each of the monomodal ones, provided they have been properly aligned. And this alignment is based on information that is common to all the modalities used, *redundant* information.

With all these different situations, defining what *multimodal* means in general is difficult. All *multimodal systems* extract meaning from multiple sources of information. *Multimodal signals* can be defined as signals originating from the same physical source [16] or phenomenon, but captured through different means. Such signals manifest some dependency,

which is present even if they might have been distorted, affected by noise or changed in other ways which would make it difficult to emphasize their common origin. The signals involved can be spatially varying, temporally varying, or both, but synchronized in some sense, and this aspect of temporal and spatial variation also needs to be taken into account when extracting the information from these signals. Multimodal signals can be at times redundant, complementary or even contradictory.

In this thesis, we will present methods used for both the extraction and the fusion of information from multiple modalities. In the next sections we will show why these problems are important and what we are aiming for when developing solutions to them.

1.2 Motivation

As shown above, the need to process signals from different modalities is becoming more and more common in various application fields. There are two reasons for this. The first reason for the complementarity of the information contained in those signals, that is, each signal brings extra information which can be extracted and used, leading to a better understanding of the phenomenon under study. The second reason is that multimodal systems are more reliable than monomodal ones, since, if one modality becomes corrupted by noise, the system can still rely on the other modalities to, at least partially, achieve its purpose.

To offer an example for the first advantage of multimodal systems, the complementarity of information, in multimodal medical image analysis, information that is missing one modality may be clearly visible in another. The same is true for satellite imaging. Here, by missing information we mean information that exists in the physical reality but was not captured through the particular modality used. By using several different modalities we can get closer to the underlying phenomenon, which might not be possible to capture with just one type of sensor.

Another example could be audio-visual speech recognition. Indeed, some speech sounds are easily confusable in audio, but easy to distinguish in video, and vice-versa. In conditions of noise, humans themselves lip-read subconsciously, and this also shows that there is useful information in the movement of the lips, information that is missing from the corrupted audio.

For the second advantage, the improved reliability of multimodal systems, consider the case of a three-modal biometric identification system, based on face, voice and fingerprint identification. When the voice acquisition is not optimal, because of a loud ambient noise or because the subject has a bad cold, the system might still be able to identify the person correctly by relying on the face and fingerprint modalities. Audio-visual speech recognition can also be given as an example, since, if for some reason the video becomes unavailable, the system should seamlessly revert to audio-only speech recognition.

There are two essential challenges in multimodal signal processing. The first is the extraction of relevant features from each modality, features that contain all the information needed while at the same being compact. The second is the fusion of this information, ideally in such a way that variations in the reliability of each modality stream affect the

final result as little as possible. We will now analyze in detail why each of these challenges is important.

The first challenge is the extraction of features which are at the same time relevant and compact. By *relevant* features we mean features that contain the information required to solve the underlying problem, thus the term *relevance* is always tied to the context of the problem. To offer a simple example, for both speech recognition and speaker identification some features need to be extracted from the audio signal. But their purpose is different. Features extracted for speech recognition should contain as much information as possible about what was being said, and as little as possible about who was speaking. Thus, some hypothetic ideal features that would respect this requirement would be very relevant for the speech recognition task, but irrelevant for speaker identification. For speaker identification, the situation is reversed, that is, the features should contain as much information as possible about who is speaking, and nothing about what was said.

Obviously, we would like features to contain all the relevant information in the signal, and at the same time retain as little superfluous information as possible.

The second requirement that we impose on the features is that they should be *compact*, that is, the feature vector should have a low dimensionality. This is needed because of the *curse of dimensionality*, a term defining the fact that the number of equally-distributed samples needed to cover a volume of space grows exponentially with its dimensionality. For classification, accurate models can only be built when an adequate number of samples is available, and that number grows exponentially with the dimensionality. Obviously, for training, only a limited amount of data is available, so having a low input dimensionality is desirable in order to obtain optimal classification performance.

Dimensionality reduction techniques work by eliminating the *redundancy* in the data, that is, the information that is common to several modalities, or, inside a modality, common to several components of the feature vector. The quest for compact features needs however to be balanced by the need for *reliability*, that is, the ability of the system to withstand adverse conditions like noise. Indeed, multimodal systems can be tolerant to some degree of noise, in the case when there is redundancy between the modalities. If one of the modalities is corrupted by noise, the same information may be found in another modality. In conclusion, reducing redundancy between the features of a modality is a good idea, however reducing redundancy between modalities needs to be done without compromising the reliability of the system.

Therefore, for the system to attain optimal performance, the features need to be relevant and compact.

The second challenge that we mentioned is multimodal *integration*, or *fusion*, which represents the method of combining the information coming from the different modalities. As the signals involved may not have the same data rate, range or dimensionality, combining them is not straightforward. The goal here is to gather the useful information from all the modalities, in such a way that the end-result is superior to those from each of the modalities individually. Multimodal fusion can be done at different levels. The simplest kind of fusion operates at the lowest level, the signal level, by concatenating the signals coming from

each modality. Obviously, for this, the signals have to be synchronized. High-level decision fusion operates directly on decisions taken on the basis of each modality, and this does not require synchronicity.

The multimodal integration method has an important impact on the reliability of the system. Ideally, the fusion method should allow variations in the importance given to the data streams from each modality, since the quality of these streams may vary in time. If one modality becomes corrupted by noise, the system should adapt to rely more on the other modalities, at least until the corrupted modality returns to its previous level of quality. This type of adaptation is however not possible with simple low-level fusion, but only with decision fusion.

In conclusion, a good integration method in a multimodal system can offer both higher performance compared to the monomodal systems and some tolerance to noise.

In the following we will detail our aims with respect to dealing with the two challenges.

1.3 Aim of the thesis

The aim of the thesis is two-fold, and closely related to the two challenges mentioned above. First, we want to analyze the methods to extract relevant features from multimodal signals, features which ideally capture the complementarity of the signals, while reducing the redundancy when needed. Second, we aim to develop multimodal fusion methods that can work in varying conditions, adapting to any changes in the environment.

Our aim for feature extraction was to find quantitative measures for both the relevance of features and their redundancy, and then develop methods to extract relevant and compact sets of features according to these measures. We found our needed measure in information theory. Mutual information between features and class labels is a good quantitative measure for the information content of features, that is, for the relevance of features to the problem. When applied only between features, mutual information is also a measure of redundancy. Intuitively, mutual information can be seen as a measure of dependency between random variables, but, unlike statistical correlation which measures only linear dependency, mutual information can also show if two variables are related in a nonlinear way.

Let us go back to our example of the difference between speech recognition and speaker identification. Assume that we could find a feature transform that could separate the feature set into features relevant for speech and features relevant for identification. In this case, mutual information would be a good measure to separate the two types of features. Indeed, the mutual information of the speech-specific features with the phoneme label variable would be high, while the mutual information of the same features with the speaker variable would be low.

For the case of redundant features, consider two features for lipreading, computed from the movement of the mouth, one on the left half of the mouth and the other on the right. This is an extreme case, as, because of the symmetry of the mouth, the two features should be very similar. The high value of the mutual information between these two features will reflect their high redundancy.

For multimodal fusion, our aim was to find a way to determine the instantaneous reliability of a stream in order to be able to estimate the importance that needs to be assigned to it. We also wanted to find a modality-independent estimator, thus having a method which is general and works for a large family of applications. The measure that we use also comes from information theory, the entropy of the class posterior probabilities for each stream. This measure shows how flat or "peaky" this distribution is. We argue that a flat posterior distribution shows that the modality is not very reliable and that we should have a low confidence in the probability distribution arising from it. This reasoning leads to a method in which weights are assigned to each modality, weights that vary in time according to our reliability estimates.

In the case of audio-visual speech recognition, the class posterior probabilities that we mention are the probabilities of individual phonemes. For biometric identification, they could represent the probabilities of a match between the test subject and each of the individuals enrolled in the database. Going back to our three-modal biometric identification example, a few pages back, consider that this posterior distribution is flat for the fingerprint modality, but has a clear peak for the other two modalities. This would show that the fingerprint modality is corrupted and our confidence in it should be low, while we can have a high confidence in the other two modalities, speech and face. The entropy for the fingerprint modality would in this case be high, while for the other modalities it would be low. There is thus an inverse relation between the entropy of the posteriors for a modality and the confidence assigned to it.

While we focused our efforts on audio-visual signals and in particular speech as an application, the methods developed here are very general and can be easily applied on a multitude of problems where different modalities are used.

1.4 Main contributions

The main contributions of this thesis can be summarized as follows:

- A new low-dimensional set of visual features for audio-visual speech recognition derived from the motion of the mouth.

  Aiming to find visual features for audio-visual speech recognition which are simple, very low-dimensional but also very robust, we develop a method of extracting motion features from the difference of optical-flow vectors on the region of the mouth. Our features are robust in the sense that they should tolerate small errors in the localization of the mouth, some degree of head turning and partial mouth occlusion. Although our feature is only two-dimensional, a significant improvement over audio-only speech recognition is reported.

- A novel approach to information-theoretic multimodal feature extraction, one that also takes into account the redundancy between features.

We approach the problem of multimodal dimensionality reduction from an information theoretic perspective, using mutual information as a measure of both relevance of an individual feature and also redundancy between features. Our application is audio-visual speech recognition, a field in which information-theoretic methods have been previously explored, but only for evaluating the relevance of features, not for eliminating redundancy. We prove through experimental results that better recognition rates can be achieved when redundancy is also eliminated from the feature set.

- A multimodal fusion method which can adapt automatically to the degradation of a stream.

  We develop a method of multimodal fusion which uses time-varying weights automatically adapting to the relative quality of each stream. If one of the streams is degraded, the weights immediately adjust by reducing the degraded stream's importance and favoring the reliable stream. This adaptive adjustment is done based on reliability measures derived from entropy, a method which is modality-independent and could in theory be applied to any multimodal integration problem.

- An audio-visual speaker localization method based on the correlation between optical flow variations in the video and the sound.

  Based on our low-dimensional motion features, we also develop a method of multimodal speaker localization, which is able to find the mouth of the speaker in an image by finding correlations between the relative movement in the video and the sound. Our method is simple and does not require detecting the face prior to localizing the mouth of the speaker. We compare our results with another method from the literature, on the same database, and show significant improvements.

1.5 Outline

This dissertation is organized as follows:

Chapter 2 An overview of audio-visual speech recognition.

Our main application for both multimodal feature selection and also multimodal integration is audio-visual speech recognition, which uses the video of the speaker's lips to improve the quality of audio speech recognition. In this chapter we give the context of the problem of AVSR and present the types of preprocessing done on both audio and video, the recognizers which are used and the typical methods for audio-visual integration. We will place more emphasis on the methods actually used in our experiments and to methods similar to ours.

Chapter 3 An overview of feature selection methods, and in particular selection methods using information theoretic measures.

We present an overview of feature selection methods used in the general classification domain. We present the broad groups of feature selection and extraction methods, and then focus on one particular type of methods, the ones using information theoretic measures. The measure used here is mutual information, which can estimate both the relevance of one particular feature with respect to the classification problem, but also the redundancy between two features.

Chapter 4 Our feature extraction methods applied on AVSR.

Here we present two novel methods of feature extraction for AVSR. The first method extracts very simple features with a low dimensionality from the motion of the speaker's lips. These features are differences of optical flow vectors, and should be more robust than other types of features typically used in AVSR. The second method uses the information theoretic measures presented in the previous chapter to find features which are individually informative, but also complementary, that is, each brings information that is not already present in the other chosen features.

Chapter 5 Our multimodal fusion method.

We devise an integration algorithm which adjusts automatically to the perceived reliability of each stream and varies their relative importance accordingly. The algorithm uses time-varying weights which are derived from the entropy of classifier outputs for each stream.

Chapter 6 Our multimodal speaker localization method.

This chapter shows how a very simple method can find the correlation between the movement of a speaker's mouth and the speech sounds. Once trained, a probabilistic classifier can find this type of correlation anywhere in the image, locating the active speaker without the need for a face detector.

Chapter 7 Conclusion.

We give a summary of our achievements and a discussion of possible future applications and research topics.

Part I

State of the art and background

An Overview of Audio-Visual Speech Recognition

2

2.1 Introduction

As mentioned in the previous chapter, our aims are to find methods to extract relevant information from different modalities and analyze ways to fuse this information. Audio-visual speech recognition (AVSR) [86] is a multimodal classification problem which is well-suited to this analysis, since it involves time-varying signals from two modalities, audio and video, which have different rates and properties. For example, they have different temporal resolutions, audio having a sampling rate of tens of kHz, while video has thousands of times less, that is, only tens of temporal samples per second. They also have a different dimensionality, as video has two spatial dimensions and a temporal one, while audio only has a temporal dimension.

AVSR uses visual information derived from the video of the speaker, in particular from the mouth region, to improve the audio speech recognition results, especially when the audio is corrupted by noise. This can be done because the audio and the video are complementary in this case, that is, the phonemes that are easily confused in the audio modality are more distinguishable in the video one, and vice-versa. Pioneered by Petajan in 1984 [78], AVSR is a quite active area of research.

AVSR performance can be affected by the curse of dimensionality since the dimensionality of the visual feature vectors is obviously quite high. Taking into account the fact that the visual modality contains less relevant information than the audio, it's clear that reducing its size should be a priority in AVSR.

Silence	/sil/, /sp/
Lip-rounding based vowels	/ao/, /ah/, /aa/, /er/, /oy/, /aw/, /hh/ /uw/, /uh/, /ow/ /ae/, /eh/, /ey/, /ay/ /ih/, /iy/, /ax/
Alveolar semivowels	/l/, /el/, /r/, /y/
Alveolar-fricatives	/s/, /z/
Alveolar	/t/, /d/, /n/, /en/
Palato-alveolar	/sh/, /zh/, /ch/, /jh/
Bilabial	/p/, /b/, /m/
Dental	/th/, /dh/
Labio-dental	/f/, /v/
Velar	/ng/, /k/, /g/, /w/

Table 2.1 — Phoneme to viseme mapping.

2.2 The multimodality of speech perception

Humans usually deal with the problem of noise by using visual information. It is a well-known fact that human speech perception is bimodal in nature. For example, showing a video of a person saying /ba/ while, at the same time, playing a recording of the same person saying /ga/ would make the audience perceive the sound /da/. This is called the *McGurk effect* [63]. It is also known that some hearing impaired and deaf persons can reach almost perfect speech perception by only seeing the speaker's face, or particularly the region of the mouth [114]. For everyone, visual information complements the audio signal not only when noise is present, but also in clean environments [113].

The reason why the visual modality is important is that it offers complementary information about the place of articulation. This is because the articulators (lips, teeth, tongue) are visible. Seeing them can help distinguish for example /p/ from /k/, /b/ from /d/ or /m/ from /n/, since all these pairs are easy to confuse from audio only.

The basic unit describing how speech conveys linguistic information is the *phoneme*. A phoneme is a speech sound generated by a particular configuration or movement of the voice tract articulators. There are approximately 42 phonemes in American English. But since not all the articulators are visible, many phonemes are indistinguishable visually. The units that can be visually distinguished are called *visemes* and their number is much smaller than that of the phonemes. Phoneme to viseme mappings can be derived from human speechreading studies or through statistical clustering techniques. Unfortunately no universal agreement about the precise mapping has been reached. A typical mapping [86] is depicted in Table 2.1.

In real-world AVSR systems, phonemes are the unit of choice. Having two segmentations, one into phonemes for an audio recognizer, and another into visemes, is considered too costly because it complicates audio-visual fusion and is also not bringing any clear benefit [86]. Few methods of audio-visual fusion allow for different segmentations and models to be used for the two modalities. In our system the speech classes are identical for both

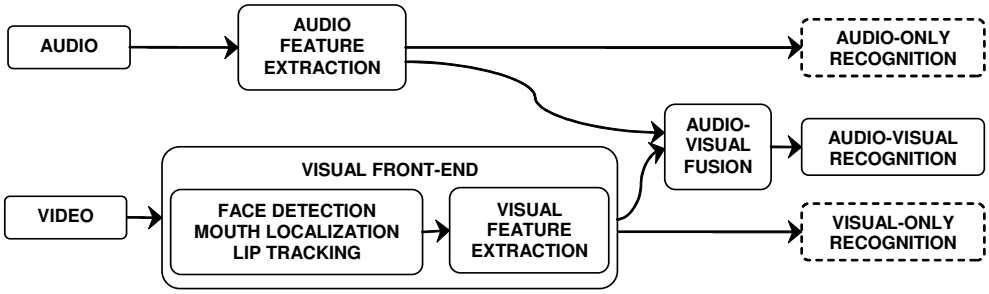

Figure 2.1 — The general structure of an audio-visual speech recognition system.

visual and audio recognition.

2.3 The structure of an audio-visual speech recognizer

In this section we briefly present the structure of an audio-visual speech recognition system. While all such systems share common traits, they can differ in three major respects. The first one is the visual front-end; i.e., the part of the system that tracks the region of the mouth and extracts the visual features. The second one is the audio-visual integration strategy, that is, the way audio and visual information are put together in order to reach a decision about the recognized word. Finally, the type of speech recognition system can differ depending on the particular task (isolated-word recognition, continuous speech or large-vocabulary speech recognition).

In Figure 2.1, we present a typical AVSR system with its components. First, the audio processing unit extracts mel-frequency cepstral coefficients (or similar attributes well-suited for speech recognition). In parallel, in the video modality, the face of the speaker has to be localized, the region of the mouth segmented and normalized and relevant features need to be extracted. Typically, in order to include some temporal information, the first and second derivatives of these features (for both modalities) are added to the feature streams. Then, both these feature streams are "run through" phoneme recognizers, to obtain phoneme-level likelihoods. The models used here are typically gaussian mixtures, but neural networks are also quite common [14]. Finally, the likelihoods are used to find the most probable path through a set of Hidden Markov Models [88], to obtain in the end the recognized words. The fusion of information between modalities can happen at any of these three stages, but, irrespective of this, in all cases having low-dimensional features is important.

2.4 The audio front-end

The audio front-end of an AVSR recognizer is identical to that of an audio-only speech recognition system. We present here a brief overview on audio feature extraction for speech recognition, since this subject is not within the scope of this thesis. A much more detailed presentation can be found in [34] and [89].

Like in any classification problem, features that are relevant to the given task need to be extracted from the underlying signal, in this case, the audio. These features should ideally contain all the information about the speech and as little as possible about the audio environment (audio noise, reverberations) or the identity of the speaker. However, the latter goal is typically not accomplished, as the same features used for speech recognition can also be employed in speaker identification, meaning that a large amount of the speaker-specific information is still left in the extracted features.

Another goal of the audio front-end is reducing the bit rate of the signal, so that it is more easily processed by the classifier. This is possible since there is a lot of redundancy in the initial audio signal. For example, starting with a single-channel audio signal digitized at 22kHz with 8 bit samples (for a total rate of 176kbps), the typical features extracted are 13 single-precision floating-point coefficients at the much lower frequency of 100 Hz, for a total rate of 21kbps. The rate of the extracted coefficients remains the same even for higher sampling rates of the audio, as for example CD-quality sound at 44kHz with 16 bit sampling, which has a bit rate of 704kbps.

Audio features used for speech recognition need to estimate the spectral envelope of the speech signal, which can be interpreted as a smoothed version of the magnitude spectrum. It can be obtained using one of two main methods, either through *cepstral analysis* or through *linear predictive coding*. Both methods are applied on short temporal windows between 10 and 40 ms, overlapping so that the frequency at which features are extracted is 100 Hz. Typically Hamming windows are used, and the overlap is around 75%.

The cepstrum of a signal is obtained by applying the discrete cosine transform (DCT) on the logarithmic spectrum [89]. This leads to coefficients which are uncorrelated and thus easily modeled with diagonal gaussians, as opposed to spectral coefficients which are highly correlated. Another advantage of the cepstrum is that, by truncating it, a smooth version of the spectrum can be obtained. The truncation also leads to a smaller number of coefficients, which reduces dimensionality.

Linear Predictive Coding [62] [61] is a method to estimate an autoregressive all-pole model of the short-term power spectrum of speech. A good model can approximate the high-energy areas of the spectrum, while smoothing out the less relevant spectral details. However, there are several shortcomings. First, only non-nasalized sounds can be modeled closely, as nasal sounds require zeros. Second, it approximates equally well at all frequencies, which is inconsistent with human speech perception, although that is also true for the cepstrum.

Modern speech features also take into account the way that human hearing works. The justification would be that speech evolved in a way that is adapted to the ear, so any details that are indiscernible for a human would also be irrelevant for a speech recognizer. The features that include properties of hearing are also based on LPC or the cepstrum.

Mel frequency cepstral coefficients (MFCCs) [65] [20], introduced in 1976, are probably the most commonly used audio features in speech recognition. MFCCs are cepstral coefficients derived from a triangular filter bank, spaced on a linear logarithmic frequency axis, the *Mel scale* [112] [111]. The Mel scale is a scale of pitches subjectively judged by

listeners to be equally spaced. From around 500Hz, larger and larger intervals are perceived to produce equal pitch increments, and this property is exploited when building the MFCC filter bank.

Perceptual Linear Predictive (PLP) coefficients [45] are based on three concepts from hearing psychophysics: the critical-band spectral resolution, the equal loudness curve and the intensity-loudness power law. This is achieved by warping the spectrum into the Bark frequency [123] and convoluting with the power spectrum of the simulated critical band, pre-emphasizing the samples by the simulated equal loudness curve [93] and finally applying a cubic-root amplitude compression, which approximates the power law of hearing [110]. An extension of the PLP method is RASTA-PLP (RelAtive SpectTrA PLP) [46], where the conventional critical-band short-term spectrum is replaced with a spectral estimate in which each frequency channel is band-pass filtered by a filter with a sharp zero at zero frequency. This has the effect of eliminating any constant or slowly varying component in each frequency channel. This is again inspired from human perception, and makes the recognizer relatively robust to variations in noise conditions.

The two types of audio features presented last, MFCCs and PLP coefficients, show better performance in the recognition of speech than the ones mentioned before and are most commonly used in such systems. However, there are also other aspects that need to be taken into consideration when preparing the input data for a speech recognizer.

As speech is a time-varying signal, there is also a lot of information in its relative variation, justifying the need for dynamic features. Introduced in [30], dynamic features are typically first and second order temporal derivatives generated by applying a polynomial regression over a temporal window of the signal. They are used in virtually all speech recognition systems together with static features, as they improve the accuracy of speech recognition especially in mismatched conditions.

Another type of dynamic speech features are TempoRAl Patterns (TRAPs) [47] [44], which are long (500-1000ms) and frequency-localized (1-3 Bark) overlapping time-frequency regions of the signal, spanning more than a whole phoneme, whose length is at least 300ms. Taking an even longer interval is justified by the need to remove information about slowly varying noise.

Time-invariant frequency distortions in the data can also be eliminated through other means. Cepstral Mean Normalization (CMN) introduced in [3] is a post-processing method which implies simply removing the cepstral coefficient mean computed over the time of the entire utterance. This has the effect of removing the influence of the microphone and transmission system on the data, making the recognition system more robust.

To conclude, the audio front-end of speech recognition systems has been a research theme for tens of years already and the types of features that are used are quite well established. In this thesis we focus our efforts on the extraction of visual features in a way that complements the audio. Our audio features are the ones commonly used for audio-only speech recognition, MFCCs with first and second temporal derivatives, with CMN applied on them.

2.5 The visual front-end

All audio-visual speech recognition systems require the identification and tracking of the region of interest (ROI), which can be either only the mouth, or a larger region, like the entire face. This typically begins with locating the face of the speaker, using a face detection algorithm. The second step is locating the mouth of the speaker and extracting the region of interest. This region can be scaled and rotated such that the mouth is centered and aligned.

Once the ROI has been extracted, the useful information that it contains needs to be expressed using as few features as possible. This is because the high dimensionality of the ROI impairs its accurate statistical modeling. Three main types of features are used for visual speech recognition [86]:

- Appearance based features, extracted directly from the pixels of the ROI.
- Shape based features, extracted from the contour of the speaker's lips.
- Joint appearance and shape features, the result of combining the two previous types.

In the following we will present each feature type in detail.

2.5.1 Appearance-based visual features

In the appearance-based approach, the pixels of the ROI themselves are used as features. The problem of finding the relevant information in the image is left for the classifier to solve. The ROI can be strictly the smallest rectangle which contains the mouth, or a larger rectangle containing the cheeks and the chin.

Although it may seem that locating the ROI does not need to be done with very good precision in this case, it has been proven that the normalization of the ROI improves recognition rates [75]. This normalization means that the mouth is centered and horizontally aligned, and also scaled. This should make recognition be invariant to small movements of the speaker's head and to the distance between the speaker and the camera. The normalization is done by tracking at least the two corners of the mouth.

Usually the dimensionality of the obtained feature-vector is too large to allow statistical modeling. Dimensionality reduction is necessary, and can be achieved for example by low-pass filtering and subsampling. But the most popular dimensionality reduction methods are the image transforms. Principal components analysis (PCA), the discrete cosine (DCT) or wavelet (DWT) transforms, the Haar or Hadamard transforms, all can be used to achieve this goal [86]. Their use is inspired from image compression, where they are used to reduce the size of images by eliminating redundancy while also keeping the perceived quality constant. However, there is no guarantee that they are appropriate for dimensionality reduction in classification, as they might not preserve the relevant information for speech recognition.

A transform which does capture some relevant information is the LDA (Linear Discriminant Analysis). Unlike the previously mentioned transforms, it uses information about the

classes in which the data is organized (in the case of speech, the phonemes/visemes) to maximize the ratio between the inter-class variance and the intra-class variance. The LDA is a common dimensionality reduction method in classification, and also extensively used in AVSR.

A special type of appearance-based features are those that take into account the visual motion during speech production. This motion can be represented either by delta images (the difference between two consecutive frames) or by the optical flow [37]. These dynamic features should be more robust, as they are invariant to skin color or illumination conditions. Combining dynamic and static features is also possible.

In the following, we present in more detail two types of features that we used, the DCT in section 2.5.3 and the optical flow in section 2.5.4. Since they are feature extraction methods, both PCA and LDA will be discussed in more detail in Chapter 3.

2.5.2 Shape-based visual features

For shape-based visual feature extraction, it is assumed that the speech information lies in the contours of the lips, or more generally, the contours of the lips, jaws and cheeks. Just as with delta or optical flow, these features should be invariant to skin color or illumination. But extracting them reliably proves to be a difficult task.

The contours can be defined as parametric curves: B-Spline, Bézier, ellipses, snakes, active contours. They are guided by minimizing a cost function over the area enclosed by the curves [75].

From these contours, a number of high-level features can be extracted: height, width, perimeter of the contour, or the size of the enclosed area [86] [75]. They contain a significant amount of speech information, but the contours themselves are not always correctly estimated. This happens especially in the cases when the color of the lips is close to that of the surrounding skin, or when the lighting is not uniform and shadows appear. Facial hair can also pose problems.

A way of improving the quality of the contours is by using models, like lip templates or active shape models (ASMs). ASMs are statistical models representing an object by a set of labeled points [59]. But to obtain such a model, a relatively large number of points has to be marked by hand on the images in the training phase.

Combining the low-level pixel-based features with the high-level shape features can improve recognition results. This can be done by either simply concatenating the two vectors or by deriving a model which would take them both into account.

In general, the use of shape features requires a good lip tracking algorithm and makes the limiting assumption that speech information is concentrated in the contour of the lips alone. Several articles report that DCT features outperform shape based ones [84, 92]. Both DCT features and motion-based features like the ones we will propose in Chapter 4 fall into the first category, as no lip contour is extracted.

2.5.3 The discrete cosine transform

Since the DCT is one of the most used transforms in AVSR, we present it here in more detail.

The discrete cosine transform (DCT) is a real and orthogonal transform. Its $N \times N$ cosine transform matrix $C = \{c(k,n)\}$ is defined as ([51]):

$$c(k,n) = \begin{cases} \frac{1}{\sqrt{N}}, & k = 0,\ 0 \le n \le N-1 \\ \sqrt{\frac{2}{N}} \cos \frac{\pi(2n+1)k}{2N}, & 1 \le k \le N-1,\ 0 \le n \le N-1 \end{cases} \tag{2.1}$$

where N is the width and height of a square image.

As a separable unitary transform, it is computed as:

$$v(k,l) = \sum_{m,n=0}^{N-1} \sum c(k,m)u(m,n)c(l,n), \qquad 0 \le k,l \le N-1 \tag{2.2}$$

where u is the original image, while v is the transformed one. In matrix form, $\mathbf{V} = \mathbf{CUC}^T$, where $\mathbf{U} = \{u(k,l)\}$, $\mathbf{V} = \{v(k,l)\}$ and $\mathbf{C} = \{c(k,l)\}$.

The DCT has excellent energy compaction [91] for highly correlated data and is also a fast transform [51] [24]. Because of all this, the DCT is used for video compression in the JPEG and MPEG standards [31]. Here, images are split in square 8x8 blocks, for each of which the DCT is computed. Because of the correlation in the images, many of the coefficients of the transformed image will be close to zero. The non-zero coefficients are typically grouped in the upper-left corner of the transformed image, where the lower spatial frequencies are represented. When arranging the bi-dimensional image as a one-dimensional vector, a zig-zag ordering is used, as shown in Figure 2.2, insuring that most of the zeros are grouped at the end of the one-dimensional vector. This property is used for image compression, where the resulting vector of real numbers is quantized with a variable number of bits, and the trailing zeros are ignored. For dimensionality reduction in visual speech recognition, it is common to simply use only the beginning of this vector, the part that contains the coefficients with the highest amplitudes [84].

Another way of reducing the number of coefficients, also used in AVSR, is by ordering them according to the total energy, over the whole training set, and then keeping only the highest-energy ones [84]. First the transformed images $v_j(k,l)$ with $0 \le k,l \le N-1$ are arranged into vectors $g_j(n)$, with $0 \le n \le N^2-1$ and $1 \le j \le J$, the total number of images in the training set. The energy for one position in the vector g is defined as:

$$E_n = \frac{1}{J} \sum_{j=1}^{J} [g_j(n)]^2 \tag{2.3}$$

If the M largest energies are $\{E_{n_1} ... E_{n_M}\}$, with $M \ll N^2$, then the reduced dimensionality feature vector $h_j(m)$ is

$$h_j(m) = g_j(n_m) \tag{2.4}$$

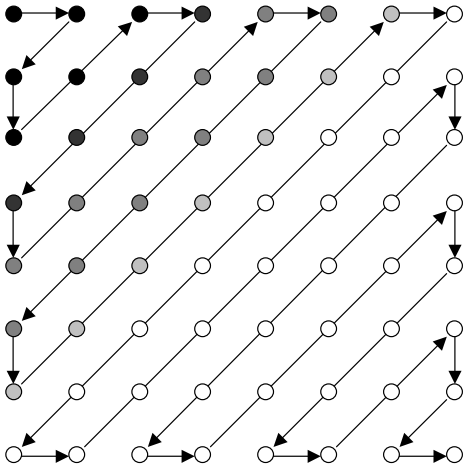

Figure 2.2 — The zig-zag coefficient ordering scheme for DCT. High-amplitude coefficients are grouped in the upper-left corner, where the lower frequencies are represented.

that is, only the coefficients which are likely to have a higher energy are kept. The resulting vector should contain most of the relevant speech information in a much reduced number of coefficients.

Typically, the DCT is only a step in the extraction of visual features for AVSR. It was shown that having a ROI that is properly centered, rotated and scaled can greatly improve recognition results [75]. Also, previous research [83] shows that the coefficients in the odd columns of the DCT have a much higher relevance, because of the symmetry of the mouth. Using only the odd columns is equivalent to imposing horizontal symmetry to the image.

Finally, a procedure similar to Cepstral Mean Normalization (mentioned in Section 2.4) can be applied on the DCT features to reduce the influence of speaker particularities and lighting conditions which are constant for a whole utterance. Feature Mean Normalization (FMN) [84] [81] is simply removing the mean from each element of the feature vector, over a time window or even a whole utterance. In [84], FMN is shown to significantly improve recognition results in speaker-independent contexts, but not for the single-speaker case.

2.5.4 The optical flow

The optical flow reflects the image changes due to motion during a time interval. The optical flow is the velocity field representing the three-dimensional motion of object points across a two dimensional image. As the base for higher level computation, optical flow can offer valuable information about the motion of the camera or of the objects in the scene, the parameters of these motions and the relative distances between the objects.

Optical flow computation is based on two assumptions [109]:

- The brightness of any object point is constant over time.
- The movements of neighboring points are similar (the *velocity smoothness* constraint).

A continuous image function $f(x, y, t)$ giving the gray-level at position (x, y) and time t, can be expressed a a Taylor series:

$$f(x + dx, y + dy, t + dt) = f(x, y, t) + \frac{\partial f}{\partial x}dx + \frac{\partial f}{\partial y}dy + \frac{\partial f}{\partial t}dt + \cdots \tag{2.5}$$

Considering that the point (x, y) is translated some small distance (dx, dy) during the interval dt, and assuming its brightness did not change, $f(x+dx, y+dy, t+dt) = f(x, y, t)$. Assuming dx, dy are small, the higher-order terms vanish:

$$-\frac{\partial f}{\partial t} = \frac{\partial f}{\partial x}\frac{dx}{dt} + \frac{\partial f}{\partial y}\frac{dy}{dt} \tag{2.6}$$

The partial derivatives of f can be approximated from the images themselves. The goal is to compute the velocity vector $\mathbf{c} = \left(\frac{dx}{dt}, \frac{dy}{dt}\right) = (u, v)$. But equation (2.6) does not specify this vector completely. To solve this, the velocity smoothness is introduced by minimizing this squared error:

$$\begin{aligned} E^2(x, y) = &\left(\frac{\partial f}{\partial x}u + \frac{\partial f}{\partial y}v + \frac{\partial f}{\partial t}\right)^2 + \\ &\lambda\left[\left(\frac{\partial u}{\partial x}\right)^2 + \left(\frac{\partial u}{\partial y}\right)^2 + \left(\frac{\partial v}{\partial x}\right)^2 + \left(\frac{\partial v}{\partial y}\right)^2\right] \end{aligned} \tag{2.7}$$

where λ is a Lagrange multiplier.

Unfortunately, in real images, the two assumptions (constant brightness and velocity smoothness) are violated quite often. A good example is a static sphere which is illuminated by a moving light source. The computed optical flow will show some movement, although the sphere is not moving. In real image sequences, the true velocity field changes abruptly around moving boundaries or depth discontinuities. Depending on the method used to compute the optical flow, the errors can propagate across the entire field, leading to poor optical flow estimates.

