Robust feature extraction for multimodal speaker ID

A multimodal speaker ID system – The experts' room

Sergi Hernanz Nogueras

University of Southern California

Signal and Image Processing Institute

Speech Analysis and Interpretation Laboratory

May 2005

Mentor: Shrikanth S Narayanan

Revision history

Revision	Date	Comments
v1.0	2005/06/25	Introduction version
		Introduction, critical reviews and Baseline Goals
v1.1	2005/07/20	Baseline version
		All baseline simulations and results
v1.3	2006/01/31	Results version
		All results plotted, final simulations needed.
V1.4	2006/02/28	All results version
		No more simulations will be done, discussion, conclusion to be done
V1.5	2007/01/15	Discussion and conlusion about Speaker ID done
V1.6	2007/03/20	Discussion and conclusion about Articulatory Features done
V1.7	2007/05/15	Discussion and conclusion about Multimodality
V1.8	2007/07/02	Multimodal meeting room discussion and conclusion done.
V1.9	2007/12/26	Further work
V1.10	2008/11/1	Review and limitations
V1.11	2008/11/30	Summary and abstract
V1.12	2008/03/31	General review and format

Abstract

Speaker recognition is traditionally achieved in speech signal analysis using voice signal. New scenarios like the meeting room at USC supply new information to the classification problem. Source localization based on a microphone array and video capture with posterior processing for people localization could be integrated into the speech-based recognition creating a multimodal system. These new systems acquire greater importance in low SNR conditions, which are frequent in the meeting rooms. Furthermore, other features from voice could be extracted and added to the final multimodal system, thinking of those being more robust against noise.

The current work demonstrates how multimodal integration of mentioned sources can improve performances for the speaker recognition issue in the meeting room. The improvement is very low and theoretical analysis of the multimodal probabilistic integration is made to set up limits for such performance. In the speech-only classification space, weakness of current approaches against noise is demonstrated, so is the higher robustness of the articulatory features. The complementarity and usability of the articulatory features is shown when joined with the baseline classifiers, obtaining small improvements on simple simulations.

The main conclusion reveals that further work using the same environment and improved methods should acquire remarkable results.

Summary

All along the current project, the speaker recognition is being reviewed. First simulations in this work use the latest ‘state of the art’ algorithms, and later new approaches and lots of modifications are used. Multimodality is the main idea to achieve better results. The new multimodal data supplied to the speaker recognition system will be articulatory features and video+voice source localization in the meeting room scenario. Some articulatory features have not been widely used for speech analysis so the correct extraction methods are still not developed. On the other hand, voice source and video spatial localization algorithms are known and only the integration methods have to be defined. Theoretical review and a study about integration will follow before finally selecting an algorithm.

Machine learning techniques are applied to extract articulatory features, which perform a surprisingly right classification. The usability of those feature extractor outputs for the speaker recognition issue is not that clear, but very important conclusions are set about how the extraction process can affect the posterior usage and how other extraction methods could be approached.

During the work, articulatory features demonstrate to be less affected by noise than the baseline MFCC+GMM approach, but the correct extraction methods are still not available. Even using the baseline extraction methods based on MLP, a classification is possible using the articulatory features, and complementarities with baseline methods are demonstrated. The improvement of the whole system adding articulatory features is very small, but demonstrates their usability. The whole process of the articulatory feature integration can surely be reviewed expecting successful results in the future.

Due to an extended analysis of how noise poisons the speech features, very concrete conclusions are set about noise rejection and affection. By plotting how the system works against different SNR conditions, behaviors of some methods are explained. In low SNR conditions, very simple changes in the

algorithms improve the overall performance, and reveal the lack of noise-oriented design of the baseline.

The most of the methods approached in the current work were finally applied to the meeting room scenario at USC. An encouraging but small performance increase was achieved, and so the aim of the current work was considered realized. The trade-off between the spent effort and the small improvement is to be reviewed with further approaches and work.

Acknowledgements

I would like to thank everybody that have contributed to this project, sharing their knowledge and devoting some of their time to help me to carry out this challenging task. I would like to especially thank the following people:

Professor Shrikanth S Narayanan., because he has motivated me to do my job better, he has been always willing to give a hand, and has led my project into a successful ending. I also wanted to thank him for his amazing talks, the sharing of his never ending experience, and his supervision.

Department and apartment partner Joaquin Lopez, who always listened to my ideas and guided so many developing decisions, was always there and supported me.

Thanks to PhD student Carlos Busso, who helped me about Signal processing issues concerning my low level problems along my thesis.

Professor Panayiotis G. Georgiou, whom practical experience concerning audio databases registration, implementation of different methods, resources available, and of course, his microphone array system; helped me a lot.

My research group; Naveen Srinivasamurthy, Soon-il Kwon, Sung Lee; who joined me at those never-ending sessions in the meeting room.

My office partners, Chartchai Meesookho, Erdem Unal and Viktor Rozgic, supporting my ideas and the blackboards' monopolizations.

Professor Antonio Ortega, who helped me about so many personal issues.

Nune Abramyam, Regina Morton and Tim Boston; who took care of us about faculty stuff.

Table of contents

Abstract 3
Summary 4
Acknowledgements 6
Table of contents 7
List of Tables 9
List of figures 11
Abbreviations 12
Abbreviations 12
1. Introduction 13
1.1. Organization of the report 14
1.2. Current state of the art 15
Speaker recognition 15
Robust Feature Extraction 20
Multimodality 22
2. Methods 25
2.1. Goals 25
Speaker Recognition 26
Robust feature extraction 33
Theoretical multimodal studies 46
Multimodal speaker recognition 49
2.2. Progress of the project 57
Requirements 57
Accomplishments 61
Limitations 62
3. Results 64
3.1. Speaker Recognition 64
BASELINE 64
REAL SCENARIO 68
ARTICULATORY FEATURES 69
3.2. Robust feature extraction 73
Speech signal 73
Noise on MFCC 74

Articulatory features 76
3.3. Theoretical multimodality studies 82
3.4. Multimodal speaker recognition 84
4. Discussion 95
4.1. Speaker Recognition 95
4.2. Robust feature extraction 104
Speech signal 104
Noise on MFCC 105
Articulatory features 106
4.3. Theoretical multimodality studies 110
4.4. Multimodal speaker recognition 114
5. Conclusions 118
6. Further work 121
7. References 123
8. Appendices 126
8.1. ICAASP' 05 Paper 126
8.2. Presentation plots 126

List of Tables

Table 1.- NIST Database. Room Microphones 27
Table 2.- NIST Database. Microphones used by participants 27
Table 3.- NIST Database. Subjects 27
Table 4.- NIST Database. Artifacts 27
Table 5.- Phoneme articulatory classification 45
Table 6.- Phoneme table for consonants 45
Table 7.- Phoneme table for vowels 45
Table 8.- NIST Database. Different single microphones for training data 65
Table 9.- NIST Database. Multiple microphones for training and testing 65
Table 10.- NIST Database. Different lengths of training data 65
Table 11.- NIST Database. Number of filters of MFCC 66
Table 12.- NIST Database. Addition of the first coefficient of DCT 66
Table 13.- NIST Database. Different length of utterance 66
Table 14.- NIST Database. Number of gaussians 67
Table 15.- NIST Database. Training Situations 68
Table 16.- NIST Database. Training the background model 68
Table 17.- Room Data. Different microphone info 68
Table 18.- Room Data. Training with clean data 69
Table 19.- Articulatory baseline. All pairs 70
Table 20.- Articulatory baseline. Lucky pairs 70
Table 21.- Dynamic articulatory classification 71
Table 22.- Pair classification using pitch and energy only 72
Table 23.- Pair classification excluding pitch and energy from articulatory features 73
Table 24.- Confusion matrix. Manner I 78
Table 25.- Confusion matrix. Manner II 78
Table 26.- Confusion matrix. Place 79
Table 27.- Confusion matrix. Voiced-voiceless 79
Table 28.- Confusion matrix. Vowel 79
Table 29.- Confusion matrix. Height 79
Table 30.- Confusion matrix. Round 79

Table 31.- Articulatory baseline performance 79
Table 32.- MLP complexity analysis for articulatory baseline 80
Table 33.- MPL input vector analysis for articulatory baseline 80
Table 34.- Articulatory baseline performance 82

List of figures

Figure 1.- General scheme 25
Figure 2.- Evaluation problem 29
Figure 3.- ACS Evaluation 30
Figure 4.- ACT Evaluation 31
Figure 5.- CCT Evaluation 31
Figure 6.- GMM Performance Vs SNR 34
Figure 7.- GMM Performance Vs Test Utternace Length 30
Figure 8.- Speech and AWGN distribution over MF Filterbank 38
Figure 9.- Speech and AWGN distribution over log-MF Finterbank 38
Figure 10.- Speech and AWGN distribution over cepstral coefficients 39
Figure 11.- Possible scheme for noise reduction 40
Figure 12.- Speech and AWGN distributions over each cepstral coef. 41
Figure 13.- Pitch evaluation values and pitch extractor results 43
Figure 14.- Simple example for joint classification 46
Figure 15.- Microphone array output 53
Figure 16.- Microphone array pdf 54
Figure 17.- Video output 55
Figure 18.- Covariance calculation from video 56
Figure 19.- NIST Database. Number of filters of MFCC 66
Figure 20.- NIST Database. Different length of utterance 67
Figure 21.- NIST Database. Number of Gaussians 67
Figure 22.- Discard high frequencies in MFFB 75
Figure 23.- Pitch extraction accuracy Vs SNR 77
Figure 24.- Performances of Bayes, sum, product and SE 82
Figure 25.- Multimodal performance against previous performances 83
Figure 26.- State-transition graph used in video 86
Figure 27.- State-transition graph used in video 88
Figure 28.- Model pdf and classification spaces 91

Abbreviations

FB	Feature based (approach)
MM	Multimodal (approach)
MFCC	Mel frequency cepstral coefficients
MFFB	Mel-frequency filterbank
LPC	Linear predictive coefficients
SNR	Signal to Noise Ratio
VQ	Vector quantization
GMM	Gaussian Mixture Models
HMM	Hidden Markov Models
EM	Expectation Maximization
NIST	National Institute Of Standards and Technology
MLP	Multi-Layer Perceptron
RBF	Radial Basis Functions
DBNN	Dynamic Bayesian Neural Networks
MOE	Mixture Of Experts
TDNN	Time Delay Neural Network,
SRN	Simultaneous Recurrent Neural Network,
BPTT	BackPropagation Through Time
DCT	Discrete Cosine Transform
AWGN	Additive White Gaussian Noise
SL	Semantic Level
FL	Feature Level
SE	Squared Error
PDF	Probability Density Function
STT	Speech To Text

1. Introduction

New human computer interfaces need to extract ‘who speaks and when’. This information is used to create personalized services, grant remote access, guarantee security policies or for indexing purposes.

It’s all about recognizing a speaker, and traditionally it has been divided into two different problems: speaker recognition or speaker identification. The first problem concerns security purposes, where a sample of voice can be admitted as one of the allowed users or rejected as an intruder. Background models are usually acting as intruders, moving the problem to a highest likelihood approach. Then the particular aim of these systems is to train the models exhaustively, specially background ones. Second issue tries to select the most likely model of a set. The training of those models is less demanding, and can be done in non ideal conditions. These systems use to deal with higher noises and disturbing scenarios, such as meeting rooms.
Both systems share techniques and theoretical keys; then methods can be extracted from both of them for any purpose. Obviously, there are important differences; for example segmentation of speech does not make sense in the identification problem, where a piece of voice is assumed to belong to the same speaker. However, the study will be based on a recognition problem, assuming that techniques for identification should be similar except from background modeling, speech segmentation and some other specific issues.

The experts’ room is a good example of Speaker recognition application. A set of people meet in the room, and the system must be able to extract the id of the currently speaking user. This information can be extracted to save meeting reports when joint with a speech recognition system, or to gain remote access to the room, sending the speaker ids and voice information to remote assistants. Problem has been approached recently by some researchers [17] [18] [19] and it is becoming an interesting situation for the community. Allowing the users to move free of carrying microphones or sensors is the most important request of this project. People can go into the room and act in a natural way,

without being disturbed by gadgets. These requirements imply a set of real constraints to the signal processing problem.

Constraints of this scenario, actually found in most of the real scenarios, are; very low SNR, use of speech expressions, strong noises such as blows or movements, fast speaker changes, overlapping of speakers or important differences on speech volumes. All these problems modify implementation or evaluation methods.

This study focuses on finding a solution to the speaker recognition problem on difficult scenarios. Work can be divided into some important objectives: analyze voice features and its robustness to noise, suggest and evaluate new voice features, analyze methods for combining different sets of features, and finally apply used methods to the specific problem.