Two of the most popular optical flow computation methods are the Horn-Schunck [49] [55] and the Lucas-Kanade [58] methods.

In AVSR, a downsampled optical flow field can simply be used instead of other features, as shown in [37]. Downsampling is necessary as the optical flow field has two components per pixel, vertical and horizontal, effectively doubling the resolution.

2.6 Hidden Markov models

Many different classifiers have been applied to the area of speech recognition, which is a difficult classification task due to the fact that the signals involved are time varying and of different temporal lengths.

The simplest classifier that can be used for isolated word recognition is template matching, where classification is based on the distance between the test word and stored temporal

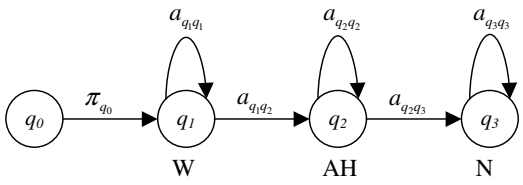

Figure 2.3 — A left-right HMM for the word "one".

patterns representing the learned words. This technique was used for AVSR in [78]. However, the speaking rate needs to be more or less the same, otherwise their length would differ and the method would not work.

Dynamic Time Warping (DTW) [95] is a time normalization technique which aims to solve the problem of nonlinear temporal fluctuations caused by speech rate variation. DTW works best on small tasks and for isolated word recognition, and can also be applied to connected speech recognition, as shown in [106]. However, it was replaced by Hidden Markov Models (HMMs) which became the dominant classifier used in virtually all speech recognition systems.

We will give here a very brief overview of HMMs and their use in speech recognition. More detail can be found in [87] [88].

HMMs can model the behavior of systems which can switch between states stochastically. In a discrete, first order *Markov chain*, the probability of being in a particular state at a particular time depends only on the state itself and the previous state. In each of these states, the system emits a symbol which can be observed, the *observation*. In HMMs, the state itself is always hidden and only the observation is visible, thus the name *Hidden* Markov Models.

In speech recognition HMMs can be used to model the temporal evolution of the speech signal. In small vocabulary systems each word can be modeled by an HMM, while in large vocabulary tasks each speech sound has a corresponding HMM. Most recognition systems use left-right models, or Bakis models [4], in which states that have been left are never visited again and the system always progresses from an initial state towards a final one. The observations are audio feature vectors as described in Section 2.4.

The purpose of the recognition process is to choose the most likely word model, given an observation sequence. The word attached to this model is the recognized word. To this end the Viterbi algorithm is employed [88].

Let us take as an example an isolated word speech recognizer, having left-right HMMs as word models. Such a word model is given in fig. 2.3. Let us define $a_{q_iq_j}$ as the transition probability from state q_i to state q_j. The initial transition probability is considered equal to one ($\pi_{q_0} = 1$). Let us also define the likelihood that observation O_t was emitted by state q_t as $b_{q_t}(O_t)$. The likelihood of the observation sequence $\mathbf{O} = O_1O_2 \cdots O_T$, given a path $\mathbf{Q} = q_1q_2 \cdots q_T$ in the model ω, is:

$$p(\mathbf{O}|\mathbf{Q},\omega) = b_{q_1}(O_1)b_{q_2}(O_2)\cdots b_{q_T}(O_T), \tag{2.8}$$

assuming the statistical independence of the observations. The probability of the path itself is given by:

$$P(\mathbf{Q}|\omega) = \pi_{q_0}a_{q_1q_2}a_{q_2q_3}\cdots a_{q_{T-1}q_T}. \tag{2.9}$$

The joint probability of $\mathbf{O}$ and $\mathbf{Q}$ occurring simultaneously is the product of the two:

$$p(\mathbf{O},\mathbf{Q}|\omega) = \prod_{q_i\in\mathbf{Q}} b_{q_i}(O_i)\cdot\pi_{q_0}\prod_{(q_i,q_j)\in\mathbf{Q}} a_{q_iq_j}. \tag{2.10}$$

The likelihood of the observation sequence given the model is the sum of these joint probabilities over all possible state sequences $\mathbf{Q}$:

$$p(\mathbf{O}|\omega) = \sum_{\text{all }\mathbf{Q}} p(\mathbf{O},\mathbf{Q}|\omega)). \tag{2.11}$$

This "full" likelihood can be replaced by the "Viterbi" approximation, considering only the most likely path in the model:

$$p(\mathbf{O}|\omega) \simeq \max_{\mathbf{Q}}[p(\mathbf{O},\mathbf{Q}|\omega)]. \tag{2.12}$$

This often-used approximation does not lead to a significant performance loss, while facilitating the numerical computation. In the end, the recognized word is given by the most likely word model:

$$\omega_{\text{recognized}} = \arg\max_{\omega}[p(\mathbf{O}|\omega)]. \tag{2.13}$$

The observation probabilities are modeled using Gaussian Mixture Models (GMMs) [13], in this way:

$$b_j(o_t) = \sum_{m=1}^{M} c_{jm}N(o_t;\mu_{jm},\Sigma_{jm}) \tag{2.14}$$

where $N(o;\mu,\Sigma)$ is the value in o of a multivariate gaussian with mean μ and covariance matrix Σ. M gaussians are used in a mixture, each weighed by c_{jm}. Typically, in speech recognition systems, the gaussians have a diagonal covariance matrix, as the coefficients used are assumed to be uncorrelated.

For the training phase of the HMMs, an iterative method called the Baum-Welch algorithm [6] is used. Depending on how well the environment conditions, that is, the acquisition equipment, the room acoustics and ambient noise, match between training and testing data, experiments can be performed in *matched* or *mismatched* conditions. The largest gain from AVSR comes especially in mismatched conditions.

Since the GMMs used to model speech units are not aimed to discriminate between them, a lot of attention has been given to the use of more discriminative models in conjunction

with HMMs. In hybrid ANN-HMM systems [70], GMMs are replaced with Artificial Neural Networks (ANNs) [13]. Such models have also been applied to AVSR, as in [42].

Support Vector Machines (SVMs) [17] [102] have also been used as an alternative to GMMs, both in audio-only speech recognition [32] and in AVSR [35]. The main advantage of using discriminative models like ANNs and SVMs is that the scores of the speech units will be better separated and the distinction between these units will be clearer.

Although widely used for speech recognition, Hidden Markov Models have several inherent limitations with respect to their use for modeling speech [88]. The first one is the assumption that successive speech frames are independent, as seen in equation 2.8. The use of first and second temporal derivatives partially compensates for this shortcoming, as it includes information about the correlation between frames. A second limitation is the assumption that the probability distribution of observations can be well represented by a GMM, a limitation which is addressed by hybrid systems. Finally, the Markov assumption itself, that is, that only the previous state influences the choice of current state, is flawed, as temporal dependencies for speech can extend for several states.

2.7 The multimodal integration methods

The integration of audio and visual information [86] can be performed in several ways. The simplest one is *feature concatenation* [1], where the audio and video feature vectors are simply concatenated before being presented to the classifier. Here, a single classifier is trained with combined data from the two modalities.

Although the feature concatenation method of integration does in some cases lead to an improved performance, it is impossible to model the reliability of each modality, depending on the changing conditions in the audio-visual environment.

A second family of integration methods is *decision fusion*. In this method separate audio and video classifiers are trained, and their output log-likelihoods are linearly combined with appropriate weights. There are three possible levels for combining individual modality likelihoods [86]:

- Early integration, in the case when likelihoods are combined at the state level, forcing the synchrony of the two streams.
- Late integration, which requires two separate HMMs. The final recognized word is selected based on the n-best hypothesis of the audio and visual HMMs.
- Intermediate integration, which uses models that force synchrony at the phone or word boundaries.

In the following we present one of the most common integration methods, and the one we chose for our experiments, the Multi-Stream HMM. It belongs to the early integration category, forcing synchrony at the frame level. Our choice is justified by the fact that this type of integration allows very rapid changes in the importance given to each modality, allowing the implementation of systems which can very quickly adapt to changing conditions.

2.7.1 Multi-stream hidden Markov models

The Multi-Stream HMM (MSHMM) is a statistical model derived from the HMM and adapted for multimodal processing. Unlike typical HMMs which have one gaussian mixture (GMM) per state, the MSHMM has several GMMs per state, one for each input modality.

The emission likelihood b_j for state j and observation o_t at time t is the product of likelihoods from each modality s weighted by stream exponents λ_s [121]:

$$b_j(o_t) = \prod_{s=1}^{S} \left[\sum_{m=1}^{M_s} c_{jsm} \mathcal{N}(o_{st}; \mu_{jsm}, \Sigma_{jsm}) \right]^{\lambda_s} \tag{2.15}$$

where $\mathcal{N}(o; \mu, \Sigma)$ is the value in o of a multivariate gaussian with mean μ and covariance matrix Σ. M_s gaussians are used in a mixture, each weighed by c_{jsm}. The product in eq. 2.15 is in fact equivalent to a weighted sum in logarithmic domain. In practice, the weights λ_s should be tied to stream reliability, such that, when environment conditions (e.g. SNR) change, they can be adjusted to emphasize the most reliable modality. Our weighting strategy is detailed in Chapter 5.

The product seen here comes from the more general probability combination rules [54], and is one of the most widely used, along with the sum rule, the min rule or the max rule. These rules are compared in [53], with the purpose of combining the outputs of classifiers trained on different types of audio-only features. The product rule was found to be the best performer.

There are two possibilities to train MSHMMs. The first one is to start with normal one-stream HMMs, one per modality, and train each of them on single modality data. These models have to have the same number of states, but the parameters of the GMMs can differ between modalities. No weights are required while training. After the training stage, the models can be combined into a MSHMM. However, there are some drawbacks with this training strategy, one being that there is no guarantee that the models are trained on the same speech segments across modalities. Another drawback is that it is not clear how the transition probabilities should be combined.

The second method of training MSHMMs is to begin with a MSHMM and train it directly on multimodal data. In this way the synchronicity is guaranteed. However, weights are required in training, and a poor choice of training weights can lead to poor performance in testing.

At testing time, the weights given to each stream can be fixed for the whole duration of the test sequence, or can vary dynamically in time. In either case, the weights should be set according to the estimated reliability of each stream. In the following, we present some common stream reliability estimation methods.

2.7.2 Stream reliability estimates

The choice between weights which are constant in time and weights which vary dynamically, adapting to the conditions in the environment, is taken in the beginning based on

the assumptions made on the context of the problem. If it is assumed that the acoustic and visual environment remains more or less the same, and that the training conditions reasonably match those in testing, fixed weights can be used. However, it is more realistic to assume that conditions will not be matched, and that they will not be constant in time. Indeed, sudden bursts of noise can happen anytime in the audio, like a cough, a pop from a microphone, the door of the conference room that is swung against the wall or any other similar situation can lead to a sudden degradation in the quality of the audio. Similarly, for the video, a lighting change, a speaker that turns his head or gestures with his hands making his mouth invisible to the camera, all such situations can lead to sudden degradation in the quality of the video stream. These degradations can be temporary as in the case of the cough, or permanent, as the lighting change. In all these conditions, having the stream weights adapt automatically to the conditions which change in time should be beneficial for the performance of the system.

Fixed stream weights can be derived with discriminative training techniques, applied on training or held-out data. They will only be relevant for the particular environment conditions in which that data was acquired. From the methods that are applied directly on the training data, some minimize a smooth function of the word classification error [79], [72]. Another approach is to minimize the frame classification error, as in [36] where the maximum entropy criterion is used. From the methods that use a held-out data set, the simplest is the grid search, when the weights are constant and constrained to sum to 1, as is the case in [36] or [67]. More complex methods need to be employed in other cases, for example when weights are class-dependent, however, this dependency was not proved to have a beneficial effect on recognition results, as shown in [36] or [52].

Yet another approach is to use a small amount of unlabeled data, as in [96], to estimate the stream weights in an unsupervised way. Class specific models and anti-models are first built, and then used to initialize a k-means algorithm on the unlabeled data. The stream weights ratio is then expressed as a non-linear function of intra- and inter-class distances.

In [115], the testing set itself is used in an iterative likelihood-ratio maximization algorithm do determine the stream weights. The algorithm finds the weights that maximize the dispersion of stream emission likelihoods $P(o|c)$, which should lead to better classification results. The measure to be maximized is:

$$L(\lambda_c^A) = \sum_{t=1}^{T} \sum_{c \in C} \left\{ P(o_t^A | c_t) - P(o_t^A | c) \right\} \tag{2.16}$$

where c is the class or HMM state out of the set C of classes, and o^A is the audio observation vector. The measure is computed over a time interval of T frames.

An extension of this algorithm is based on output likelihood normalization [116]. Here, the weights are class-dependent, and the weight for one class is the ratio between the average class-likelihood for a time period T and the average likelihood for that particular class over the same time period, that is:

$$l_{vt}^{A} = \frac{\frac{1}{NT}\sum_{t=1}^{T}\sum_{c\in C}\log P(o_t^A|c)}{\frac{1}{T}\sum_{t=1}^{T}\log P(o_t^A|v)} \tag{2.17}$$

Both these methods optimize the audio weights first, and then set the video weights relative to the audio ones.

Stream reliability estimates are however not limited to the AVSR field. For example, in [29], reliability measures are used for the fusion of three streams for multimodal biometric identification. The three streams are audio speech, visual speech and the speaker's face, while the reliability measure used is the difference between the two highest ranked scores, normalized by the mean score.

Dynamical stream weights are however better suited for practical systems, where the noise can vary unexpectedly. Such examples of sudden degradation of one modality can be loud noises in the audio, or loss of face/mouth tracking in the video. Events like these can happen in a practical setup and they prove the need for temporally-varying stream weights. The weights can be adjusted for example based on the estimated signal to noise ratio (SNR), as in [1] [19] [41] [117] [43] [64] [23], or based on the voicing index [10] used in [33]. However these methods are based on reliability measures on the audio only, and the video degradation is not taken into account. Other weighting methods are modality-independent, based only on indicators of classifier confidence, as presented in the following.

In [80] and [85], several classifier confidence measures are used. The first one is the N-best log-likelihood difference, based on the stream emission likelihoods $P(o|c)$. The measure is defined as follows. If o_{st} is the observation for stream s at time t and c_{stn} are the N-best most likely generative classes (HMM states), $n = 1...N$, then the log-likelihood difference at time t for stream s is:

$$l_{st} = \frac{1}{N-1}\sum_{n=2}^{N}\log\frac{P(o_{st}|c_{st1})}{P(o_{st}|c_{stn})} \tag{2.18}$$

The justification is that the likelihood ratios give a measure of the reliability of the classification. The same justification can be given for the second measure used, the log-likelihood dispersion, defined as:

$$d_{st} = \frac{2}{N(N-1)}\sum_{m=1}^{N}\sum_{n=m+1}^{N}\log\frac{P(o_{st}|c_{stm})}{P(o_{st}|c_{stn})} \tag{2.19}$$

Other methods for stream confidence estimation are based on posteriors, not on likelihoods. In [105], the class-posterior probability of the combined stream $P(c|o^{AV})$ is computed as the maximum between three posteriors, derived from three observation vectors, the audio-only one o_t^A, the video-only o_t^V or the concatenated observation o_t^{AV}, that is:

$$P(c_{tn}|o_t) = \max(P(c_{tn}|o_t^A), P(c_{tn}|o_t^V), P(c_{tn}|o_t^{AV})) \tag{2.20}$$

The stream reliability estimation framework is not only applicable on AVSR, but also in audio-only speech recognition, in the case when multiple feature streams are used in order to exploit their complementarity. For example, in [66] the entropy of the class-posterior distribution is used as a reliability estimator:

$$h_{st} = -\sum_{i=1}^{C} P(c_i|o_{st}) \log P(c_i|o_{st}) \tag{2.21}$$

where C is the number of classes. The entropy is also a measure of dispersion, but this time used on all the classes, not only the N-best ones.

2.8 Summary

We presented an overview of state of the art AVSR, with an emphasis on the particular algorithms that we will use in the next chapters. The DCT is our method of choice for an initial preprocessing of the image and reduce its dimensionality to a level where we can use more discriminative selection methods. We chose the DCT for its good compaction properties and as it is the standard preprocessing method for visual features in AVSR. The DCT is faster to compute than the PCA, and its results are easier to interpret.

As for the integration of modalities, we chose MSHMMs as models for their ability to vary the importance given to each individual stream at a very fine temporal level. This allows our system the flexibility to adjust to temporally varying conditions.

In the next chapter we will present feature selection methods that are general for classification, and that we will apply on multimodal data, aiming to find the relevant information.

An Overview of Feature Selection Methods

3

3.1 Introduction

Feature selection and extraction are important problems in the classification field. A good overview of dimensionality reduction methods in the context of classification can be found in [57]. In this chapter we will give an introduction to some feature selection and extraction methods and their application on multimodal signals.

First, let us define the context of the problem. *Features* or *attributes* are different types of measures that can be taken on the same physical phenomenon. In the case of speech recognition, they can be the audio signal itself or measures extracted from it, like the spectrum, cepstrum or LPC coefficients. An *instance* or *sample* is a collection of feature values representing simultaneous measurements. An example from speech could be the MFCC feature vector with first and second derivatives, computed for a 40ms segment of speech. Each sample belongs to a class, identified by a *label*, which for speech could be the phoneme to which the particular sample belongs. *Classification* is the process of assigning class labels to samples, belonging to a data set which we will call the *test set*. This can only be done after a training phase, which consists of building models representing the relation between samples in the *training set* and their known class labels.

It may not be obvious in the beginning which are the appropriate measures that need to be taken on a particular signal for a particular classification task. Some features may be redundant, i.e. containing the same information as other features in the set, while others may be completely irrelevant. Feature selection and extraction methods focus on measuring the relevance of individual features for the given task and then keeping only the features which are necessary, resulting in a feature vector of reduced dimensionality.

In the following, we first give the motivation for dimensionality reduction techniques and give some examples from each category. We describe in more detail the information

theoretic methods, as they will be extensively used in our experiments. Finally, we present which of the feature selection methods have been applied before on multimodal signals.

3.2 Motivation

The primary motivation for dimensionality reduction is the "curse of dimensionality", i.e. the fact that, to obtain accurate models of data, the number of samples required increases very fast with the dimensionality of the data. Originally, the term was introduced by Richard Bellman to show that the number of points necessary to uniformly sample a volume of space grows exponentially with the dimensionality [7]. In machine learning problems, where only a limited sample of the data is available, this means that a too high dimensionality can adversely influence the results, since the data space is not adequately covered. Most machine learning models need to attribute some parameters to cover the variability in the input features. The more parameters, the higher the complexity or capacity of the classifier. If some of the features are just noise, the capacity allocated for them is practically wasted. It is also possible that false regularities are found in the unneeded features, also leading to a waste of capacity and decreased performance.

There are several benefits from dimensionality reduction techniques [57]. First, the result contains less data, so the classification algorithm can learn faster. Second, the classifier can generalize better from the data, leading to higher accuracy. Indeed, redundant or irrelevant data may mislead learning algorithms, or cause them to overfit, reducing classification accuracy. Third, the results are simpler and so, easier to understand and interpret. Finally, if there is another round of data collection, the unneeded features would simply not be collected resulting in a faster and less costly data collection process.

Of course, the danger is that important information may be lost in the dimensionality reduction process. However, in most cases some type of dimensionality reduction is inevitable as the data can not be directly used in its initial form.

Feature selection and extraction is very important in multimodal signal processing, where the dimensionality of the modalities is added up and can get particularly high, especially when one of the modalities is video.

3.3 Dimensionality reduction techniques

Feature *selection* means choosing from an initial set of features only the ones which are relevant for the classification task, having as a result a reduced set of features. Feature *extraction* is different in the sense that although the end-result is similar, a shorter feature vector, the features are obtained through a transform, either linear or non-linear, of *all* the features in the initial feature set. We will present examples from both categories and go into more detail into information theoretic methods, as they hold the promise of directly measuring the relevance of each feature for a particular classification task.

Another aspect that distinguishes dimensionality reduction methods is the use of class information. If no class information is used, the process is *unsupervised*, and only statistical

information of the features is used. On the other hand, if class information is included, the dimensionality reduction process is called *supervised*.

Finally, supervised methods can be divided into two. The first, most obvious category, are the *filters*, where the criterion to assess the quality of a feature subset is some measure computed directly on the features. The second category are the *wrappers*, which use the final accuracy given by the classifier itself as a measure, requiring for this the complete training and testing of a classification system for each subset of features taken into consideration. This requires significantly more resources, but potentially will lead to feature sets which are better adjusted to the specific classifier used.

3.3.1 Selection methods

Feature selection methods aim to find an *optimal*, in some sense, subset of features, from a large initial set. Formally, assume that we have a set F of n features, out of which we want to select a subset S of m features. The total number of possible subsets S, $\binom{n}{m}$, is very high, so processing every possible subset is typically impossible. The goal here is to obtain a subset S which retains as much of the information in F as possible.

Feature selection methods differ mainly on two aspects. The first is the search strategy, which specifies how the subset S is built, while the second is the measure used to evaluate the quality of each feature or even the subset S.

The search strategy [57] refers to the method used to generate candidate subsets of features for evaluation. An *exhaustive* search will evaluate all possibilities, although at a very high cost. When the evaluation measure has certain properties, as for example monotonicity, the search can be *complete*, i.e. no optimal subsets are missed, without needing to be exhaustive [101]. An example of such a method is the "branch-and-bound" algorithm [108]. *Heuristic* search is a guided search which avoids the brute force approach, but also risks losing optimal subsets. An example of such a strategy is the *greedy* selection algorithm, where the subset S is built iteratively by selecting the "best" feature at each step, according to the evaluation measure. Finally, *nondeterministic* search will choose subsets of features at random, meaning that it is not known when the optimal set is found, but it will be clear when a better one appears.

The direction of the search can also discriminate feature selection algorithms. *Forward* search starts with an empty subset S and adds features sequentially. *Backward* search starts with a full set F and removes features which are found irrelevant.

The evaluation measures used to determine feature quality can be very varied [8] [57]. Information gain is a measure that shows the reduction in the uncertainty about the class label when a feature is known. The distance between class-conditional probabilities can also be useful, as features that discriminate well between classes are desirable. Dependence measures show the correlation between the feature and the class labels. Yet another measure can be the number of inconsistencies, i.e. the number of cases when two samples have the

same feature values but different class labels. And finally, the accuracy of classification itself can be used as a measure for the quality of the features, in the case of wrappers.

In Section 3.4 we will focus on sequential forward selection using information theoretic measures, which can be used to find both linear and nonlinear dependency in data. These measures can be used to determine how relevant a set of features is for a certain task, and at the same time how much redundancy is present in the set.

3.3.2 Feature transforms

Feature transforms, or feature extraction methods [57], produce a set of new features based on all the features in the original set. This means that all the original features are needed, and there is no reduction in the requirements for data collection. The new features are obtained as the result of applying a transform, either linear or nonlinear, on the initial feature vectors. This leads to a change in the representation of the data itself, meaning for example that it could be better visualized and understood.

Feature transforms can also be supervised or unsupervised. We present in the following two very popular transforms which are used to reduce the dimensionality of data.

Principal Components Analysis is an unsupervised transform, that is, no information about the classes present in the data is used, even if this information is available. PCA aims to find the directions in space in which the variance of the data is largest. It performs a rotation such that the new coordinate axes are these directions. Also known as the Karhunen-Loeve transform, this transform was known since 1901, when it was introduced by Pearson [76].

PCA is computed from the data covariance matrix, by diagonalizing it, sorting the eigenvectors by decreasing eigenvalue and then projecting all the data on this new basis. To reduce the dimensionality of the resulting data, only the features which correspond to a large enough eigenvalue are kept. This is done by setting a threshold on the eigenvalue.

PCA is the optimal linear transform, in mean-square error sense, for compressing high-dimensional data into a lower-dimensional representation, and then reconstructing it. This could be a good justification for its use as a dimensionality reduction technique, but this implies the assumption that the directions of high variance are also informative, or discriminant. Since the PCA includes no class information at all, for speech recognition this assumption typically only holds on clean data with a very good SNR. There is no guarantee that the new features are relevant for the ulterior classification task.

Linear Discriminant Analysis is, by contrast, supervised, as it uses the class labels aiming to find the transform that best separates the classes, while at the same time compacting them. LDA is also a very old and popular technique, deriving from Fisher's linear discriminant introduced in 1936 [25] for two-class problems and extended to multiclass problems by Rao [90].

The criterion which is maximized by LDA is the ratio of between-class scatter and

within-class scatter. This has the advantage of requiring only simple matrix arithmetics. The result of the transform is a lower-dimensional representation, just like for the PCA, but where the classes would ideally be separable and compact, simplifying the classifier's task.

However, there are several limitations to LDA, like the assumption that the class-conditional probabilities are normal distributions and the classes are homoscedatic, i.e. they have the same covariance. Heteroscedatic LDA [56] [22] was developed to deal specifically with the problem of class distributions having different covariances.

3.4 Information theoretic methods

3.4.1 Basic information theoretic notions

The concept of mutual information is derived from the notion of *entropy* [18], which represents the uncertainty that we have about the value of a random variable. The entropy is maximal when the probability density function is flat, that is, the variable can take any of the values in its support interval with equal probability. At the other extreme, the entropy is zero when the pdf is a Kronecker delta function placed on one of the values, that is, the random variable's value is certain. Formally, the entropy of a random variable X with the pdf $p(x)$ is defined as:

$$H(X) = -\sum_{x \in X} p(x) \log_2 p(x) \tag{3.1}$$

Conditional entropy is the uncertainty that remains about the value of X when another random variable Y is known. Formally

$$H(X|Y) = -\sum_{x \in X} \sum_{y \in Y} p(x, y) \log_2 p(x|y) \tag{3.2}$$

where $p(x, y)$ is the joint probability distribution of X and Y while $p(x|y)$ is the conditional probability.

With these notions, mutual information is defined as:

$$I(X;Y) = H(X) - H(X|Y) \tag{3.3}$$

leading to

$$I(X;Y) = \sum_{x \in X} \sum_{y \in Y} p(X, Y) \log \frac{p(x, y)}{p(x)p(y)} \tag{3.4}$$

Equation 3.3 gives the definition of mutual information, that is, that mutual information between two random variables X and Y is the reduction in the uncertainty about the value of the random variable X brought by the knowledge of Y.

An important result from information theory, the *data processing inequality* states that no processing of the data can increase the amount of information in it.

The theorem is formally stated as follows [18]: if X, Y and Z are three random variables forming a Markov chain $X \rightarrow Y \rightarrow Z$ (that is, X and Z are conditionally independent given Y, $p(x, z|y) = p(x|y)p(z|y)$), and $I(X;Y)$, $I(X;Z)$ are the Shannon mutual information values measured between X and Y, respectively X and Z, then $I(X;Y) \geq I(X;Z)$.

Assuming that the relevant information that is sought in the signal is represented by the random variable X and the data by Y, there is a dependency between them. If the data passes through some transformation $g(Y) = Z$, these three random variables form a Markov chain $X \rightarrow Y \rightarrow g(Y)$. This means that $I(X;Y) \geq I(X;g(Y))$. The inequality shows that no function applied on the data can possibly increase the amount of relevant information, but only decrease it (or keep it unchanged).

Note that this does not mean that preprocessing data is wrong or futile. The inequality just states what can or can not be obtained from the data. Indeed, processing the data can reduce its dimensionality, remove noise or redundancy, but it can never add information that was not there initially. However, knowing how much relevant information was contained in the original data is usually impossible to compute in practice, because of the difficulties of computing information theoretical measures.

3.4.2 Mutual information used for feature selection

In feature selection, mutual information between features and class labels has often been used as a measure to evaluate the quality of the features for the classification task, or their *relevance*. Formally, if $I(Y;C)$ is the mutual information between a feature Y and the class labels C, it represents the amount of information gained about the class if the feature Y is used. A high mutual information here shows that the feature is relevant for our classification task and should be part of the subset of selected features.

However, computing the mutual information between each one feature and the class labels may not be enough. What would really be required is the mutual information between a whole subset of features and the class labels. The justification for this concept comes from Fano's inequality [18], which gives the probability of error p_e when trying to estimate one random variable from another. In our particular case, as we are trying to find the correct class label from the features, this equation can be written as:

$$p_e \geq \frac{H(C|F) - 1}{logN} = \frac{H(C) - I(C;F) - 1}{logN} \tag{3.5}$$

where N is the number of classes, F is the feature set and H is the entropy. The equation gives a lower bound for the probability of error, but does not guarantee that this lower bound will be reached by the classifier. It is clear that with a "bad" feature set, one which has a low mutual information with the class label, the bound on the probability of error will be high, forcing it to be high itself. However, when the bound is low, it is up to the actual classifier to come as close as possible to this bound. This shows that bad features lead to bad classification, but good features do not necessarily lead to good classification.

This all shows that a feature set with a high mutual information with the class labels is desirable. However, computing mutual information from data is not trivial. The estimation

of probability density functions is required, which can not be accurately done in high dimensions. This is why most feature selection algorithms that use mutual information actually use two or three-dimensional measures, not more. This means that at most two features are used together with the class label to compute the joint probability density.

We will present now a few information-theoretic feature selection methods. They all belong to the class of sequential forward selection methods, starting from an empty set of features and adding at each step the feature that is "best" according to the evaluation criterion.

First, let us introduce the formal framework for these algorithms. Let $F = \{Y_1, Y_2 \dots Y_n\}$ be the initial set of features. Let $\{\pi_1, \pi_2 \dots \pi_m\}$ be a permutation on a subset of dimension m of the set of feature indices $\{1 \dots n\}$. Then the set of selected features can be written as $S = \{Y_{\pi_1}, Y_{\pi_2} \dots Y_{\pi_m}\} \subset F$.

The simplest way to obtain a subset S iteratively would be to pick at each step the feature with the highest mutual information with the class labels. Formally, this means choosing at step $k+1$ the feature [57, 92]:

$$Y_{\pi_{k+1}} = \arg \max_{Y_i \in F \setminus S_k} I(Y_i; C) \tag{3.6}$$

where $S_k = S_{k-1} \cup \{Y_{\pi_k}\}$ is the set of features selected at step k. This is equivalent to assuming that the mutual information that we want to maximize, $I(S; C)$, can be approximated with the sum of individually computed mutual information values $I(Y_k; C)$, with $Y_k \in S$.

However, this does not take into account any redundancy that may be present in the features. At the extreme, if two features have identical values and a high mutual information with the labels, they will both be chosen, even if the second feature does not bring any new information. So, in order to keep the set of relevant features small, redundancy should be penalized.

Redundancy between features can also be expressed in information-theoretic terms. Indeed, the redundancy between features Y_i and Y_j is measured by their mutual information, $I(Y_i; Y_j)$. However, as the set of selected features grows, we need to compute the redundancy of the candidate feature with the whole set of previously selected features, that is $I(Y_k; S_{k-1})$. This again requires high-dimensional probability density functions. The same assumption as for equation 3.6 can be used, that is, assuming that $I(Y_k; S_{k-1})$ is the sum of individual mutual information values $I(Y_k; Y_i)$ with $Y_i \in S_{k-1}$. An algorithm that does just that is the MIFS (Mutual Information based Feature Selection) algorithm [5]:

$$Y_{\pi_{k+1}} = \arg \max_{Y_i \in F \setminus S_k} \left[I(Y_i; C) - \beta \sum_{Y_{\pi_j} \in S_k} I(Y_i; Y_{\pi_j}) \right] \tag{3.7}$$

Here the redundancy is approximated not with the sum, but with a proportion β of the sum, which the authors recommend setting to between 0.5 and 1.

A similar approach is to penalize the average redundancy [77]:

$$Y_{\pi_{k+1}} = \arg \max_{Y_i \in F \setminus S_k} \left[I(Y_i; C) - \frac{1}{|S_k|} \sum_{Y_{\pi_j} \in S_k} I(Y_i; Y_{\pi_j}) \right] \tag{3.8}$$

where $|S_k|$ is the size of set S_k. In the end, none of these methods has a good theoretical justification, since the high-dimensional mutual information values simply can not be approximated with lower-dimensional ones.

Perhaps a little better justified theoretically are the information-theoretic methods based on the conditional mutual information, (CMI) as a measure [28], $I(X; C|Y) = I(X, Y; C) - I(Y; C)$. This shows how much the random variable X increases the information we have about C when Y is given. The selection criterion is the following:

$$\begin{aligned} Y_{\pi_{k+1}} &= \arg \max_{Y_i \in F \setminus S_k} \left[\min_{Y_{\pi_j} \in S_k} I(Y_i; C|Y_{\pi_j}) \right] \\ &= \arg \max_{Y_i \in F \setminus S_k} \left[I(Y_i; C) - \max_{Y_{\pi_j} \in S_k} I(Y_i; Y_{\pi_j}; C) \right] \end{aligned} \tag{3.9}$$

using $I(X; Y; C) = I(Y; C) - I(Y; C|X)$ [18]. For a certain Y_i, the particular Y_{π_j} is found with which Y_i is most redundant, that is, which has the minimum conditional mutual information with the class label. By taking the maximum over this CMI, the feature that adds the most relevant information to this feature, and, implicitly, to the set S_k, is found.

This method is better as it takes into account only the relevant information added by a new feature to the set, that is, it penalizes redundancy with respect to the class labels, not any type of redundancy. However, this is also an approximation, since, as the set S_k grows, we should compute the conditional mutual information with respect to the whole set, not its one most redundant feature relative to the candidate, which is impossible for the same reasons mentioned before.

In the end, the goal of all these algorithms is to maximize the joint MI between the set S and the classes C, which could be expanded like this (chain rule [18]):

$$\begin{aligned} I(S; C) &= I(Y_{\pi_1}, Y_{\pi_2}, \ldots, Y_{\pi_m}; C) \\ &= \sum_{k=1}^{m} I(Y_{\pi_k}; C|Y_{\pi_1}, \ldots, Y_{\pi_{k-1}}) \\ &= \sum_{k=1}^{m} \left[I(Y_{\pi_k}; C) - I(Y_{\pi_k}; C; Y_{\pi_1}, \ldots, Y_{\pi_{k-1}}) \right] \end{aligned} \tag{3.10}$$

An iterative algorithm could maximize the terms of this sum one by one.

$$Y_{\pi_k} = \arg \max_{Y_i \in F \setminus S_k} \left[I(Y_i; C) - I(Y_i; C; Y_{\pi_1}, \ldots, Y_{\pi_{k-1}}) \right] \tag{3.11}$$

Since Y_{π_k} is the particular Y_i that maximizes the k^{th} term of the sum, all previously mentioned criteria (Eq. 3.7, 3.8, 3.9) can be interpreted as approximations of this general optimization. They all maximize the difference between $I(Y_i;C)$ and an approximation of the redundancy $I(Y_i;C;Y_{\pi_1},\ldots,Y_{\pi_{k-1}})$ between Y_i, S_{k-1} and the class labels C. However, nothing can be said about which of these approximation is actually better, since it all depends on the particularities of the high-dimensional probability density function which can not be estimated.