This introduction continues with a succinct explanation of the report's organization, summary of previous work about the speaker id or recognition approaches, some referring to the extracted and used voice features until the moment, and the combination methods for multimodal systems. Further in the document, methods and results will be explained

1.1. Organization of the report

This introduction is created to involve the reader into the speaker multimodal recognition issue. Please note it has been divided into three different topics; 'speaker recognition', 'robust feature extraction from speech' and 'multimodality', and so have been all the rest of the project. Later on 'Methods', experimental goals and accomplishments are explained in detail. Results and discussions do not appear before its corresponding section, and finally, conclusions are explained and possible future work analyzed.
Since there are a lot of approaches and simulations all along the project, and they are concerning different goals, every objective, simulation and result will be

enumerated to simplify the search of the whole process on a specific experiment through the paragraphs for ‘Methods’, ‘Results’ and ‘Discussion’

This organization of the report could bring to some confusion; a recommended reading method is navigating the Goals, Progress of the project (Requirements, Accomplishments and Limitations), Results and Conclusions section by section. Otherwise, one could keep reading through the different experiments related to the different topics, and get into the next section with no record of the first topic achievements.

1.2. Current state of the art

Speaker recognition

Basic approach to the speaker recognition has been based on segmentation or clustering of speech segments. Main problems have been the decision of features to be used, the feature extraction methods, the error measure criteria, the cluster/model selection or the cluster/model training methods.

All techniques capture speech samples from audio sources in their first algorithm step, and then features are extracted from that data, usually cepstral or LPC coefficients. Different features from different natures are extracted from the voice is this very important step. MFC coefficients has shown to be more robust to noise than LPC and derived features [1]. The most of the studies have been using MFCC, and only a few of them apply new voice characteristics to the speaker recognition issue. Current approaches combining information from different features [5] classify them into acoustic features, prosodic, phonetic, lexical and conversational features. They take into account from low-level characteristics of speech signal to high-level ones. [7] and [8] show results about usability of prosodic and conversational characteristics avoiding text transcription.

Once a vector of inputs is obtained from recorded sound, input vectors are segmented into speaker turns. Whatever speaker turns model can be applied later, it is necessary to use a set of speaker models. Then the input vector will be compared to each of the models in order to select the closer one to the sample, solving the problem in a classification space. Distances or likelihood functions based on modified distances functions have been classically applied. Kullback-Leibler distance, Generalized Likelihood Ratio and Bayesian Information Criterion has been used, besides models, likelihood ratios or VQ distortion measures for clustering data.

About statistical models for speakers, Reynolds [2] set important conclusions and demonstrated that GMM outperformed other stand-alone techniques such as VQ or Radial Basis Functions. GMMs take into account speaker-dependent cepstral data and discard time variation information, which was represented by transitions in HMMs. Independence of the speech dynamics among speakers is one of the important conclusions all over bibliography. Initialization and training for the models are studied and solved for off-line systems as well as noisy channel issues are mentioned in Reynolds' work. It is important to remember these conclusions all over this work, because they are revisited in some of the incoming methods, and sometimes assumed for other experiments. When HMM were used, and transitions were discarded, models performed better. That means than first-order time-dependent statistics relying on phoneme transitions are more associated to language than to speaking manners. Some other features and temporal statistics of higher order could be used, though first results were very discouraging.

Some approaches use a modified EM training algorithm, taking GMM as base-technique [3]. This technique showed the best performances in cepstral domain, and bad results when applied to other features, thought they are useful in speaker discrimination, as shown when other models are applied. Pitch values, pitch contours, and statistics from energy are best prosodic features for speaker discrimination. Their distribution and contours in [4] showed well stand-alone performance but more important for our purpose was the complementarity of this information with cepstral results.

There are some projects using acoustic pitch. Approach in [4] studies its contours. Parameters such as logarithms of F0 maximum, minimum and mean have demonstrated to be useful [8] as well as its distributions, normally assumed tied Gaussian. However, its time dependent contours reveal the most of the information. [4] estimates contours using the piecewise stylization of pitch, adopted by NIST as relevant feature [7]. Behavior information is encapsulated in bigrams of increasing, decreasing and voiceless segments of speech. [5] presents a different way of modeling pitch and energy. Indifferently of how pitch and energy are used, baseline demonstrates usability of those voice characteristics, still not sufficient to avoid MFCC-GMM collaboration, but its results are remarkable.

Critical Review

GMM is the best system found on literature for modeling speakers in the acoustic domain. It uses cepstral coefficients extracted with MFFB which are more robust to noise than LPC. It assumes Gaussian distribution for cepstral coefficients in different realizations of same utterances, and uncorrelation between coefficients. Time variation is omitted in these models because of its independency from speaker and higher relation to the language. It can be empirically demonstrated that models with higher numbers of uncorrelated added Gaussians can accomplish similar performances than a lower number of correlated mixtured Gaussians. Then in this work, the inter-correlation of the Gaussian coefficients will not be considered, so the major parameter affecting the complexity of the models will be the number of Gaussians. A stand-alone system with GMM presents three basic problems; order selection, EM initialization and calculation. Basically, those are specific application problems less important than our main goal; but getting solutions for them achieving similar performances with lower demands in terms of training processes or offline necessities would also be a remarkable result in the scope of the current project. Optimum search of minimum error estimation with EM is guaranteed when initialization of the model is correct, but finding minimums is not synonym of good representation of speakers. Initialization and order selection are highly

important, to avoid performance decrease and misclassification. These constraints force applications to get its speaker models trained offline with some good speech information, though final application would involve hard noise conditions.

Issues related to on-line behavior of the system are still being studied. One of the approaches [6] consists on creating a wide database of speaker models, which will be picked, assigned and adapted to speakers in real time. It's based on speaker-change detection and later classification. As long as online model adaptation is not one of the aims of this project, a separately trained database will be assumed. The commented reference discusses about an interesting and assumed fact: turns over-detections can be corrected by posterior clustering or turns modeling, but misdetection can not be repaired later.

Accuracies on low SNR conditions are only mentioned as future work. One could just find comments at the end of the cited references approaching high noise conditions. Cepstral mean subtraction is the solution given by Reynolds on GMM-based works. Anyway, there is not a study about which features are more robust to noise, in terms of Speaker recognition.

Acoustic pitch underlies in cepstral coefficients, but its temporal behavior is not considered on static models such as GMM. Combination of these features with other speech-based classifiers is still a baseline in current literature.

Pitch database used has not low SNR conditions, which we are interested in. The correct extraction of pitch from these constrained audio data can be an obstacle for future applications not being analyzed. Variances for the length of pitch and energy contours models are also basic problems in the previously commented work. Stronger extraction methods against noise and time-variations exist and have not been used, they will be analyzed in this work.

Transcription of text could maybe solve these issues, as some studies have tried; with the addition of the STT system: suitable to errors and with higher computational requirements. Approaches on text-transcription used for speaker

recognition are present on bibliography. One could find usual speaker dependent expressions in speech, word rates, pause rates and some other prosodic information. Authors of the current work think it would be better to solve the problem in a simpler way; specially avoiding those high requirements.

Current study follows up the experts' room project, considering a text recognition system may be added to the multimodal system in future, but for the moment, this system and its possibilities in speaker recognition will not be considered. Then these items will follow; first try to extract the best information from feature level, later from larger term features and finally from conversational-length characteristics. In other words; acoustic, word and sentence-level features and conversational turns.

Conversational turns have been studied too. In [7], a theoretic approach to usability of conversational statistical parameterization showed positive results, but complementarity of this performance with previous systems is not clear.

Later on this episode, multimodal discussion is the main theme; there are solutions and ideas about how this information and previous one can be joined. Extensive database is needed for speakers' turns modeling. The best parameters seem to be static ones, used when there are not wide databases available or when systems can work on-line. They are turns length, #pauses/turn… a unigram is sufficient to model these parameters, bigrams should take into account dynamic properties for turns, which are considered meeting-dependant and then not usable. It will be important to disguise which information extracted from the turn modeling is not dependent on people's mood or on meeting themes.

Other contributions to speaker recognition comprise high-level information such as characteristic pronunciation of phones or characteristic/incorrect lexical use of words. These features are extracted from phonemes segmentation and transcriptions. Good results have been estimated for the union of these features into basic systems [5]. It is still presented as just an approach to demonstrate complementarities of information from different context; acoustic, prosodic,

phonetic, lexical and conversational. It takes into account too many parameters for on-line computation and necessarily uses lots of redundant information, but as long as it's presented like a basic approach throwing good results, it's remarkable. Huge feature vectors used for high-level studies such as [5] or [8] result from an elementary combination of shorter vectors; combination should be studied in depth to improve performances or to save requirements.

There are no studies about articulatory features applied to speaker recognition [7][8]. So it's a good start point for research.

Before leaving this section, considering all the read bibliography, it can be assumed that larger speech segments become more reliable for feature extraction. When thinking about posterior usage of segments for classification, segments should necessarily be shorter than the smallest speaker turn. If 1 second is considered the shortest of the speaker turns, 1 second should be the length of the speech segment to be extracted for analysis.

On the methods section, the drawbacks found in the current "state of the art" in speaker recognition will be transformed in experiments and simulations. The main conclusion of the review about feature extraction has been the lack of usage of articulatory features. See the corresponding section of Methods, called ARTICULATORY CLASSIFICATION for further details.

Robust Feature Extraction

No relevant studies have been found about noise effects on the feature extraction methods. These studies would answer questions such as; which of our features are most affected by noise? Rejecting weak features would increase or decrease the performance? Is there any feature extraction method more resistant to noise?

Reynolds in [5] showed cepstral mean subtraction increased performances on high noise conditions, and one of the methods of that study demonstrates the

reason for that, and explains why some other researchers have proposed noise removal on the MFFB. On bibliography and further, the number of cepstral coefficients used is reported as an accurately chosen design parameter involving robustness, with no special mention about the election method.

Critical Review

The mentioned literature, lacks an analysis about the behavior of the different audio features when contaminated with noise. The robustness of a system against noise is normally reported using results, and no specific pre-analysis is made. Systems are not designed thinking on predefined noise rejection ratio; when a system outperforms some other, then it is backwards studied in terms of noise and possible reasons are explained. Although designing specially focusing on noise rejection is not a good strategy, maybe there are important conclusions on the approach, or maybe implications about how features and extraction methods behave against noise show up.

On the methods episode, addressing this discussion, these questions are to be approached;

- Given a group of features from voice, can we define its behavior when noise is added? (For example, would likelihood borders be affected because of SNR? Or how much affected are features by noise?)
- Given a group of methods for extracting the same set of features, which is the best method against noise?
- Given a vector of features, which of them are more important for the speaker recognition system (there exists a low dependency between features on the model)? And which of them are more/less robust to noises?
- A model or some models for noise are needed all along this study. Can we define a good one addressing the meeting room scenario?
- Can we compare different feature extraction methods using a noise rejection measure?

And linking with the review of the previous section…

- Are articulatory features more robust to noise compared to MFCC?

In the Robust feature extraction subsection, inside the methods section, some simulations, hopefully answering the previous questions, are explained. This mentioned subsection will be responsible of the articulatory feature extraction baseline and methods. Some discussion about the path to follow in the project is there developed, though this could be the right place for it, the arguments expressed are considered part of the progress of the project.

Multimodality

Combining information from different sources has demonstrated an improvement on information systems. There are on literature two ways to integrate multimodal information. They combine information at feature level or at semantic level, meaning they mix features before using them in a classifier, or they are classified separately and outputs of these classifiers are used. Mixed feature models would take advantage of all the information available on input parameters. On the other hand, using the outputs of previous well-known subsystems could mean easy implementations with similar results. Anyway, they both obey front-end processing and feature extraction of source signals and try to decide from source parameters. So the multimodal systems deal with designing joint models for features from different sources and problems such as modular design and information bottlenecks at the output of subsystems.

Some approaches of feature level multimodal integration have been using statistical techniques like Hidden Markov Models [9], Neural Networks (MLP, RBF, DBNN, MOE) and Temporal Neural Networks [10] (TDNN, SRN, BPTT). They provide good solutions and an easy human interpretation for sources with close relationship between them, as speech and gestures relations. One of the multimodal combination problems studied in depth is the speech and lip movement interaction [11], which is usually solved from feature point of view, because the correlation between this information is high. Less nice relations can be found in other problems, solved at semantic level. These systems work with processed information which is expected to be less capable of reliable

solutions, and accuse the lack of test databases, while feature level systems require computation power and have scalability problems.

Semantic-based systems must combine decisions to “solve subsystems’ errors”. Used techniques have been Neural Networks [11], Dynamic Bayesian Networks [12][13] or some kinds of probabilities’ weighting solutions. There are many systems on literature combining sources information but they just solve their problems and it is done in very different ways. Owiatt [14] tried to set lower and upper bounds for subsystems combination efficiency and measure the disambiguation between sources, where the lower bound equals to separate decisions and upper bound can be calculated from sources inter-correlations. These are not available from the start point but it can be approximated by training.

Along bibliography [9]-[16] it is known that appropriate systems for feature-based integration would take strongly correlated input parameters and a wide training database. If softly correlated input signals are present, or there's a lack of training database one should use other solutions. Known problems in both techniques are: determining relevant features to be extracted from the input signals and the synchronization of these sources. Multiple articles discuss features relevance or their joint behavior for specific problems, even one could find studies about sources’ correlations. Being not aware of design rules for integration, it seems to be useful to take advantage of previous experience on subsystems design, and set up some procedures for selecting the data at its outputs and combining it.

Critical Review

Although exist a lot of papers about multimodal combination, there are few of them studying the problem from a theoretical point of view. A theoretical approach would be very useful for setting some high level rules to deal with multimodal problems. Few conclusions are given by papers about using semantic or feature level systems, and the classical product or sum rules are

not suitable for systems where the whole performance is expected to reach its best because of the multimodal integrator.

There are also few explanations about empirical results trying to analyze which systems perform better with which solutions. Some simple problems could be demonstrated to be best solved by one method instead of using some others. Neither a specific application has been demonstrated to be outperformed with one specific method. Given the classification nature of these problems, most of the methods used acquire previous (from previous subsystems) likelihood responses to achieve new classification spaces. That’s for sure a mathematical non-sense working because of the previous good behavior of classifiers, but theoretically inconsistent.

The peculiar nature of the data in the meeting room scenario brings up a non classical problem. Final system will be receiving data from other systems running on different machines, each of them reaching misdetection and over-detection errors continuously, and indeed system falls. Finding about similar projects was an impossible objective, and apply classical digital signal processing algorithms has been tricky.

Before proceeding to apply a method to the specific problem, the mathematical background of the multimodal issue will be discussed, and the results will follow. Although this work is not pretended to be a mathematical theory, the formulation of the problem will guide to an easier understanding of the results of the experiments.

2. Methods

2.1. Goals

The global system approached in this work will combine the results of specific subsystems: these subsystems are: people's localization, sound source localization and voice classification over speaker models. This information is clearly separated into two spaces; physical one; where sound source localization and people localization relies; and the speech space, where spectral information of voice is used. Then there are two clustering results; based on spatial and voice decision boundaries. For sure, voice data is not giving any information to space classification, in terms of likelihood or distance functions used, meaning; distances used for each classification are completely independent. The variation along time of the classification weights is more suitable to be used for the combination purpose. Although they will not be synchronous, because time slots used for classification will be pretty different, the prior probabilities of each input vector related to a specific class in every space will be functions behaving similar along time. Then it's obvious to assume a semantic level combination of these two results. Each subsystem will throw its likelihood or distances vector to the multimodal integrator, defining the main scheme this way:

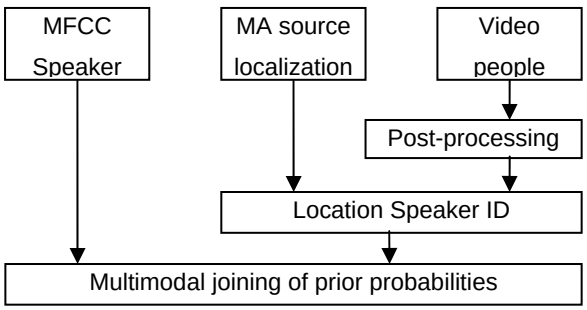


Figure 1.- General scheme

A feature-level combination of the system is not considered. Features of the system are; the MFCC coefficients, the cross-correlations of the microphone array and the video localization shapes. As stated, it is nonsense to combine

MFCC with other features. However, the estimation of the microphone array delays and the video localization is also theoretically tricky. There are several algorithms for the source localization based on microphone arrays. The mostly used are the ones based on correlation.

Once the main scheme has been stated and discussed, this ‘Goals’ episode will be divided into 4 categories involving different parts of the previous scheme. ‘Speaker Recognition’ and ‘Robust Feature Extraction’ will deal with the speech space clustering, each of them carrying out the task of classifying a set of input vectors and extracting them, respectively; ‘Theoretical multimodality studies’ and ‘Multimodal speaker recognition’ will approach the space classification and the high level combination, concerning theoretical and implementation specific issues.

Speaker Recognition

BASELINE

As stated before, this chapter is responsible of classification. GMM is the most used model, then the baseline of this work will be a first approach based on it and on MFCC extraction. Some simulations will be performed to accomplish these goals:

- Familiarize with the MFCC extraction method, evaluating the parameters of the extraction method, how they affect the final performance.
- Use the GMM models, working on its parameters and training methods
- Create a background environment to work with in the next simulations.

The NIST database will be used for the first baseline. It is a meeting between four speakers; two male and two female, and one unmiked component:

Topic:	news gathering scenario 1
Type:	focus group discussion
Date:	29/07/2003 - 15:13
Duration:	23 minutes

Participants: 5

Unmiked Participants: 1

Bleeps: 0

Location: NIST/225/B243

Room Microphones:

Name	Type	Stats	Notes	Location* (m)
ARRAY-1	Array	OK		(2.88, 0.00, 1.20)
ARRAY-2	Array	OK		(-0.40, 3.26, 0.40)
ARRAY-3	Array	OK		(4.54, 6.60; 1.20)
OMNI-1	Table	OK		(1.51, 3.26, 0.74)
OMNI-2	Table	OK		(3.30, 3.26, 0.74)
OMNI-3	Table	OK		(4.40, 3.26, 0.74)
QUAD-1	Table	OK	Direct. SE	(2.95, 3.26, 0.74)
QUAD-2	Table	OK	Direct. NE	(2.95, 3.26, 0.74)
QUAD-3	Table	OK	Direct. NW	(2.95, 3.26, 0.74)
QUAD-4	Table	OK	Direct. SW	(2.95, 3.26, 0.74)

Table 1.- NIST Database. Room Microphones

Microphones used by the participants

Name	Type	Subject ID	Status	Notes
HM-1	Head	25	OK	
LM-1	Lapel	25	OK	
HM-2	Head	27	OK	
LM-2	Lapel	27	Corrupted	Problem, no signal first few minutes
HM-3	Head	40	OK	
LM-3	Lapel	40	OK	
HM-4	Head	19	OK	
LM-4	Lapel	19	OK	

Table 2.- NIST Database. Microphones used by participants

Subjects

Subject ID	Gender	Native	Notes
25	F	Yes	
27	M	Yes	
40	M	Yes	
19	F	Yes	
6	F	Yes	Unmiked

Table 3.- NIST Database. Subjects

Artifacts

Artifact Name	Location* (meters)
Projector Screen	(-0.40, 1.70, 0.74)
Whiteboard	(2.52, 0.00, 0.90)

Table 4.- NIST Database. Artifacts

There is a considerable amount of information on this meeting database. Spatial information will be discarded as well as delays on the microphone array. The aim of this particular step is to get familiar with a noisy speech database. There are three omnidireccional microphones and lapel microphones for each speaker. The speaker turns and speech are transcribed. The environment is very noisy, so it is perfect for the goal of the thesis.

The current baseline should set conclusions on basic questions such as:
- The correct selection of training data and simulation data, in terms of microphones, which involve SNR conditions and channel filtering.
- The correct parameterization of the MFCC, particularly how the number of filters variation performs, selection of the first DCT coefficient, length of the utterances….
- The selection of GMM parameters; number of Gaussians and performances in front of different input features and training situations, testing the training algorithm exhaustively
- Methods to model the background, in order to add the silence as a new model into the speaker's group.

To accomplish these goals, evaluation methods are needed. These are the suggested and used along this work:

1.- ACS. Approximately classified segments

When a timeline is segmented into different speaker turns, each instant belongs to a speaker, assuming there are not overlappings. Along time, a classification is performed in each cluster/period, due to signal analysis issues every decision is based on information belonging to the current period and the surrounding data, because the longer the speech segment is taken, the better the classification works. The upper time axis in the next diagram shows that issue:

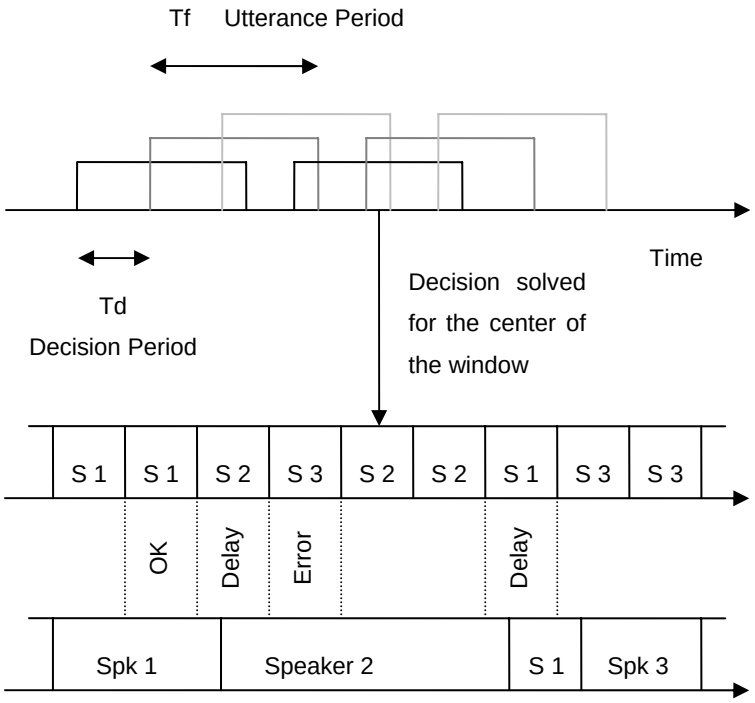


Figure 2.- Evaluation problem

The utterance period Tf is the length of the window used in the time domain to extract speech features', and the decision period Td is the time stepped to move the window. A decision is taken when a speech window is settled in a time instant; a classification label is applied to the result. Note that the classified period is Td, so the solution of the algorithm will be compounded by labeled periods of Td seconds, as showed in the second horizontal axis of the drawing above. Considering that timeline the final decision of the system, how to evaluate its performance?

Advancing some of the preliminar simulations, one second seems to be the shorter utterance period to perform a valid decision. Larger periods would exhaustively take periods shared by two speakers, and shorter ones do not behave statistically the way stated models do. Otherwise, there are not important conversation turns shorter than one second, it has been demonstrated than significant utterances for a conversation are longer than one second or at least they have large pauses before and after when important. This reason brings up an upper bound for Td. Speaker models are based on the statistics of spectral characteristics or voice source parameters, and 50 ms is

the average time vocal tract needs to emit a phoneme; so we use it as a lower bound for Td. So we know Td must be in between 50ms and 1s. Values from 50 ms to 500 ms would imply; redundancy of the decisions, the need for post-processing when isolated turn errors appear; and a great precision on speaker changes. Finally: decision period Td will be considered half a second, though shorter values for redundancy will be simulated to evaluate its contribution.

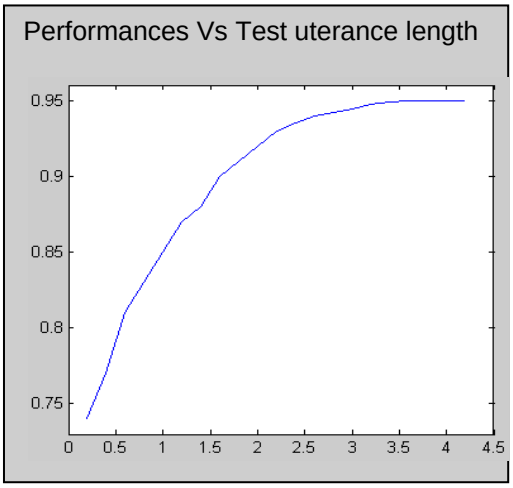


Figure 3.- GMM Performance Vs Test Utternace Length

ACS evaluation algorithm doesn't mind about speaker change detection accuracy. If a time segment is classified into a speaker model present in some part of the window, individually or overlapped, it's taken as a correct classification, as showed in this scheme:

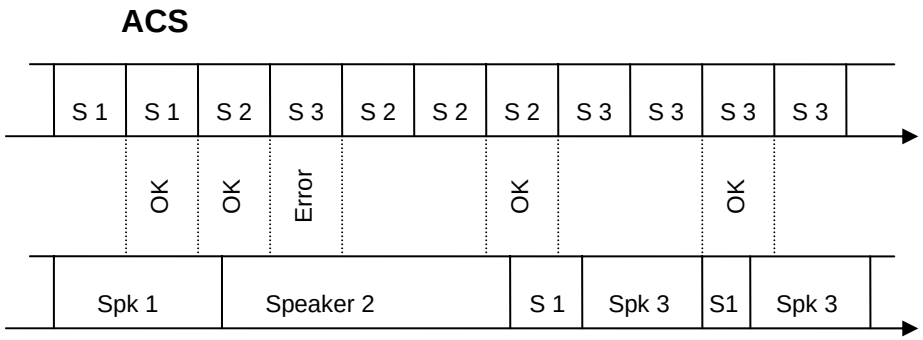


Figure 4.- ACS Evaluation

2.- ACT. Approximately classified Turns

Similarly to ACS evaluation method, ACT will not mind about exactness of speaker change detection but missing a speaker turn will be considered an error. Where ACS missed turns because of its relaxed evaluation measures, ACT is stricter and throws an error.