3.5 Application in AVSR

The most commonly used dimensionality reduction technique for visual features in AVSR is linear discriminant analysis (LDA). Although the classes do not have gaussian probability density functions, which is a basic assumption for the LDA, the transform performs quite well for dimensionality reduction in conjunction with pre-applying a transform like DCT or PCA. For example, a cascade application of LDA, first on each modality separately, and then on a concatenated multimodal feature vector, is the basis of the hierarchical LDA (HiLDA) transform [82] [86].

Mutual information is a measure that can find dependencies in data which does not necessarily have a gaussian probability distribution. This makes it promising for the analysis of audio and visual features in both audio-only speech recognition and AVSR.

Indeed, mutual information has been used in audio-only speech recognition [120] to assess the quality of audio features. The first finding of the analysis was that critical-band spectral energy observations are not gaussian distributed, justifying the use of a non-linear measure such as the mutual information. Second, it was shown that the phonetic information is spread across the frequency domain and also in time. In [98] and [100] mutual information is used to analyze the time-frequency structure of phones, and subsequently to select individual time-frequency features for separate phoneme classifiers.

Such an analysis can also be performed in the visual domain, where the type and number of features to be used are less established. In [99] [97], the authors select the features used for visual speech recognition based on either the mutual information between features and class labels, or the joint mutual information between two features and the class label. Formally, they use either:

$$Y_{\pi_{k+1}} = \arg \max_{Y_i \in F \setminus S_k} I(Y_i;C) \tag{3.12}$$

or

$$Y_{\pi_{k+1}} = \arg \max_{Y_i \in F \setminus S_k} I(Y_i;C) + \frac{1}{|S_k|} \sum_{Y_j \in S_k} I(Y_i;C|Y_j) \tag{3.13}$$

where $|S_k|$ is the number of elements in S_k. The second term in equation 3.13 comes from the joint mutual information $I(Y_i,Y_j;C)$. Neither measures contain a penalty for redundancy.

Another widely used transform for reducing the dimensionality of visual data is the PCA. Here, dimensionality reduction is typically achieved by only choosing the features

which correspond to the largest eigenvalues. Selecting PCA coefficients from mouth images based on mutual information gives rise to "mutual information eigenlips" [2], leading to an improved speech recognition performance. The measure used is again the maximum mutual information as in equation 3.12, with no penalty for redundancy.

3.6 Summary

We presented an overview of feature selection and extraction techniques for classification in general, and the algorithms that have been applied to AVSR. Methods based on information theory have been treated in more detail as these are the methods we use in our analysis. As has been shown, although the subject of using MI as an evaluation method for feature selection has been extensively explored during the years, only the simplest methods have been applied on multimodal signal processing and in particular AVSR. In particular, none of the approaches that use MI for AVSR take into account the redundancy between features or treat it as a penalty when building the feature set.

In the next chapter we will present two novel approaches to visual feature selection for AVSR, the first one based on the optical flow on the region of the mouth, and the second using mutual information as a measure of both relevance and redundancy. Our mutual information approach differs from those presented in the previous section in the fact that it penalizes the redundancy in visual features, choosing not only the most informative ones, but those that add the most information over what was already present in the previously chosen set.

Part II

Multimodal feature selection and fusion

Selecting visual features for AVSR

4

4.1 Introduction

In this chapter we present our two rather different approaches for visual feature selection applied on AVSR. The first one is based on the assumption that most visual speech information is concentrated in the opening and closing motion of the mouth. The goal here is to find the lowest dimensional features that can give us useful information about what was said, without necessarily capturing all the detail. The novelty of the method consists in the fact that it uses spatial differences of optical flow vectors on the region of the mouth, which are better-suited for modeling mouth movement and also more tolerant to noise. The second approach uses forward selection schemes using feature evaluation measures based on information theory. Here we want to find how we can reduce the dimensionality of the data while losing as little as possible from the useful information. This approach produces features which have a higher dimensionality and are more complex, but also lead to better results. This method is novel in the field of AVSR, as we are showing that the redundancy of features, which was previously ignored, is an important factor in the final performance of the system.

To begin with, we present the database that we will be using throughout the following chapters and the preprocessing that was applied on the visual and audio parts. We show how the region of interest was extracted and how noise was added in the audio to simulate different acoustic environments. Then, we present our AVSR system in detail. We continue with our feature extraction method for low-dimensional motion-based features. Finally, we present our methods for feature selection using mutual information as the evaluation measure. We compare here several different information-theoretic approaches, some that take into account feature redundancy, others that do not. We will also include a comparison to LDA, which is the transform most commonly used for dimensionality reduction in AVSR.

Figure 4.1 — Six sample frames from the single-speaker part of the CUAVE audio-visual database.

The content of this chapter is partially based on work that we have published in [119], [118] and [21].

4.2 The database

Few audio-visual databases exist, a lot less than in the domain of audio-only speech recognition, and even fewer are freely available. The difficulty of collecting a large amount of video data and the size of the resulting files may be the explanation for the scarcity of data in this domain. With a single notable exception, most AV databases have a very limited vocabulary and a quite limited number of speakers.

A full review of AV databases used for speech recognition can be found in [85]. We will mention here a few such databases used around the world for AVSR, then go into more

detail about our particular database choice.

- The Tulips1 [71] is maybe the smallest audio-visual database in use. It consists of 12 speakers saying the first four English digits. The video consists of only the mouths of the speakers.

- The CUAVE database [75] has a larger vocabulary, the digits from "zero" to "nine", repeated five times by 36 speakers. Here the whole head of speaker is visible, together with the shoulders, not just the mouth.

- The CMU database [50] has an even larger vocabulary, 78 words, but the number of speakers is reduced to 10. Only the heads are visible here.

- Finally, the IBM ViaVoice Audio Visual database [85] is the largest AV database collected. It is intended for Large Vocabulary Continuous Speech Recognition tasks. It consists of continuous sentences read by 290 subjects.

We opted to use CUAVE as the database on which to perform our experiments, because it offers us a good balance between the variability in the data and the amount of time spent to run the experiments. Indeed, as there are 36 speakers, there is a fair amount of variability between them, and this means that this database may be more representative for the general population than, for example, a database with just 4 speakers. And, as we are running a lot of experiments, with different SNRs, dimensionality values and many different feature extraction algorithms, the duration of the experiments would have become too long with a larger database.

There are two parts in the CUAVE database, the "individuals" part where there is only one speaker in the image, and the "groups" part where there are pairs of speakers. In the second part, the speakers are taking turns, and finally they are speaking together. We used the first part of the database for speech recognition experiments here and in the next chapter. The second part is also suitable for speaker detection experiments, and was used in Chapter 6.

The words are spoken in sequence, with short silences in-between. Although the words are isolated, we treat the task like a continuous speech recognition problem - the recognizer receives a whole sequence comprising of five repetitions of all the digits, and has to recognize what is being said. We use a very simple syntax, which allows any combination of digits and silence, in any order.

The database was recorded in an isolated sound booth at a resolution of 720x480 with the NTSC standard of 29.97 fps [75]. Sample images from the database can be seen in Figure 4.1. The resulting video files are MPEG2 compressed. The audio is 16-bit, stereo at a sampling rate of 44 kHz. There is also word-level labeling at millisecond accuracy, done manually, for all sequences of the database.

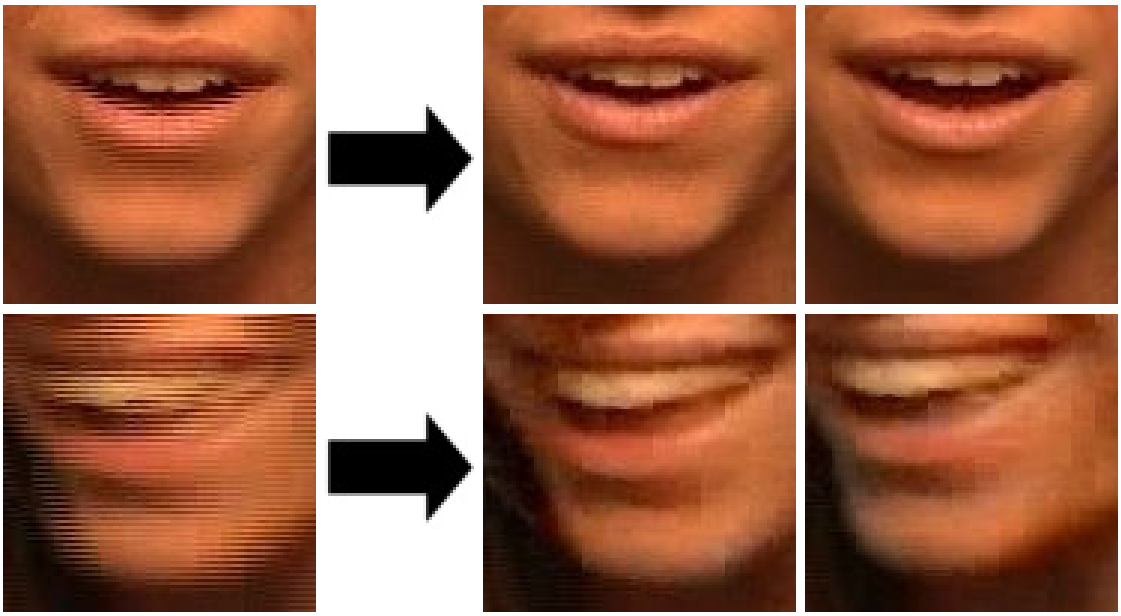

Figure 4.2 — Mouth regions from the CUAVE database, before and after deinterlacing.

4.3 Preprocessing methods

4.3.1 Deinterlacing

Interlacing is a technique used in video acquisition where the final frame is a blend of two *fields*, each of them a complete image of the scene that is filmed, but at half the vertical resolution. The fields are captured one after the other, and then blended into one frame, such that the odd lines of the frame come from one field, and the even lines from the other. The acquisition rate of the fields is double that of the frames, leading to motion blur which makes the eye believe that the frame rate is higher. This is a standard procedure in all major analog television standards, NTSC, PAL and SECAM, as well as the 1080i HDTV standard. The reason for this is that the motion in the video will appear more fluid than in non-interlaced or *progressive scan* videos at the same frame rate, while keeping the bandwidth requirements unchanged.

As the videos in our database are filmed in NTSC, there are some artifacts related to interlacing which are visible where there is movement. Since we are particularly interested in the movement in the image, we chose to remove these artifacts through deinterlacing.

In the case of our database filmed at 30 frames per second, the fact that the video is interlaced means that there are actually 60 images per second, at half the vertical resolution. This is in fact an advantage, as we can use the higher temporal resolution for a finer analysis of the motion of the mouth.

To obtain the original temporal resolution of 60fps from the 30fps movie, the simplest method would be *bob deinterlacing*, which is simply separating the even and odd lines of each image and grouping them in the two fields. However, this leads to objects moving up and down in the image (*bobbing*), as there is a vertical shift of half a (field) pixel between the

Figure 4.3 — Four sample ROIs, centered, rotated and scaled with respect to the corners of the mouth.

two fields. This can be fixed by upscaling the two fields to the original vertical resolution and compensating for the shift through interpolation.

More advanced deinterlacing methods detect the areas of the image where there is motion and separate the two fields only on those areas, leaving the rest of the image at full vertical resolution. This is called *adaptive deinterlacing* and it is the method that we applied on our database. This results in both high spatial and high temporal resolution. In Figure 4.2 we show two examples of deinterlaced frames, one where the speaker is static, the other when the speaker is turning her head. In both examples it is clear that at least for analyzing motion, the deinterlaced frames are more appropriate. In the second example, the images have a blocky appearance due to the MPEG2 compression. These kind of artifacts are impossible to eliminate, as the original detail has been lost.

4.3.2 Region of interest extraction and temporal upsampling

The initial part of the visual front-end for AVSR consists of Region of Interest (ROI) detection. In our case, as the focus of our work is finding the best way to extract information from images, having a good initial ROI is important. The ROI is extracted based on the positions of the corners of the mouth. First, the average width of the mouth for a particular speaker is computed over a whole sequence. Then a rectangular area is cut around the mouth, in such a way that the mouth is centered, rotated and scaled relative to this average width. Bilinear interpolation is used to obtain the final images.

The mouth corners are obtained with a semi-automatic approach. A correlation-based tracker is used, initialized with a model of the corner of the mouth. The correlation value is used as an indicator for the quality of the match between the current image and the model. If the correlation drops under a certain threshold, this means that the match is bad and the tracker stops. At this point, the tracker is re-initialized manually with a mouse click on the mouth corner. This approach requires much less user input than a full manual approach, while also being more reliable compared to a fully automatic one. As our goal here is the off-line analysis of features, requiring some degree of manual control in the ROI extraction is acceptable.

As mentioned before, the extracted ROIs have the mouth centered, rotated and scaled. As this is done after deinterlacing, the frame rate of the video is 60fps. However, the integration method that we use requires synchronous streams, and the audio features are extracted 100 times per second. To bring both streams to the same rate, we upsample the

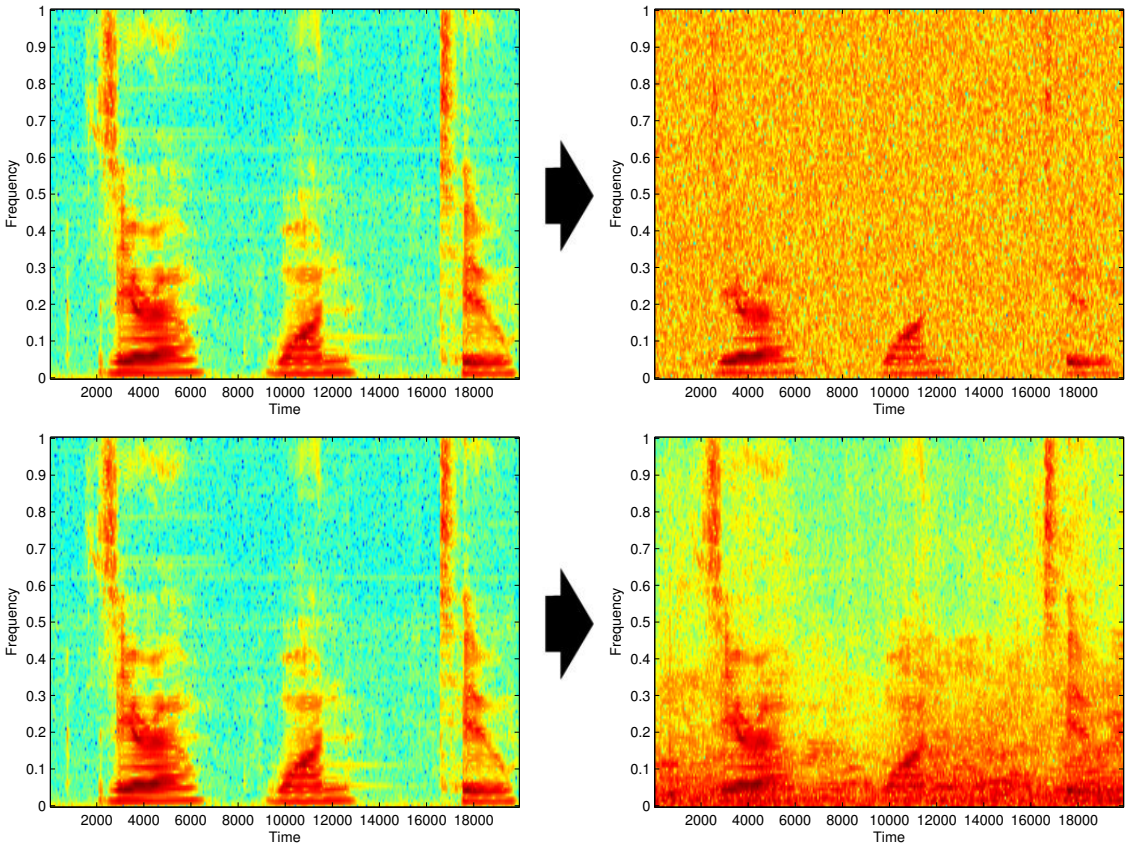

Figure 4.4 — Spectrograms of a short portion of speech, before and after adding noise. Two types of noise are used, white noise and babble noise. The SNR for the noisy signal is 0 dB. The time units are 10 ms frames, while the frequency range is 0 - 4 kHz.

video features to 100fps through linear interpolation.

4.3.3 Audio noise addition

To be able to analyze the effect of multimodal integration on speech recognition across different acoustic environments, we added two types of audio noise to the audio data, at several signal to noise ratios.

AVSR works best when the audio stream is corrupted. Although there is a small performance gain even for clean audio, the biggest influence of the video stream is seen when there is noise in the audio. And indeed, studio-quality sound with very high SNR is very seldom found in practical circumstances.

The first type of noise is additive *white* noise, where the noise is completely random, its energy split equally in all frequency bands. This type of noise is typically not encountered in real settings, however it can give us an idea of how audio stream degradation influences final audio-visual results.

The second type of noise is *babble*, which is continuous speech added over the clean signal. This is a particularly difficult case, as the noise has the same characteristics of the

original signal.

In both cases, the SNR is computed only over the speech segments, not the whole sequences. This was done as we wanted to ignore the silent segments in the computation of the SNR, as the ratio of silence to speech is particularly high for the CUAVE database. Including the silent segments in the computation would make the SNR artificially low, as there would be a lot of noise energy spread in the silent samples. Even worse, as the ratio of silence to speech differs between sequences, the SNR of the speech segments would vary from sequence to sequence depending on the amount of silence taken into account, which is clearly not desirable.

We run our experiments at 9 different SNR levels: clean, 25 dB, 20 dB, 15 dB, 10 dB, 5 dB, 0 dB, -5 dB and -10 dB. In Figure 4.4 a comparison is given between the spectrograms of a clean segment of speech with the same segment with added noise. Both types of noise are presented, white and babble.

4.4 Our AVSR system

Our speech recognition system is based on Multi-Stream Hidden Markov Models, which have been detailed in Chapter 2. We chose MSHMMs for their ability to vary the relative importance of the streams very quickly, a capability which will prove its usefulness in Chapter 5.

Each Markov state is modeled with only one gaussian with a diagonal covariance. The number of states is chosen empirically to 32 emitting states, after several validation experiments with both audio and video. However, the 32 states are linked in pairs, so that two of them share one gaussian. This allows a better modeling of the duration of the states, without requiring more parameters. For the actual implementation we used the widely popular HTK library [121].

The recognition rate is computed as the number of correctly recognized words C minus the number of insertions I, divided by the total number of words N [121], that is:

$$PercentAccuracy = \frac{N - D - S - I}{N} \times 100\% = \frac{C - I}{N} \times 100\% \tag{4.1}$$

where S is the number of substitutions and D the number of deletions.

When performing the initial experiments to decide the size of the models that will be used, we noticed that the recognition rates were sometimes very low, even negative because of high insertion rates. Increasing the size of the models solved this problem, as insertions are typically very short words which are falsely recognized in the silent segments.

We run our experiments in a mismatched speaker-independent scenario. Training is always done only on clean conditions, while testing is done on all SNRs. As mentioned before, we are using the CUAVE database for all our experiments. Since this database is quite small, with only 36 speakers, there is barely enough speaker variability for speaker-independent tests. We tried to compensate for this problem through leave-one-out crossvalidation. For each experiment there are 36 runs, with one speaker left aside for testing and 35 speakers

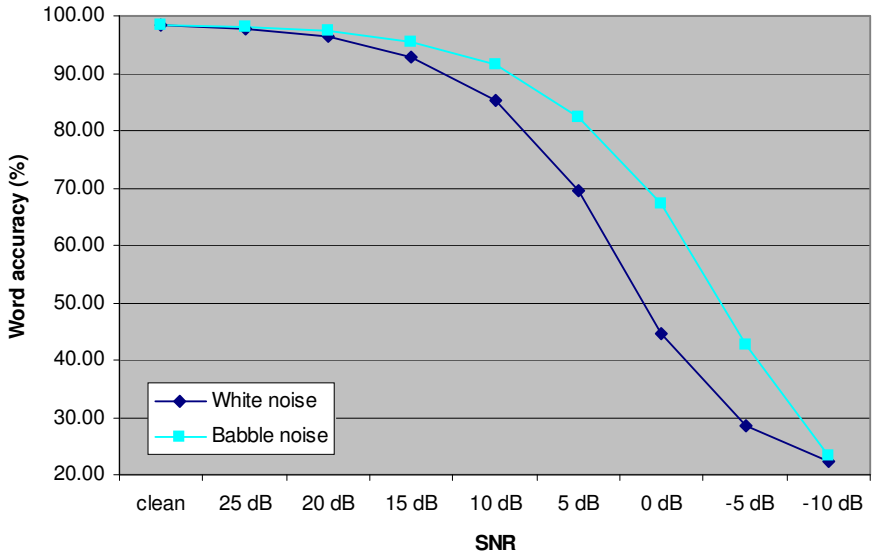

Figure 4.5 — Audio-only results with 2 noise types, white and babble, for different SNRs.

used for training. The accuracy reported at the end is an average of the result of the 36 runs. Even in the controlled conditions of the CUAVE database, the spread of the results for the 36 different speakers is quite high, possibly because of the variability, both audio in the pitch, rate of speech and different accents, and video, in skin tone, mouth shape and even in head pose. Variations as high as 20% are seen sometimes between different training/testing selections for the same conditions.

Training of the MSHMMs is done separately for each stream. The models are then joined, with transition probabilities chosen as a weighted sum of transitions from each stream, with weights being the same as the ones used in testing. At test time, the likelihood is computed as shown in Chapter 2, that is:

$$b_j(o_t) = \prod_{s=1}^{S} \left[\sum_{m=1}^{M_s} c_{jsm} N(o_{st}; \mu_{jsm}, \Sigma_{jsm}) \right]^{\lambda_s} \tag{4.2}$$

or

$$\log b_j(o_t) = \sum_{s=1}^{S} \left[\lambda_s \cdot \log \sum_{m=1}^{M_s} c_{jsm} N(o_{st}; \mu_{jsm}, \Sigma_{jsm}) \right] \tag{4.3}$$

Testing is done with weights λ_s constant in time, or fixed, with $\lambda_a + \lambda_v = 1$. We run experiments with several pairs of weights, from $(\lambda_a; \lambda_v) = (0.0; 1.0)$ to $(\lambda_a; \lambda_v) = (1.0; 0.0)$ with step 0.05, and then choose the best weights for the particular conditions of the test. Although this can not be done in a practical setup, it gives us a good estimate of multimodal performance in practice. In Chapter 5 we will present dynamic weighting methods that achieve more or less the same performance as the fixed weights.

Our audio features are 13 mel-frequency cepstral coefficients, with first and second temporal derivatives. Figure 4.5 shows the audio-only results obtained at all SNRs for the

two types of noise that we use.

For the visual features, we used either motion features derived from the optical flow on the ROI, or the odd columns of the 2D-DCT of the ROI, on which we apply different selection and extraction algorithms. The reason for using motion features is the assumption that most of the visible speech information is in the motion of the lips. As was shown in Chapter 2 the DCT is a transform borrowed from the image compression field, used to reduce the dimensionality of the feature vector while preserving most of the information contained in the image. The even columns of the transform are removed as a way of imposing symmetry on the ROI. Both motion and DCT-based features will be detailed in the following sections. We always include monomodal results for comparison, and also results with the LDA transform applied on the initial DCT features, since this is the most commonly used method.

4.5 Feature normalization

There is one operation that was applied on all features, both audio and visual. This operation is feature mean normalization (FMN) which is also called cepstral mean normalization (CMN) when applied on audio cepstral features. Both FMN and CMN have been mentioned in Chapter 2. The method consists in removing the temporal mean of the features for each sequence, that is, for each individual speaker in the database.

This has the effect of reducing the impact of differences between the sequences themselves, either from different acquisition hardware, or from different illumination conditions.

On the audio signal CMN improves the recognition ratios at all SNRs, starting from clean, where the improvement is from 97.58% to 98.25%, all the way to the lowest SNR. The largest improvement is at 0dB, from 2.79% without normalization to 67.35%. The audio-only comparison between recognition with CMN and without it is given in Figure 4.6.

On the video stream the same effect can be seen with different numbers of features. For DCT features without temporal derivatives, the visual speech recognition rate for 6 features with zig-zag ordering is 15.9% without normalization and 27.67% with FMN. The difference grows for more features. For example, for 50 visual features, the recognition rate is 23.65% without FMN and 50.44% with. The comparison between the two cases for different feature vector lengths is given in Figure 4.7.

4.6 Optical-flow features

The first type of visual features that we present are motion features based on the optical flow. A brief overview of optical flow is given in Chapter 2. Here we want to extract very simple features that are related to speech to see how low we can reduce the dimensionality of the features while still getting useful information from the visual modality.

Our algorithm is as follows. First, we compute optical flow on the mouth regions, using the Horn-Schunck algorithm [49], as shown in figure 4.8. Two strips of pixels are selected, above and below the center of the mouth, and only the vertical components of the motion

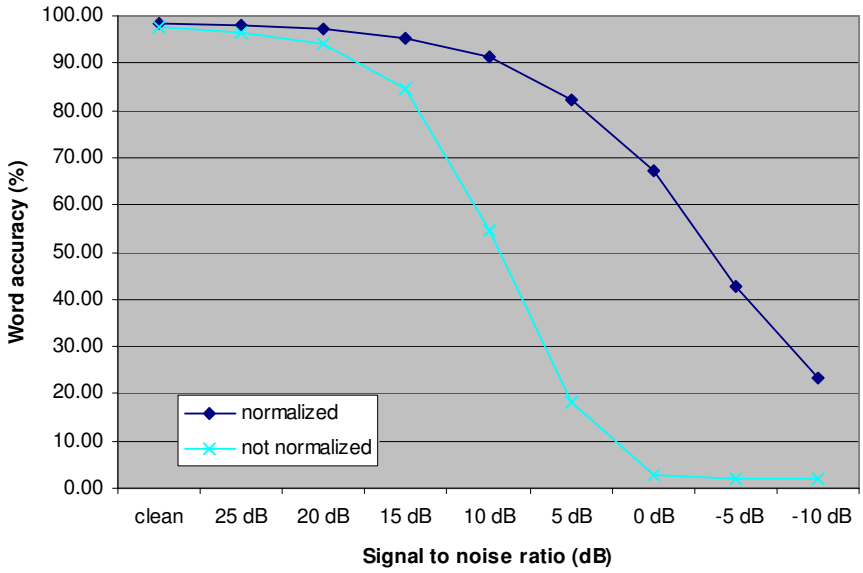

Figure 4.6 — The effect of cepstral mean normalization on audio-only speech recognition on our database.

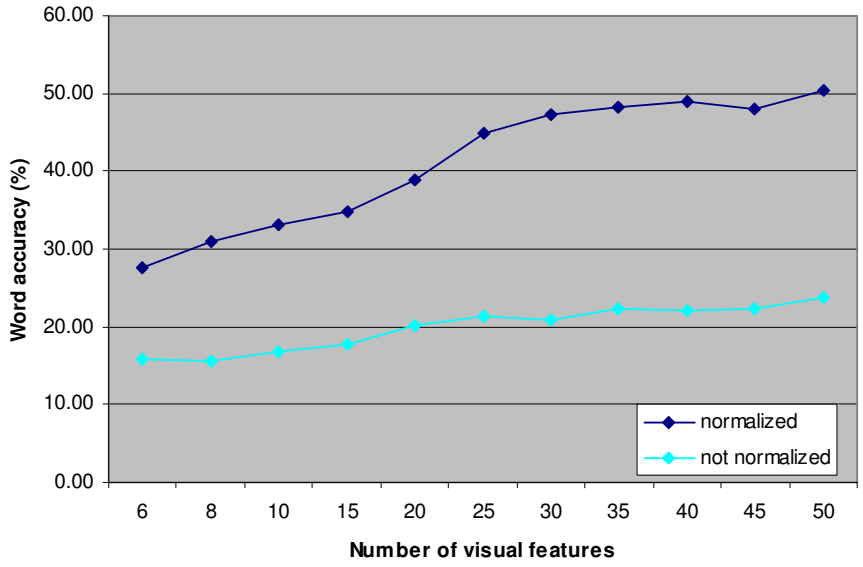

Figure 4.7 — The effect of feature mean normalization on video-only speech recognition on our database.

field are retained. Our first visual feature is the difference between the maximum optical flow values on the top and bottom strips. The same is then done on the horizontal, to obtain a second feature describing the horizontal movement of the lips.

Taking the maximum over a strip of pixels has a very simple justification. We want the algorithm to work even when there is partial occlusion of the mouth, or when the center of the mouth is not detected correctly. The value of the vertical motion feature would be the same even if the center of the mouth is shifted to the left or the right, if a part of the mouth is partially covered, or, in fact, even if the head is partially turned. The optical flow

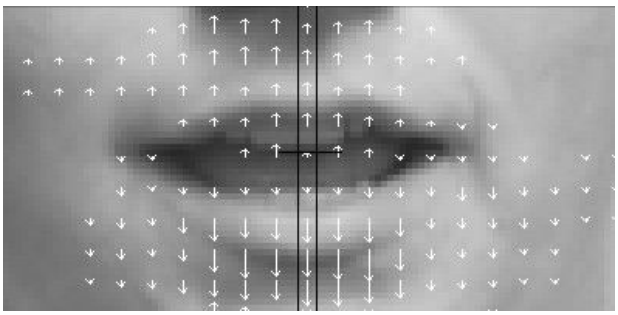

Figure 4.8 — A ROI with the corresponding optical flow.

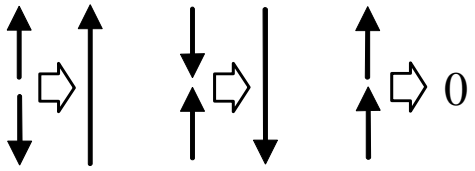

Figure 4.9 — An illustration of the difference of motion vectors.

itself is not precise, since its assumptions, mentioned in Chapter 2, are more often than not violated. Indeed, the smoothness assumption is not valid in the case of the mouth, as the teeth and the tongue can appear and disappear from the image, making the motion at the edge of the lips not smooth. The constant brightness assumption can also be violated, as a small change in the angle of the head can lead to a variation brightness on the entire face. Taking the maximum may also compensate for some errors in the optical flow estimation, when the movement is not detected on the entire region above and below the lips.

All this means that our motion features are quite robust. However, we work with sequences where these adverse conditions conditions are not commonly encountered, so this robustness can not be effectively proven. Indeed, the lighting is uniform and diffuse, and the mouth is never occluded, while ROI detection is quite precise, as it is done with the help of user input.

What we observe is that the optical flow difference is closely related to the movement of the mouth. When the mouth is opening, the result is a large positive number, while when it is closing, the result is negative. However, when both vectors point in the same direction, they cancel each other out, as shown in figure 4.9. The advantage of this approach is that small movements of the head are neutralized. When the head is moving, the upper and lower components of the head motion cancel out, yielding only the mouth movement. This is also valid for the horizontal motion feature. In the end, we obtain a simple 2D feature vector describing the mouth opening and closing.

For comparison purposes, we also include results with the LDA coefficients. LDA is applied with the purpose of reducing the dimensionality on the set of DCT coefficients, together with their first and second temporal derivatives. This means that the LDA also includes temporal information, just like the optical flow, so the comparison is fair.

We compared our two motion features with the first two features computed with LDA.

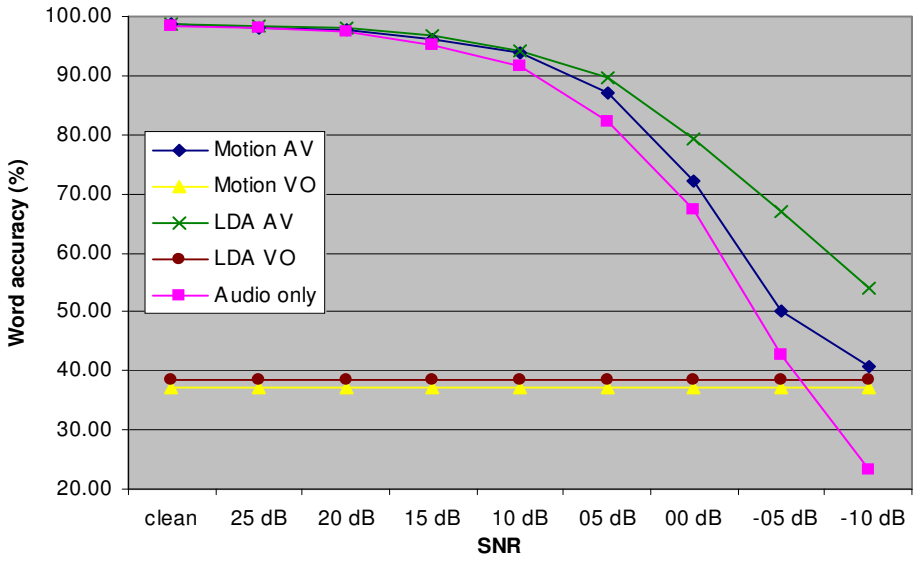

Figure 4.10 — Audio-visual results with 2 motion features, compared with LDA features at the same dimensionality. Audio-only and video-only results are also included.

Features / dim / SNR	Optical flow 2	LDA 2	Audio only
clean	98.58	98.65	98.25
25 dB	98.14	98.49	97.92
20 dB	97.86	98.21	97.36
15 dB	96.25	96.81	95.25
10 dB	93.83	94.28	91.49
05 dB	86.99	89.60	82.24
00 dB	72.19	79.18	67.35
-05 dB	50.14	66.89	42.54
-10 dB	40.69	54.13	23.18
video-only	37.29	38.49	

Table 4.1 — Results with 2 motion features, both audio-visual and video-only. LDA and audio-only results are added for comparison.

For video-only, results are quite similar, around 38% word accuracy for both types of features. However, when combined with the audio, the LDA features perform much better, as there is an absolute performance gain of 14% compared to optical-flow at the lowest SNR. We present here experiments with white noise. The results can be seen in Table 4.1 and Figure 4.10.

It should be noted that it is quite clear from all our experiments that visual-only performance should not be considered a good estimator for audio-visual performance. In this particular case, although video-only accuracies are very close, audio-visual performance is much higher with LDA. This may mean that the LDA coefficients contain more information complementary to the audio, that is, they contain information which is not present in the audio coefficients. This is confirmed at all SNRs, from clean to -10 dB.

Although the motion features are not performing as well as the LDA ones, there is still a large gain from employing them, around 17% at the lowest SNR, compared to audio-only. The motion features are simpler than the cascade of DCT and LDA transforms, and more tolerant to a bad localization of the mouth. Indeed, in cases where the mouth is partially occluded or the head of the speaker is turned, the DCT would not perform that well, while the motion features would remain practically the same. The motion features can also remove the effect that a shift of the ROI could have, due to tracking errors, by canceling the shift movement when computing the vector difference. Finally, the motion features have a very low dimensionality, so the computational cost for using them should be quite low. They could be employed in applications where the use of the DCT - LDA cascade would be impractical, due to lack of detail in the ROI, high variability, high motion, occlusions or any number of other problems.

In our best knowledge, the optical flow difference has not been used before in visual speech recognition. The use of such features is novel. Typically the optical flow has been used either as it is or downsampled, as shown in [37].

In conclusion, the low-dimensional motion features that we extract lead to significant performance gains compared to audio-only speech recognition, they are simpler and more robust than the LDA.

4.7 Selecting features with mutual information

4.7.1 General theory

As shown in Chapter 3, mutual information has been quite extensively used as an evaluation measure for feature quality for classification. There are two reasons for this. The first comes from the definition of MI, as it can expose a dependency between two random variables, even when that dependency is nonlinear. This makes MI a much more powerful measure of dependency than correlation. The second reason comes, as has been mentioned before, from Fano's inequality, which can be interpreted in a way that shows that the bound on the probability of classification error can be lowered by choosing features with higher MI with the class label.

A large majority of MI feature selection algorithms aiming to find k features from an initial set of n follow these basic steps [5]:

1. (Initialization) Start with a complete set F of initial features and an empty set S of selected ones.

2. (MI computation) Compute $I(f;C)$, the MI between each feature $f \in F$ and the class variable C.

3. (First choice) Choose $f = \arg\max_{f \in F} I(f;C)$, set $F \leftarrow F \setminus \{f\}$, $S \leftarrow \{f\}$

4. (Greedy selection) Repeat until $|S| = k$:

(a) (Evaluation) Compute the evaluation measure as $I(f;C) - penalty$ for each $f \in F$ based on the previously selected features $s \in S$ and the class labels

(b) (Choice) Choose feature f which maximizes the evaluation measure, set $F \leftarrow F \setminus \{f\}$, $S \leftarrow S \cup \{f\}$

5. (End) The set S contains the *best* k features according to the evaluation measure.

The evaluation step, 4.(a), is the step that changes between different MI selection algorithms. As shown in Chapter 3, most algorithms try to maximize an approximation of the MI between the set of chosen features S and the class labels, a problem which reduces to computing the redundancy between feature f, the set S and the class labels C. This approximation of the redundancy is used as a penalty on the class MI term $I(f;C)$.

Although a rich diversity of algorithms exists, only the simplest one has been previously applied to AVSR. Previous approaches to the problem of feature selection for AVSR using MI choose the features with maximum MI with the class labels [99] [97] [2]. That is, the evaluation measure is simply $\max I(f;C)$, and no measure of redundancy is taken into account. By contrast, we show that penalizing features for their redundancy can improve the recognition accuracy.

In all our following tests, we use the greedy selection method outlined here. We employ several evaluation measures, starting with maximum MI mentioned above, and then exploring different measures which also include a penalty for redundancy between features. Our contribution here is twofold. First, we prove from our experiments that reducing redundancy between features is essential when building a feature set. Second, we make an extensive evaluation of our proposed feature selection methods based on MI, showing both monomodal and multimodal results, for a wide range of dimensionality values. We compare our results to two common methods for dimensionality reduction in AVSR, the zigzag DCT ordering and the LDA.

A natural question that might arise is why not use all the available features for recognition. We will also answer this question, showing that, due to the curse of dimensionality, multimodal recognition performance decreases when the dimensionality becomes too big. In the end, practice confirms the intuition that small sets of features with little redundancy perform better. Redundancy needs to be reduced, as it is possible, when not adding a penalty for it, to select features that contain the same information, to the detriment of others which would bring complementary information.

Throughout all the following we will use the same notations as in Chapter 3.

4.7.2 Redundancy, complementarity and synergy

Before presenting our information-theoretic feature selection methods, we go into more detail showing what the redundancy between features is, and the different types of redundancy.

Figure 4.11 shows a Venn diagram of entropies and mutual information values between three variables. Assume that X and Y are two features, and C is the class label variable.

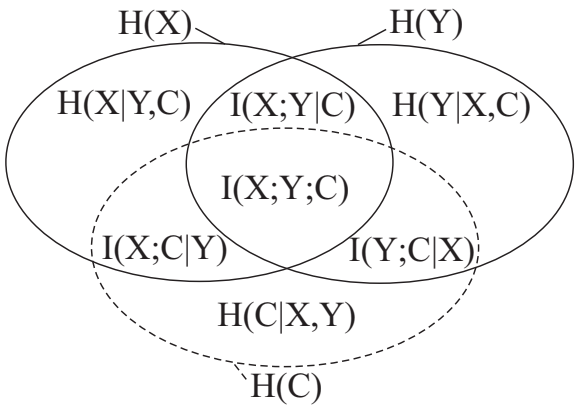

Figure 4.11 — A Venn diagram showing entropies and mutual information for two features X, Y and the class labels C.

Then $I(X;C) = I(X;Y;C) + I(X;C|Y)$ and $I(Y;C) = I(X;Y;C) + I(Y;C|X)$ are the individual mutual information values between each feature and the class label. Some selection algorithms choose features which individually have high MI values, irrespective of the other features in the set. However, the information contained in features X and Y could be the same, which on the graph would make the two "bubbles" for X and Y superposed. In a set of correctly selected features, the *redundancy*, or, on the figure, the intersection of the X and Y "bubbles", $I(X;Y) = I(X;Y;C) + I(X;Y|C)$, should be minimal. In this way the information brought by one feature that is supplemental to the other is maximized. This *complementary* information is measured by the Conditional Mutual Information (CMI), $I(X;C|Y)$ for X and $I(Y;C|X)$ for Y.

In the end, the measure that we want to maximize is the joint mutual information between the pair of features (X,Y) and the class C:

$$\begin{aligned} I(X,Y;C) &= I(X;C) + I(Y;C|X) && (4.4)\\ &= I(Y;C) + I(X;C|Y) && (4.5)\\ &= I(X;C) + I(Y;C) - I(X;Y;C) && (4.6) \end{aligned}$$

This shows that the three-way mutual information $I(X;Y;C)$ is the measure that should be reduced. However, by contrast to the other information measures, $I(X;Y;C)$ can even be negative. This leads to the following interpretation. When $I(X;Y;C)$ is positive, it means that the two features X and Y are redundant with respect to the class label, that is they have some common information about the class. In this case $I(X;Y;C)$ could be called *relevant redundancy*, since it is the part of the feature redundancy which counts for our particular task. Since $I(X,Y;C) = I(X;C) + I(Y;C) - I(X;Y;C)$, this means that $I(X,Y;C) < I(X;C) + I(Y;C)$, the joint mutual information is smaller than the sum of the two individual MI values.

The case when $I(X;Y;C)$ is negative is however more interesting. A negative $I(X;Y;C)$ would mean that $I(X,Y;C) > I(X;C) + I(Y;C)$, which can be interpreted as a *synergy* between the two features, that is, the information contained jointly in the two features is

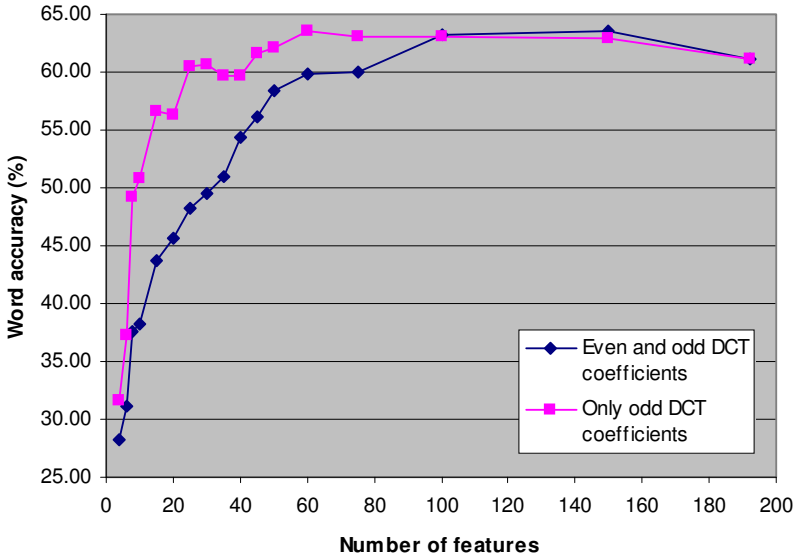

Figure 4.12 — Visual-only recognition results showing the effect of removing the even DCT columns.

greater than the sum of individual information values. Intuitively, this would be the case when, for example, two classes are impossible to separate on any of the two one-dimensional axes, but become easily separable on the two-dimensional plane.

Ideally, for feature selection, such synergy between the features should be exploited whenever possible. However, for higher dimensionality, the MI can not be estimated and we are forced to employ lower-dimensional approximations. As could be seen from the example above, the two values $I(X;C)$ and $I(Y;C)$ do not offer any information on the value of the joint MI, $I(X,Y;C)$, which can be either larger or smaller than the sum $I(X;C) + I(Y;C)$. In this particular case, we can solve this problem by estimating the three-way MI, $I(X;Y;C)$. However, in higher dimensions this becomes impossible and approximations will be used.

4.7.3 Implementation details

In all the following, MI values are approximated with the use of histograms. Only 2D (feature value and class variable) and 3D (two feature values and class variable) probability density functions (pdfs) are estimated this way, as we consider that with the limited number of samples available, higher-dimensional pdfs are impossible to estimate reliably. The classes that we use are groups of HMM states, which we consider to be close to the true speech units.

However, it should be noted that, since we are always approximating MI between high-dimensional pdfs with MI between 3D pdfs, it is also possible to overestimate the redundancy between them. This means that there will be cases when features are penalized too much. A balance has to be found between eliminating true redundancy and penalizing useful features which were wrongly deemed too redundant.

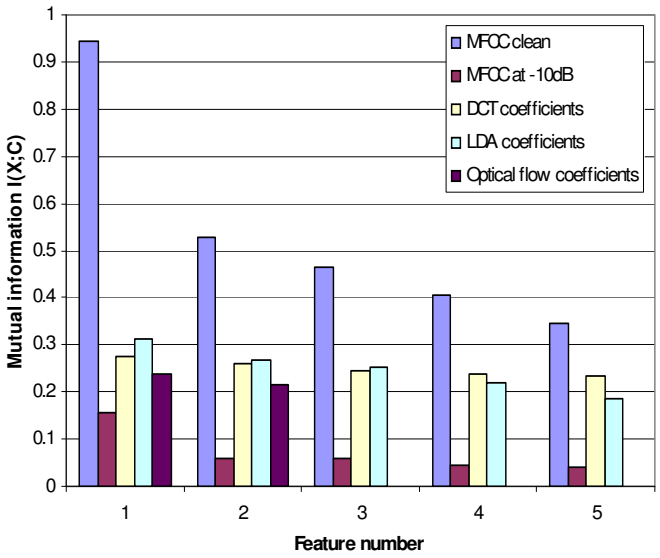

Figure 4.13 — The amount of relevant information (MI between the feature and the class label) for the best five features, for five types of features: clean audio, noisy audio, DCT, LDA and optical flow.

We start with an initial set of 64 DCT features, obtained from applying the DCT transform on the segmented ROIs. The initial resolution of the rotated, centered and scaled ROIs is 128x128. After applying the 2D DCT, only the odd columns are kept, as it was shown that they have a higher relevance for visual speech recognition [83], and removing the even columns is equivalent to imposing horizontal symmetry to the image. Figure 4.12 shows that this operation greatly improves speech recognition results at low dimensionality.

The remaining coefficients are ordered according to the zig-zag ordering scheme detailed in Section 2.5.3, and the first 64 are kept. First and second temporal derivatives are computed on all coefficients, increasing the dimensionality of the initial feature set F to 192.

The zig-zag ordering itself is also included in our experiments as a reference, however with an important change, that is, coefficients are interleaved with delta and delta-delta values (first and second temporal derivatives). This is done because all the other selection algorithms are allowed to freely choose between normal coefficients, deltas and delta-deltas. Indeed, this interleaving scheme greatly improves the performance of the zig-zag ordering for low dimensionality values.

4.7.4 Maximum mutual information selection

The only algorithm that has been used before for visual feature selection in AVSR is the Maximum Mutual Information. Basically, it means features are selected only by their MI value with the class, irrespective of previously selected features. Using the same notation as in the previous chapter, the following equation is used:

$$Y_{\pi_{k+1}} = \arg \max_{Y_i \in F \setminus S_k} I(Y_i; C) \tag{4.7}$$

The selection algorithm is reduced to sorting the features in F by their MI value and picking only the top part of the list for inclusion in the subset S.

Figure 4.13 shows the first 5 MI values for the best features. We analyzed five types of features: MFCCs from clean audio, MFCCs from noisy audio, DCT features, LDA and finally our 2D optical flow features. What can be seen from the graph is that, as expected, the clean audio contains the highest amount of information about classes. However, when the audio is corrupted by noise, the amount of information decreases drastically, and below the level in any of the visual features. Between these visual features, the LDA coefficients have the highest MI for the first few features. However, the MI for the latter ones decreases faster in the case of LDA coefficients than in the case of the DCT. Finally, the two motion features have the lowest MI from all the visual features, in spite of having virtually the same visual-only performance as the two best LDA features. This could be explained by the fact that this graph does not include the redundancy between the features. Indeed, two features could contain exactly the same information about the class, or even have identical values, and still both of them would be selected if their MI with the class is high. In the case of optical flow and LDA features, it is possible that although the MI values are higher for the first two LDA coefficients compared to the optical flow, the redundancy may also be higher.

This hierarchy is also reflected in the monomodal classification results, confirming that MI is a good measure for feature relevance.

Comparing only the DCT and LDA coefficients' MI values, it would seem that the data processing inequality is violated. Indeed, the LDA coefficients are obtained directly through the application of a transform (the LDA) on the DCT coefficients, the same which are shown on the graph. The data processing inequality claims that information can not be created when a transform is applied on the data, so the LDA MI values should not be higher, while, in fact, they are. What happens in fact is that we do not have access to the whole picture. The inequality is valid for the whole set of features $I(F_{LDA}; C)$ should be smaller or equal to $I(F_{DCT}; C)$, however these measures can not be estimated. For some individual coefficients, MI values can increase perhaps through a redistribution of information caused by the application of the transform, which is allowed by the data processing inequality.

Figure 4.14 shows the visual-only recognition results for the maximum MI selection algorithm, for feature dimensionality ranging from 4 to 192. Results for the zig-zag ordering and for LDA are also included for comparison. What can be seen from the graph is that both DCT feature types outperform the LDA at higher dimensionality, but the LDA is better between 4 and 20 features. The maximum MI method obtains a maximum performance for around 60 features. When dimensionality increases beyond that, the accuracy begins to decrease.

Figure 4.15 shows the audio-visual recognition performance for the same features, for an audio SNR of -10 dB with babble noise. Although the audio-only accuracy in this case is

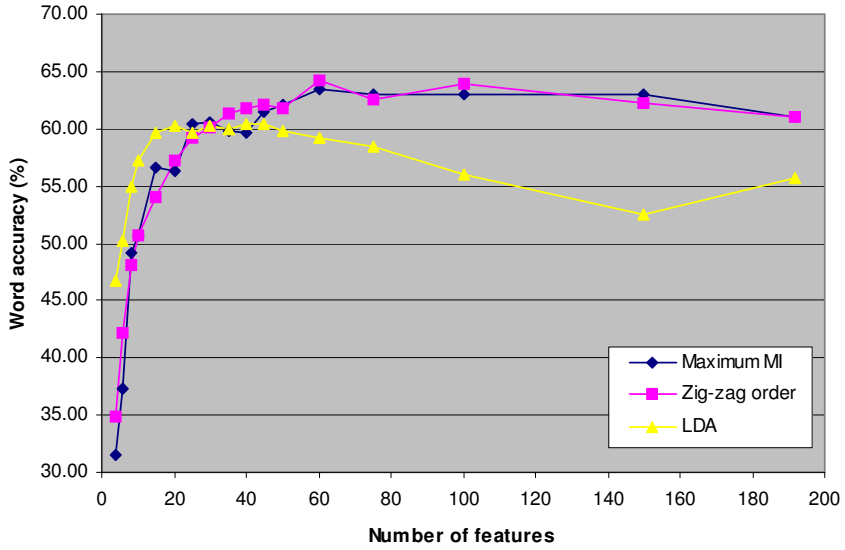

Figure 4.14 — The visual-only recognition results for the maximum MI selection algorithm.

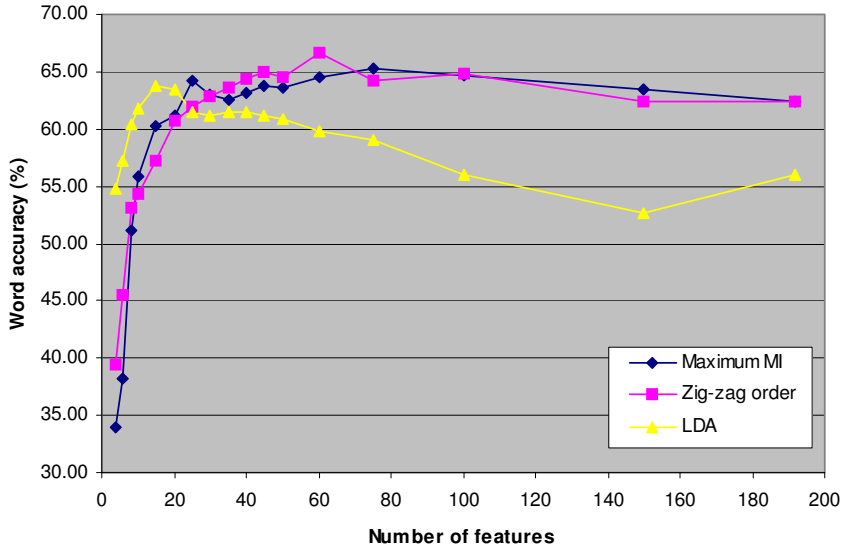

Figure 4.15 — The audio-visual recognition results at -10 dB SNR with babble noise for the maximum MI selection algorithm. The audio-only accuracy is only 22.3% .

	Number of features																
	4	6	8	10	15	20	25	30	35	40	45	50	60	75	100	150	192
MaxMI AV -10 dB	34.0	38.2	51.1	55.9	60.2	61.2	64.2	63.1	62.5	63.1	63.8	63.6	64.5	65.3	64.6	63.5	62.4
Zigzag AV -10 dB	39.4	45.5	53.1	54.3	57.2	60.7	61.9	62.9	63.7	64.4	64.9	64.5	66.6	64.2	64.8	62.4	62.4
LDA AV -10 dB	54.8	57.2	60.4	61.8	63.8	63.4	61.5	61.1	61.4	61.5	61.2	60.8	59.9	59.0	56.0	52.7	55.9
MaxMI VO	31.6	37.3	49.2	50.7	56.6	56.3	60.4	60.6	59.7	59.7	61.5	62.0	63.5	63.1	63.0	62.9	61.1
Zigzag VO	34.9	42.1	48.1	50.7	54.0	57.2	59.2	60.1	61.3	61.9	62.1	61.8	64.2	62.5	63.9	62.2	61.1
LDA VO	46.7	50.2	54.9	57.2	59.7	60.3	59.6	60.3	60.0	60.5	60.5	59.9	59.2	58.5	56.0	52.5	55.7

Table 4.2 — Results with the maximum MI selection algorithm, both visual-only and audio-visual.

only 22.3%, audio-visual recognition rates are in all cases better than both audio-only and visual-only ones. The exact values can be found in Table 4.2. What can be noticed from the table is that there is a larger gain from multimodality at lower dimensionality values compared to higher ones. For example, for 10 visual features, 5.2% are gained, while for 100, the gain is of only 1.6%. This could mean that the curse of dimensionality is impeding the gains in performance with a high number of features.

As can be seen, the Maximum MI selection method is not a very good performer compared to the two state of the art methods, neither for visual-only nor for audio-visual recognition. In the next section we will present methods that, by contrast, also penalize redundant features, which leads to better performance.

4.7.5 MMI with weighted redundancy penalty

An algorithm introduced for classification, MIFS (Mutual Information Feature Selection) [5], penalizes features for their redundancy with other features in the selected set. The equation used to select features at each step is:

$$Y_{\pi_{k+1}} = \arg \max_{Y_i \in F \setminus S_k} \left[I(Y_i; C) - \beta \sum_{Y_{\pi_j} \in S_k} I(Y_i; Y_{\pi_j}) \right] \tag{4.8}$$

The parameter β is the proportion of the redundancy that is penalized by this algorithm, and is recommended by the author in [5] to be set between 0.5 and 1. The justification for this is that we are trying to penalize the redundancy of the feature Y_i with respect to the whole set S_k, that is $I(Y_i; S_k)$, which unfortunately we can not compute. The sum $\sum_{Y_{\pi_j} \in S_k} I(Y_i; Y_{\pi_j})$ is actually an upper limit for that redundancy, reached in the case when all the features in the set S_k are disjoint, that is, MI between any of them is zero. Of course this is not the case, so the real penalty should be lower then the sum, and hence the parameter β. However, the true value of the parameter depends on the particularities of the data, and it fact, the optimal β is different at each selection step. We choose the parameter β heuristically, as in [5].

Figure 4.16 shows a comparison between the results of the algorithm for visual-only recognition for three values of the parameter β, 0.5, 0.75 and 1.0. As can be seen the best is $\beta = 0.5$, which outperforms not only the other values, but LDA and zig-zag features as well, as shown in Figure 4.17. All visual-only results are listed in Table 4.3.

For audio-visual speech recognition, the same tendency is seen as in the previous section, that is, results improve more at low dimensionality than at high dimensionality when moving from single modality to multimodal processing. For example, for 10 features with $\beta = 0.5$, the gain is 5.5%, while for 50 it is only 0.3%.

Figure 4.18 shows audio-visual results for the lowest SNR audio, -10 dB, with babble noise. The MIFS selection method performs better than either the zig-zag ordering, or the LDA, at low dimensionality.

The results obtained with the MIFS method show how important it is to eliminate the redundancy between features. However, the MIFS method has an important drawback,

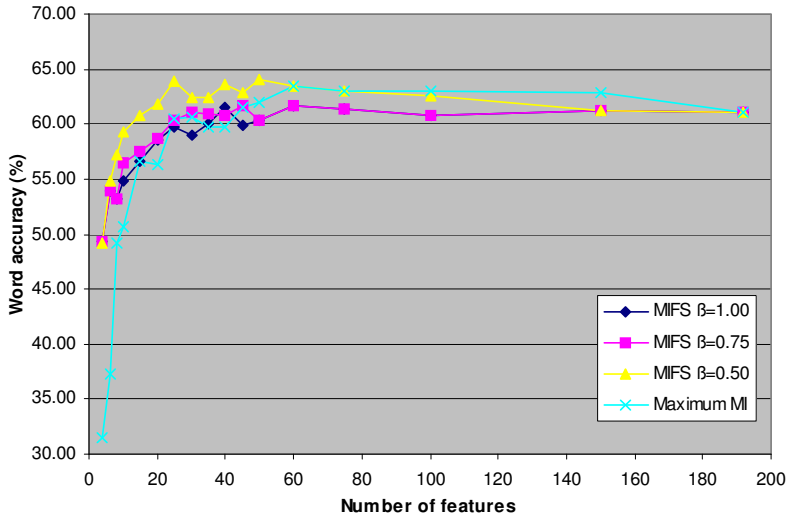

Figure 4.16 — Visual-only results with MIFS, using three different values for the parameter β, 0.5, 0.75 and 1.0 . Results with the previous method presented, maximum MI, are included for comparison.

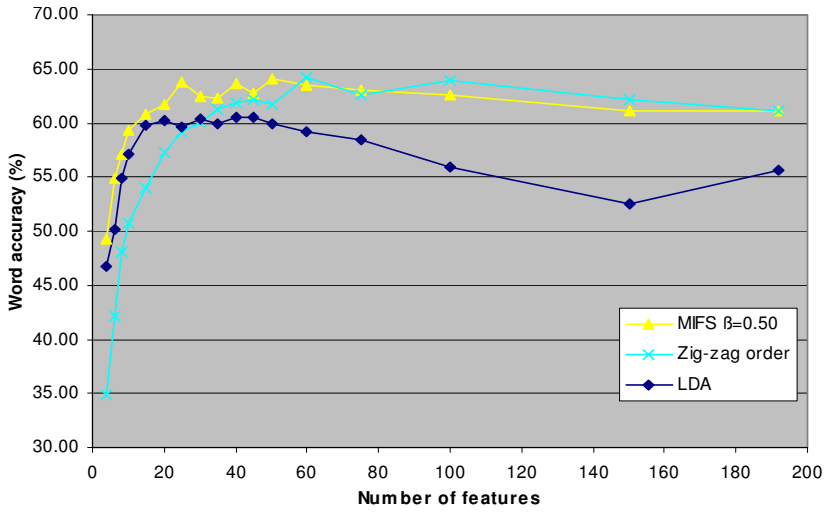

Figure 4.17 — Visual-only results with MIFS, with $\beta = 0.5$ compared to LDA and the zig-zag order.

	Number of features																
	4	6	8	10	15	20	25	30	35	40	45	50	60	75	100	150	192
MIFS β=1.00 VO	49.3	53.9	53.1	54.9	56.6	58.6	59.7	59.0	60.1	61.6	59.9	60.4	61.6	61.3	60.7	61.2	61.1
MIFS β=0.75 VO	49.3	53.9	53.1	56.4	57.5	58.7	60.3	61.0	60.9	60.7	61.7	60.4	61.6	61.4	60.8	61.2	61.1
MIFS β=0.50 VO	49.3	54.9	57.2	59.4	60.8	61.8	63.8	62.5	62.4	63.6	62.8	64.1	63.5	63.1	62.6	61.2	61.1
MaxMI VO	31.6	37.3	49.2	50.7	56.6	56.3	60.4	60.6	59.7	59.7	61.5	62.0	63.5	63.1	63.0	62.9	61.1
Zigzag VO	34.9	42.1	48.1	50.7	54.0	57.2	59.2	60.1	61.3	61.9	62.1	61.8	64.2	62.5	63.9	62.2	61.1
LDA VO	46.7	50.2	54.9	57.2	59.7	60.3	59.6	60.3	60.0	60.5	60.5	59.9	59.2	58.5	56.0	52.5	55.7

Table 4.3 — Results with the MIFS selection algorithm, visual-only.

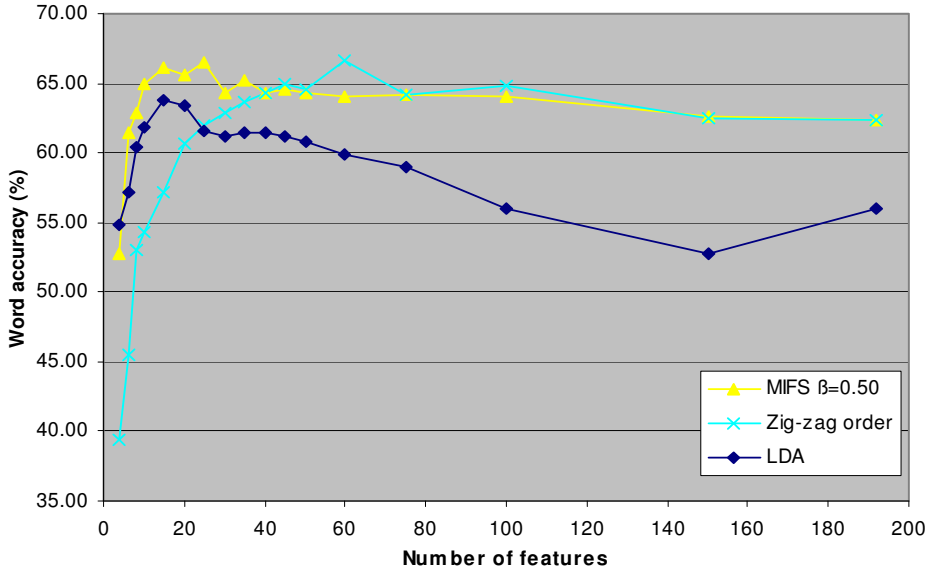

Figure 4.18 — Audio-visual results with MIFS, with $\beta = 0.5$, compared to LDA and the zig-zag order.

	Number of features																
	4	6	8	10	15	20	25	30	35	40	45	50	60	75	100	150	192
MIFS β=0.5 AV -10dB	52.7	61.4	62.9	64.9	66.1	65.7	66.5	64.3	65.2	64.3	64.6	64.4	64.1	64.1	64.1	62.6	62.4
Zigzag AV -10 dB	39.4	45.5	53.1	54.3	57.2	60.7	61.9	62.9	63.7	64.4	64.9	64.5	66.6	64.2	64.8	62.4	62.4
LDA AV -10 dB	54.8	57.2	60.4	61.8	63.8	63.4	61.5	61.1	61.4	61.5	61.2	60.8	59.9	59.0	56.0	52.7	55.9

Table 4.4 — Results with the MIFS selection algorithm, audio-visual. The audio is at -10 dB SNR with babble noise, and audio-only recognition is only at 22.3% .

that is, the parameter β needs to be chosen correctly for the method to give good results. In the next section we present a method which does not depend on any parameter to set the penalty for redundancy.

4.7.6 Selection with conditional mutual information

Another possibility when choosing features would be to maximize the conditional mutual information (CMI), that is, the information that is added by a feature to what was already known about the class label through the other features. The equation used by the CMI algorithm is:

$$Y_{\pi_{k+1}} = \arg \max_{Y_i \in F \setminus S_k} \left[\min_{Y_{\pi_j} \in S_k} I(Y_i; C | Y_{\pi_j}) \right] \tag{4.9}$$

which, by using $I(X;Y;Z) = I(X;Y) - I(X;Y|Z)$ [18], becomes:

$$Y_{\pi_{k+1}} = \arg \max_{Y_i \in F \setminus S_k} \left[I(Y_i; C) - \max_{Y_{\pi_j} \in S_k} I(Y_i; Y_{\pi_j}; C) \right] \tag{4.10}$$

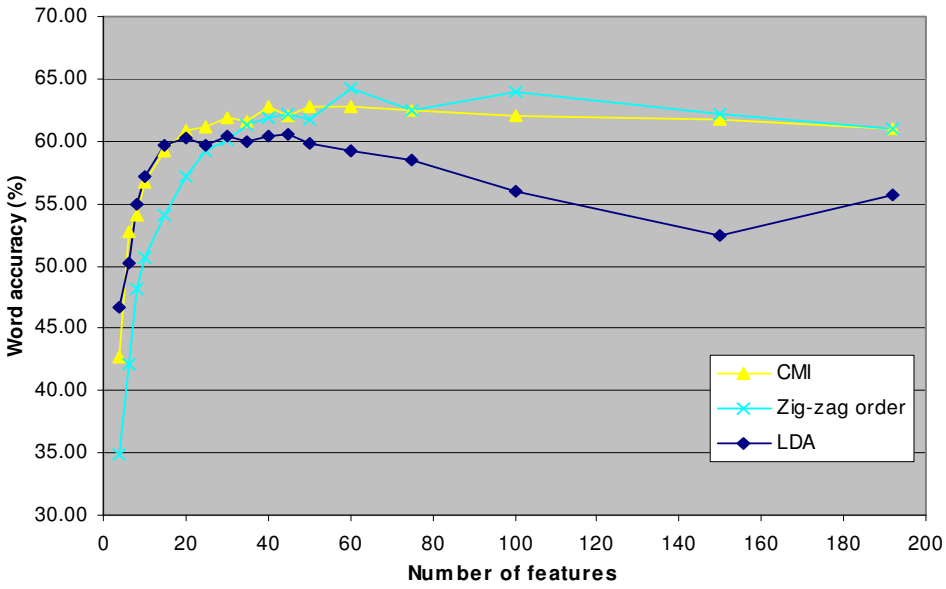

Figure 4.19 — Visual-only results with CMI.

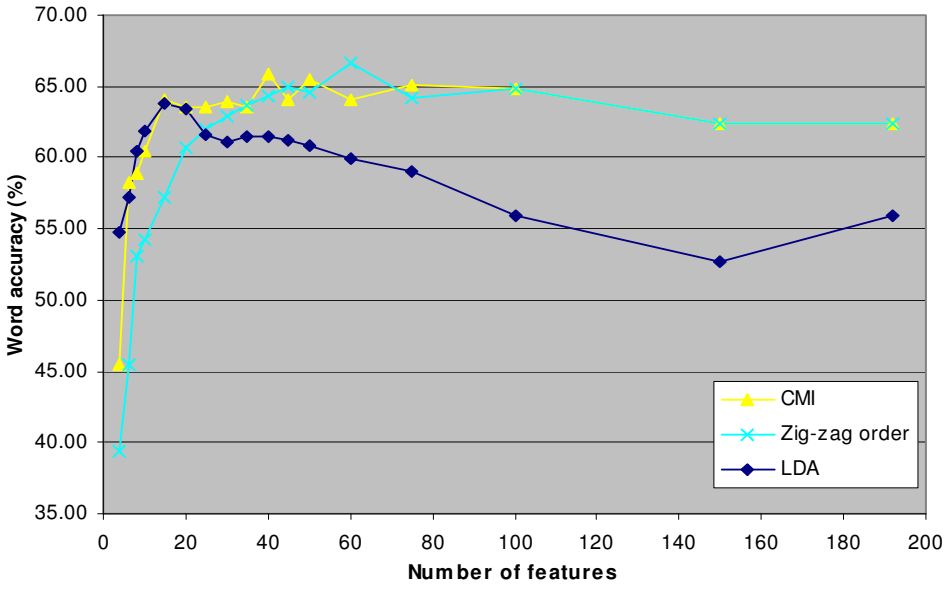

Figure 4.20 — Audio-visual results with CMI.