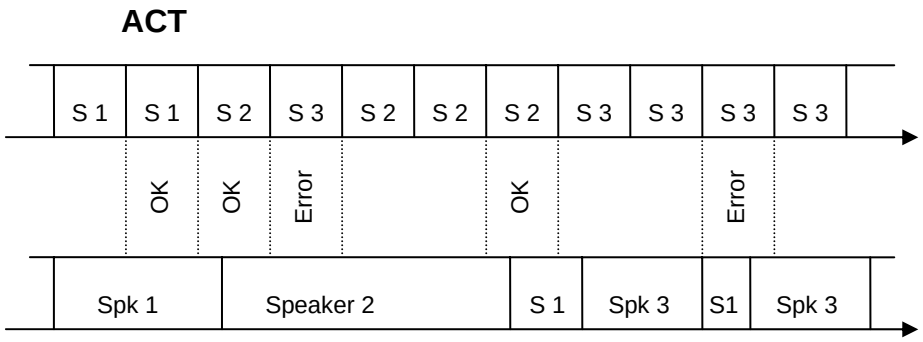


Figure 5.- ACT Evaluation

These errors are supposedly avoidable with the correct selection of Td and Tf, then ACT will be used to be compared to ACS and extract conclusions about the selection of these parameters.

3.- CCT. Correctly classified Turns

Taking a look to the GMM approach and the ML algorithm, it is supposed on the segments where 2 speakers are present, to throw a decision belonging to the most present of the speakers. Then CCT will evaluate the shared segments to belong to the speaker present on more than 50 % of the segment.

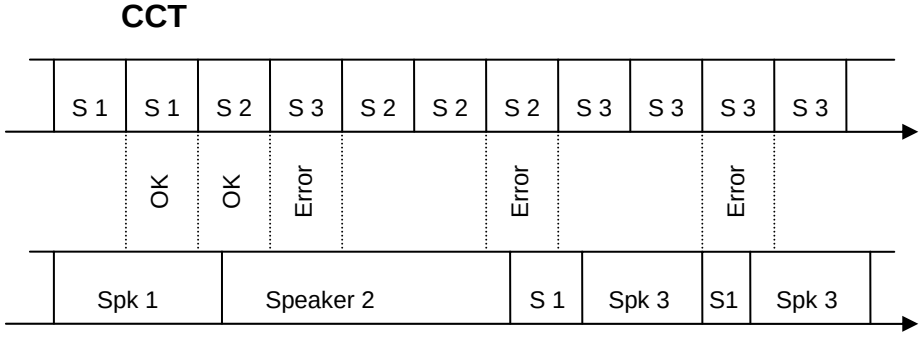


Figure 6.- CCT Evaluation

The overlappings will be also computed using a percent. As an example, if the first half of a segment belongs to the speaker 1, and spk1 and spk2 are overlapped on the second half, then 33% of the segment would belong to spk2 and 66% to spk1. Correct decision for CCT would be spk1 because in the ML domain; probability of spk1 given the utterance is greater or equal to 50%.

4.- CCTS Correctly Classified Turns with Silence

Based on the CCT method, this evaluation method will not extract silence period from the speech signal, taking it as a new model to be classified with the same accuracy than others.

Previous work on speaker classification was working on perfectly cut speech segments. The models were applied to segments of voice belonging to just one speaker. Online requirements forced us to set these performance measures, which will let us detect problems such as little delays of the goal segmentation, correct selection of parameters, exactness of the speaker change detection, and exactness of the silence model. Another problem is the large delays between the classification labels, and the real speech signal. These errors are present on some databases, and are not necessarily continuous along database, that means, there are errors on the classification goal. This will necessarily be studied manually and exhaustively.

REAL SCENARIO

Once having data relative to the Real Scenario, new Simulations with different goals should be run. Special care should be taken to accomplish these goals;

- Extract conclusions about the location of the microphones and the preprocessing necessary for the correct classification
- Conclusions about model training methods.
- Analyze the background model performance.

ARTICULATORY CLASSIFICATION

The next goal of this section will be using new acoustic features to classify speakers. Again, two baselines for the NIST database and for the real scenario will be stated and discussed.

The methods for extracting the features will be commented in this section, but further explanations are in the Robust Feature Extraction section.

Once the baseline gets working, how to improve its performance will be studied and how to join it into the GMM model in a multimodal scheme. The basic plan for the models is to start with static models, like GMM, and move to dynamic ones like HMM. It's well known in bibliography than major differences in pitch between different speakers are basically the temporal variation, nor the instant values. That's why every feature will be specially considered and studied about this concern, trying different approaches like static models, models based on derivatives, models based on temporal variations and some mixtures.

In the Results section, the Speaker Recognition::Baseline subsection contains the explanation and results of the run simulations about MFCC (Simulation 1 and Simulation 2). Further related studies on this topic, is in the Progress of the project section, inside Requirements::Speaker Recognition. Simulation 5 and Simulation 6 are the baselines for the articulatory usage. See Requirements::Speaker Recognition for other articulatory approach explanations.

Robust feature extraction

Speech signal

The accuracy of the systems decay fast as noise is added to the speech signal. Next plot shows it, it responds to a preliminary simulation with the same corpora being contaminated with white, colored and real noise.

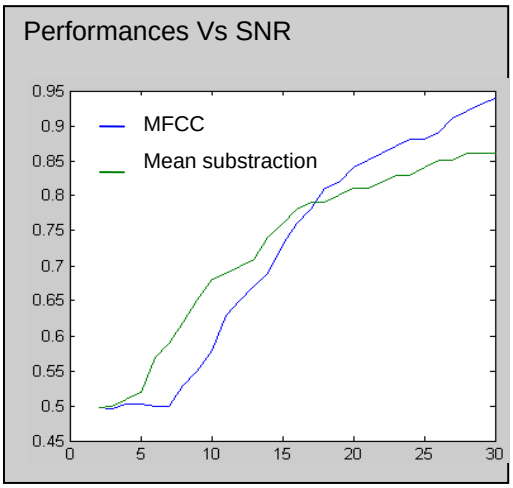


Figure 7.- GMM Performance Vs SNR

Speech has been traditionally modeled as the sum of a source from the vocal cords and the spectral filtering created by the vocal tract. Two major approaches to model speaker dependant features have based the classification problem on the source or on the tract characteristics. In terms of spectral shape, MFCC are the most commonly used features. Its properties has been largely analyzed and exalted by speech analysis lectures. Otherwise, when using source characteristics, pitch and energy are the important features; speaker dependency has been demonstrated in the way pitch evolves on time. However, the speaker ID problem is usually solved with statistics of the spectral shapes, which need long test utterances and high SNR. Since that method is considered in the limit of its capabilities, addition of new features is the way to improve it. New features can income from a similar nature or from a very different one, such as voice source localization. This section is based on extracting the most from the sound signal.

It's hard to find features capable to represent the acoustic characteristics of an utterance better than MFCC, so the most of the researches focuses on prosodic information. Before that, an analysis of robustness of MFCC against noise (please view next section) could throw some light over the problem.

The aim of this section is to work on the MFCC avoidance, and extract other source features which could contribute with information different from MFCC. The MFCC have been used for speech recognition, since the human ear works in the spectral domain, then the human voice tries to reproduce sounds distinguishable by the human ear. There is a strict true underneath the whole process; the different sounds of the human voice are classified by the speech recognition systems using the same methods the human ear uses; spectral shape, and it's surely the best way. But in terms of speaker recognition that is completely erroneous, meaning; the human voice is not trained to be different than other voices; and even worse; it is trained to be equal and understandable. So when looking at MFCC for the speaker recognition purpose, we are looking into the small differences users have when creating a sound, that is the resilient error in the speech recognition algorithm. For sure that measures the inability of some vocal tracts to reproduce standard sounds. That is for sure a very good baseline, since the human ear is capable of distinguish the person speaking by using its ear, meaning using a spectral shape analyzer, but doesn't need to be the unique solution to the speaker recognition issue.

Pitch is the main feature that can be used to look directly into the speech generation process. It's a clean feature different among all speakers, genuine without any doubt. But the aim of this project is finding other similar characteristic features on the speech generation process, specially focusing on those ones not captured by the MFCC algorithm.

Looking for the vocal tract bibliography and speech generation process lessons, the speech is known to be generated differently if a voiced or unvoiced phoneme is being pronounced. The voiced segments are modeled as a source followed by a filter, where the vibrating vocal chords and the vocal tract are responsible of each role respectively. The MFCC catch the filtering process successfully, filtering is always considered linear because second order components are considered inexistent. The unvoiced phonemes are commonly assumed to be generated by some source coming from the lumps, and the vocal tract generating some temporal predefined events. That reveals the need of using something like the HMM transition modeling in the speech recognition

problem, which joins the GMM to accomplish the recognition of the whole phoneme corpora.

By using the mentioned ideas, which are a brief description of a very large theory, one could assume than the only information included in the human voice that is not firmly captured by the MFCC algorithm is: the source model, the non-linear effects of the vocal tract filters or the characteristics of the high-frequency components of voice. Source models can be approached using pitch extractions, or calculating background noise models using the mean of the unvoiced segments. The non-linearity of the vocal tract is minim, and the catch of that kind of events needs a correct modeling and understanding of the non-linearity order, and furthermore; online extraction. The high-frequency particularities are not captured by MFCC since they are focused in the human ear listening procedure. The Mel-Frequency is based in the loss of spectral accuracy of human ear as frequencies are higher. That is a very logical escalation of the filters for the speech recognition, but as mentioned, the speaker recognition is different, and the human ear behavior has not to be copied. The particular resolution scale used by MFFB from low to high frequencies is to be analyzed

Simulations in the Results section will try to answer some of the approached questions, using pitch, energy and some other articulatory features, and see if plain usage reveals hints for next steps.

Noise on MFCC

In Figure 7, the performances of a speaker ID system against additive Gaussian are plotted[1]. The same graph adding real world noise or conversation interferences was plotted during the progress of this baseline and the results

[1] Classification performance on the USC1 database, 14 MFCC coefficients (first discarded), 16 gaussian per model, silence periods discarded

were similar. It is obvious how the speaker dependency using MFCC is lost when noise or interferences are present.

To evaluate how the noise is merged with the real data in MFCC, let's take again a simple example by adding AWGN to speech signal:

$$y(n) = x(n) + w(n) \quad \leftrightarrow \quad Y(w) = X(w) + W(w)$$

And carrying the new signal through each step of signal preprocessor, starting with pre-emphasizer filter;

$$y(n) = x(n) - 0.95x(n-1) + w(t) - 0.95w(n-1)$$
$$Y(w) \cong \left(X(w) + W(w)\right) 2\sin(w) e^{j w/2 + \Pi/4}$$

passing the signal through the filter bank and computing the power at the output means the same than computing spectral power and multiplying per filter bank;

$$S_y(w) = E\{Y(w)Y^*(w)\} \cong S_x(w) 4\sin^2(w) + S_w(w) 4\sin^2(w)$$

the width of the filter i being a total of N filters is

$$\Delta_i = w_i^{end} - w_i^{init} = 2\pi 700(1 - e^{2\psi}) e^{(i+2)\psi} - 2\pi 700(1 - e^{2\psi}) e^{i\psi} =$$
$$= 2\pi 700(e^{i\psi} - e^{(2+i)\psi} e^{2\psi} - e^{i\psi} + e^{(2+i)\psi}) = 2\pi 700(1 - e^{2\psi}) e^{(2+i)\psi}$$

where

$$i = 0 \ldots N-1$$
$$\psi = \frac{\ln(1 + \frac{w_{\max}}{2\pi 700})}{N+1}$$

we want to evaluate how the noise goes through the filter bank. The amplitude of the noise can be approximated in each filter as the amplitude in the center of the filter:

$$W(f_i^{center}) = 4\sin^2\left(2\pi 700(1 - e^{2\psi}) e^{(i+1)\psi}\right)$$

Then, at the output of the filter there will be;

$$P_i = \Delta_i W(f_i^{center}) = Px_i + Pw_i =$$
$$= Px_i + 2\pi 700(1 - e^{2\psi}) e^{(2+i)\psi} 4\sin^2\left(2\pi 700(1 - e^{2\psi}) e^{(i+1)\psi}\right)$$