	Number of features																
	4	6	8	10	15	20	25	30	35	40	45	50	60	75	100	150	192
CMI AV -10 dB	45.4	58.2	58.9	60.5	64.1	63.6	63.6	63.9	63.5	65.8	64.1	65.5	64.0	65.0	64.8	62.4	62.4
CMI VO	42.6	52.7	54.1	56.7	59.3	60.9	61.1	61.8	61.5	62.8	62.0	62.8	62.8	62.5	62.0	61.8	61.1

Table 4.5 — Results with CMI.

.

This method is thus penalizing only *relevant* redundancy, $I(Y_i; Y_{\pi_j}; C)$. As was shown in Section 4.7.2, this measure can be either positive or negative. When it is positive, it can be interpreted as the relevant redundancy, that is, the information about the class label that is shared, or redundant, between two features. However, when this measure is negative, it can have a different interpretation, as a measure of synergy, that is, how much information about the class label is added by taking two features together, compared to just taking them individually.

The CMI $I(Y_i; C|Y_{\pi_j})$ represents the information about C that is brought by Y_i that is supplementary to what was already known through Y_{π_j}. Obviously the measure that should be used in fact is $I(Y_i; C|S_k)$, the information about C brought by Y_i that is complementary to the information in the whole set of selected features, S_k. However, because of the high dimensionality, this can not be computed, so we have to rely on approximations.

The CMI algorithm works as follows. First, for a certain candidate feature Y_i which was not included in the selected feature set yet, a corresponding feature Y_{π_j} is found, the one that is maximally redundant to it. This can be expressed either by minimizing $I(Y_i; C|Y_{\pi_j})$ or by maximizing $I(Y_i; Y_{\pi_j}; C)$, as can be seen from Eq. (4.9) and Eq. (4.10). The second step is choosing the feature Y_i which adds the most information about C to its most redundant corresponding feature, Y_{π_j}. This basically assumes that if the feature Y_i adds a lot of information even compared to its most redundant counterpart, it will also add information not present in the whole set S_k.

Figure 4.19 shows visual-only results with the CMI selection algorithm. The performance is on the same level as the LDA for the low dimensionality values, and similar to the zig-zag order for the higher dimensionality values. Compared to the maximum MI algorithm performance is better, however compared to MIFS it is a little worse. The same tendencies can be seen on the audio-visual performance graph, Figure 4.20. It is possible that there are cases where the CMI algorithm penalizes features too much, that is, it considers them as having too much redundancy, although by themselves they contain a lot of information.

4.7.7 Performance in clean conditions

The previous analysis was done only with corrupted audio, with a SNR of -10 dB. However, our goal is to have a system which performs well across all conditions. In the following, we will present results at other SNRs as well.

Figure 4.21 and Table 4.6 show audio-visual performance with LDA, CMI and MIFS, compared to audio-only. As can be seen, the CMI method is the best in this case, although by a very slight margin. For clean audio, the monomodal recognition accuracy is already very high, at 98.3%. Multimodal recognition improves this, irrespective of the visual features' dimensionality. Both the CMI and the MIFS algorithms further improve performance compared to the LDA, although by very little in absolute terms. However, given the very small number of recognition errors, the relative reduction in error is impressive, 16.6% pass-

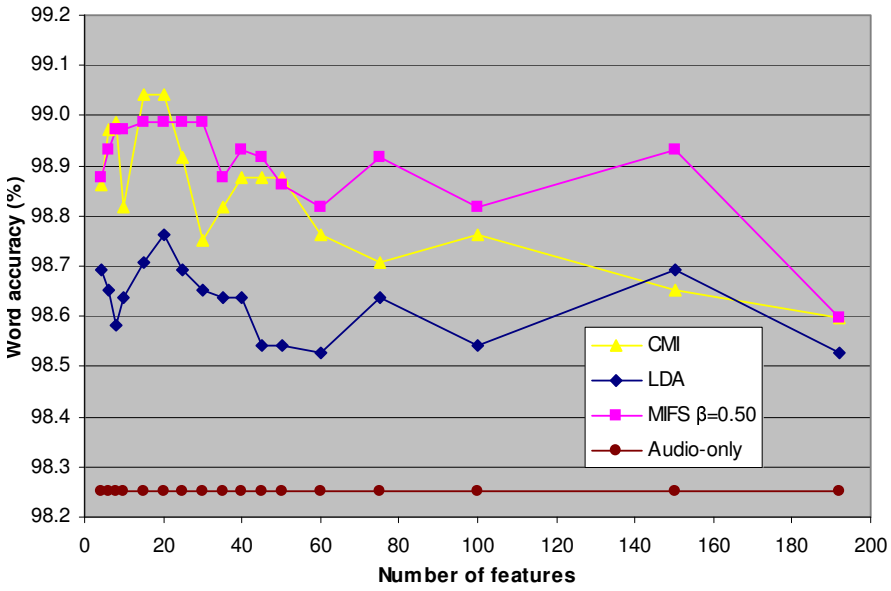

Figure 4.21 — Audio-visual results with clean audio, compared to audio-only.

	Number of features																
	4	6	8	10	15	20	25	30	35	40	45	50	60	75	100	150	192
MIFS β=0.5 AV clean	98.9	98.9	99.0	99.0	99.0	99.0	99.0	99.0	98.9	98.9	98.9	98.9	98.8	98.9	98.8	98.9	98.6
CMI AV clean	98.9	99.0	99.0	98.8	99.0	99.0	98.9	98.8	98.8	98.9	98.9	98.9	98.8	98.7	98.8	98.7	98.6
LDA AV clean	98.7	98.7	98.6	98.6	98.7	98.8	98.7	98.7	98.6	98.6	98.5	98.5	98.5	98.6	98.5	98.7	98.5
audio-only clean	98.3	98.3	98.3	98.3	98.3	98.3	98.3	98.3	98.3	98.3	98.3	98.3	98.3	98.3	98.3	98.3	98.3

Table 4.6 — Audio-visual results with clean audio.

ing from LDA to MIFS for example. Although there is a lot of variability in the results, the same tendency is seen across all dimensionality values, that is, MI-based methods are better than the LDA.

Another tendency worth mentioning is the fact that optimal dimensionality of the visual features for AV recognition is lower for clean audio. While for very corrupted audio, a visual dimensionality between 40 and 60 would lead to optimal results, for clean audio the situation changes and only around 20 visual features are needed to obtain the best results. This would seem to suggest that varying the modality weights may not be sufficient for an optimal audio-visual integration. Indeed, a method that would be able to change the number of features used from each modality "on the fly", according to the reliability of each stream, would probably be better than only adjusting the weights.

The fact that less visual features are necessary to obtain an optimal performance for clean audio could be explained by the redundancy between the audio and visual features, which is lost when the SNR decreases. This means that, when the audio is degraded, information that was common to audio and video is now only present in the video, and more features are necessary in order to include all this information in the feature set.

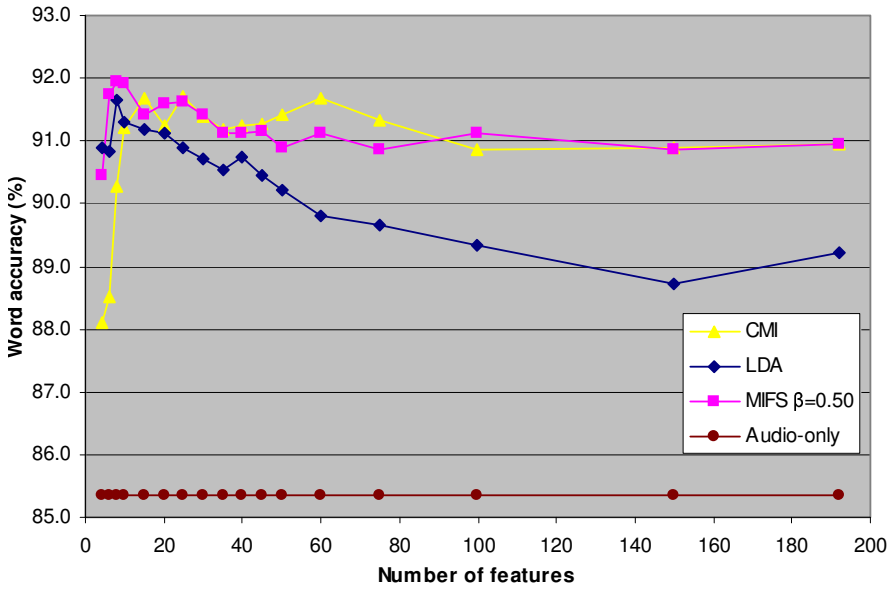

Figure 4.22 — Audio-visual results with audio SNR at 10 dB with babble noise, compared to audio-only.

	Number of features																
	4	6	8	10	15	20	25	30	35	40	45	50	60	75	100	150	192
MIFS β=0.5 AV clean	90.5	91.7	92.0	91.9	91.4	91.6	91.6	91.4	91.1	91.1	91.2	90.9	91.1	90.8	91.1	90.8	91.0
CMI AV 10 dB	89.9	91.3	91.4	91.6	91.8	92.0	91.1	91.2	91.7	91.8	91.4	91.5	90.8	91.0	91.0	91.0	91.0
LDA AV 10 dB	90.9	90.8	91.6	91.3	91.2	91.1	90.9	90.7	90.5	90.7	90.4	90.2	89.8	89.7	89.3	88.7	89.2
audio-only 10 dB	85.3	85.3	85.3	85.3	85.3	85.3	85.3	85.3	85.3	85.3	85.3	85.3	85.3	85.3	85.3	85.3	85.3

Table 4.7 — Audio-visual results with audio SNR at 10 dB with babble noise.

4.7.8 Performance at 10 dB SNR

At a SNR of 10 dB, the audio is quite corrupted and performance decreases to 85.3% for monomodal recognition. We used babble noise for all experiments, as this scenario is more challenging.

Audio-visual performance is presented in Figure 4.22 and Table 4.7. The gain from multimodal recognition is impressive, 6.7% in absolute terms, or a relative reduction in the number of errors of 45.6%. Again the best-performing method is MIFS. Both MI-based methods are better then the LDA at all dimensionality values.

The ideal number of visual features, the dimensionality that gives the highest performance for AV recognition is, surprisingly, 8. This is true for the MIFS features, which, when used alone for monomodal recognition, give the best results when the dimensionality is 25. The reduction in the number of features which give the best result compared to video-only may be explained by the curse of dimensionality, but also, as mentioned before, by the fact that there is redundancy with the audio, which means that only a reduced number of features bringing complementary information is actually required to augment the audio.

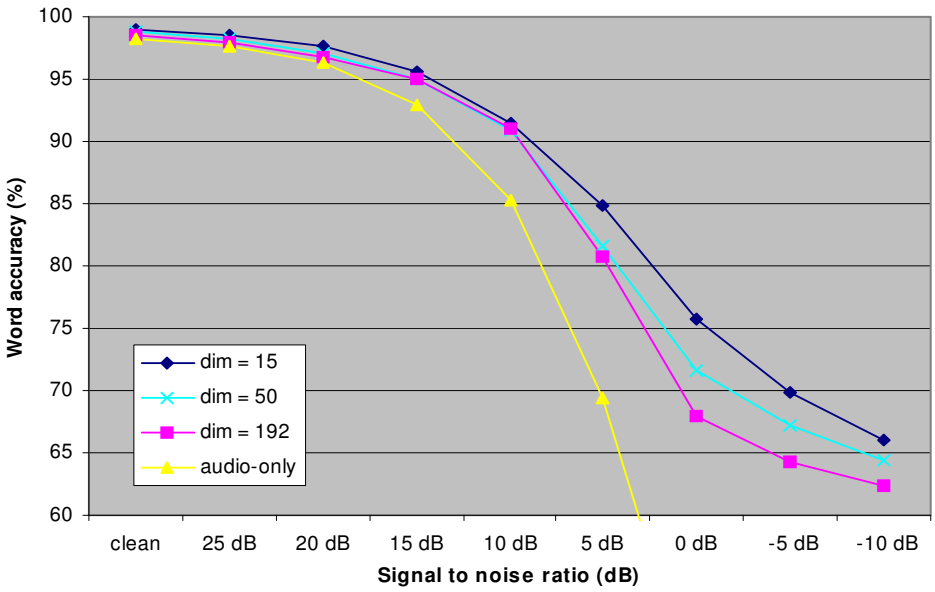

Figure 4.23 — Audio-visual results with 15, 50 and 192 visual features chosen with MIFS, compared to audio-only results, at all SNRs, with babble noise.

	SNR clean	25 dB	20 dB	15 dB	10 dB	5 dB	0 dB	-5 dB	-10 dB	VO
dim = 15	99.0	98.5	97.7	95.6	91.4	84.8	75.7	69.8	66.1	60.83
dim = 50	98.9	98.3	97.1	95.1	90.9	81.6	71.7	67.1	64.4	64.07
dim = 192	98.6	97.9	96.8	94.9	91.0	80.8	68.0	64.2	62.4	61.07
AO	98.3	97.7	96.3	92.9	85.3	69.4	44.8	28.4	22.3	

Table 4.8 — Audio-visual results with 15, 50 and 192 visual features. Audio-only and visual-only results are also included.

4.7.9 How many features to pick?

Up until now, we analyzed the performance of the audio-visual recognizer for different numbers of visual features, between 6 and 192. As mentioned above, the optimal number of visual features differs from SNR to SNR and even between the different types of features. However, there is a simple conclusion that can be drawn from all these experiments, and that is that having less features is preferable. Figure 4.23 shows audio-visual results for 15, 50 and 192 visual features chosen with MIFS. The audio is corrupted with babble noise. It is obvious from the graph that the higher dimensionality features have, in general, a lower performance.

However, the optimal number of visual features is different for each SNR, and grows as the SNR decreases. This number also depends on the particular selection algorithm which is chosen. For example, while for maximum MI selection the optimal performance is obtained for 25 visual features with clean audio and 75 at -10 dB, the situation is completely different for MIFS, where the optimal number of visual features with clean audio is 15 and for noisy audio 25.

Finally, the best number of visual features will depend on the application. Ideally, the

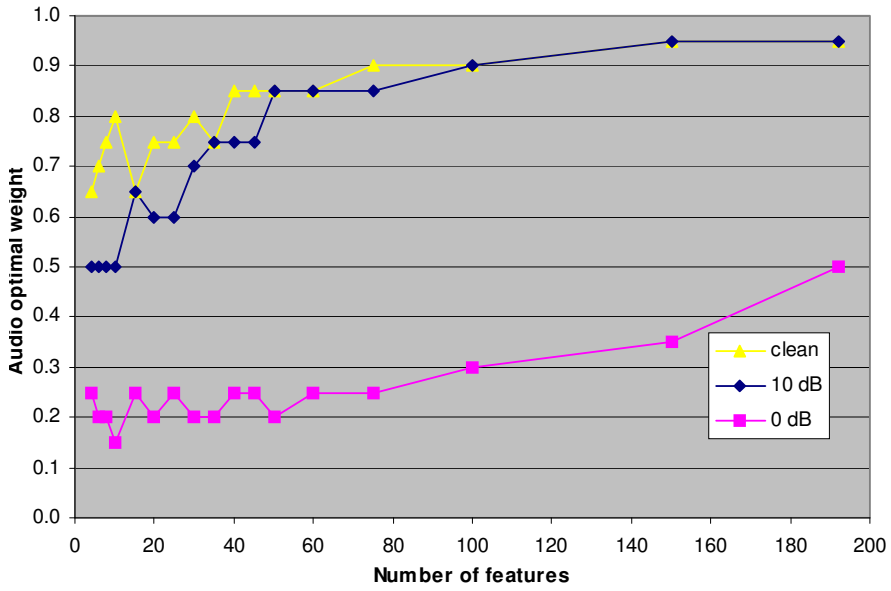

Figure 4.24 — The evolution of the optimal audio weight as the dimensionality of the visual features increases, for audio-visual experiments with features selected with MIFS, at different SNRs with babble noise.

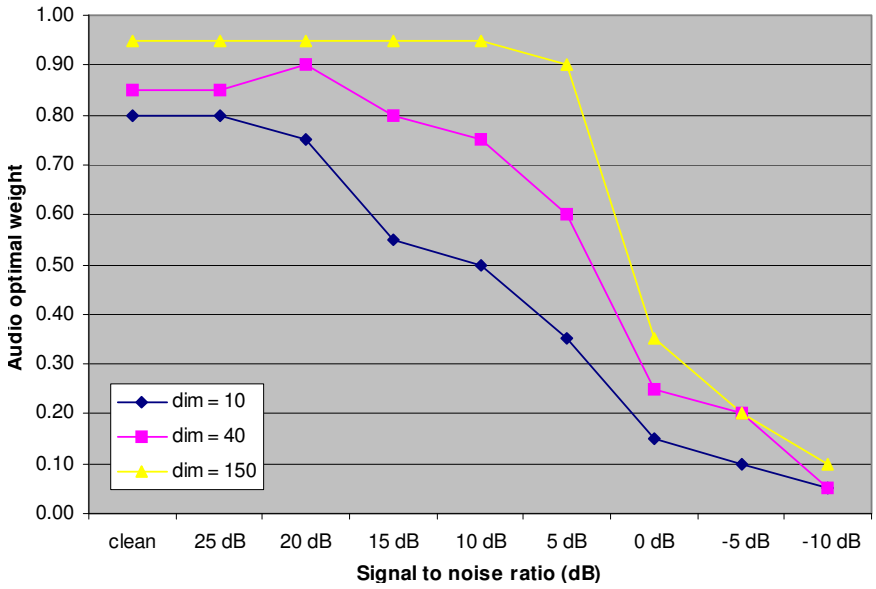

Figure 4.25 — The evolution of the optimal audio weight with the SNR, for 10, 40 and 150 visual features.

SNR of the target environment can be more or less established and the application can be built to optimize performance in the given SNR range.

4.7.10 The influence of feature dimensionality on the modality weights

In the previous analysis, the values of the modality weights were not mentioned. Intuitively, the audio weight should decrease as the SNR decreases, while the video weight would

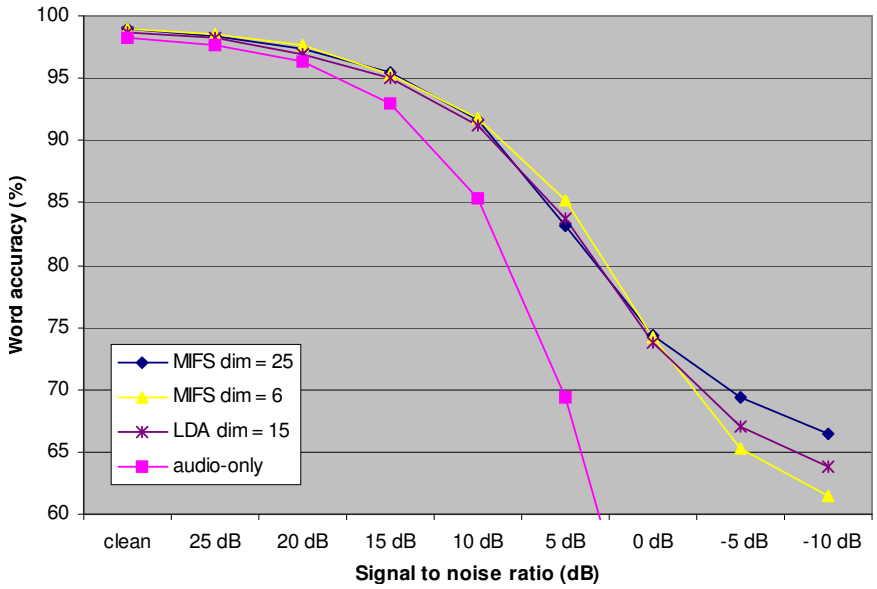

Figure 4.26 — Audio-visual results with MIFS at all SNRs, with babble noise.

increase, as their sum is fixed to 1.0. But how does varying the dimensionality of the visual features influence the weights?

Figure 4.24 shows the evolution of the optimal audio weight, the one that gives the best performance, with the number of visual features. The visual weight is always $\lambda_v = 1 - \lambda_a$. The graph shows that, as the dimensionality of the visual features increases, the weight assigned to the audio needs to increase, while the video one decreases accordingly. This basically shows that the likelihood values are dependent of the dimensionality. This effect is seen at all SNRs.

Figure 4.25 shows the evolution of the optimal audio weight with the audio SNR, with 10, 40 and 150-dimensional features. As expected, the optimal audio weight decreases with the SNR, but its actual values depend on the dimensionality of the features. The same trend can be seen, as higher dimensionality requires a lower weight for the video.

In the end, the conclusion is that the stream weighting algorithm needs to be adapted to the particular dimensionality of the streams involved, or the likelihoods need to be normalized in a dimensionality-independent manner.

4.7.11 Discussion

Given the fact that the particular dimensionality of the visual features that leads to the best performance depends on the SNR, a balance has to be found, choosing a dimensionality which is best suited to the conditions in which the application will be used. Figure 4.26 shows the audio-visual recognition results for MIFS at for 6 and 25 features selected. As can be seen, the lower-dimensional features have a slight advantage for all SNRs down to 0 dB, where the two curves cross and the higher-dimensional vector begins to perform better. Although the performance gain is large at very low SNRs, these extreme conditions

would likely not be encountered in practice, so the low-dimensional feature vector is still preferable. Comparing to the performance of the LDA in the same conditions, there is little difference at SNRs above 0 dB, however the MIFS features have a slight advantage.

Our contribution here is two-fold. First, we propose two methods of visual feature selection for AVSR, methods based on maximizing mutual information with the classes while at the same time minimizing redundancy. We prove that penalizing features for their redundancy is important and obtain significant performance gains compared to the state of the art.

Secondly, we perform an extensive analysis of information theoretic feature selection methods applied on visual features for AVSR, for different audio SNRs, with two types of noise, white and babble. As opposed to many approaches in the literature, where a typically a visual feature vector of dimensionality 40 is used, we present results across a wide range of dimensionality values. We show that, in some cases, a small vector of only 15 features can outperform higher-dimensional ones, proving that obtaining low-dimensional features is always desirable.

4.8 Summary

This chapter details our work in feature selection for audio-visual speech recognition. Two types of methods are presented, one based on the motion of the lips, the other on information theory.

The first part of the chapter presents a method based on using only the motion of the lips, without requiring a very accurate tracking of the mouth, or even a complete image of the mouth. The goal was to extract the most information possible with very simple and low-dimensional features. The features that we extract do indeed lead to a significant improvement compared to audio-only recognition, while being at the same time simple and robust.

The second part of the chapter is dedicated to information-theoretic feature selection methods. We show that methods that take into account the redundancy between features perform better than the ones that do not. We also prove that lower-dimensional feature vectors are beneficial for recognition accuracy, although the particular value of the optimal number of features depends on the SNR.

Intuitively, the two types of features presented could be considered as the results of two opposite approaches. The first one is the prior knowledge approach, where we try to include the information that we consider as important in the extracted features, that is, the motion of the lips. The second is the statistical approach, where we assume no knowledge on the particular type of information that is important, and use a statistical method based on measuring information content to choose the features. Between the two, the statistical approach gives better results and is also more flexible. However, the prior knowledge may help in adverse conditions, like occlusions or severe tracking errors. It may be possible to combine the two, for example by using prior knowledge to extract more speech-specific features and then measuring their information content with statistic methods.

In the next chapter we will consider the second challenge of multimodal signal processing systems, the integration of modalities. We present a method to combine the audio and visual information from the HMM states with temporally varying weights which adjust dynamically to environment conditions.

Dynamic weighting for AVSR

5

5.1 Introduction

Merging information from different modalities is not trivial. First, the data streams may have different representations, different temporal rates and different ranges of variation. Second, they may be corrupted by noise in varying ways and at different moments. This is why multimodal fusion or integration is a very active area of research.

The case of AVSR is more complex from the multimodal integration point of view as compared to, for example, multimodal person identification. In AVSR both modalities are time-varying, which is not the case in identification, where typically at least some of the modalities are static, like face images or fingerprints.

The simplest multimodal integration method is feature concatenation, putting together the features from all modalities into one high-dimensional vector. However this has the big disadvantage of weighting all modalities equally and not allowing for variations in their relative importance. This problem is solved in decision fusion by assigning weights to each of the modalities. This also allows the dynamic adjustment of the importance of each stream through the weights, according to its estimated reliability.

The multimodal integration methods typically used for AVSR have been presented in Chapter 2. We are using a dynamic weighting scheme in which the weights are derived from the posterior entropies of each stream, and at each frame. The reason is that, in this way, the weights are allowed to vary quickly in a large range, thus being able to adapt to sudden changes in the quality of the streams.

In the following, we will present our method in detail. We begin by presenting the general algorithm and the justification behind it. We present our results with dynamic weights constrained to sum to one. We continue by justifying that the constant sum constraint is not necessary and show that consistently better results can be obtained by allowing the

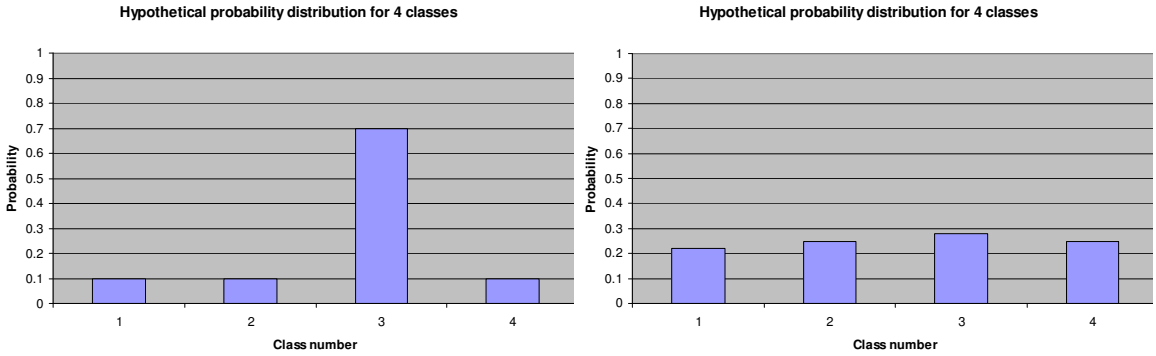

Figure 5.1 — Two hypothetical four-class posterior distributions. The one from the left has a very distinct peak, so the confidence that we can have in the result of the classification is high, while the one to the right is flat, leading to a low confidence in the classification result.

sum itself to be variable.

The content of this chapter is partially based on work that we have published in [39] and [40].

5.2 Estimating stream reliability with posterior entropies

As mentioned before, we are using multi-stream HMMs to recognize words. These recognizers require synchronous audio and visual feature streams, and perform multimodal fusion at the frame level, that is, a decision on the fusion is taken every 10 ms. As shown in Chapter 2, the emission likelihood b_j for state j and observation o_t at time t is the product of likelihoods from each modality s weighted by stream exponents λ_s [121]:

$$b_j(o_t) = \prod_{s=1}^{S} \left[\sum_{m=1}^{M_s} c_{jsm} N(o_{st}; \mu_{jsm}, \Sigma_{jsm}) \right]^{\lambda_s} \tag{5.1}$$

The stream weights λ_s are computed based on the estimated stream reliability, which is derived from the entropy of the class-posterior distributions for each stream.

The reasoning is as follows. Consider the case in Figure 5.1, where a simple hypothetical situation is presented. Assume that, in a multimodal 4-class problem, the posterior distribution has a very clear peak, as in the left figure. This means that there is a very good match between the test sample and the class model for the recognized class, and a very bad match with all the other classes. The confidence that we have in assigning the sample to the class is high, meaning that the confidence in the corresponding stream should also be high. On the other hand, when the posterior distribution is flat or nearly flat, like in the right figure, there is a high possibility that the sample was assigned to the wrong class, so the confidence is low, both in the classification result and the stream. To measure if the distribution is peaky or flat, we use the entropy:

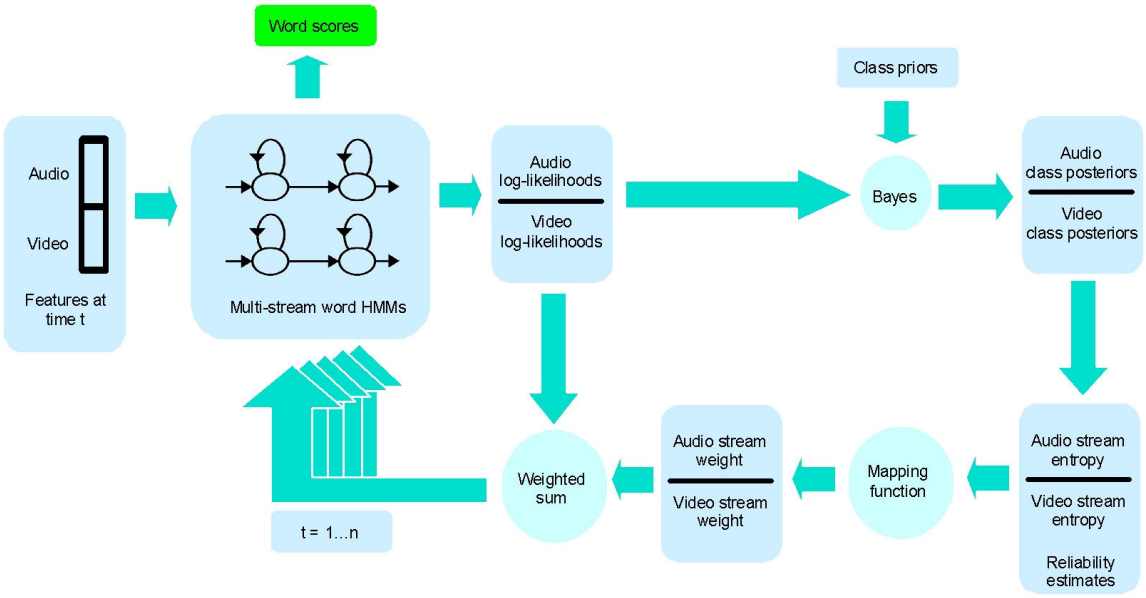

Figure 5.2 — The steps in the dynamic weights multimodal integration algorithm.

$$h_{st} = -\sum_{i=1}^{C} P(c_i|o_{st}) \log P(c_i|o_{st}) \tag{5.2}$$

The steps of the algorithm are presented in Figure 5.2. First, the audio and visual log-likelihoods are obtained for each class, that is, for each gaussian mixture in each of the states of the HMMs. Then, using Bayes' formula, we obtain the posteriors:

$$P(c_i|o_{st}) = \frac{P(o_{st}|c_i)P(c_i)}{\sum_j P(o_{st}|c_j)P(c_j)} \tag{5.3}$$

Here, the class priors $P(c_i)$ are the relative durations of the classes, obtained from the training set. The entropies of both distributions are then computed, and finally the weights are obtained through a mapping function.

The mapping function is required since the weights are the inverse of the entropies, and scaling also needs to be applied. Indeed, since a high entropy signifies a low confidence in the corresponding stream, a low weight should be assigned to it. The mapping functions we use will be detailed in the next section.

The big advantage of this algorithm is its flexibility in different environments. If one of the modalities becomes corrupted by noise, the posterior entropy corresponding to it would increase, making its weight, and so its importance to the final recognition, decrease. This is also valid in the case of a complete interruption of one stream. In this case, the entropy should be close to maximum, and the weight assigned to the missing stream close to zero. This practically makes the system revert to one-stream recognition automatically. This

process is instantaneous, and also reversible, that is, if the missing stream is restored, the entropy would decrease and the weight would increase to the level before the interruption.

Even in the case of static noise, the relative importance that should be given to each of the modalities may vary. As shown in Chapter 2, some speech sounds are easier to distinguish in the visual modality, while others are more distinguishable in the audio. This means that the posterior distribution peaks corresponding to these speech sounds will be more pronounced in one modality than the other, leading to a reduced entropy and a higher weight. Thus, in theory, our algorithm automatically favors the modality in which the sound is easier to distinguish.

All this means that the system can dynamically adapt to all levels of noise, including even the loss of one of the streams.

5.3 Choosing a mapping function from entropies to weights

5.3.1 Three static mapping functions

We have established that the entropies of the posterior distributions are suitable reliability estimators for the audio and visual streams. What is still needed is a function to transform these estimators into stream weights. This function is required since the weights and the reliability measure are not on the same scale, and their relation is non-linear.

Other approaches use training on a held-out subset of the training dataset, but, as mentioned before, we try to avoid this, as we consider that the type and intensity of noise in testing conditions is uncertain. This means that having a held-out set for weights training that matches the target noise conditions is very unlikely.

The approach that we take is to find a mapping that satisfies a few basic conditions. In our case, the relationship between the entropies and the weights is inverse, that is, when the entropy is low, the weight should be high, and vice-versa. We also impose for the moment that the sum of the weights be 1, that is $\lambda_a + \lambda_v = 1$.

Let the audio and video streams' entropies be H_a and H_v, and their associated weights λ_a and λ_v. The maximum value that the entropy can reach in our case is $H_{max} = log_2 83 \simeq 6.3$ since we have 83 classes. The mapping should ensure that when $H = 0$, $\lambda = 1$ and when $H = H_{max}$, $\lambda = 0$. Obviously, when $H_a = H_v$, $\lambda_a = \lambda_v = 0.5$.