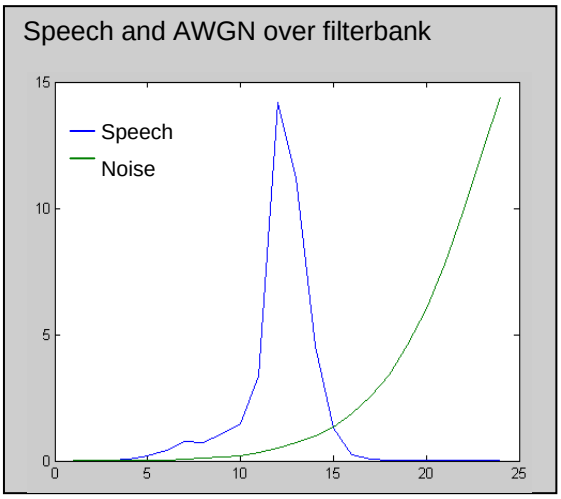


Figure 8.- Speech and AWGN distribution over MF Filterbank

Applying the logarithm:

$$\log(P_i) = \log\left(Px_i + 2\pi 700(1-e^{2\psi})e^{(2+i)\psi}\,4\sin^2\left(2\pi 700(1-e^{2\psi})e^{(i+1)\psi}\right)\right) =$$

$$= \log\left(Px_i\right) + \log\left(1 + \frac{2\pi 700(1-e^{2\psi})e^{(2+i)\psi}\,4\sin^2\left(2\pi 700(1-e^{2\psi})e^{(i+1)\psi}\right)}{Px_i}\right)$$

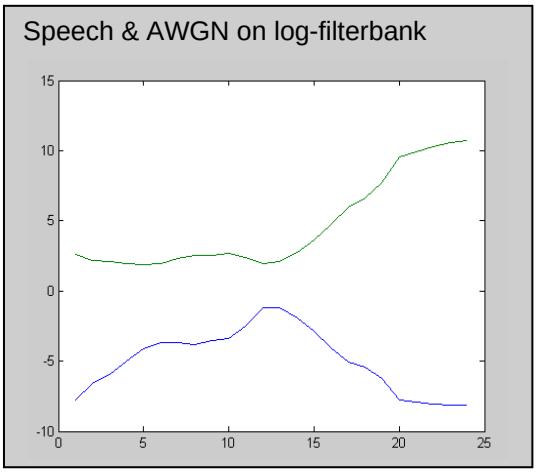


Figure 9.- Speech and AWGN distribution over log-MF Finterbank

The last step previous to obtain the cepstral is the DCT:

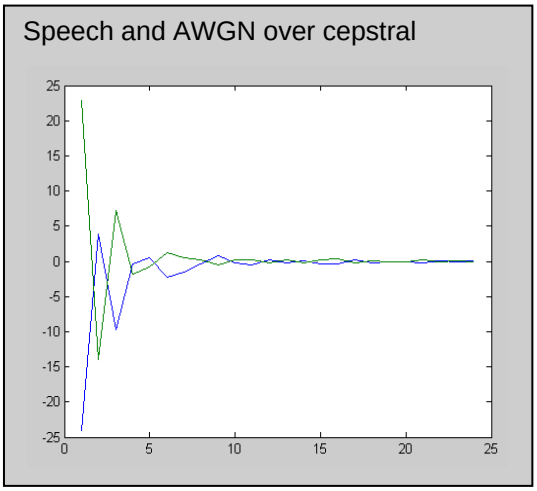


Figure 10.- Speech and AWGN distribution over cepstral coefficients

The plots show how the power of noise is distributed in the same way as the power of signal is. This compacted distribution is the great advantage of DCT, but it makes impossible to separate useful information from the noisy data in this domain. It is not that way in the previous step, before transformation. Logarithmic outputs of filterbank have more power when noise is added in high frequencies, where speech power starts to decay. Having a vector with a reliability scale over its coefficients, would give the possibility of discarding some coefficients on hard noise environments. That's not doable after the DCT, and DCT is one of the most important steps for classification.

These are the suitable modifications of the system revealed by the study:

-Take out the high frequency coefficients of the log-filterbank

-Find a transformation different than DCT mapping energy of noise and speech to different coefficients could lead to a similar situation in the transformation domain, then coefficients with higher SNR after the transformation could be discarded.

First solution works on spectral domain, and second one on cepstral domain. Two simulations will be performed in order to set a baseline on these solutions:

1.- Discarding of high frequency coefficients

From previous plots, high frequency coefficients are demonstrated to suffer lower SNRs. Then, it's intended to compare how its avoidance performs in front of its use. The general scheme of the simulation will be:

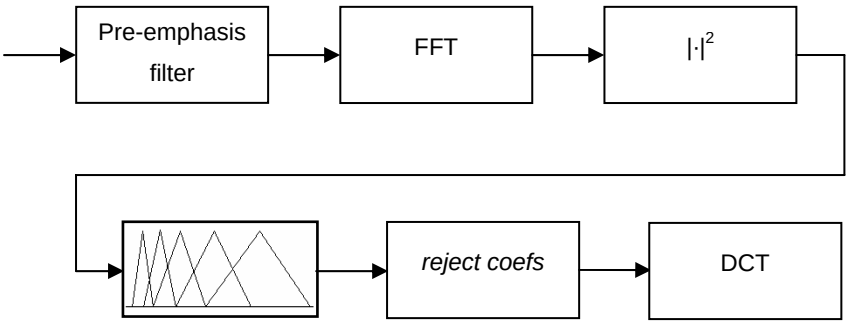


Figure 11.- Possible scheme for noise reduction

After the DCT, Gaussian mixture models will be applied. As long as GMM are working when all the coefficients are used, GMM are supposed to perform similar when some of them are discarded, no other classification methods will be considered. The baseline stated for speaker classification (Simulation 1) will be taken to evaluate this simulation.

Since the particularity of speakers given frequency information exists for spectral shapes and not simply spectral values, the histograms of log-filterbank don't show any speaker dependency. Only specific interlocutors having big differences such as pitch, the classification could be performed. The DCT or any other transformation is needed to locate the differences.

After the transformation, these are the histograms for a pair of speakers taking 14 mel cepstral coefficients with energy included:

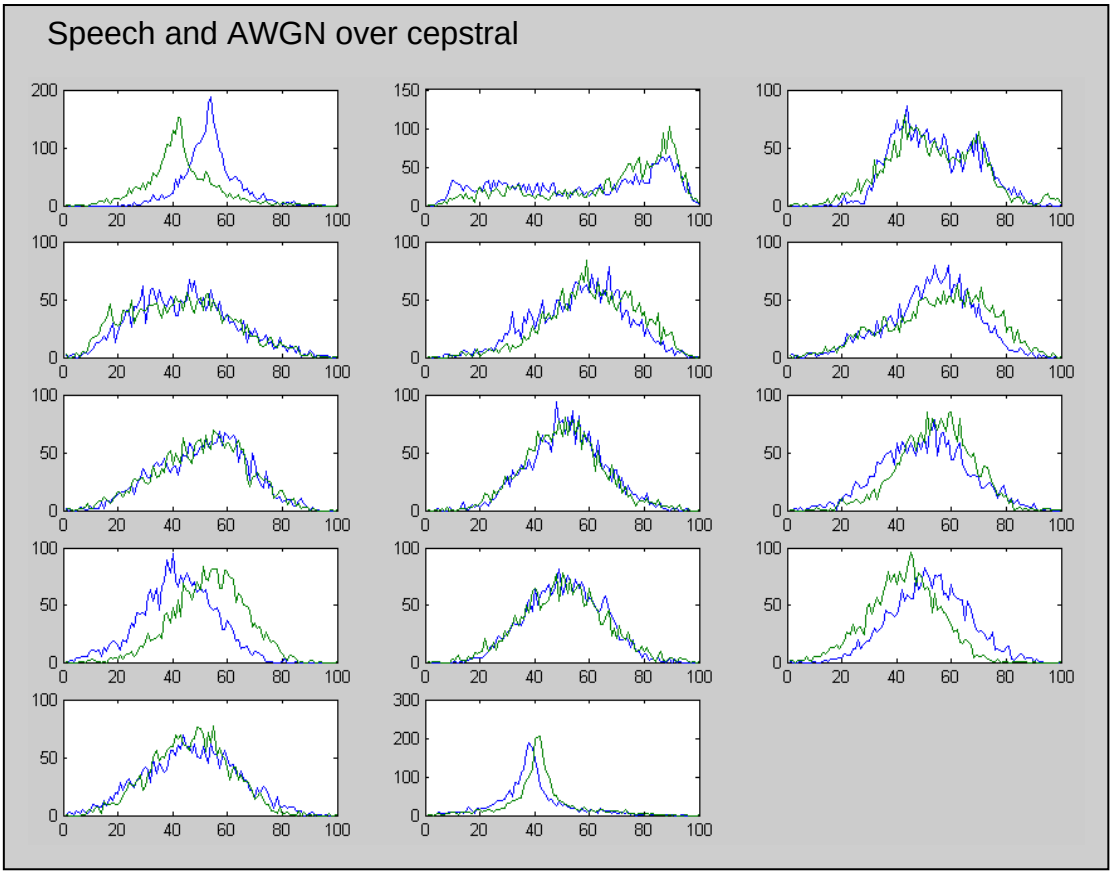


Figure 12.- Speech and AWGN distributions over each cepstral coef.

Green color has been used for the histograms of signal with added noise. Because of the non-linear operation; the logarithm, it's impossible to disambiguate PDFs shapes of signal and noise. The results end up with limited possibilities when trying to get cleaner MFCC coefficients.

Simulation 9 in the Results section will play with the MFCC algorithm and the mentioned distributions of noise so proposed solutions can be analyzed. After those simulations, the Progress of the project will indicate how to continue the study.

Articulatory features

There are different methods to accomplish the extraction of the articulatory features. Pitch, energy, voicidity, rounding, place, open, manner and point

(low/medium/high) are the considered features. Since bibliography is full of methods for the pitch and energy calculation, and the performance of these algorithms are good enough to train speaker models, the pitch and power methods will be explained and compared, and later its performance examined for high noise constraints. No new methods will be approached in this work. Otherwise, other articulatory characteristics have not been widely studied, neither extraction methods. First goal of this section is to extract those features with a simple algorithm, and later when the features get a classification result the extraction method will be reviewed

For the pitch extraction, the KEELE pitch database is used, where we can find transcribed valid values of pitch for training and testing the pitch calculation algorithm. These methods are based on autocorrelation, maximum likelihood, spectrum based product or the YIN method. Being the pitch the more powerful periodic signal in the human voice; a simple filtering, correlation and peak search algorithm is enough for an accurate extraction, and first results showed it being sufficient. However, to evaluate and compare the methods, we could use a simple squared error measure. Being x the expected values of the pitch, and y the resulting values for the pitch:

$$SE = \sum_{i=0}^{N-1} (y_i - x_i)^2 \quad where\ i \in voiced$$

For those pieces of signal where the pitch can’t be calculated because of unvoicidity of the phonemes, the error will be avoided. There are some considerations about the effects caused by small delays; next plot can help to understand it:

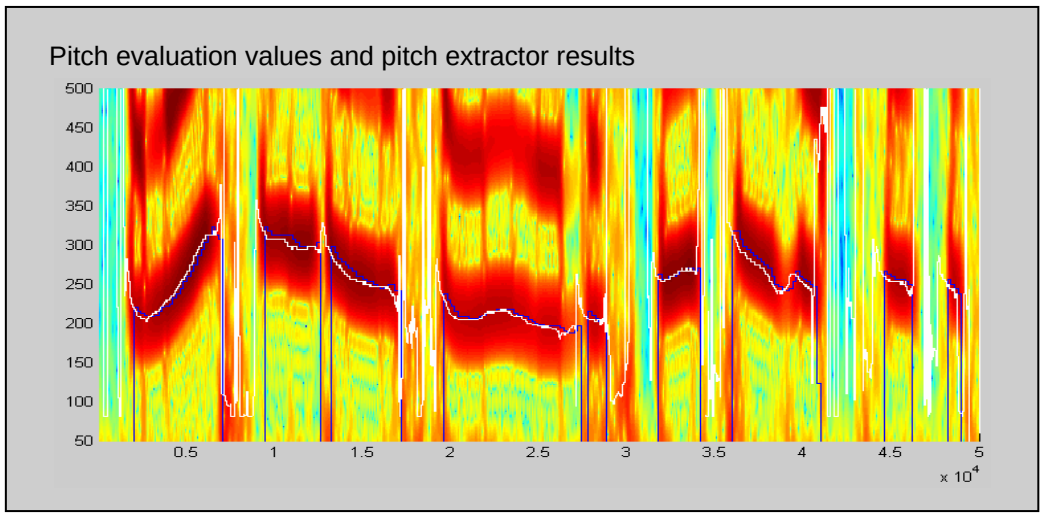


Figure 13.- Pitch evaluation values and pitch extractor results

This plot is extracted from the results sections; it shows the spectrogram of a piece of voice, with the pitch values saved in the KEELE database in blue color, and the values from the pith extractor in white color. Discarding voiceless pieces in order to keep the performance measure calculated with only voiced segments is plainly done by reading the zero values of the KEELE's pitch. But in the areas close to VOICE-VOICELESS changes, there's a small delay between the expected values and the real values. Then we can analyze the accuracy of the systems with two measures; let's call them SE (Squared error) or CSE (Corrected Square Error). The second one allows a delay without considering it an error, varying from half the length of the analyzing window to the whole window.