There are several possible mappings that can be used. Two possibilities are presented below:

$$\lambda_a(t) = \frac{H_{max} - H_a(t)}{\sum_s H_{max} - H_s(t)} \tag{5.4}$$

$$\lambda_a(t) = \frac{1/H_a(t)}{\sum_s 1/H_s(t)} \tag{5.5}$$

We will refer to equations 5.4 and 5.5 as the "negative entropy" and "inverse entropy" mappings. The first one uses the difference to reverse the entropy, while the second uses

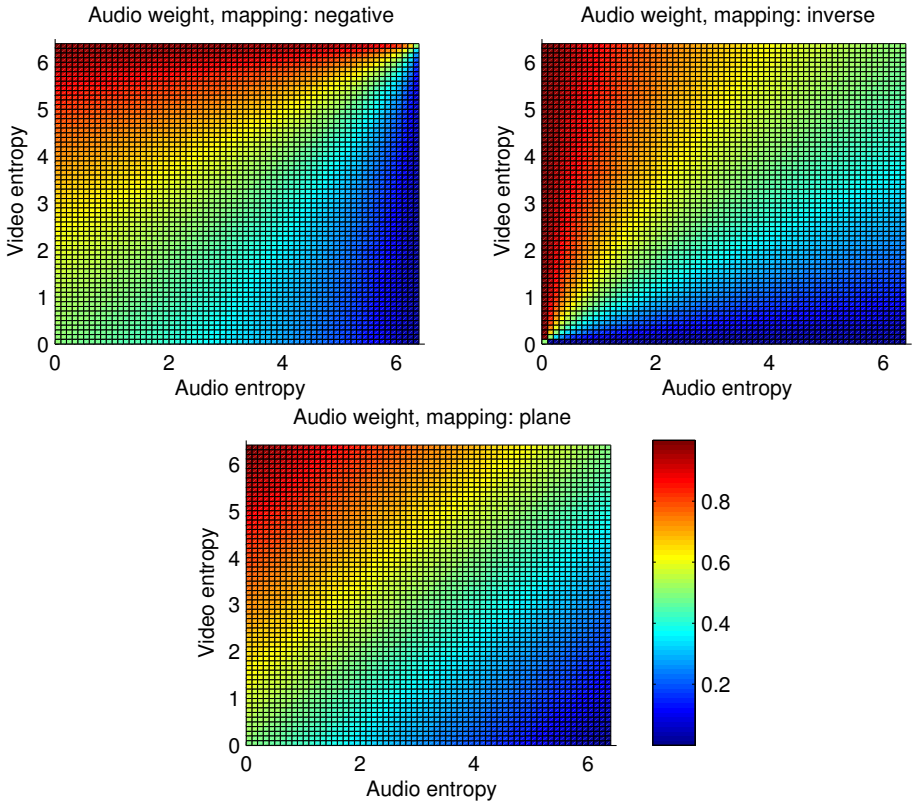

Figure 5.3 — Three possible mappings from posterior distribution entropy to stream weight.

the inverse. The difference between them is that the inverse mapping has a bias towards low values of the entropy, while the negative mapping has a bias towards high values.

The two mappings have a common shortcoming: if one of the entropy values is close to an extreme (either zero or H_{max}), a variation in the other entropy's value will have no effect. This can be seen in figure 5.3. For example, for the negative mapping, when the video entropy is close to H_{max}, the audio weight will be close to 1, irrespective of the value of the audio entropy. The preferable behavior would be for the audio weight to vary with the audio entropy, for example, when the audio entropy is also high, the weight to be close to 0.5. The inverse mapping has the same problem, since, when the audio entropy is close to zero, the audio weight is 1, even if the video entropy is also close to zero.

To avoid these problems, we derived a third mapping, which represents a plane in 3D space. The plane was derived using the following four points:

- $H_a = H_v = 0 \rightarrow \lambda_a = \lambda_v = 0.5$
- $H_a = 0; H_v = H_{max} \rightarrow \lambda_a = 1; \lambda_v = 0$
- $H_a = H_{max}; H_v = 0 \rightarrow \lambda_a = 0; \lambda_v = 1$
- $H_a = H_v = H_{max} \rightarrow \lambda_a = \lambda_v = 0.5$

The resulting equation is:

$$\lambda_a(t) = \frac{H_v - H_a}{2H_{max}} + \frac{1}{2} \tag{5.6}$$

As can be seen from the figure, using this mapping, the audio weight always varies with both entropies.

In the next section, we present our results with these three mappings.

5.3.2 Results with the static mappings

We compare our method with the Maximum Stream Posterior (MSP) method described in [104] [105]. The method is based on the premise that a stream weighting algorithm should outperform when possible either of the two streams when they are reliable, and perform at the same level as one of the stream when the other one is corrupted. The class-posterior probabilities for each stream are used, $P(s|o_a)$ and $P(s|o_v)$, where s is the state and o the observation vector. They are computed from the likelihoods with Bayes' formula. The combined stream class-posterior probability is also used, $P(s|o_{AV})$, derived as follows:

$$P(s|o_{AV}) = \frac{p(o_a|s)p(o_v|s)P(s)}{\sum_{s'} p(o_a|s')p(o_v|s')P(s')} \tag{5.7}$$

where $p(o_a|s)$ and $p(o_v|s)$ are the likelihoods. This would be equivalent to using the weights $\lambda_a = \lambda_v = 1.0$ in our case. The MSP method consists of using either the combined stream, or one of the monomodal streams, in order to maximize the stream posterior at each frame, as follows:

$$P(s|o) = \max\left[P(c|o_a), P(c|o_v), P(c|o_{AV})\right] \tag{5.8}$$

Figure 5.4 shows audio-visual results for dynamic weighting with the three mapping functions. As can be seen from the figure, the negative and plane mapping are practically equivalent, while the inverse mapping is performing slightly worse. For the first two SNR levels, clean and 25 dB, the performance is equivalent to audio-only recognition, but for lower SNRs there is an ever-increasing gain from audio-visual recognition.

The reason for the fact that the negative and plane mappings lead to the same results may be that the entropy values are not covering the whole space between $(0.0, 0.0)$ and (H_{max}, H_{max}). Indeed, in Figure 5.3, if splitting the images along the second diagonal, between the points $(0.0, H_{max})$ and $(H_{max}, 0.0)$, the lower halves for the negative and the plane mapping images are very similar. If most entropy values are confined in that interval, the results should also be similar indeed.

Figure 5.5 compares our dynamic weights method with negative mapping with the MSP method and the fixed weights method used in the previous chapter. The fixed weights methods consists in searching for the optimal pair of weights for a given SNR and using it for the entire sequence, with no time variation. This method is not applicable in practice, as typically the test SNR is not known at training time and is also not necessarily constant. The fixed weights results are presented here as a measure of how well multimodal integration can perform with ideal weights.

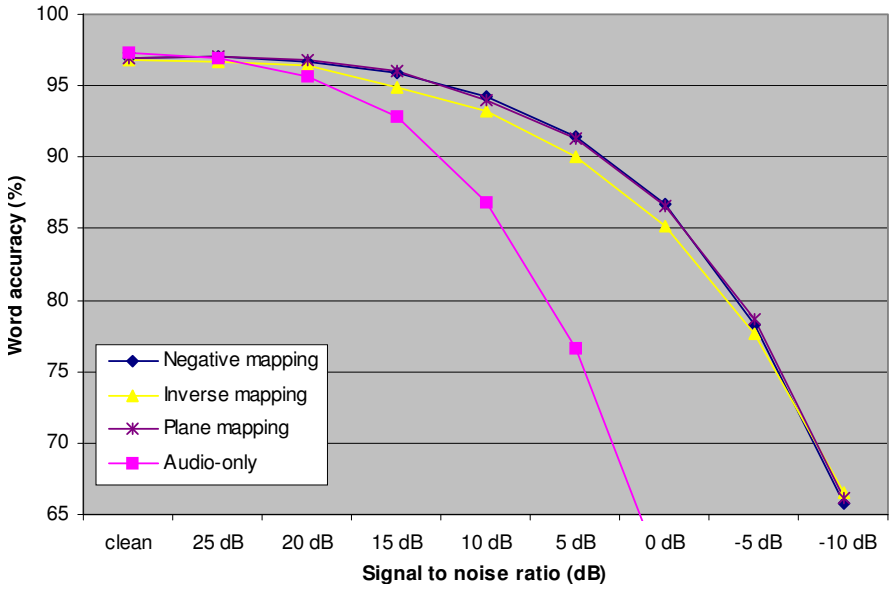

Figure 5.4 — Audio-visual results with dynamic weights and the three mapping functions, inverse, negative and plane, for all SNRs, with white noise.

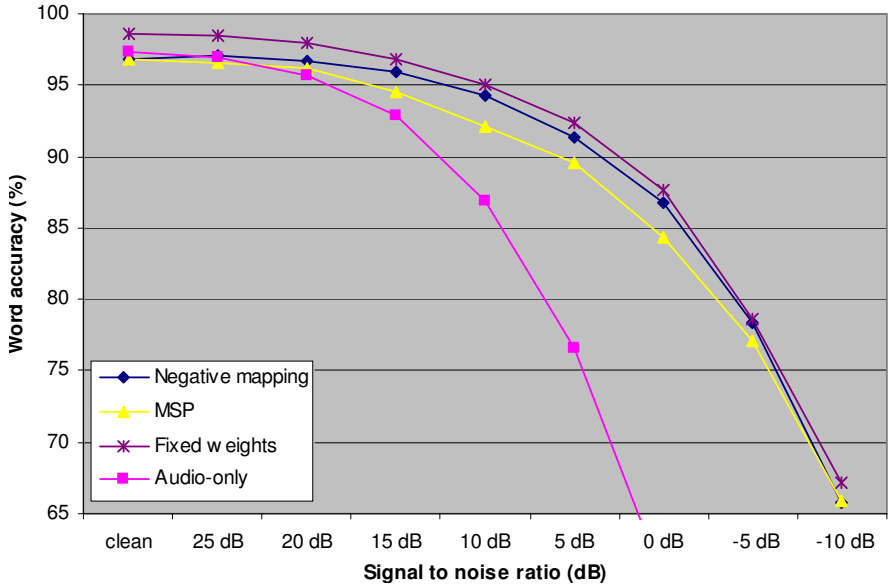

Figure 5.5 — Results for audio-visual recognition with dynamic weights, fixed weights and the MSP method, for all SNRs, with white noise.

	SNR								
	clean	25 dB	20 dB	15 dB	10 dB	5 dB	0 dB	-5 dB	-10 dB
negative	96.88	97.10	96.65	95.93	94.30	91.39	86.71	78.33	65.82
inverse	96.82	96.65	96.48	94.91	93.18	90.10	85.20	77.70	66.55
plane	96.99	97.10	96.82	95.98	94.02	91.28	86.54	78.61	66.15
MSP	96.76	96.54	96.15	94.47	92.12	89.61	84.41	77.11	65.86
fixed	98.66	98.44	97.93	96.81	95.02	92.34	87.65	78.60	67.17
AO	97.31	96.97	95.61	92.87	86.88	76.64	60.38	40.35	19.55

Table 5.1 — Audio-visual results for dynamic weights, fixed weights, and the MSP method, for all SNRs, with white noise.

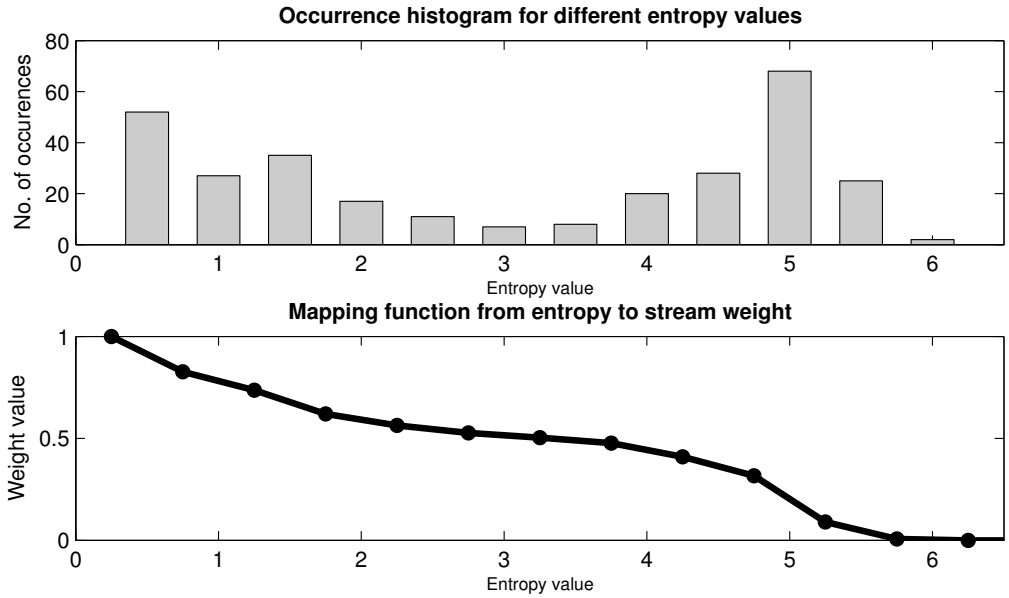

Figure 5.6 — A flexible mapping function from entropy to weight.

The MSP method performs worse than our method at most SNR levels, with the highest difference at 10 dB, 2.2%. The reason for this poor performance may be the fact that the MSP method takes extreme decisions, either considering a 50% - 50% combination of the two streams, or ignoring one of the streams entirely. Our method, by contrast, is more flexible, allowing any weighted combination of scores from the two streams, at any moment.

Both methods perform worse than the fixed weights for clean audio and high SNRs, as the entropy of the clean audio does not become low enough compared to the video to allow the high difference in weight values which leads to high performance for these SNRs.

In fact, the entropies for both audio and video do not cover the entire range from 0.0 to H_{max}. It may be better to use a mapping which is more "sensitive" in the parts of this range where there actually are a lot of audio and video entropy values. The next section investigates such a mapping.

5.3.3 A dynamic mapping

The mappings presented earlier cover uniformly the entire space between 0 and H_{max}. If, hypothetically, the entropy values are concentrated in only a small region of this space, the variations in the weights values would be very small. An ideal mapping would, by contrast, concentrate its discriminative power only in that region, making small variations in entropy lead to large variations in the stream weights. If the entropy values of the two streams are consistently close, this type of mapping would strongly favor the stream which has an even slightly higher entropy.

Intuitively, this mapping should be more sensitive for some entropy value intervals com-

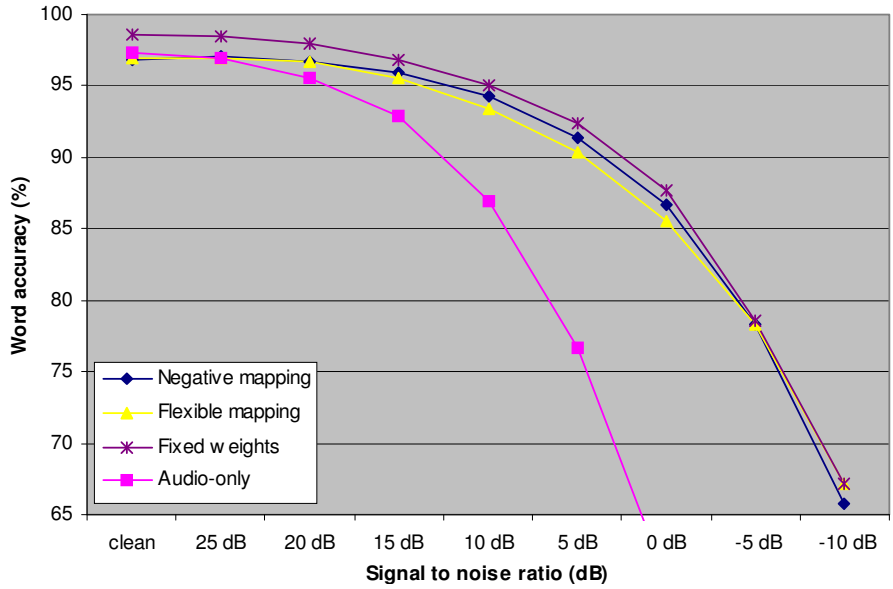

Figure 5.7 — Audio-visual results with dynamic weights and a flexible mapping, for all SNRs, with white noise.

	SNR								
	clean	25 dB	20 dB	15 dB	10 dB	5 dB	0 dB	-5 dB	-10 dB
negative	96.88	97.10	96.65	95.93	94.30	91.39	86.71	78.33	65.82
flexible	96.93	96.93	96.71	95.53	93.40	90.37	85.53	78.37	67.21
fixed	98.66	98.44	97.93	96.81	95.02	92.34	87.65	78.60	67.17
AO	97.31	96.97	95.61	92.87	86.88	76.64	60.38	40.35	19.55

Table 5.2 — Results for audio-visual recognition with dynamic weights and a flexible mapping, for all SNRs, with white noise.

pared to others, and those "sensitive" intervals should be the ones that include the entropy values that occur most often. This intuition lead to the following method of constructing a entropy to weights mapping.

First, a histogram of past entropy values is built for both streams. In our case, the histogram has 15 bins and comprises 150 past entropy values from both streams, for a total of 300 samples. Then, a piecewise-linear function is built, mapping low entropy values to high weights and vice-versa. This is done in such a way that the slope of each piece is proportional to the number of points contained in the corresponding histogram bin. Figure 5.3 shows an example of such a mapping and the histogram from which it was built.

This mapping is dynamic itself. It adapts to the particular configuration of entropy values, with the purpose of having the best discriminating power between the most occurring ones. As can be seen from the figure, the mapping is flat for the intervals where the number of frames with corresponding entropies is low, and steep where the number of frames is high. Its shape changes all the time, according to the particular distribution of entropy values.

Figure 5.7 and Table 5.2 show our results obtained with this mapping. As can be seen from the figure, the results are rather disappointing. The flexible mapping performs identical to the negative mapping for high SNRs and slightly worse for lower ones. A notable

exception is the -10 dB level, where this mapping performs better than all the others. However, overall the flexible mapping does not bring any improvement to performance compared to the previously presented methods.

5.4 The weight sum

5.4.1 The role of the weight sum

As mentioned before, most approaches in the literature impose that the sum of the weights should be equal to one. The reason might be to keep the result in the same range as the original likelihoods. However, the score that is computed by combining the likelihoods is not a likelihood (or emission probability) anymore. As shown before, in logarithmic domain the likelihoods are combined as follows:

$$\log b_j(o_t) = \sum_{s=1}^{S} \left[\lambda_s \cdot \log \sum_{m=1}^{M_s} c_{jsm} N(o_{st}; \mu_{jsm}, \Sigma_{jsm}) \right] \tag{5.9}$$

For our particular case, with two streams, audio and video, the equation becomes:

$$\log b_j(o_{AV}) = \lambda_a \log b_j(o_a) + \lambda_v \log b_j(o_v) \tag{5.10}$$

Indeed, there are no guarantees that the combined score would still integrate to 1 over the value range of the features, as the mono-modal likelihoods do, with or without the constraint $\lambda_a + \lambda_v = 1$. So, in fact, the constraint is not required.

To explain the effect that a different weights sum could have on the decoding, we recall the expression of the score of an observation sequence O and a path Q through a model ω, with respect to the emission likelihoods b_{q_i} and the transition probabilities $a_{q_i q_j}$.

$$\log p(O, Q|\omega) = \sum_{q_i \in Q} \log b_{q_i}(o_i) + \sum_{(q_i, q_j) \in Q} \log a_{q_i q_j} \tag{5.11}$$

which, when replacing the combined score with its expression from Eq. (5.10), becomes:

$$\log p(O, Q|\omega) = \lambda_a \sum_{q_i \in Q} \log b_{q_i}(o_i^A) + \lambda_v \sum_{q_i \in Q} \log b_{q_i}(o_i^V) + \sum_{(q_i, q_j) \in Q} \log a_{q_i q_j} \tag{5.12}$$

In a typical speech recognition system, the emission likelihoods b_{q_i} tend to become very small, making log-likelihoods large in absolute terms. Obviously, all log-likelihoods and log-probabilities are negative. In fact, the reason for using the logarithm in the first place is to prevent underflow errors that might be caused when multiplying likelihood values. By contrast, the transition probabilities $a_{q_i q_j}$ are larger than the emission likelihoods, and thus, have a much smaller absolute value in the logarithmic domain. This makes the emission likelihoods have a larger influence on the recognition results compared to transition probabilities. Indeed, if the difference in the range of variation is large, the path through the HMM and in consequence the score of the corresponding word could be influenced only

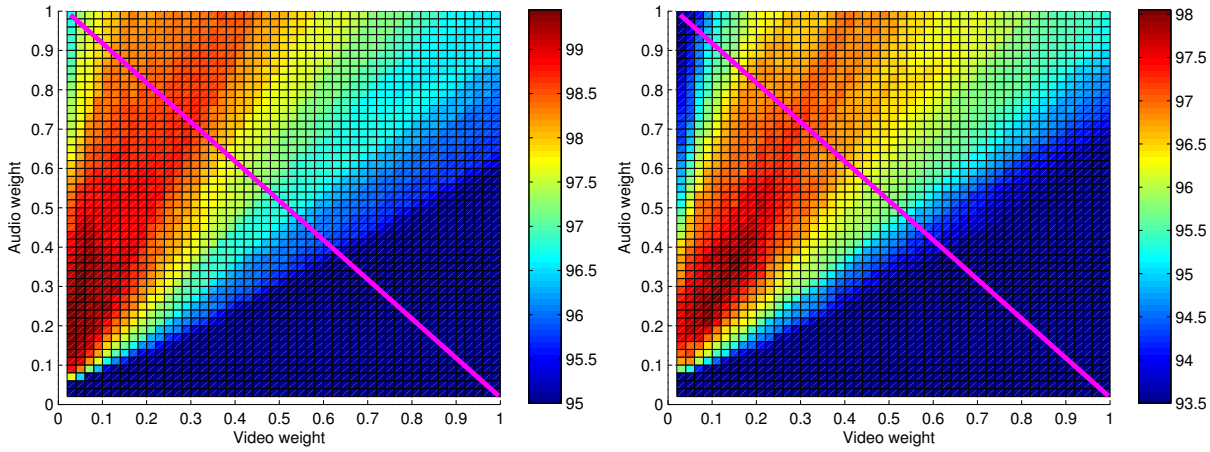

Figure 5.8 — AV accuracy, with unconstrained weights, for clean and 20 dB.

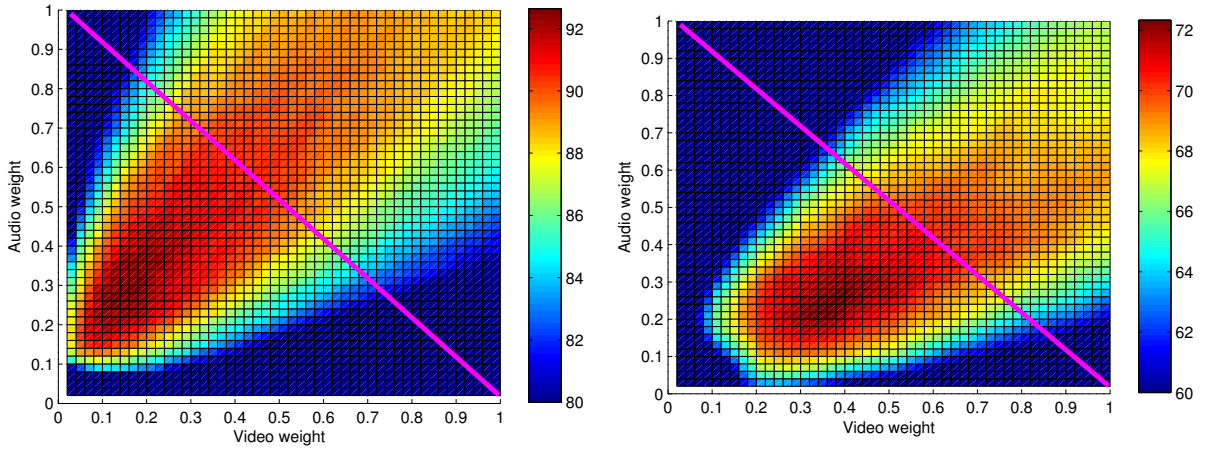

Figure 5.9 — AV accuracy, with unconstrained weights, for 10 dB and 0 dB.

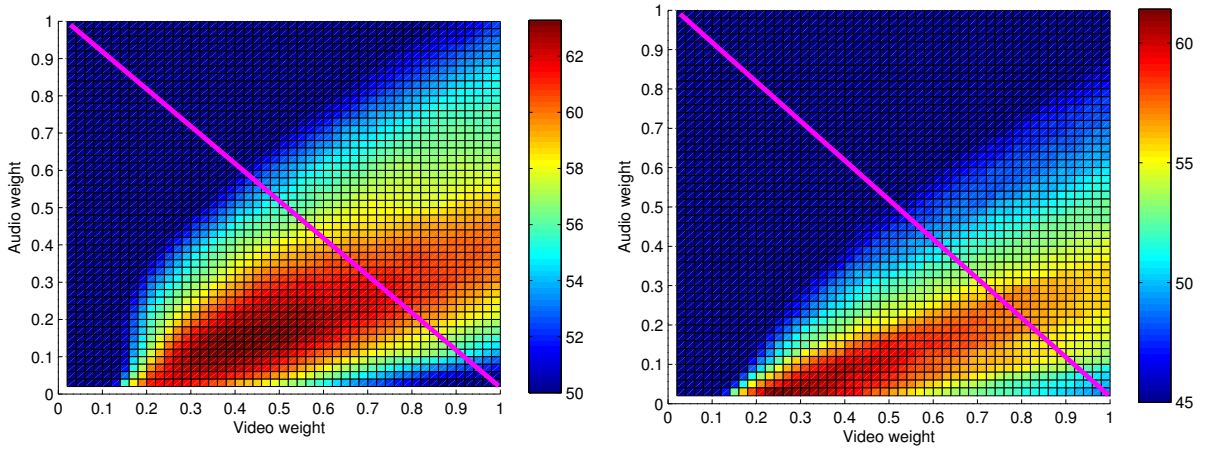

Figure 5.10 — AV accuracy, with unconstrained weights, for -5 dB and -10 dB.

by the emissions, with transitions having only the effect of allowing or not allowing certain paths in the model.

In our AVSR setup, the weights λ_a and λ_v can also be a factor in the balance between emission and transition probabilities, as can be seen from Eq. (5.12). Indeed, if the sum $\lambda_a + \lambda_v$ is allowed to change, this balance will also change. If, for example, the sum is reduced, the effect of the transition probabilities would be increased.

5.4.2 Results with an unconstrained sum

Figures 5.8, 5.9 and 5.10 show the audio-visual performance with audio and visual weights varying from 0.0 to 1.0 with a step of 0.02, without any constraint on their sum. The audio SNRs are as follows: clean, 20 dB, 10 dB, 0 dB, -5 dB and -10 dB, with babble noise. The purple line drawn on all the figures shows the weights for which the sum is equal to 1, $\lambda_a + \lambda_v = 1$, a constraint common to most approaches in the literature. As can easily be seen on all the figures, the optimal performance is always attained for sums which are significantly smaller, that is, the maximum is never on the purple line. In all figures, the lower value on the color bar is also the lower threshold applied on the accuracy values. This thresholding is applied for visualization purposes, as much lower accuracy values are also obtained for some combinations of weights, which lead to a widening of the range which has to be covered by the color map, making the peak indistinguishable in the absence of the threshold.

For clean audio, the value of the video weight at the point where maximum accuracy is very small, $\lambda_v = 0.02$, while the audio weight is $\lambda_a = 0.14$. However, the video still has a significant influence on the final result. Indeed, the AV accuracy is 99.45%, while audio-only performance is just 97.31%.

The same is seen at the other extreme, for an SNR of -10 dB. Here the audio weight is extremely small, $\lambda_a = 0.01$ compared to a video weight of $\lambda_v = 0.24$. As seen with clean audio, such a small weight can still have a big influence on the result. In this case, AV performance is 61.93%, while video-only is just 54.8%.

Figure 5.11 and Table 5.3 show the results obtained with fixed weights when using all combinations of weights, not only the ones summing to 1, at all SNRs with babble noise. The results for $sum \neq 1$ are practically the peak performance values from Figures 5.8, 5.9 and 5.10. It is clear from the figure that sums that are smaller than 1 lead to improved results at all SNRs. Although the gains are small, they are consistent across all noise levels. While for clean audio only 0.8% are gained, the number of errors is reduced by more then half. This is also true at the 25 dB SNR level. The relative reductions in error levels, shown in Table 5.3, are more modest for lower SNRs, but the trend is consistent at all noise levels. Allowing the sum to vary improves recognition results.

The justification for these accuracy improvements should be the factor mentioned above, the balance between transition probabilities and emission likelihoods. Indeed, for sums which are smaller than 1, the influence of the transitions is enlarged, as seen in Eq. (5.12).

The balance between audio and video is given in this case by the ratio between the

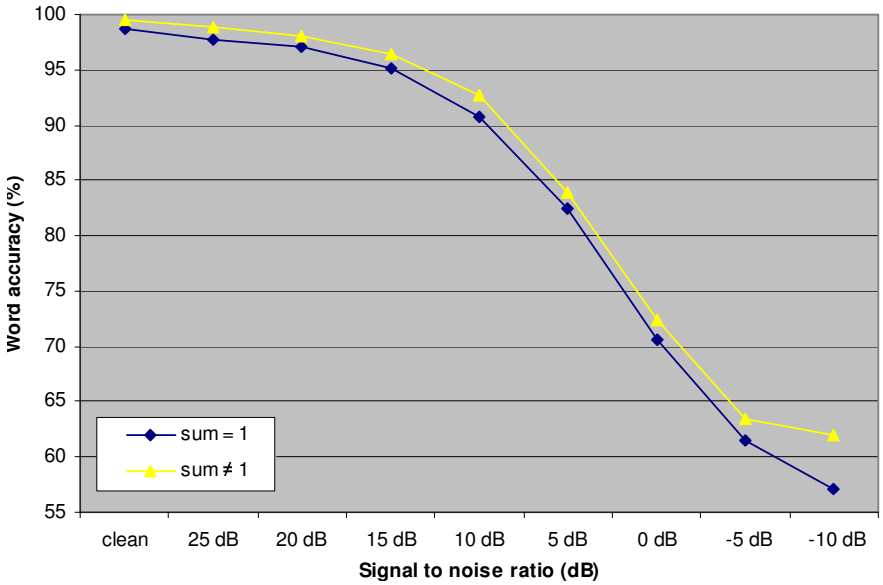

Figure 5.11 — The impact of removing the constraint on the sum of the weights on audio-visual results, with fixed weights.

	SNR clean	25 dB	20 dB	15 dB	10 dB	5 dB	0 dB	-5 dB	-10 dB
sum=1	98.66	97.77	97.11	95.15	90.74	82.41	70.63	61.53	57.18
sum≠1	99.45	98.89	98.05	96.37	92.64	83.87	72.37	63.42	61.93
gain	+0.79	+1.12	+0.95	+1.22	+1.9	+1.45	+1.74	+1.9	+4.76
% error reduction	58.88	50.26	32.83	25.13	20.52	8.24	5.92	4.94	11.12

Table 5.3 — Results for audio-visual recognition, with fixed weights and an unconstrained sum.

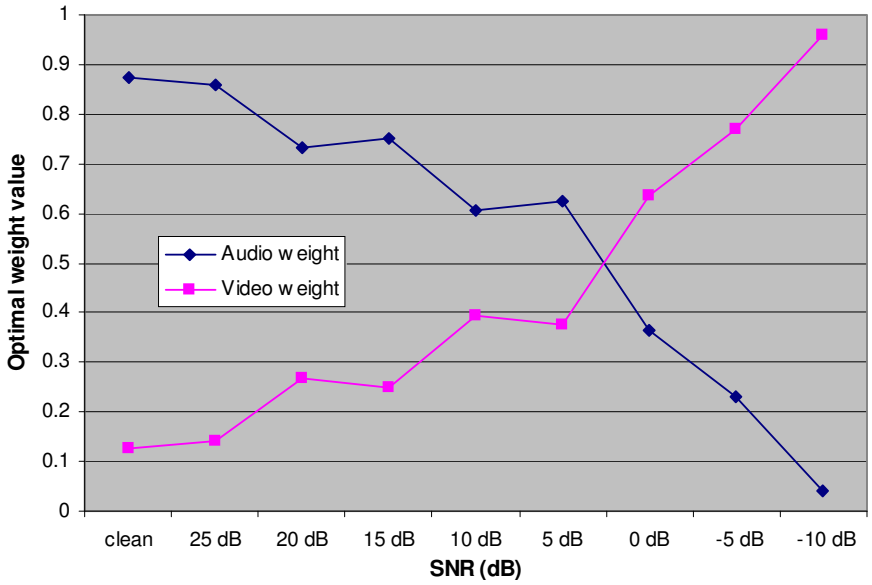

Figure 5.12 — The values of audio and video weights, normalized by the value of their sum, for all SNRs.

audio and the video weights, not their absolute values. Indeed, if we normalize the weights by dividing them to their sum value, we obtain an image which is very similar to that obtained with constrained sum. Figure 5.12 shows the values of the audio and the video weight, divided by their sum. The audio normalized weight decreases with the SNR, from 0.88 to 0.04, while the video normalized weight increases from 0.12 to 0.96.

5.4.3 Adapting the weight sum dynamically

Since a smaller weight sum is beneficial to the recognition results, it might be a good idea to allow the sum itself to vary dynamically. This would lead to a system in which the ratio of the weights is estimated from the individual reliability of each stream, while the sum is estimated globally. But how would a dynamical sum be able to influence recognition results?

In a situation where the noise is variable, there might be instances where both modalities are corrupted simultaneously. In such cases, both emission likelihood distributions may be unreliable, and the only source of reliable information left would be the transition probabilities. In such a case, it may be convenient to reduce the weight sum in such a way that decoding continues based mostly on the transitions, that is, the states in the maximum-likelihood path through the HMM are chosen on the basis of the most likely transitions. This may be better than allowing likelihoods from corrupted modalities to influence the result.

Even when the noise is constant, in instances when the two modalities are contradicting each other, it might be better to ignore the emission likelihoods for a few frames and just continue the decoding based on the transitions. A contradiction between audio and video would be for example if one modality has a peak in the posterior distribution for one phoneme, while the peak in the other modality corresponds to a different phoneme.

To estimate when a reduction in the value of the weight sum may be necessary, we need to detect the instances when both streams are unreliable, or when they are contradicting. For this purpose, we use the entropy of class posteriors again, but on the combined likelihoods. We compute the combined class-posteriors with a formula similar to Eq. (5.7), that is:

$$P(s|o_{AV}) = \frac{p(o_a|s)^{\lambda_a} p(o_v|s)^{\lambda_v} P(s)}{\sum_{s'} p(o_a|s')^{\lambda_a} p(o_v|s')^{\lambda_v} P(s')} \tag{5.13}$$

where the weights λ_a and λ_v are computed as before, based on the stream posterior entropies, and with $\lambda_a + \lambda_v = 1$. The entropy of this combined posterior distribution is then computed:

$$h_{av}^t = -\sum_{i=1}^{S} P(s_i|o_{av}^t) \log P(s_i|o_{av}^t) \tag{5.14}$$

This entropy will be low in the case when the modalities are in agreement, that is, if there are definite peaks in both the audio and the video posterior distributions, and

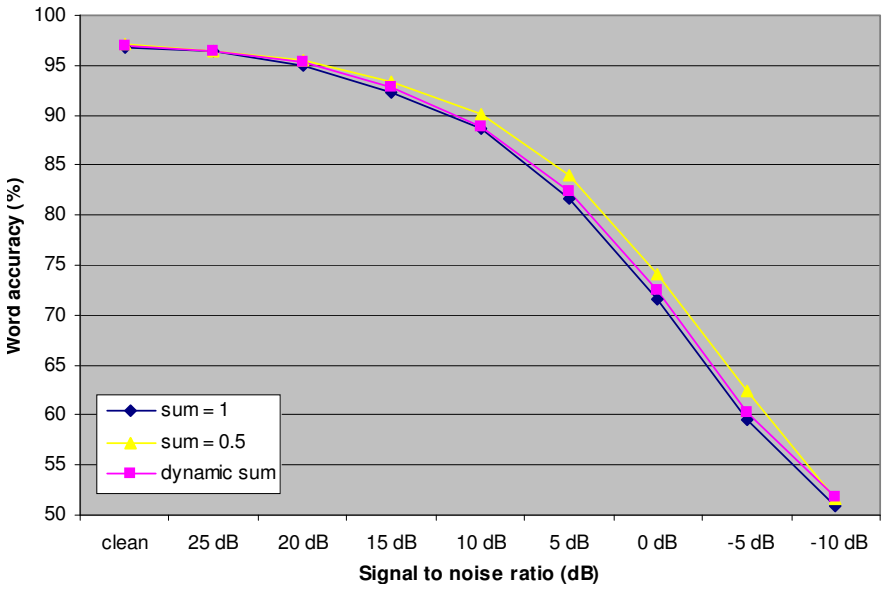

Figure 5.13 — The impact of removing the constraint on the sum of the weights on audio-visual results, with dynamic weights, compared to the sum fixed to 1 and 0.5.