The database used for other articulatory features has been the TIMIT database, which was the only free database that was soon available for the simulations. We could find marked phonemes on it, which could be easily translated into marked articulatory features, since every phoneme has unique articulatory characteristics. Once done, first baseline will be based in MLP. To evaluate how good the classifiers are, we will plot the probability of error in every simulation. Basically the MLP is trained to write at its output the value 1 when the input signal is voiced (for example) and -1 when it is not. The amount of voiced phonemes classified with a value <0 and the amount of voiceless phonemes classified with >0 result is the probability of error.

Here we plotted the translation table for the phonemes transcribed in the TIMIT database to convert them to articulatory training/testing vectors. The phonemes are transcribed with the ARPABET phoneme alphabet.

Phoneme	Manner	Place	Voic	Vowel	Height	Round
Consonants						
[b]	stop	lab	voiced	-	-	-
[d]	stop	alv	voiced	-	-	-
[g]	stop	vel	voiced	-	-	-
[p]	stop	lab	voiceless	-	-	-
[t]	stop	alv	voiceless	-	-	-
[k]	stop	vel	voiceless	-	-	-
[dx]	flap	alv	voiced	-	-	-
[jh]	fric	alv	voiced	-	-	-
[ch]	fic	alv	voiceless	-	-	-
[z]	fric	alv	voiced	-	-	-
[s]	fric	alv	voiceless	-	-	-
[zh]	fric	vel	voiced	-	-	-
[sh]	fric	vel	unvoiced	-	-	-
[v]	fric	lab	voiced	-	-	-
[f]	fric	lab	voiceless	-	-	-
[dh]	fric	den	voiced	-	-	-
[th]	fic	den	voiceless	-	-	-
[m]	nas	lab	voiced	-	-	-
[em]	nas	lab	voiced	-	-	-
[n]	nas	alv	voiced	-	-	-
[nx]	flap	alv	voiced	-	-	-
[ng]	nas	vel	voiced	-	-	-
[en]	nas	alv	voiced	-	-	-
[hh]	fric	glo	voiceless	-	-	-
[q]	stop	glo	voiceless	-	-	-
Semivowels						
[l]	-	cen	voiced	-	-	-
[el]	-	cen	voiced	ten	mid	-
[r]	-	ret	voiced	ten	mid	-
[hv]	-	cen	voiced	lax	mid	-
[er]	-	alv	voiced	lax	mid	-
[axr]	-	alv	voiced	lax	mid	-
Vowels						
[w]	-	bak	voiced	ten	hi	unr
[y]	-	fro	voiced	ten	hi	rounded
[iy]	-	fro	voiced	ten	hi	unr
[ih]	-	fro	voiced	lax	hi	unr
[eh]	-	fro	voiced	lax	mid	unr
[ey]	-	fro	voiced	ten	mid	unr
[ae]	-	fro	voiced	ten	lo	unr
[aa]	-	cen	voiced	ten	lo	unr
[aw]	-	cen	voiced	ten	lo	rounded
[ay]	-	cen	voiced	ten	lo	unr
[ah]	-	cen	voiced	ten	lo	unr
[ao]	-	bak	voiced	ten	lo	unr
[ow]	-	bak	voiced	ten	mid	rounded
[uh]	-	bak	voiced	lax	hi	unr

[uw]	-	bak	voiced	ten	hi	rounded
[ax]	-	cen	voiced	lax	mid	unr
[ix]	-	fro	voiced	lax	hi	unr

Table 5.- Phoneme articulatory classification

[voiced] [unv]	**Labial (lab)**	**Dental (den)**	**Alveolar Postalv. (alv)**	**Retrof lex (ret)**	**Vowel (bak,cen,f ron)**	**Velar (vel)**	**Glotal (glo)**
stop	[b] [p]		[d] [t]			[g] [k]	[q]
Nasal	[m] [] [em] []		[n] [] [en] []			[ng] []	
flap		[dx] []	[nx] []				
fri	[v] [f]	[dh] [th]	[jh] [ch] [z] [s]			[zh] [sh]	[] [hh]
-			[er]* [axr]*	[r]*	[l]* [el]* [hv]*		

Table 6.- Phoneme table for consonants

* Semivowels; all voiced

[rounded] [unr]	**Frontal (fro)**	**Central (cen)**	**Back (bak)**
Height (hi)	[] [y] [] [iy] [] [ih] [] [ix]		[w] [] [uw] []
Middle (mid)	[] [eh] [] [ey]	[er]* [axr]* [r]* [el]* [hv]* [l]* [] [ax]	[ow] [oy]
Low(lo)	[] [ae]	[aw] [aa] [] [ay] [] [ah]	[[] [ao]

Table 7.- Phoneme table for vowels

The input of the MLP is a key factor of the system. It can't be a high dimensional vector, and it needs to encapsulate all the necessary speech info. For the baseline, the MFCC or the plain MF Coefficients will be used. The MLP complexity is another key factor, and two layers of 16 and 8 nodes respectively are being used for the most of the features, except for the point of articulation, which needed a higher complexity of the network. The results showed those dimensions were enough, at least for a correct feature extraction, which doesn't necessarily mean the results were good for speaker classification. Simulation 11 shows the results of the MLP baseline.

After the baseline is accomplished, we would like to analyze different methods for extracting these same features, based on speech signal analysis, and trying

to avoid the MFC analysis, which is surely a bad start point when the main aim of this work is to extract different information than the MFCC. In the progress of the project section how to approach that extraction is discussed

Theoretical multimodal studies

In order to present the objectives of this discussion, a simple example will be taken, it will show empirically and analytically the exposition. The example consists on a classification problem. Given two classes, two sources are being measured by two different one-dimensional signal readers, which we are calling modes. Classification could be done in everyone of the spaces, or using a multimodal algorithm. To be analytically easy and to represent generalized issues, measures in both spaces are contaminated by additive Gaussian noise. The aim of the exercise is to achieve the best joint classification possible.

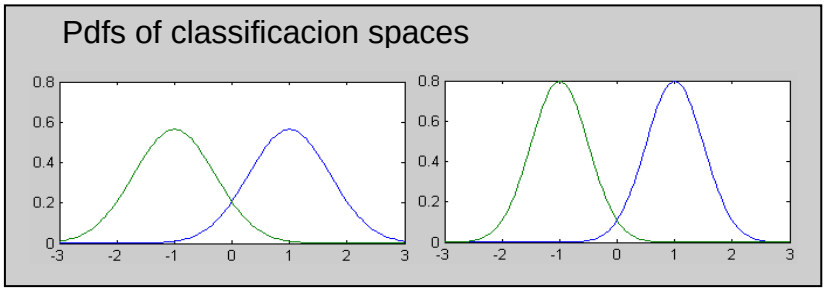


Figure 14.- Simple example for joint classification

It's a multimodality example that can be approached from a semantic level (SL) combination or a feature level (FL) combination. FL uses the input features to solve the classification problem, while SL uses prior probabilities of the features given the classes or any other classification measure to compute a final prior probability.

Let's show how a feature level combination is based on Bayes decision for the given example.
Taking joint probability distribution in both spaces, and assuming it to be joint Gaussian:

$$L_{X_0,X_1}(x_0,x_1\mid\theta_0)=\frac{1}{2\cdot\pi\sqrt{\sigma_{00}\sigma_{11}-\sigma_{01}^2}}\exp\left(-\frac{1}{2}\begin{pmatrix}x_0-1\\x_1-1\end{pmatrix}^{|}\begin{pmatrix}\sigma_{00}&\sigma_{01}\\\sigma_{01}&\sigma_{11}\end{pmatrix}^{-1}\begin{pmatrix}x_0-1\\x_1-1\end{pmatrix}\right)$$

$$L_{X_0,X_1}(x_0,x_1\mid\theta_1)=\frac{1}{2\cdot\pi\sqrt{\sigma_{00}\sigma_{11}-\sigma_{01}^2}}\exp\left(-\frac{1}{2}\begin{pmatrix}x_0+1\\x_1+1\end{pmatrix}^{|}\begin{pmatrix}\sigma_{00}&\sigma_{01}\\\sigma_{01}&\sigma_{11}\end{pmatrix}^{-1}\begin{pmatrix}x_0+1\\x_1+1\end{pmatrix}\right)$$

The Bayes decision taking equally probable classes and based on posterior probabilities is equivalent to the decision of prior probabilities.

$$\arg\max_j\left\{P(\theta_j\mid x_0,x_1)\right\}=\arg\max_j\left\{\frac{P(x_0,x_1\mid\theta_j)P(\theta_j)}{P(x_0,x_1)}\right\}=\arg\max_j\left\{P(x_0,x_1\mid\theta_j)\right\}$$

It will assign the received sample to the maximum of the likelihoods
The Bayes decision rule sets a bound. It's the optimal statistical solution. All other approaches will perform lower than this solution.

There are a couple of questions to be answered about semantic level point of view.
-What can be reached with a semantic level approach? We know that the accuracy will be for sure lower than the Bayesian one. But, can we set a bound lower than that?
-Performance of alone classifiers also set a bound for the solution. Before that bound it's not necessary to use a multi-classification scheme, because there are stand-alone classifiers working better than the given solution. But, can we find a good solution over this bound for a wide range of conditions?
-Is there a method to find solutions to a problem near the upper bound? Is there a method to find solutions far from the lower bound?

Solutions to joint classifiers

The given semantic solution will be some function of prior probabilities:

$$P(x_0,x_1\mid\theta_j)=f\left(P(x_0\mid\theta_j),P(x_1\mid\theta_j)\right)$$

Known classical functions have been product, sum, min and max functions. These classical functions are supposed inefficient and there's no way to analyze which one is the best for each problem. All of those functions are usually tested for the current problem, and the best of them is selected.

We need to define a criterion to establish which functions are closer to the one we expect than others. Integrating the difference between the solution and the reference PDF along all the possible input values, we get a measure of how far are the functions one from the other.

$$SE = \iint \left[P(x_1, x_2 \mid \theta) - f\left(P(x_1 \mid \theta), P(x_2 \mid \theta)\right) \right]^2 \partial x_1 \partial x_2$$

Minimizing this value is a good way to proceed to fit the solution to the PDF. The basic problem is usually the availability of the real PDF. Let's take the simple example: Because of the shape of the density function used to compute prior probabilities, the shape of the approximated function (centered in 0) is attached to:

$$f(x_1, x_2 \mid \theta) = f(-x_1, x_2 \mid \theta) = f(x_1, -x_2 \mid \theta)$$

It can be shown than minimizing SE for a given even-even f function is the same than minimizing the function against the even-even part of the joint density

$$SE = \iint \left[P_{even-even}(x_1, x_2 \mid \theta) - f\left(P(x_1 \mid \theta), P(x_2 \mid \theta)\right) \right]^2 \partial x_1 \partial x_2$$

Then the most appropriate function to extract is the even-even part of the joint density function expressed in terms of prior probabilities. And it is:

$$P_{ee}(x_1, x_2 \mid \theta) = \frac{1}{4\pi \sqrt{\sigma_{X_1X_1}\sigma_{X_2X_2} - \sigma_{X_1X_2}^{\ 2}}} \exp\left(\frac{\sigma_{X_2X_2} x_1^{\ 2} + \sigma_{X_1X_1} x_2^{\ 2}}{2(\sigma_{X_1X_1}\sigma_{X_2X_2} - \sigma_{X_1X_2}^{\ 2})} \right) \left[\exp\left(-\frac{\sigma_{X_1X_2} x_1 x_2}{\sigma_{X_1X_1}\sigma_{X_2X_2} - \sigma_{X_1X_2}^{\ 2}} \right) + \exp\left(\frac{\sigma_{X_1X_2} x_1 x_2}{\sigma_{X_1X_1}\sigma_{X_2X_2} - \sigma_{X_1X_2}^{\ 2}} \right) \right]$$

$$x_1 = \sqrt{-2\sigma_{X_1X_1} \ln\left(\sqrt{2\pi\sigma_{X_1X_1}} P(x_1 \mid \theta)\right)} \quad x_2 = \sqrt{-2\sigma_{X_2X_2} \ln\left(\sqrt{2\pi\sigma_{X_2X_2}} P(x_2 \mid \theta)\right)}$$

To see where this solution is located between product rule and Bayes rule, probabilities of error of the final decision for different values of noises' correlations can be plotted. These figures are in the Results section, and their implications are discussed on Methods and Conclusions.

Multimodal speaker recognition

There are two main objectives for this section of the project. The first one will try to accomplish the best speaker classification achievable by using only speech data.. The second objective is related to the intelligent meeting room, where not only speech info is used, a microphone array and the output from a set of cameras are considered a new input for the system, as mentioned above.

Speech only classifiers

About the speech only multimodal classification, further sections are expected to result in two main classifiers; the one based on GMM, and another based on Articulatory characteristics. Both classifiers focus on feature extractor and posterior clustering/modeling method capable of calculating prior probabilities given the set of models to be considered. This is a good scheme where the feature or semantic level approaches can be applied. Which one will perform better will depend on the temporal constraints related to each kind of feature. The goal of the speech only multimodal classifier will be to apply the different multimodal combinations known, and select the one working better.

Simulation 14 tries to mix features from different nature, all based on speech information.

Multimodal speaker classification in the meeting room

For an accuracy improvement of the current speaker Id systems, new sources capable of speaker identification are joining acoustic algorithms. Spatial

dimensions are the most informative feature spaces; with the prior knowledge of the people's positions and the estimation of the voice source coordinates, a decision is possible. Next section explains how people and source localization are added to the current multimodal speaker ID system. Basically, some previous input modules will output data to feed the multimodal fusion, and the front modules are:

-Video system; supported by 4 cameras focusing the meeting room. Using background subtraction techniques, segmentation and creating three dimensional models, video system estimates the number of objects present in the room and its coordinates.

-Based on time-delay estimation over a microphone array (MA), and a ML criterion, the most probable voice source coordinates are given at the output of the MA system.

-Spectral-shape based speaker ID, commented in previous sections provides prior probabilities of the received voice utterances.

These systems have some errors, then a post-processing is expected to filter their noises. It is considered necessary to analyze the received signal from these front modules, and next sections will create some models for those inputs with two different purposes; to explain the behavior of the systems and to create mathematical usable formulas describing them.

Input models

The names chosen for the basic features, coming from source voices, microphone array and cameras, are:

$x_a(t)$ Information from audio; vector of 39 components; 12 MFCC + energy, first and second derivatives

$s_{MA}(t)$ Information from time delay estimation, vector of 15 elements corresponding to each of the pairs in a 16-channel microphone array including the origin microphone

$s_V(t)$ Information from video cameras

Previous modules are processing this information to output new data directly to the multimodal processing. GMM models for each speaker in the speaker ID module are able to extract prior probabilities of the received vector. Source localization in the MA computes the most probable voice location, we can't get prior probabilities for several positions because computational requirements doesn't allow us to sample likelihood function on-line. Coming from the video system, positions and number of detected objects are available.
Then the final information received will be;

$L(x_a(t) \mid S_i)$ Information from audio; vector of 5 components; log-probabilities of the input signal vector given each of the 4 speaker models of the database and given the silence model (1Hz)

$x_{MA}(t)$ Most probable location of the audio source (12Hz)

$n_V(t), x_V(t)$ Number of objects found in the room and coordinates of the objects. (15Hz)

Acoustic Speaker ID

The system called 'acoustic speaker id' is the one giving log-likelihoods of the input speech vector given the speaker models in the database. For each input vector of cepstral coefficients, it internally computes the prior log-likelihood values. In order to get better performances, and to decrease the output rate, the sum of the logarithms along 1 second is the final output, what lineally means the product.
There are 4 speakers in the database and a model for the silence. Then a vector of 5 values is outputted each second. Under the assumption than one of the speakers should be speaking or there is silence, the probabilities should be forced to comply with:

$$P(\theta_{S1} \mid x_a) + P(\theta_{S2} \mid x_a) + P(\theta_{S3} \mid x_a) + P(\theta_{S4} \mid x_a) + P(\theta_{sil} \mid x_a) = 1$$

But the outputs of the system are log-likelihoods, so using Bayes, and equally probable speakers, the vector will comply:

$$\alpha \cdot \left(10^{P(x_a|\theta_{S1})} + 10^{P(x_a|\theta_{S2})} + \ldots + 10^{L(x_a|\theta_{sil})}\right) = 1$$

Where $\alpha \propto \frac{P(\theta_{Si})}{P(x_a)}$

Otherwise, the alpha factor can be adjusted in every input sample, or every 1s period, once the product rule is applied. Then the best way to get the desired probabilities is using a scalar factor, which involve the summation of all the log-likelihoods during 1 second, and to avoid the lost of resolution caused by logarithms in the floating point calculations;

$$P(\theta_{Si} \mid x_a) = \alpha \frac{\beta 10^{L(x_a|\theta_{Si})}}{\beta\left[10^{L(x_a|\theta_{S1})} + \ldots + 10^{L(x_a|\theta_{S1})}\right]}$$

Product per beta will ensure resolution when logarithmic values go to linear ones, when the above equation is calculated online by summing the denominator and later inverting the result

There appears a new problem: margin between higher and lower extracted probabilities results too high for a posterior use of the values. However, posterior normalization of the probabilities has been discarded for theoretical inconsistencies.

Microphone Array

The result from the MA is basically distributed near a centre called “noise centre”, and moves slightly near to the current speaking person.

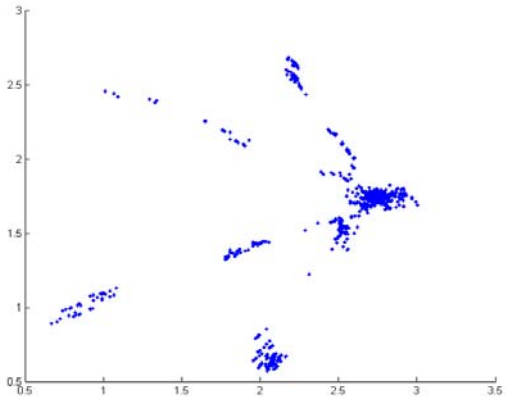


Figure 15.- Microphone array output

To create a statistic model for these samples, a three dimensional pdf will be fitted with the shown characteristics. With the current amount of data, a correct statistical study is still not available, but some conclusions can be extracted.

The model will be speaker-independent. The only speaker-dependent property of the MA behavior is how easy it catches some of the speakers and it doesn't catch some others, mainly because of the power of the voice and its good statistical properties. A speaker-dependent selection of models would just assign the samples closer to the silence to the less detectable of the speakers.

The plot shows the correctness of the coordinates in terms of direction of incoming sound, and the error on distance. The pdf will be parameterized by the noise centre and the current speaker location. The first approach will transform the three-dimensional coordinates to the basis of the vectors:

$$\bar{s}_1 = \frac{\bar{v}_s - \bar{v}_0}{\left\|\bar{v}_s - \bar{v}_0\right\|^2} \qquad \bar{s}_2 = \frac{\bar{s}_1 / \left\|\bar{s}_1\right\| \times \hat{z}}{\left\|\bar{s}_1 / \left\|\bar{s}_1\right\| \times \hat{z}\right\|} \qquad \bar{s}_3 = \frac{\bar{s}_1}{\left\|\bar{s}_1\right\|} \times \bar{s}_2$$

Where s1 is the vector going from the noise centre to the speaker location, s2 is the vector parallel to the plane XY and perpendicular to the first, and s3 is the perpendicular to others.

Assuming these dimensions as uncorrelated, the statistical models for each of the coordinates will be;

$$L(s_1) = \alpha_1 \mathrm{N}(s_1 \mid 0, \sigma_{S10}) + \alpha_2 \mathrm{N}(s_1 \mid 1, \sigma_{S11})$$
$$L(s_2) = \mathrm{N}(s_2 \mid 0, \sigma_{S2})$$
$$L(s_3) = \mathrm{N}(s_3 \mid 0, \sigma_{S3})$$

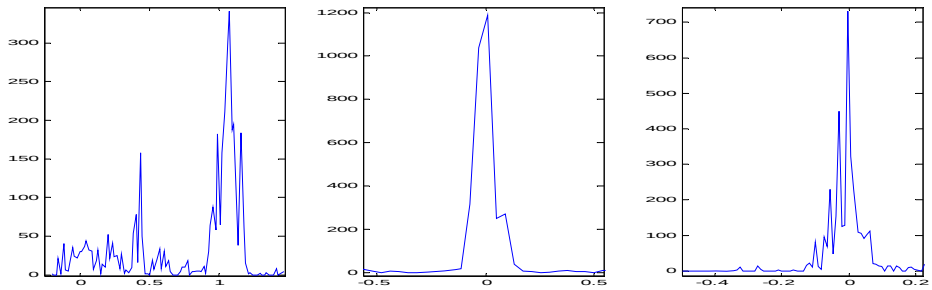


Figure 16.- Microphone array pdf

Gaussian assumption for the second and third coordinates is justified by the histograms. From the histogram of the first coordinate, a Gaussian located on 1 is visible, some Gaussian distribution close to 0 caused by misdetection and some uniform statistic from 0 to 1 also. From the training data, the next values for the mentioned pdf parameters are extracted:

$$\alpha_1 = 0.65 \quad \alpha_2 = 0.35 \quad \sigma_{S10} = 0.2 \quad \sigma_{S11} = 0.01$$
$$\sigma_{S2} = 0.39$$
$$\sigma_{S3} = 0.11$$

Now this information is enough to build several pdf`s of the voice source locations, given the speaker models, based on coordinates, where likelihoods can be extracted and moved to the multimodal step.

The independence assumption is not real, and the distribution along the first coordinate is not accurate, but this model is expected to be accurate enough for the purpose.

Video

The signal received from the video is hard to model. Each person in the room can or can not be detected. Coordinates of each one (when detected) are assumed contaminated by an additive Gaussian noise. In addition to that, there are also false alarms.

The problem is based on receiving an input stream represented by the next plot:

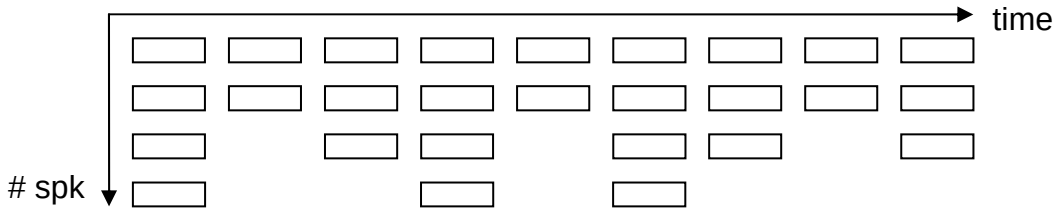


Figure 17.- Video output

Where the boxes are vectors of three coordinates, each time step is a column, the number of vectors received is variable, and those vectors correspond to some of the current speaker or to false alarms

Taking independent realizations as the first simpler model (statistics are not conditioned to the number of received coordinates neither the number of people on the room), each of the 3-coordinates input vectors can be modeled as:

$$\bar{x}_{V1}(t) = \zeta \cdot \bar{v} + (1-\zeta)\bar{\chi}$$

Where:

ζ is a binary random variable being 0 for a false alarm with probability P_{fa}, and 1 otherwise.

$\bar{\chi}$ is a three coordinates uniform random variable distributed along all the room space

$$L(\bar{v}) = \sum_{i=0}^{N} \alpha_i \mathrm{N}(\bar{v} \,|\, \bar{\mu}_i, \bar{\bar{\Sigma}}_i)$$

Where:

N is the number of speakers in the room

α_i is the global probability of being speaking speaker i

$\overline{\mu_i}$ is the position of speaker i

$\overline{\Sigma_i}$ is a fixed covariance matrix analyzed from the error generated by the video system

About detection or misdetection:

P_{nd} is the probability of a present speaker being not detected in a frame

In order to continue working with the model, it's necessary to know its parameters. Probability of false alarm, probability of misdetection, probabilities for each of the speakers and covariance matrix can be computed form the total training data. Speakers' positions are missing and it's necessary to compute them.

From the training data:

$$P_{fa} = 0{,}15$$

$$P_{nd} = 0{,}2$$

$$\overline{\overline{\Sigma}} = \begin{pmatrix} 0.0232 & 0 & 0 \\ 0 & 0.0246 & 0 \\ 0 & 0 & 0.0330 \end{pmatrix}$$

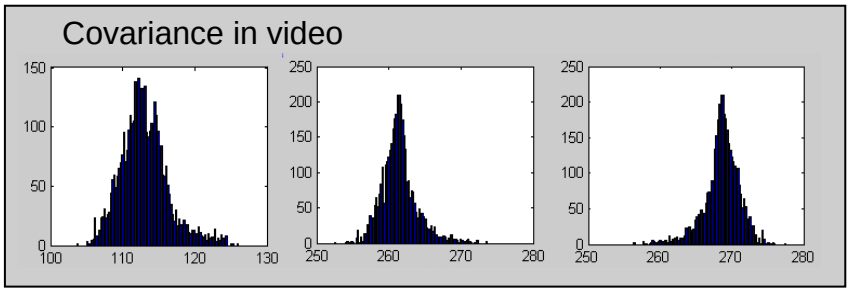


Figure 18.- Covariance calculation from video

Once having these input models, which will be considered valid for all the rest of this work, the Results section will apply classification algorithms to this input models and data, and discussions will follow.