	SNR								
	clean	25 dB	20 dB	15 dB	10 dB	5 dB	0 dB	-5 dB	-10 dB
sum=1	96.82	96.32	95.04	92.35	88.62	81.64	71.62	59.55	50.81
sum=0.5	97.10	96.49	95.59	93.41	90.07	83.92	74.03	62.36	51.65
dynamic sum	96.93	96.32	95.26	92.74	88.90	82.31	72.46	60.18	51.76

Table 5.4 — The impact of removing the constraint on the sum of the weights on audio-visual results, with dynamic weights, compared to the sum fixed to 1 and 0.5.

the peaks coincide. When the peaks do not coincide however, the combined probability distribution $P(s|o_{AV})$ will be flatter than the two mono-modal distributions, leading to a higher entropy. The entropy will also be high when both the audio and the video posterior distributions are flat themselves, because this also leads to a flat combined distribution.

However, in the case when one posterior distribution for one modality is flat, while the other has a definite peak, the weights λ_a and λ_v will be heavily biased toward the reliable modality, making the combined posterior look very similar to its posterior distribution. This means that when one modality is reliable but the other is not, the combined entropy will also be low.

This all shows that the combined entropy should be a good measure for the purpose mentioned above, detecting when both modalities are unreliable or when they are contradicting. The inverse of this entropy can be mapped to a an adjustment factor which will be applied afterwards on both weights, effectively reducing their sum.

Figure 5.13 and Table 5.4 show results with dynamic weights, for three cases. The first two are with with the sum of the weights constrained to be either 1 or 0.5. The third case is our algorithm of applying an adjustment factor to the sum, based on the entropy of the combined audio-visual posterior probability distribution.

As can be seen, results with the dynamic algorithm are rather disappointing. The performance improves only slightly, but the improvement is present across all SNRs. The fact that there is a consistency in the results shows at least that this improvement is not random. However, a larger improvement and just as consistent across the SNRs can be obtained by simply halving the weights after their estimation from the respective mono-modal entropies. This reduces the weight sum to 0.5 and also has significant influence on the final results.

In our best knowledge, the variation of the sum of stream weights was not attempted before in the literature, neither with fixed weights nor with dynamic ones.

In conclusion, the constraint on the sum of the stream weights is unnecessary, and removing it can lead to significant performance gains across all the SNRs. However, a reliable method of estimating the level at which this sum should be reduced still needs to be found.

5.5 Summary

This chapter presents our multimodal integration method, based on the estimated reliability of each stream. The problem of multimodal integration is solved by merging frame-level probabilities at each time instant, with weights reflecting each stream's importance.

Our stream reliability estimates are based on the entropies of the class-posterior distributions for each stream and for each frame. Since the weights need to be the inverse of these entropies and they also need to be scaled, several mappings from entropies to weights are investigated, both static and dynamic. Experimental results show that dynamic weights perform well in a variety of conditions, leading to improved accuracy compared to audio-only results across a wide range of SNRs, with both white noise and babble noise.

We also investigate the role played by the constraint typically imposed in other work from the literature on the sum of stream weights in multi-stream HMMs. Our findings are that reducing the sum of the weights can lead to significant improvement in the word-level accuracy of the recognizer, across all SNR levels.

The framework presented here is general, since the reliability measure that we use is not particular to either audio or video. The entropy of the posterior distribution can be used in any multimodal context to evaluate the reliability of each modality dynamically, allowing the adjustment of each stream's importance to temporal variations of its quality. The loss of one of the streams is also naturally handled in this framework, as the weights would automatically adjust to ignore the missing stream, reverting to a mono-modal situation.

In the next chapter we depart from the speech recognition framework to analyze another application of multimodal feature selection and integration, speaker localization. We apply concepts from both Chapter 4 and this chapter to build a simple multimodal algorithm which is able to find a speaker's mouth in the video based on the correlation between the movement in the video and the audio energy.

Multimodal speaker localization

6

6.1 Introduction

In this chapter, we present a different application to our visual feature selection method based on optical flow, speaker localization. Speaker localization consists in finding the active speaker in a video sequence containing several persons. This can be useful especially in videoconference applications, and automatic annotation of video sequences. Consider for example a case where the videoconferencing software only sends the image of the person that's actually speaking, not of everyone in the view of the camera. In a smart meeting room, a system controlling several cameras could switch views or change the focus automatically depending on who was speaking. Another example may be a software that creates an automatic transcript of a meeting, using speaker localization in conjunction with speech recognition.

Typically, speaker localization can either be done in the audio modality using a microphone array, or in the visual modality by detecting movement. Our approach uses the correlation between audio and video to find the mouth of the speaker. This multimodal approach should be able to make the difference between the active speaker and a person murmuring something inaudible, something that is impossible with a video-only approach. At the same time, the multimodal method should be less affected by noise in the audio, compared to the microphone-array approach.

As a multimodal application, speaker localization is simpler than speech recognition, in the respect that only correlations are sought between the modalities, without requiring a deeper understanding of the information in the two streams. However, understanding these simple correlations may help other domains that also analyze the voice and mouth movement together.

Our method of speaker localization has the advantage of being simple to apply and also

being quite general. We train a joint probability model of the movement of the mouth and the audio energy, and then, at test time, we search through all the image locations where the optical flow correlates to the audio in the same way that was learned during training. Compared to other approaches which do not make use of any training, our method has the advantage of requiring less computation at test time, since there is a trained model of the type of correlation which is sought.

We compare our method to a similar approach from the literature, applied on the same database, and show that our method has a much higher performance.

The content of this chapter is based on work that we have published in [38].

6.2 Prior work on speaker localization

We will focus here on the problem of finding the active speaker in a video sequence, using a single audio channel together with the video. We will give a brief overview of previous approaches from the literature that attempt to solve this problem, concentrating on multimodal methods and leaving aside solutions based on microphone arrays, since this is not the focus of our work. Our intention is to develop a method which can function without requiring hardware more specialized than a camera and a single microphone.

Previous approaches to audio-visual speaker localization either assume the gaussianity of the data, or rely on complex and computing-intensive operations at test time to detect correlation between the audio and the video.

Hershey and Movellan [48] use an estimate of the mutual information between the average acoustic energy and the pixel value, which they assume to be jointly gaussian. Slaney and Covell [107] use Canonical Correlation Analysis to find a linear mapping which maximizes the audio-visual correlation on training data. Applying the same mapping on test data and measuring the audio-visual correlation in the transformed space gives a quantitative measure of audio-visual synchrony. This approach too implicitly makes the assumption that audio and visual information are jointly gaussian.

Audio-visual synchrony is also analyzed by Nock et al [73, 74]. The mutual information between the audio and the video is estimated using two methods, one using histograms to estimate the probability densities, the other using multivariate gaussians. For the second measure, the data is assumed to be jointly gaussian only locally, on short windows.

Fisher et al. [27] use a nonparametric statistical approach to learn maximally informative joint subspaces for multimodal signals. This method requires neither a prior model nor training data. In [26], the method is further developed, showing how the audio-visual association problem formulated as a hypothesis test can be related to mutual information based methods.

Monaci et al. [68] decompose the image sequence in video atoms and then search for correlations between the audio and the movement of these atoms, in order to find the speaker. The framework is extended [69] such that a multimodal dictionary of audio-visual atoms can be learned directly.

Butz and Thiran [15, 16] propose an information theoretic framework for the analysis of

multimodal signals. They extract an optimized audio feature as the maximum entropy linear combination of power spectrum coefficients. They show that the image region where the intensity change has the highest mutual information with the audio feature is the speaker's mouth. Besson et al. [11] use the same framework to detect the active speaker among several candidates. The measure that they maximize is the efficiency coefficient, i.e. the ratio between the audio-visual mutual information and the joint entropy. They use optical flow components as visual features, extracting them from candidate regions identified using a face tracker.

The disadvantage of methods that attempt to maximize an information theoretic measure *at test time* is that they need to use some time-consuming optimization procedure, such as gradient descent or a genetic algorithm. This means that, although these methods do not require a training procedure, the amount of computation that is needed during testing is important, making a real-time implementation unfeasible.

By contrast, our multimodal approach does use a training procedure. The joint pdf of the audio energy and a visual feature based on optical flow is learned. This ensures that the number of operations performed while testing is reduced, and thus a real-time implementation would be possible.

Another advantage of our approach is that, in contrast to methods that consider the audio and video of speech to have a gaussian joint pdf, we can model any kind of probability density. The gaussian mixture model that we use is an universal approximator of densities, even when using only diagonal covariance matrices, provided that enough gaussians are considered.

Moreover, in our case, no face tracker needs to be used, as testing is done on the entire image, not only the face or mouth region. An extracted mouth region is required, but only in the training step, when the joint pdf is estimated.

Finally, although the optical flow has been used before for speaker localization, our visual feature, which is the difference between vertical components of the optical flow, is novel. This feature is the same that was presented in Chapter 4. We argue that it is better at representing the movement of the mouth, and, at the same time, more tolerant to the motion of the head, compared to simple optical flow, pixels or pixel differences (deltas).

6.3 Our speaker localization method

6.3.1 Feature extraction

As we want to model the dependency between the audio and the video signals in the case of speech, we need to extract temporally synchronized features from both streams. The audio feature that we use is the logarithm of the energy (log-energy) of the audio signal. From the video, in the training phase, we only use the rectangular region of the mouth. We extract visual features as follows. We compute the optical flow from the luminance component of the images. A single vertical column of points is selected at the center of the mouth region, and only the vertical components of the motion field are retained, as shown in figure 6.1.

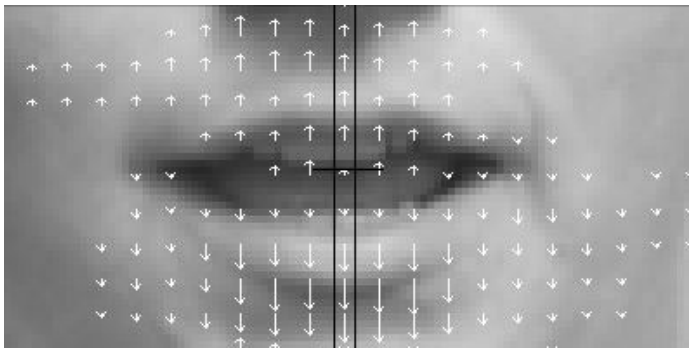

Figure 6.1 — A frame from the training sequences, with the corresponding optical flow.

Our visual feature is the difference between the average optical flow on the top and bottom halves of this column.

This visual feature is similar to the motion features used in Chapter 4. The optical flow algorithm used here is Horn-Schunck [49]. As was mentioned in Chapter 4, computing the difference of optical flow vectors has the effect of neutralizing the movement of the head and only shows the opening and the closing of the mouth. The feature is tolerant to both vertical and horizontal displacement, meaning that the extraction of the mouth region, required for training, does not need to be very accurate.

6.3.2 The probability distribution

In order to estimate the joint pdf of features extracted from the training sequences, we need an appropriate model. If $F_v^{train}(t)$ is the visual feature for the training frame t, and $F_a^{train}(t)$ the corresponding audio feature, we want to estimate the probability density function $p(F_a^{train}, F_v^{train})$. Assuming that $p(F_a, F_v)$ is gaussian is too restrictive. Instead, we use a gaussian mixture model (GMM), trained with an Expectation-Maximization (EM) procedure [13]. As mentioned before, the GMM can be used to represent any type of pdf, provided that enough components are included. Our trained model consists of four gaussians with diagonal covariance matrices, which proved to be a good representation for our data without overfitting it.

The distribution of the audio-visual samples taken from the training sequence, as shown in figure 6.2, has a high concentration of points around zero audio energy. This is caused by pauses between words. As can be seen, the estimated pdf has a high peak in the same area, while the distribution of the remaining points is poorly modelled.

When searching the correspondence between the sound and the movement of the mouth, the silent samples (low audio energy) do not convey any useful information. Therefore, we removed these samples through thresholding.

In general, image points with low relative movement (low value of the video feature) are characteristic for a static background, even when associated with a high audio energy. However, such points are also present in the training set consisting of mouth regions only. They appear either as a result of errors in the optical flow, or during the pronunciation of

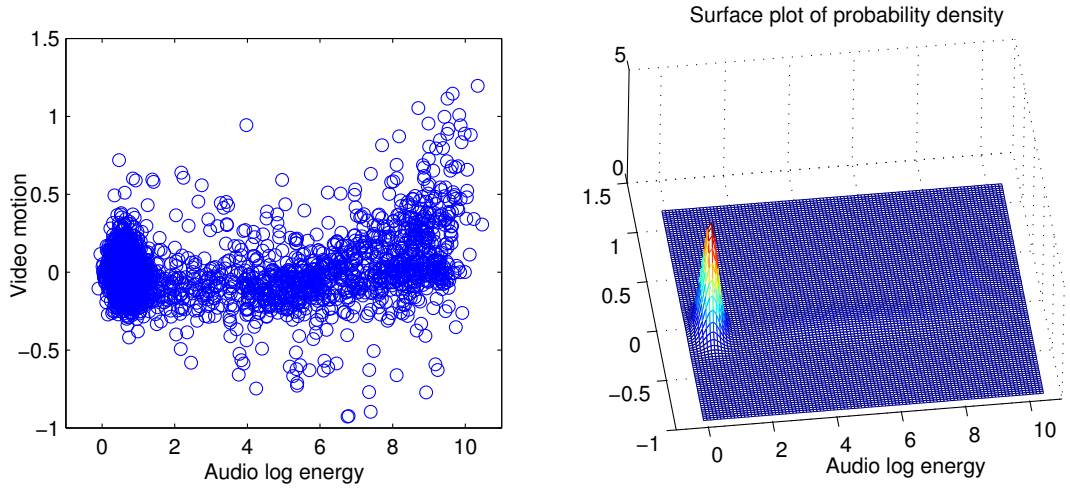

Figure 6.2 — The distribution of audio-visual samples and their estimated pdf.

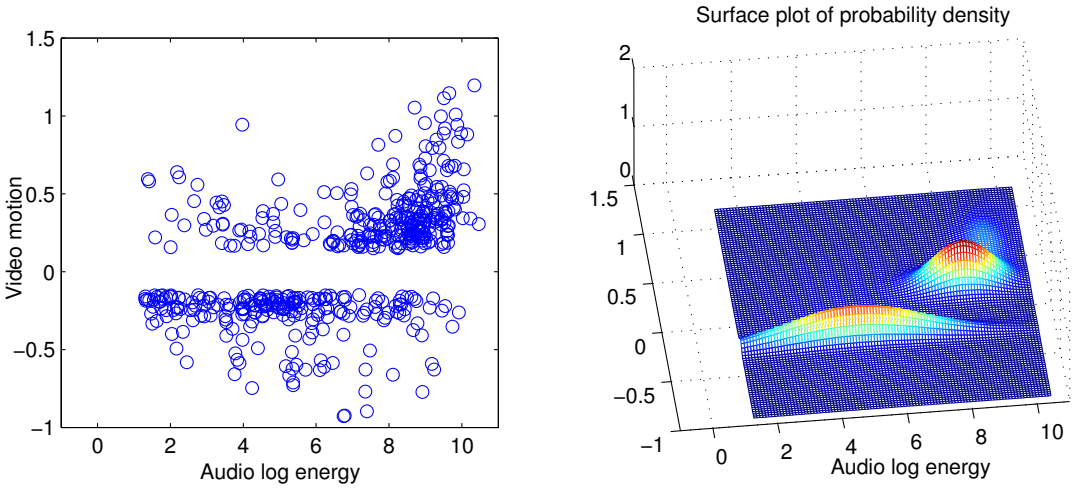

Figure 6.3 — The distribution of audio-visual samples and their estimated pdf, after removing the parts where there is either silence, or very little motion.

long vowels, when the mouth does not move much. As these samples can not help determine the location of the speaker, we removed them as well.

Figure 6.3 shows the distribution of the remaining samples. Their pdf has an interesting property, that is, high audio energy is more often associated to positive values of the visual feature, while lower audio energy is associated to negative values. Since our visual feature is the difference of vertical optical flow vectors, a positive value in the training samples represents the action of opening the mouth, while a negative one represents closing it. This confirms the intuition that opening the mouth should lead to louder sounds than closing it.

We can infer from the discrimination, based on the audio, between positive and negative values of the visual feature, that the audio-visual approach can offer more information than video only. This clearly shows the advantages of multimodal analysis. Our method does more than just detecting regions of high relative motion. By associating this motion

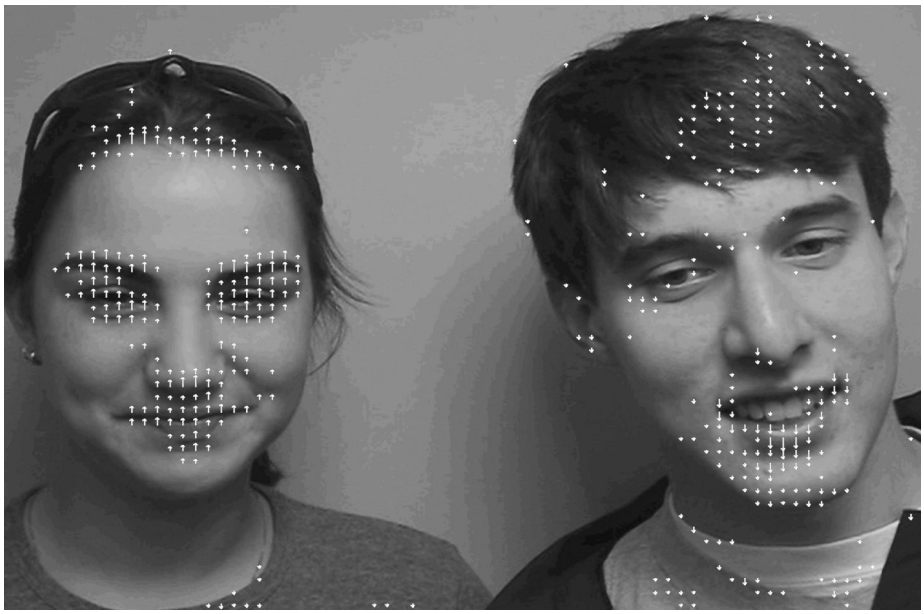

Figure 6.4 — A frame from the test sequence, with the corresponding optical flow

with a corresponding audio value, our algorithm can find the combination that most likely represents a speaking mouth.

However, the speed of the mouth's movement, as measured with the optical flow, can vary depending on the distance from the speaker to the camera. We normalized the values of the visual feature by scaling them with a factor proportional to the distance between the speaker's eyes. This scaling factor was computed once for each speaker, as the distance to the camera remains constant in our sequences.

As the audio energy level is used as a feature, and included in a joint probability model, we also make the implicit assumption that the different speakers' voices will be more or less equally loud, which holds for our database.

6.3.3 Finding the active speaker

Our method of speaker localization is based on a maximum likelihood approach. We find the region of a test image where samples have the highest likelihood to have originated from our learned pdf. Our tests show that this region corresponds quite accurately to the active speaker's mouth.

The testing sequences consist of two speakers side by side, taking turns at speaking. They pronounce series of connected digits. Since we do not model the words themselves, it is not a requirement for testing to have the same vocabulary as the training set, but generally the same set of phonemes.

Our testing procedure is as follows. We compute the optical flow from the luminance of the frames. One such frame with the corresponding optical flow is shown in figure 6.4. Only vectors larger than 10% of the maximum motion vector in the image are represented. We used the LTI-Lib computer vision library (`http://ltilib.sourceforge.net`) to compute

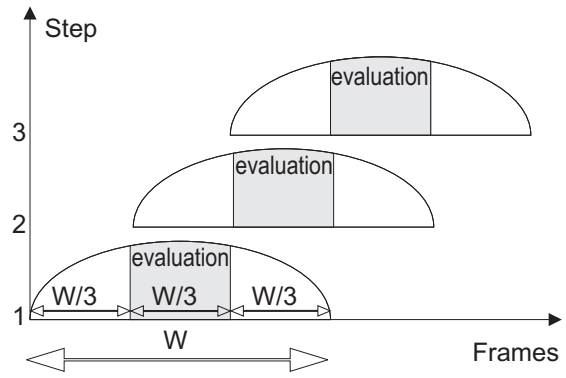

Figure 6.5 — Overlapping temporal windows used for testing and the corresponding evaluation intervals. In our case, the length of the window is W = 2 seconds.

the optical flow and display it.

From the optical flow, only the vertical components are retained. We compute the value of the visual feature in all points on a grid (with a 10-pixel spacing), using the same method as in training. After selecting columns having the same height as the mouth regions from training, we compute the difference of average vertical optical flow between their top and bottom halves. The reason for using a grid is that the value of the visual feature does not differ much between neighboring points, and we considered the 10-pixel accuracy as sufficient for speaker localization.

For each video frame, the corresponding audio energy, together with the visual feature values on the points of the grid, are used to compute log-likelihoods from the learned joint pdf. If $F_a^{test}(t)$ is the audio feature for the test frame t, and $F_v^{test}(t,x,y)$ is the visual feature value at coordinates (x,y) in the same frame, then the obtained log-likelihood is:

$$l(t,x,y) = \log\left[p\left(F_a^{test}(t), F_v^{test}(t,x,y)\right)\right] \tag{6.1}$$

where p is the pdf obtained from training.

We sum the log-likelihoods resulting from several consecutive frames at each image coordinate on the grid. We use temporal windows of length W (2 seconds), with a $2W/3$ overlap, as shown in figure 6.5. The result of the summation is a 2D map, representing the likelihood that the active speaker's mouth is located at a certain coordinate in the image, during the time interval W. The algorithm outputs the location of the detected active speaker as the (x,y) coordinates of the likelihood maximum:

$$L(x,y) = \sum_{t \in W} l(t,x,y) \tag{6.2}$$

$$(x_{speaker}, y_{speaker}) = \arg\max_{x,y}\left[L(x,y)\right] \tag{6.3}$$

Figure 6.6 shows the isocontours of such likelihood maps $L(x,y)$, superimposed on frames from the corresponding temporal windows. For the first image, the maximum likelihood point is emphasized by a cross, and, in this case, lies on the speaker's mouth, as

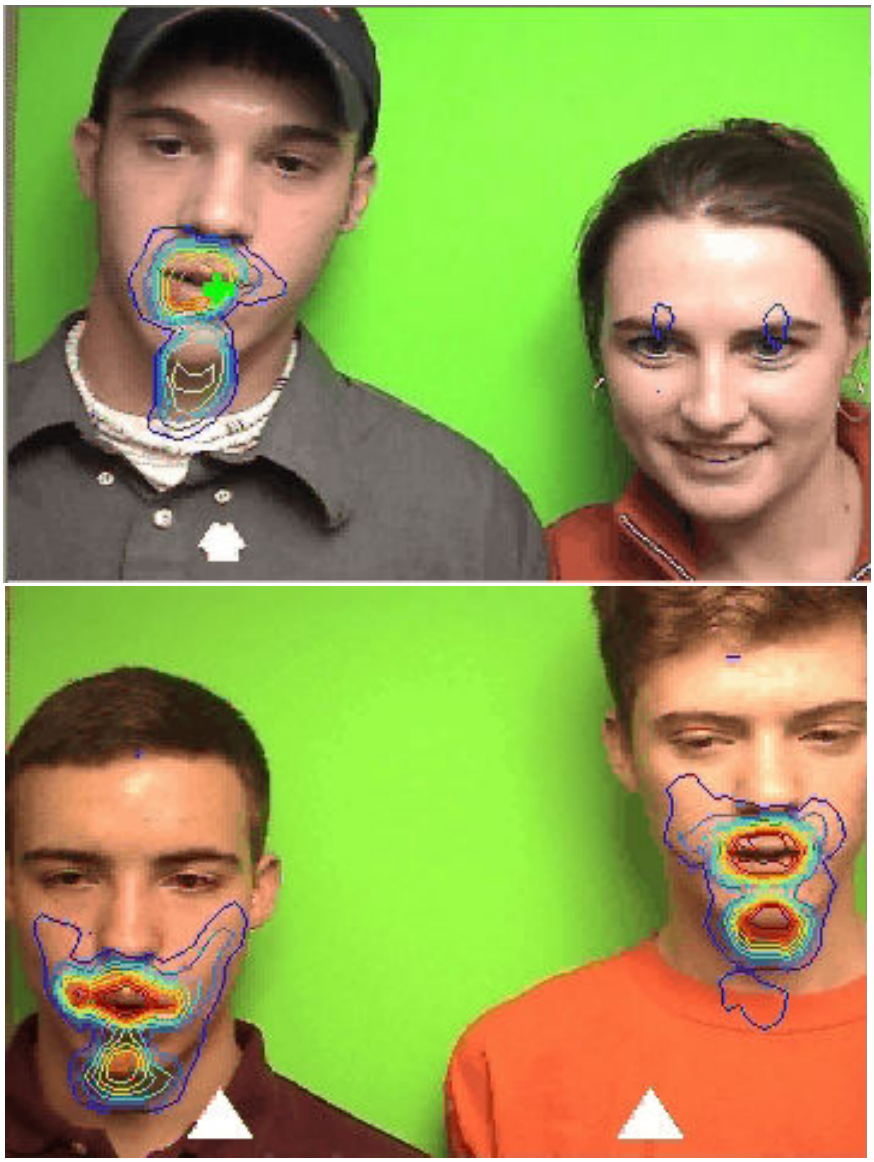

Figure 6.6 — Isocontours of likelihood maps, superimposed on frames from the corresponding temporal windows.

expected. In the second image, both the left and the right person are simultaneously speaking, and, as can be seen, the two biggest local maxima of the likelihood function are on the speakers' mouths.

6.4 Speaker localization results

For our experiments, we use sequences from the CUAVE audio-visual database [75] described in Chapter 4. The sequences are deinterlaced as previously detailed, filtered to remove noise and downsampled to half their resolution to speed up the processing.

Although the sampling rate of the audio is higher than the video frame rate, we need

Seq. no.	Including silence detection (%)	Only speaker localization (%)		Nock et al. [74] (%)
1	90	97		-
2	84	89		-
3	80	86		-
4	82	97		-
5	88	88		-
6	93	93		-
7	82	89		-
8	77	85		-
9	89	92		-
10	76	84		-
11	96	96		63
12	83	90		64
13	89	93		50
14	89	100		91
15	90	97		75
16	97	97		85
17	93	93		94
18	79	83		64
19	88	88		47
20	88	95		93
21	91	94		83
22	100	100		95
Avg.	87.4	92.1		75.3

Table 6.1 — Speaker localization accuracy on the "groups" sequences of the CUAVE database, both with and without silence detection. Results from Nock et al. [74] are also included.

synchronized features. To this end, we compute the audio energy on short temporal windows, so as to obtain one audio feature value for every video frame.

The training sequence that we use belongs to the "individuals" part in the CUAVE database. The speaker utters the English digits from "zero" to "nine" separately, for five times, with pauses between the repetitions. Testing sequences are from the "groups" section of the database. They consist of two speakers taking turns, and finally speaking simultaneously for a short time at the end. We ignored this final part of each sequence in testing.

For quantitative results, we use the frame-level ground truth established by Besson et al. [12] for the "groups" sequences. They assign to each frame one of these three labels: *silence*, *left speaker* or *right speaker*. This is the ground truth used to obtain one set of results, for which we also detect silence in the audio.

A second type of ground truth is derived with the purpose of showing the performance of the speaker localization algorithm itself, without the silence detection. To obtain this

different ground truth, we split every silent period marked in the old one, labelling each half with the nearest speaker label. With this second ground truth, we obtain a second set of results.

Although our method can quite accurately detect the position of the mouth, we only distinguish between the left and right speaker in our quantitative test. We base our choice on the horizontal position of the likelihood maximum. If it lies in the left half of the image, then we consider that the left speaker is active, and vice-versa.

We compare the detected speaker with the frame-level ground truth. This evaluation is done on the central part of the temporal window, as shown in figure 6.5. At the same time, the audio energy in the evaluation window is compared to the silence threshold used in training. If the majority of samples in the window are silent, we label it as silent. Otherwise, the label given is the detected active speaker. The detected label is compared to the one that forms the majority in the ground truth, within the temporal window. The first set of results presented in table 6.1 is obtained with this method. For the second set of results, the silence detection step is skipped, and the second type of ground truth is used.

The results obtained by Nock et al. [74] for multimodal speaker localization, using a gaussian mutual information measure on the same sequences, are also included in table 6.1. They make no attempt to detect silence, so their results should be compared to our second set, based on the two-label ground truth. The comparison is fair, since they also only distinguish between the left and right halves of the image for the results included in the table. The average performance they obtain is 76%. This result is lower than the 81% reported in the same paper using a visual-only method, so the multimodality did not improve performance in their case.

By contrast, our multimodal method did increase performance. Taking only the last 11 sequences into account, we obtain an average accuracy of 93.7%. This is better than the 81% reported [74] for visual-only speaker localization, confirming that we are able to profit from the extra information present in the audio.

6.5 Discussion

The good results that we obtained on the CUAVE database could be justified by the small amount of background movement present in the sequences. On sequences with more movement in the background our method may perform worse, depending on the movement's amplitude and direction. To influence the results, there should be a motion difference oriented vertically and correlated in sign and amplitude with the audio energy, according to the joint pdf. However other methods of speaker localization, such as the ones using pixel differences, would be influenced by any type of movement, be it horizontal or vertical, and of any amplitude.

It is possible for simple movements, like a hand moving upward against a static background to lead to positive or negative motion differences, because, at the edge of the hand, the motion field would contain vectors pointing upwards next to points with zero motion. Their difference can be positive or negative depending on their placement. However, this is

where the temporal averaging comes into effect. Those differences will appear on different points on our grid as the hand continues its movement, and thus a single point from the grid will only catch movement for a short period of time, not enough for a false positive.

An improvement of both the visual and the audio features would make detection more reliable. The Horn-Schunck optical flow extraction algorithm [49] has difficulties correctly extracting the motion field of the mouth. While the previous motion field is used as an initialization at each step, only two frames are analyzed. Methods that take into account larger time intervals, that is, more frames, might lead to more precise motion estimation. For the audio, using features more related to speech, such as mel-cepstrum coefficients, would make our method invariant to differences in the loudness of the speakers' voices.

6.6 Summary

We presented a method for finding the active speaker in a video sequence, based on both the audio and the video. We look for correlations between relative vertical movement in the image and the audio energies which are similar to those learned in training.

Our method leads to improved results compared to other approaches from the literature. Moreover, we are able to exploit the multimodality of speech, obtaining a better performance than that reported for a visual-only method. The training procedure makes the number of operations required for testing small, leading to a fast implementation.

Our novel visual feature is well-suited to represent the movement of the mouth, and is tolerant to some degree to both horizontal and vertical movement of the head. As testing is done on the entire image, there is no need for a face tracker.

To conclude, we showed that it is possible to find dependencies between modalities even with very simple one-dimensional features, by training non-gaussian joint probability density functions. This general framework gives good results on speaker localization, allowing us to locate the mouth of the speaker in a video sequence without any priors other than the trained model.

7 Conclusions and future directions

7.1 Discussion and conclusions

The goals of this thesis were the analysis of feature extraction methods from multimodal signals and the development of multimodal integration methods that would work in a wide range of conditions.

Throughout the thesis, we aimed to validate our work by performing many tests in varied conditions, and always comparing with alternative methods from the literature. We use leave-one-out validation to minimize errors caused by particularities of the data. We ran tests with many different feature dimensionality values and feature types, for both video-only and audio-visual recognition, in order to be able to offer a comprehensive picture of the problem.

In Chapter 4, we present several feature selection methods applied on visual features typically used in AVSR. We investigate methods of selection based on mutual information, an information theoretic measure used both to evaluate the relevance of individual features, but also the redundancy between them. We always compare our methods with state of the art alternatives, as well as with results obtained for mono-modal recognition. We show, as was to be expected, that multimodal processing outperforms mono-modal methods in all situations, across all noise levels and for different types of noise. As for feature selection, all information-theoretic methods outperform the LDA, which is the method of choice for dimensionality reduction for visual features in the literature. Between the information theoretic methods, we show that methods that penalize the redundancy between features perform much better that features that do not, an aspect that had previously not been investigated in the literature.

The use of redundancy-penalizing methods for visual feature selection was our main contribution in Chapter 4. We showed how this improved recognition results across a wide

range of SNRs and with two types of noise.

In the same chapter, we investigate an alternative path to obtain visual features, one based on the prior assumption that most the relevant speech information in the visual domain is contained in the movement of the lips, and more particularly, the opening and closing of the mouth. We aimed to obtain a very low-dimensional feature vector extracted with simple operations, but which would be however very robust to several typical problems that can appear in the visual domain, like errors in mouth tracking, partial occlusions and changes of angle. Indeed, the motion features that we use, the differences of maximum values of optical-flow vectors on the region of the mouth is robust to some degree of head movement and head tilt, as well as partial occlusions. Our simple visual features perform on the same level as the LDA, while also being more robust.

Our contribution in this case was the introduction of optical flow differences as a low-dimensionality visual feature for audio-visual speech recognition.

In Chapter 5 we develop a method of multimodal integration which can dynamically adjust to changes in the quality of the streams. Our method uses the entropy of the class posteriors per stream as an estimator of reliability, weighting the streams according to it. A decision is taken at each time instant on the values of these weights, allowing the system to adapt very quickly to changes in the quality of the streams. We investigate four different functions for mapping the entropies to stream weights.

Our method performs well in a variety of conditions, at a large range of SNRs and with different types of noise. We compare it to a similar method from the literature, one based on maximum class posteriors, and find that our method consistently outperforms it. Furthermore, we investigate the role of a constraint typically imposed on the sum of the weights, that is, that the sum should be 1, and show that, by removing it, performance can be further improved. We show that the justification lies in the fact that changing the sum of the weights changes the balance between the ranges of the emission likelihoods and those of the transition probabilities.

Our contribution is developing a dynamic weighting system which can quickly adapt to any changes in the quality of the streams, including even the temporary loss of one stream, case in which the system reverts to monomodal processing. The analysis of the role of the weight sum is also novel, since in virtually all previous work the sum was imposed to be constant.

Finally, in Chapter 6 we apply concepts from both feature selection and multimodal integration to the problem of speaker localization. We develop a method that can find the mouth of the speaker in a video sequence by finding correlations between the motion in the image and the audio, based on a joint probability model. Comparing our results with a similar method from the literature we find that we were able to achieve a significant improvement.

Our contribution consists here in deriving a novel method for speaker localization, one that has small computational requirements at test time compared to other methods in the literature, while also being a very good performer.

With this thesis, we offered a comprehensive look at the problems of feature selection for

multimodal signal processing and multimodal integration. These are important problems and very active research fields.

Feature selection is important because using only relevant information for any classification problem will always improve results. Having less features means that more accurate models can be built with the same amount of initial data, in shorter time and with less of a computational burden at test time. Knowing which features are relevant for a classification task can even, in some instances, offer us insight on the physical phenomenon that is studied.

Our work shows that, for audio-visual speech recognition, mutual information is not only a good measure for the relevance of features, but also a good way of estimating the redundancy between them. We proved that penalizing features for their redundancy leads to better sets of features, outperforming even the LDA.

Knowing which features are relevant and which are not, which features contain the same information and which complement each other can be very useful not only for building better feature sets, but also for our own understanding of the problem. We could, for example, globally evaluate the quality of feature transforms, and choose the transform which gives us features having the most information. The information theoretic analysis might also show that features obtained with two different transforms are complementary, and using more than one feature type might increase performance. It is common for example in AVSR to use both appearance-based features and shape features together, as they complement each other well. MI is a tool that could be used to identify similar situations where more than one feature type may be useful.

Multimodal integration is an even more important problem. Indeed, it would be very useful in many domains to be able to use information from several sources, merging them seamlessly to profit from their complementarity while also having the possibility to replace information that is missing from one stream with information from another. Our work shows just that, that it is possible to combine the information from two streams seamlessly, augmenting the performance when both streams are reliable, while still obtaining the performance of a single-stream system when one of the streams is corrupted.

7.2 Future research directions

As with any other research, there is still a lot of room for improvement. We will present here a few ideas that could be the subject of further research.

For feature selection, starting with a bigger initial set of features may offer the selection algorithm more choice when picking features, potentially allowing smaller sets of features having even more complementary information. This could be done for example by including other types of features in the initial set, not only features based on the DCT. Indeed, it might be interesting to see the performance of contour-based features, the height and width of the mouth, or even the motion-based features that we presented. Our algorithm based on mutual information should be able to show which of these types of features contain more information and are thus better suited for the recognition task.

Another possible area of research could be in the field of feature transforms. Indeed, it might be impossible to obtain an optimal set of features only by picking from DCT features together with their temporal derivatives. A transform which also uses relevant information might lead to better sets of features. LDA is such a transform, however it is based on very strong assumptions and our work showed that its performance is not on the same level as the MI-based algorithms.

Finally, the time intervals considered in the visual domain may be too short to capture the entire dynamic of the speech sound formation. As was proposed for audio-only speech, using longer time intervals to extract visual features may improve the quality of AVSR.

On the integration side it may be possible to find even better estimates for the quality of the streams. It is true that the entropy out-performs other measures such as the maximum posterior, but computing the entropy only at the frame level may mean missing some important information in the time evolution of the stream. A better reliability measure might take into account this evolution, looking not only at the state of the stream at one particular time instant, but over a larger time window.

Our results with the varying weight sum were promising, however we were unable to find a suitable criterion to estimate when the sum should be decreased, and by how much. Integrating a dynamic adjustment of the sum itself in the system should lead to a significant increase in performance, if such a method can be developed.

In all cases, our methods are not limited only to audio and video, and definitely not only to speech. The methods that we present, for both feature selection and for multimodal integration, are general and can be applied to any problems where information from multiple sources needs to be analyzed and fused.

7.3 Possible practical applications

At any thesis presentation, one question is inevitable: "What would be the practical applications of this work?" We will try here to give some examples of practical applications that could be helped by algorithms in this thesis.

As audio-visual speech recognition is our main application, we will start by giving a few examples where it may be useful. The practical applications for AVSR are a little more limited in scope than those of audio-only speech recognition, because of the supplemental hardware requirement of a video camera. However, there are a few situations where AVSR would be better suited than simple audio-only recognition, that is, applications in which the environment is very noisy and the gain from adding the video modality would be large.

Take for example the case of a voice-controlled information kiosk or an automated ticket vending machine placed in a train station, metro station or a busy town square. The noise would be unpredictable and at times very strong. Using AVSR instead of audio-only recognition would greatly improve the recognition accuracy, since it would make the system better tolerate the noise. Of course, it can be argued that speech is not necessary in this case. This is true, however speech is more desirable, since it is a much more natural and efficient interface both for asking for information and for buying public transport tickets.

Another challenging environment in which AVSR would work better than audio-only is inside a car. Here too the noise level is quite high, and the type of noise is also unpredictable, since not only the engine noise will be heard, but also quite likely the radio will be running in the background. This is a very challenging scenario, since the noise can be in this case engine noise, music or even speech. The system should recognize the driver and and his commands, while ignoring voices coming from the radio, and isolating the voice of the driver can also be helped by the video modality. In this case, the video can have a double role, first of verifying if the speech is really coming from the driver, and second, of improving the accuracy of speech recognition. Since the lighting conditions are also very unpredictable, using infrared instead of the typical visible spectrum might be a good idea.

Although the conditions are very difficult, the gains for this scenario also promise to be very high. Speech recognition inside the car would render the buttons on the dashboard useless, from radio controls to air-conditioning, and making the driver keep his hands on the wheel and be more focused to the road instead of reaching for those buttons.

Speaker localization is the second application that we used in our work. The practical application for it would be a videoconferencing system that changes focus or view automatically based on who is speaking. Another application would be the automated annotation of meetings, for which speaker localization could be used in conjunction with identification and speech recognition.

Biometric identification is another application that could benefit from improvements in feature extraction and multimodal integration. Imagine a future in which, when walking in a shop, the customer would be instantly recognized by face, gait or his fingerprints when touching the entrance door. He would be greeted by an automated message offering recommendations based on his previous purchases. Even better, he would be able to take the needed items and leave without paying, knowing that his credit card would be automatically charged for his purchase.

Multimodal signal processing offers the prospect of a future where communication between humans and computers will be simpler, more intuitive and at the same time more efficient than now. It is a very exciting domain of research, with a lot of potential for improving people's lives. It is our hope that this work will lead to a better understanding of the problems in this domain and that the methods developed here will be used, not only for research, but for practical applications as well.

Acknowledgments

First and foremost, I would like to thank my thesis advisor, prof. Jean-Philippe Thiran, for his guidance and enthusiasm, his optimism and encouraging words every time something was going wrong. The door of his office was never closed, for any of us. I will never forget his warm welcome on the day of my arrival in the lab. I could not dream of a better supervisor, and I am sure all his current and former students feel the same.

I would also like to thank prof. Murat Kunt, because without him, the Signal Processing Institute would have never existed. The Institute was not only a great forum for sharing ideas, but also a place to make good friends.

I would like to thank the Swiss National Science Foundation for providing financial support through the IM2 NCCR.

My PhD studies in Switzerland would not have been possible without the help of my father. He pushed me from a very young age to learn both English and French, paid for language courses all the way to my college years, and I had never quite understood why. Until I arrived here. Amazingly, except foreign languages, he did not impose anything else on me, allowing me complete freedom to choose the direction of my education as I saw fit. I know he wanted his son to be a doctor. Well, dad, I may not be a medical doctor, but I hope a PhD in computer science also counts.

My mother never complains about my departure to a foreign country, although I know she misses me dearly. I want to thank her for allowing me to pursue my dreams.

I would like to thank my pre-doctoral project supervisor, Ivana, as she introduced me to audio-visual signal processing, and a lot of things that I learned in the very beginning were learned from her. I would also like to thank my master students, Thomas, Andreu, Virginia and Pascal, whose enthusiasm, ideas and help brought a great contribution to this thesis.

During these 4 years, I have made a lot of friends. I want to thank all the colleagues from the Signal Processing Institute, for providing a very warm and friendly environment. I should also mention my friends from the Pre-doctoral year, some of whom have already moved to other corners of the world. They may be far away, but they are not forgotten.

Although the Romanian Students Association in EPFL, A/RO, was only born officially two years ago, we were a close group long before its creation, and the help of my Romanian friends has been invaluable. In particular I have to thank Radu and Carla, because of whom

I had a place to stay when I arrived in Lausanne, and Oana, since without her I would have never had the idea to apply for a position at EPFL.

Finally, I have to thank my girlfriend, Ramona, who had to cope with me during this long and sometimes difficult process. I am very fortunate to have found my other half of the orange, and she knows what I mean. I am happy beyond words that she was there for me during the final period of my thesis, I know I could not have done it without her.

Bibliography

[1] A. Adjoudani and C. Benoît. On the integration of auditory and visual parameters in an HMM-based ASR. In D.G. Stork and M.E. Hennecke, editors, *Speechreading by humans and machines*, pages 461–471. Springer, 1996.

[2] I. Arsic and J.P. Thiran. Mutual information eigenlips for audio-visual speech recognition. *Proceedings of the 14th European Signal Processing Conference(EUSIPCO)*, 2006.

[3] B.S. Atal. Effectiveness of linear prediction characteristics of the speech wave for automatic speaker identification and verification. *Journal of the Acoustical Society of America*, 55(6):1304–1312, 1974.

[4] R. Bakis. Continuous speech recognition via centisecond acoustic states. *Proceedings of the 91st Meeting of the Acoustical Society of America*, 1976.

[5] R. Battiti. Using mutual information for selecting features in supervised neural net learning. *IEEE Transactions on Neural Networks*, 5(4), 1994.

[6] L.E. Baum, T. Petrie, G. Soules, and N. Weiss. A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *The Annals of Mathematical Statistics*, 41(1):164–171, 1970.

[7] R.E. Bellman. *Adaptive Control Processes: A Guided Tour*. Princeton University Press, 1961.

[8] M. Ben-Bassat. Use of distance measures, information measures and error bounds in feature evaluation. In P.R. Krishnaiah and L.N. Kanal, editors, *Handbook of Statistics, Vol. 2*, pages 773–791. Academic Press, 1982.

[9] C. Benoit, J. C. Martin, C. Pelachaud, L. Schomaker, and B. Suhm. Audio-visual and multimodal speech systems. In D. Gibbon, editor, *Handbook of Standards and Resources for Spoken Language Systems - Supplement Volume*, 2000.

[10] F. Berthommier and H. Glotin. A new SNR-feature mapping for robust multistream speech recognition. *Proceedings of the International Congress on Phonetic Sciences*, pages 711–715, 1999.

[11] P. Besson, M. Kunt, T. Butz, and J.P. Thiran. A multimodal approach to extract optimized audio features for speaker detection. *Proceedings of the European Signal Processing Conference (EUSIPCO)*, 2005.

[12] P. Besson, G. Monaci, P. Vandergheynst, and M. Kunt. Experimental framework for speaker detection on the CUAVE database, EPFL-ITS Tech. Rep. 2006-003. Technical report, EPFL, Lausanne, Switzerland, 2006.

[13] C. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, 1995.

[14] H. Bourlard and N. Morgan. *Connectionist Speech Recognition - A Hybrid Approach*. Kluwer Academic Publishers, 1994.

[15] T. Butz and J.P. Thiran. Feature space mutual information in speech-video sequences. *Proceedings of the IEEE International Conference on Multimedia and Expo*, 2:361–364, 2002.

[16] T. Butz and J.P. Thiran. From error probability to information theoretic (multimodal) signal processing. *Signal Processing*, (85):875–902, 2005.

[17] C. Cortes and V. Vapnik. Support-vector networks. *Machine Learning*, 20:273–297, 1995.

[18] T. Cover and J. Thomas. *Elements of Information Theory*. Wiley Series in Telecommunications, New York, 1991.

[19] S. Cox, I. Matthews, and A. Bangham. Combining noise compensation with visual information in speech recognition. *Proceedings of the European Tutorial Workshop on Audio-Visual Speech Processing*, 1997.

[20] S.B. Davis and P. Mermelstein. Comparison of parametric representations of monosyllabic word recognition in continuously spoken sentences. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 28:357–366, 1980.

[21] T. Drugman, M. Gurban, and J.Ph. Thiran. Relevant feature selection for audio-visual speech recognition. In *Proceedings of the 9th International Workshop on Multimedia Signal Processing (MMSP)*, 2007.

[22] R.P.W. Duin and M. Loog. Linear dimensionality reduction via a heteroscedastic extension of LDA: the Chernoff criterion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(6):732–739, 2004.

[23] S. Dupont and J. Luettin. Audio-visual speech modeling for continuous speech recognition. In *IEEE Transactions on Multimedia*, volume 2, pages 141–151, 2000.

[24] E. Feig and S. Winograd. Fast algorithms for the discrete cosine transform. *IEEE Transactions on Signal Processing*, 40(9):2174–2193, 1992.

[25] R.A. Fisher. The use of multiple measurements in taxonomic problems. In *Annals of Eugenics*, volume 7, pages 179–188, 1936.

[26] J.W. Fisher III and T. Darrell. Speaker association with signal-level audiovisual fusion. *IEEE Transactions on Multimedia*, 6(3):406–413, 2004.

[27] J.W. Fisher III, T. Darrell, W.T. Freeman, and P. Viola. Learning joint statistical models for audio-visual fusion and segregation. *Advances in Neural Information Processing Systems*, 2000.

[28] F. Fleuret. Fast binary feature selection with conditional mutual information. *Journal of Machine Learning Research*, 5:1531–1555, 2004.

[29] N.A. Fox, R. Gross, J.F. Cohn, and R.B. Reilly. Robust biometric person identification using automatic classifier fusion of speech, mouth, and face experts. *IEEE Transactions on Multimedia*, 9(4):701–714, 2007.

[30] S. Furui. Cepstral analysis technique for automatic speaker verification. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 29(2):254–272, 1981.

[31] D. Le Gall. MPEG: a video compression standard for multimedia applications. *Communications of the ACM*, 34(4):46–58, 1991.

[32] A. Ganapathiraju, J. Hamaker, and J. Picone. Hybrid SVM/HMM architectures for speech recognition. *Proceedings of the International Conference on Spoken Language Processing*, 4:504–507, 2000.

[33] H. Glotin, D. Vergyri, C. Neti, G. Potamianos, and J. Luettin. Weighting schemes for audio-visual fusion in speech recognition. *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, pages 173–176, 2001.

[34] B. Gold and N. Morgan. *Speech and audio signal processing: Processing and perception of speech and music*. Wiley, 2000.

[35] M. Gordan, C. Kotropoulos, and I. Pitas. A support vector machine-based dynamic network for visual speech recognition applications. *EURASIP Journal on Applied Signal Processing*, 2002(11):1248–1259, 2002.

[36] G. Gravier, S. Axelrod, G. Potamianos, and C. Neti. Maximum entropy and MCE based HMM stream weight estimation for audio-visual ASR. *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2002.

[37] M.S. Gray, J.R. Movellan, and T.J. Sejnowski. Dynamic features for visual speechreading: A systematic comparison. In M.C. Mozer, M.I. Jordan, and T. Petsche, editors, *Advances in Neural Information Processing Systems*, volume 9. MIT Press, 1997.

[38] M. Gurban and J.Ph. Thiran. Multimodal speaker localization in a probabilistic framework. In *Proceedings of the 14th European Signal Processing Conference (EUSIPCO)*, 2006.

[39] M. Gurban and J.Ph. Thiran. Using entropy as a stream reliability estimate for audio-visual speech recognition. In *Proceedings of the 16th European Signal Processing Conference*, 2008.

[40] M. Gurban, J.Ph. Thiran, T. Drugman, and T. Dutoit. Dynamic modality weighting for multi-stream HMMs in Audio-Visual Speech Recognition. In *Proceedings of the 10th International Conference on Multimodal Interfaces*, 2008.

[41] S. Gurbuz, Z. Tufekci, E. Patterson, and J.N. Gowdy. Application of affine-invariant fourier descriptors to lipreading for audio-visual speech recognition. *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, pages 177–180, 2001.

[42] M. Heckmann, F. Berthommier, and K. Kroschel. A hybrid ANN/HMM audio-visual speech recognition system. *Proceedings of the International Conference on Audio-Visual Speech Processing*, 2001.

[43] M. Heckmann, F. Berthommier, and K. Kroschel. Noise adaptive stream weighting in audio-visual speech recognition. *EURASIP Journal on Applied Signal Processing*, 2002:1260–1273, 2002.

[44] H. Hermansky. TRAP-TANDEM: data-driven extraction of temporal features from speech. *Proceedings of the IEEE Workshop on Automatic Speech Recognition and Understanding*, pages 255–260, 2003.

[45] H. Hermansky. Perceptual linear predictive (PLP) analysis of speech. *Journal of the Acoustical Society of America*, 87(4):1738–1752, 1990.

[46] H. Hermansky and N. Morgan. RASTA processing of speech. *IEEE Transactions on Speech and Audio Processing*, 2(4):578–589, 1994.

[47] H. Hermansky and S. Sharma. Temporal patterns (TRAPs) in asr of noisy speech. *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, 1:289–292, Mar 1999.

[48] J. Hershey and J.R. Movellan. Audio vision: Using audio-visual synchrony to locate sounds. *Neural Information Processing Systems*, pages 813–819, 1999.

[49] B. Horn and B. Schunck. Determining optical flow. In *Artificial Intelligence*, volume 17, pages 185–204, 1981.

[50] F. J. Huang and T. Chen. Real-time lip-synch face animation driven by human voice. *Proceedings of the IEEE Workshop on Multimedia Signal Processing*, 1998.

[51] A.K. Jain. *Fundamentals of Digital Image Processing*. Prentice-Hall, 1989.

[52] P. Jourlin. Word dependent acoustic-labial weights in HMM-based speech recognition. *Proceedings of the European Tutorial Workshop on Audio-Visual Speech Processing*, pages 69–72, 1997.

[53] K. Kirchhoff and J. Bilmes. Dynamic classifier combination in hybrid speech recognition systems using utterance-level confidence values. *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, pages 693–696, 1999.

[54] J. Kittler, M. Hatef, R.P.W. Duin, and J. Matas. On combining classifiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(3):226–239, 1998.

[55] R. Klette, K. Schlüns, and A. Koschan. *Computer Vision*. Springer, 1998.

[56] N. Kumar and A.G. Andreou. Heteroscedastic discriminant analysis and reduced rank HMMs for improved speech recognition. *Speech Communication*, 26(4):283–297, 1998.

[57] H. Liu and H. Motoda. *Feature selection for knowledge discovery and data mining*. Kluwer Academic Publishers, 1998.

[58] B.D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. *Proceedings of the International Joint Conference on Artificial Intelligence*, pages 674–679, 1981.

[59] J. Luettin and N.A. Thacker. Speechreading using probabilistic models. *Computer Vision and Image Understanding*, 65(2):163–178, 1997.

[60] J. B. A. Maintz and M. A. Viergever. A survey of medical image registration. *Medical Image Analysis*, 2(1):1–36, 1998.

[61] J. Makhoul. Linear prediction: A tutorial review. *Proceedings of the IEEE*, 63(4): 561–580, 1975.

[62] J. Makhoul. Spectral linear prediction: Properties and applications. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 23(3):283–296, 1975.

[63] H. McGurk and J. MacDonald. Hearing lips and seeing voices. *Nature*, 264:746–748, 1976.

[64] U. Meier, W. Hurst, and P. Duchnowski. Adaptive bimodal sensor fusion for automatic speechreading. *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, pages 833–836, 1996.

[65] P. Mermelstein. Distance measures for speech recognition, psychological and instrumental. In R.C.H. Chen, editor, *Pattern Recognition and Artificial Intelligence*. Academic Press, 1976.

[66] H. Misra, H. Bourlard, and V. Tyagi. New entropy based combination rules in HMM/ANN multi-stream ASR. *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, 2003.

[67] C. Miyajima, K. Tokuda, and T. Kitamura. Audio-visual speech recognition using MCE-based HMMs and model-dependent stream weights. *Proceedings of the International Conference on Spoken Language Processing*, II:1023–1026, 2000.

[68] G. Monaci, O. Divorra Escoda, and P. Vandergheynst. Analysis of Multimodal Sequences Using Geometric Video Representations. *Signal Processing*, 86(12):3534–3548, 2006.

[69] G. Monaci, P. Jost, P. Vandergheynst, B. Mailhe, S. Lesage, and R. Gribonval. Learning Multi-Modal Dictionaries. *IEEE Transactions on Image Processing*, 16(9):2272–2283, 2007.

[70] N. Morgan and H. Bourlard. Continuous speech recognition, an introduction to the hybrid HMM/connectionist approach. *IEEE Signal Processing Magazine*, 12(3):25–42, 1995.

[71] J.R. Movellan. Visual speech recognition with stochastic networks. In G. Tesauro, D. Touretzky, and T. Leen, editors, *Advances in Neural Information Processing Systems*, volume 7. MIT Press, 1995.

[72] S. Nakamura, H. Ito, and K. Shikano. Stream weight optimization of speech and lip image sequence for audio-visual speech recognition. *Proceedings of the International Conference on Spoken Language Processing*, III:20–23, 2000.

[73] H.J. Nock, G. Iyengar, and C. Neti. Assessing face and speech consistency for monologue detection in video. *Proceedings of ACM Multimedia*, 2002.

[74] H.J. Nock, G. Iyengar, and C. Neti. Speaker localisation using audio-visual synchrony: An empirical study. *Proceedings of the International Conference on Image and Video Retrieval*, 2003.

[75] E.K. Patterson, S. Gurbuz, Z. Tufekci, and J.N. Gowdy. Moving-talker, speaker-independent feature study and baseline results using the CUAVE multimodal speech corpus. *EURASIP Journal on Applied Signal Processing*, 2002(11):1189–1201, 2002.

[76] K. Pearson. On lines and planes of closest fit to systems of points in space. In *Philosophical Magazine*, volume 2, pages 559–572, 1901.

[77] H. Peng, F. Long, and C. Ding. Feature selection based on mutual information: Criteria of max-dependency, max-relevance, and min-redundancy. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(8), 2005.

[78] E.D. Petajan. Automatic lipreading to enhance speech recognition. In *Proceedings of the IEEE Communication Society Global Telecommunications Conference*, 1984.

[79] G. Potamianos and H.P. Graf. Discriminative training of HMM stream exponents for audio-visual speech recognition. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, pages 3733–3736, 1998.

[80] G. Potamianos and C. Neti. Stream confidence estimation for audio-visual speech recognition. *Proceedings of the International Conference on Spoken Language Processing (ICSLP)*, 2000.

[81] G. Potamianos and C. Neti. Improved ROI and within frame discriminant features for lipreading. *Proceedings of the IEEE International Conference on Image Processing*, 3:250253, 2001.

[82] G. Potamianos and C. Neti. Automatic speechreading of impaired speech. *Proceedings of the Workshop on Audio-Visual Speech Processing*, 2001.

[83] G. Potamianos and P. Scanlon. Exploiting lower face symmetry in appearance-based automatic speechreading. *Proceedings of the International Conference on Audio-Visual Speech Processing*, 2005.

[84] G. Potamianos, H.P. Graf, and E. Cosatto. An image transform approach for HMM based automatic lipreading. In *Proceedings of the International Conference on Image Processing*, volume 3, pages 173–177, 1998.

[85] G. Potamianos, C. Neti, G. Gravier, A. Garg, and A.W. Senior. Recent advances in the automatic recognition of audio-visual speech. *Proceedings of the IEEE*, 91(9), 2003.

[86] G. Potamianos, C. Neti, J. Luettin, and I. Matthews. Audio-visual automatic speech recognition: an overview. In G. Bailly, E. Vatikiotis-Bateson, and P. Perrier, editors, *Issues in audio-visual speech processing*. MIT Press, 2004.

[87] L. Rabiner and B. Juang. An introduction to Hidden Markov Models. *IEEE ASSP Magazine*, 3(1):4–16, 1986.

[88] L.R. Rabiner. A tutorial on Hidden Markov Models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2), 1989.

[89] L.R. Rabiner and B.-H. Juang. *Fundamentals of Speech Recognition*. Prentice-Hall Signal Processing Series, 1993.

[90] C.R. Rao. The utilization of multiple measurements in problems of biological classification. In *Journal of the Royal Statistical Society, Series B*, volume 10, pages 159–203, 1948.

[91] K.R. Rao and P. Yip. *Discrete Cosine Transform: Algorithms, Advantages, Applications*. Academic Press, 1990.

[92] R. Reilly and P. Scanlon. Feature analysis for automatic speechreading. *Proceedings of the Workshop on Multimedia Signal Processing*, pages 625–630, 2001.

[93] D.W. Robinson and R.S. Dadson. A redetermination of the equal-loudness relations for pure tones. *Journal of Applied Physics*, 7:166–181, 1956.

[94] A. Ross and A. K. Jain. Multimodal biometrics: An overview. *Proceedings of the 12th European Signal Processing Conference (EUSIPCO)*, 2004.

[95] H. Sakoe and S. Chiba. Dynamic programming algorithm optimization for spoken word recognition. *IEEE Transactions on Acoustics, Speech and Signal Processing*, 26 (1):43–49, 1978.

[96] E. Sanchez-Soto, A. Potamianos, and K. Daoudi. Unsupervised stream weight computation using anti-models. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, 2007.

[97] P. Scanlon. *Audio And Visual Feature Analysis For Speech Recognition*. University College Dublin, 2005.

[98] P. Scanlon, D.P.W. Ellis, and R.B. Reilly. Using mutual information to design class specific phone recognizers. *Proceedings of Eurospeech*, 2003.

[99] P. Scanlon, G. Potamianos, V. Libal, and S.M. Chu. Mutual information based visual feature selection for lipreading. *ICSLP*, pages 2037–2040, 2004.

[100] P. Scanlon, D.P.W. Ellis, and R.B. Reilly. Using broad phonetic group experts for improved speech recognition. *IEEE Transactions on Audio, Speech, and Language Processing*, 15(3):803–812, 2007.

[101] J.C. Schlimmer. Efficiently inducing determinations: A complete and systematic search algorithm that uses optimal pruning. In *Proceedings of the 10th International Conference on Machine Learning*, pages 284–290, 1993.

[102] B. Schlkopf and A.J. Smola. *Learning with Kernels*. MIT Press, 2002.

[103] N. Sebe, I. Cohen, and T.S. Huang. Multimodal emotion recognition. In C.H. Chen and P.S.P. Wang, editors, *Handbook of Pattern Recognition and Computer Vision*. World Scientific, 2005.

[104] R. Seymour, J. Ming, and D. Stewart. A new posterior based audio-visual integration method for robust speech recognition. *Proceedings of Interspeech*, page 1229.

[105] R. Seymour, D. Stewart, and J. Ming. Audio-visual integration for robust speech recognition using maximum weighted stream posteriors. *Proceedings of Interspeech*, 2007.

[106] H.F. Silverman and D.P. Morgan. The application of dynamic programming to connected speech recognition. *IEEE ASSP Magazine*, 7(3):6–25, 1990.

[107] M. Slaney and M. Covell. Facesync: A linear operator for measuring synchronization of video facial images and audio tracks. *Neural Information Processing Systems*, pages 814–820, 2000.

[108] P. Somol, P. Pudil, and J. Kittler. Fast branch & bound algorithms for optimal feature selection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(7), 2004.

[109] M. Sonka, V. Hlavac, and R. Boyle. *Image Processing, Analysis, and Machine Vision.* Brooks/Cole Publishing Company, 1999.

[110] S.S. Stevens. On the psychophysical law. *Psychology Review*, 64:153–181, 1957.

[111] S.S. Stevens and J. Volkman. The relation of pitch to frequency: A revised scale. *The American Journal of Psychology*, 53:329–353, 1940.

[112] S.S. Stevens, J. Volkman, and E.B. Newman. A scale for the measurement of the psychological magnitude of pitch. *Journal of the Acoustical Society of America*, 8(3): 185–190, 1937.

[113] Q. Sumby and I. Pollack. Visual contribution to speech intelligibility in noise. *The Journal of the Accoustical Society of America*, 26(2):121–215, 1954.

[114] Q. Summerfield. Lipreading and audio-visual speech perception. *Philosophical Transactions of the Royal Society of London. Series B: Biological Sciences*, 335(1273):71–78, 1992.

[115] S. Tamura, K. Iwano, and S. Furui. A stream-weight optimization method for audio-visual speech recognition using multi-stream HMMs. *Proc. International Conference on Acoustics, Speech and Signal Processing*, 1:857–860, 2004.

[116] S. Tamura, K. Iwano, and S. Furui. A stream-weight optimization method for multi-stream HMMs based on likelihood value normalization. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, pages 468–472, 2005.

[117] P. Teissier, J. Robert-Ribes, and J.L. Schwartz. Comparing models for audiovisual fusion in a noisy-vowel recognition task. *IEEE Transactions on Speech and Audio Processing*, 7:629–642, 1999.

[118] J.Ph. Thiran, A. Valles, T. Drugman, and M. Gurban. Définition et sélection d'attributs visuels pour la reconnaissance audio-visuelle de la parole. In *Traitement et Analyse de l'Information : Methodes et Applications (TAIMA07)*, 2007.

[119] A. Valles, M. Gurban, and J.Ph. Thiran. Low-dimensional motion features for audio-visual speech recognition. In *Proceedings of the 15th European Signal Processing Conference (EUSIPCO)*, 2007.

[120] H.H. Yang, S.V. Vuuren, S. Sharma, and H. Hermansky. Relevance of time-frequency features for phonetic and speaker-channel classification. *Speech Communication*, 31(1):35–50, 2000.

[121] S. Young, D. Kershaw, J. Odell, D. Ollason, V. Valtchev, and P. Woodland. *The HTK Book*. Cambridge, Entropic Ltd., 1999.

[122] B. Zitova and J. Flusser. Image registration methods: a survey. *Image and Vision Computing*, 21:977–1000, 2003.

[123] E. Zwicker. Subdivision of the audible frequency range into critical bands. *Journal of the Acoustical Society of America*, 33(2), 1961.

Curriculum Vitae

Name:	**Mihai Gurban**
Degrees:	Bachelor of Science in Computer Science, "Politehnica" University Timisoara, Romania
Address:	Signal Processing Laboratory (LTS5) Swiss Federal Institute of Technology (EPFL) CH-1015 Lausanne Switzerland
Contact numbers:	Tel. +41 21 693 46 82 Fax. +41 21 693 76 00 E-mail: mihai.gurban@epfl.ch mihai.gurban@gmail.com
Civil status:	Single
Date and place of birth:	June 22nd 1979, Timisoara, Romania
Nationality:	Romanian

Education

Since 2003 **PhD student**
Signal Processing Laboratory (LTS5)
Swiss Federal Institute of Technology (EPFL)
Lausanne, Switzerland
Doctoral School, CGPA: 5.6/6

1998–2003 **Bachelor of Science in Computer Science**
"Politehnica" University Timisoara, Romania
High Merit Scholarship, CGPA: 9.9/10

Experience

Since 2004 **Teaching and Research Assistant**
Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland
Provided assistance for various courses, 6 semesters.

Since 2006 **Supervisor for Master students**
Signal Processing Laboratory (LTS5) Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland

2002 **Software Developer**
Caatoosee, Timisoara, Romania
The design and implementation of a report-management system, complete with database, web interface and email notifications.
Using SQL, Java Server Pages, HTML.
Teamwork, 3 months.

Publications

Conference Papers

M. Gurban and J.Ph. Thiran and T. Drugman and T. Dutoit.
Dynamic Modality Weighting for Multi-Stream HMMs in Audio-Visual Speech Recognition
Proceedings of the 10th International Conference on Multimodal Interfaces (ICMI), Chania, Greece, 2008

M. Gurban and J.Ph. Thiran
Using Entropy as a Stream Reliability Estimate for Audio-Visual Speech Recognition
Proceedings of the 16th European Signal Processing Conference (EUSIPCO), Lausanne, Switzerland, 2008

T. Drugman and M. Gurban and J.Ph. Thiran
Relevant Feature Selection for Audio-Visual Speech Recognition.
Proceedings of the 9th International Workshop on Multimedia Signal Processing (MMSP), Chania, Greece, 2007

J.Ph. Thiran and A. Valles and T. Drugman and M. Gurban
Définition et Sélection d'Attributs Visuels pour la Reconnaissance Audio-Visuelle de la Parole
Traitement et Analyse de l'Information : Methodes et Applications (TAIMA), Hammamet, Tunisia, 2007

A. Valles and M. Gurban and J.Ph. Thiran
Low-Dimensional Motion Features for Audio-Visual Speech Recognition
Proceedings of the 15th European Signal Processing Conference (EUSIPCO), Florence, Italy, 2007

M. Gurban and J.Ph. Thiran
Multimodal Speaker Localization in a Probabilistic Framework
14th European Signal Processing Conference (EUSIPCO), Antalya, Turkey, 2006

Book Chapters

M. Gurban, V. Vilaplana, J.Ph. Thiran, and F. Marques.
Face and Speech Interaction,
in "Multimodal User Interfaces, from Signals to Interaction", Dimitros Tzovaras (Ed.), Springer, May 2008.

Relevant Skills

Areas of expertise:	image processing, speech recognition, multimodal fusion, machine learning.
Programming:	C/C++, Matlab, Python, VBA, Java.

Languages

English:	fluent (Cambridge Certificate in Advanced English and ETS GRE)
French:	fluent (lived in the French-speaking part of Switzerland for 5 years)
German:	intermediate B1 (took various courses over three years)
Romanian:	native language

Hobbies

Running, skiing, swimming, reading, computers.