Since the multimodal integration of the meeting room is mainly based on the speaker recognition sections of the current work, no further simulations will be run, so no Progress of the project section neither Limitations section will be referring to the Multimodal issue. However, the Results section a Multimodal Speaker recognition section with all simulations and results.

2.2. Progress of the project

Requirements

Speaker Recognition

Some simulations were performed to accomplish the speaker recognition paradigm: Simulation 1 and Simulation 2 represent the baseline. Two algorithms were used in those both simulations; a self-programmed MATLAB code and also the SONIC software adapted to GMMs.

Next simulations were all based on the self-programmed MATLAB code, which was easier to configure than sonic though it ran slower. Simulation 3 shows first results in the real scenario. Testing and choosing algorithm parameters was also done in this simulation using the experience acquired in the previous experiments, conclusions were discarded because they were similar to the ones thrown by Simulation 1 and Simulation 2. This simulation meant the first contact with the real space which resulted far more demanding than the recorded data. The experiment was repeated many times with different combinations of the data incoming from the microphone array. The data filtering was assisted by the microphone array programmer, and finally a channel-dependent filtering and the addition of several empirically chosen microphones became the best input for the performance of the system

Simulation 4 is performed with nice training data. It will be the first simulation using separate training data (recorded during training time). It had to be

repeated several times to get the correct data. The retries are considered errors of the Training procedure so they are not errors of the system. When we finally got nice recorded data, the simulations started to work. Having cleaner training data was supposed to increase the model's accuracy. The channel transfer function will be independent of the system in this simulation, since models are not trained in the real scenario with the channel influence.

Simulation 5 is the first one using the Articulatory features. The first idea under this simulation is to use the same GMM models used in previous classification algorithms, in order to simplify the progress of the project reusing code. Then once we get encouraging results, we can move to more complicated models. Simulation 6 has applied time-dependant models which are explained in the Results section, in the introduction of the simulation. The conclusions to extract from these last couple of simulations were not clear, so Simulation 7 tried to analyze the errors/inconsistencies of previous simulations, in order to purge the possible conclusions thrown by Simulation 5 and 6.

Robust Feature extraction

At some point during the performed simulations for articulatory extraction, there appeared the need of turning the MFCC + MLP speech analysis used into a more classical speech analysis, where the place, manner… could be derived without dealing with the MFCC. MFCC extraction is a lossy analysis model, meaning that some info is lost when the MFCC is the only available information about the voice, then the natural thought would be using different methods avoiding the MFCC.

These were the suggested ideas for signal analysis to extract the articulatory features:

Voiced-voiceless: the extraction of this feature is the easiest one. By only using the same method used in the pitch extraction algorithm, the voicidity feature can easily be extracted. Other simpler methods based also in correlation can be used. It is important to mention the ability of MLP of returning a voicidity value, instead of a Boolean result, so differences between speakers could show up.

Manner: the effects of the different manners in the voice signal are sometimes clear also, as for stop for example, building a detector of stopped sounds could be easy, and analyzing the behavior of the voice in the surroundings of the glitch could show up some results in speaker recognition. There are some other manners, such as flap, that are recognizable in the speech signal. Other possibilities or features need a further study of the signal itself. Having a large database with articulatory features marked, cutting and concatenating the pieces of sound intended to classify, and plotting those pieces in front of the rest of the speech signal in several spaces and transformations, using filters and analysis methods, would be great to intuitively know how to identify the manner characteristics of phonemes. That is a large work out of the timing of this project.

Place of articulation has something to do with the spectral shape of the different sounds, maybe the MFC approach is good enough for analyzing that feature, but further studies could be accomplished for better accuracies, probably a single filter can work for every place.

About vowel characteristics; tense or relaxed, high, middle of low, and rounded or unrounded, there's no point on looking at the simple signal and try to extract specific differences between them, there a wide study should be performed, where finding specific characteristics of each feature seems to be uneasy.

On the other hand, the classification achieved using the MFC approach threw remarkable results, though they were not useful enough for speaker recognition, had a great articulatory classification success. Then it revealed in some way the importance of the spectral information in those pieces of speech. This way of thought shows again the importance of the posterior usage of the feature extraction, which is speaker recognition. Since the MLP seemed to be good in analyzing the speech signal and classify the phonemes into the articulatory groups, those outputs were not good to differentiate the speaker's speech characteristics. Then having measures about; how rounded are the most of the vowels of this speaker, or how tense or relaxed the phonemes are forced to be for a specific user…, for example, those measures can enter in a new speaker classification space that can achieve completely different information than the MFCC-GMM approach.

This work revealed nothing else than a basic MLP approach to the Articulatory feature extraction, and some theoretical ideas. These ideas would have marked the progress of the project.

Noise over MFCC

Since no interesting result was achieved from the stated theory about noise distribution among MFCC, this experiment ended up with no real progress.

Articulatory features

Pitch and energy were extracted successfully, and some classification success was reached. Other articulatories were also calculated by using MLPs, and integrated into a classification scheme including pitch and energy. The accuracies of the extractors and final classifiers were not completely satisfactory, so the current work progressed by analyzing the exact problems of the suggested baseline. Several methods were tested in Simulation 10, some different parameter validation and check was done in simulation 12. And finally, dynamic models were used instead of static ones in Simulation 13.

Theoretical multimodality studies

There were not specific situations among the project to contrast the validity of the theoretical conclusions of the multimodality studies. Though a couple of integrations demonstrated their correctness, they are still considered a theory to be demonstrated. Then this specific issue requires a more extended corpus of multimodal problems to be precisely validated.

Multimodal speaker recognition

All other results extracted from the other sections of the project were applied to the real data from the CommVision project room. All of them resulted in a performance improvement, though not remarkable. A wider database of meetings with different SNRs ratios would have been perfect for a better analysis.

Accomplishments

Speaker Recognition

A baseline algorithm for the test databases and the real scenario has been set with the previously explained simulations and the results and discussions can be found in the next sections.
The GMM approach is abandoned at that point, since that approach is widely studied and performance improvement seems to be achievable only with different ideas such the ones in the sections of this work.

Using the articulatory features in the speaker recognition problem was already accomplished, since the discussion section is throwing clear ideas about why those results did not help to the main classification improvement, the reader is invited to directly go to Robust Feature Extraction section for further results, because the methods for extraction seemed to be the worst point of the whole articulatory method.

Further accomplishments using the main ones of this section are inside the Multimodal speaker recognition section below.

Robust Feature extraction

Stated in the progress of the project, there's a big limitation in the success of this project, which is a whole study of articulatory pre-classified speech pieces to set empirical differences between articulatory features.
The whole organization of the project included the GMM system start up, the microphone array set up, and the multimodal integration algorithm. The further study of articulatory extraction methods ran out of the timing of this work.

The small achievement of the articulatory results demonstrating the success when catching slightly different information from the speech than the one caught by the old classifiers based on MFCC features is a great start point. The failure

of the suggested algorithms for mixing those speaker recognition classifiers was a small step into better algorithms and performances

Noise over MFCC

Although a better method than cepstral mean subtraction is not found, some questions about how the noise is introduced in the method were answered, and small theoretical keys discovered. It all is considered to be helping to the whole project understanding and approaching, though no result showed up.

Articulatory features

A baseline classifier using pitch, energy and other articulatory features was successfully working during this thesis.

Theoretical multimodality studies

Thanks to this analysis, the limitations of the multimodal integration are known, and then the results of several integrations are considered in their correct measure.

Multimodal speaker recognition

Although the bad results in the rest of the sections, a good integration of very different sources was build and run. This is probably the best accomplishment of the current work, which resulted in a publication at ICAASP'05 (see Appendices).

Limitations

The results of the speaker recognition based on GMM are the best effort of the author of this project. And they are still far from a good system. Throwing 65% of accuracy approximately when there are 4 speakers is a very poor result. The

main reason of that low relation is the noise in the real environment .the same algorithm used in a clean scenario resulted in approximately 90% of success.

The results of the articulatory are not far from the expected behavior, they performed not so good, but stronger against noise. The large difference between the ratios of the MFCC -GMM and the Art-GMM/Art-HMM are the culpable of the joint classification failure. So the Articulatory limitation is bind to the feature extraction method.

The noise is easily introduced in the MFCC+GMM algorithm, since it is the best method when noise is not present. Analysis has showed than mixture between white noise and voice is deep enough to ensure they are impossible to be separated again. Even those features more contaminated can't be rejected for the classification purpose. The author would probably need some other noise models, the noises found on real scenarios for example could behave different and bring this study to usable conclusions.

3. Results

3.1. Speaker Recognition

BASELINE

Simulation 1.- Selecting microphones, training and simulation data.

For this simulation, two algorithms will be used; the one supplied by SONIC and the parameters adjusted by Soon-Il, and a self-programmed MATLAB algorithm which will be more flexible and used further. The results presented are the better of the two algorithms mentioned.

The possibilities when selecting the microphones and the training or testing data are huge, but they are based on:

- Single microphones or combination of microphones. It's well known the capability of mean calculation for noise reduction.
- Sum combination or time-delayed sum combination based on correlation to perfectly adjust the sum of microphones and improve SNR. If the microphones are not completely synchronized, the sum will not act simply as a mean, it will also filter the signal. Post processing to avoid the filter or pre-processing to find the delay will be necessary. The correlation is used to find the delay manually, the method is out of the scope of this project.
- Different microphones for training and testing, which can be personal lapel mics or table omnidirectional mics. Training can be done with any microphone, but testing is necessarily performed on a microphone where all the speakers are present. One could also take every lapel microphone and calculate prior probabilities of a speaker on each audio stream. This would change the main scheme suggested. Although this is not the aim of the project because of the freedom constraint that would be lost using lapel microphones, some simulations will be focusing this problem.
- Different lengths of training data, and once selected the lengths, the part of the conversation used.

	ACS	ACT	CCT	CCTS
OMNI 1 – 1st ¼ training	74%	73%	65%	56%
OMNI 2 – 1st ¼ training	73%	71%	65%	56%
OMNI 3 – 1st ¼ training	70%	70%	61%	51%
HEAPS – 1st ¼ training and testing on OMNI 1	62%	62%	56%	48%
LAPELS – 1st ¼ training and testing on OMNI 12	58%	57%	52%	43%

Table 8.- NIST Database. Different single microphones for training data

	ACS	ACT	CCT	CCTS
ARRAY 1,2,3 – Summed directly – 1st ¼ training	70%	70%	59%	52%
ARRAY 1,2,3 – Summed delayed – 1st ¼ training	72%	70%	62%	54%
QUAD 1-4 – Summed directly – 1st ¼ training	62%	61%	51%	45%
QUAD 1-4 – Summed delayed – 1st ¼ training	63%	62%	50%	45%

Table 9.- NIST Database. Multiple microphones for training and testing

	ACS	ACT	CCT	CCTS
OMNI 1 – 1st 1/5 training	65%	63%	57%	52%
OMNI 1 – 1st ¼ training	74%	73%	65%	56%
OMNI 1 – 1st 1/3 training	73%	73%	65%	55%
OMNI 1 – 1st ½ training	76%	76%	67%	59%

Table 10.- NIST Database. Different lengths of training data

The results showed here are based on this procedure:

- First simulations are run to know the behavior of each parameter.
- A simulation was run with the best parameter combination, resulting in a 74% accuracy.
- Then simulations changing each single parameter and maintaining the other in their last 'optimal value' are executed and results presented

Simulation 2.- Selecting speech analysis parameters.

The second goal of this section was to familiarize with the MFCC-GMM algorithms, in order to analyze:

- The optimal number of filters on MFCC
- The avoidance of the first DCT coefficient
- The length of the utterances
- Number of Gaussians for different input features
- Different situations of the training algorithm
- Specific simulations for a good choice of background model

[2] One of the lapels has not data on the first period of signal. This period will be absolutely avoided.

	ACS	ACT	CCT	CCTS
10 MFCC Filters	66%	64%	59%	50%
12 MFCC Filters	72%	68%	62%	53%
14 MFCC Filters	74%	73%	65%	56,%
16 MFCC Filters	73%	71%	66%	57%
18 MFCC Filters	74%	72%	66%	57%

Table 11.- NIST Database. Number of filters of MFCC

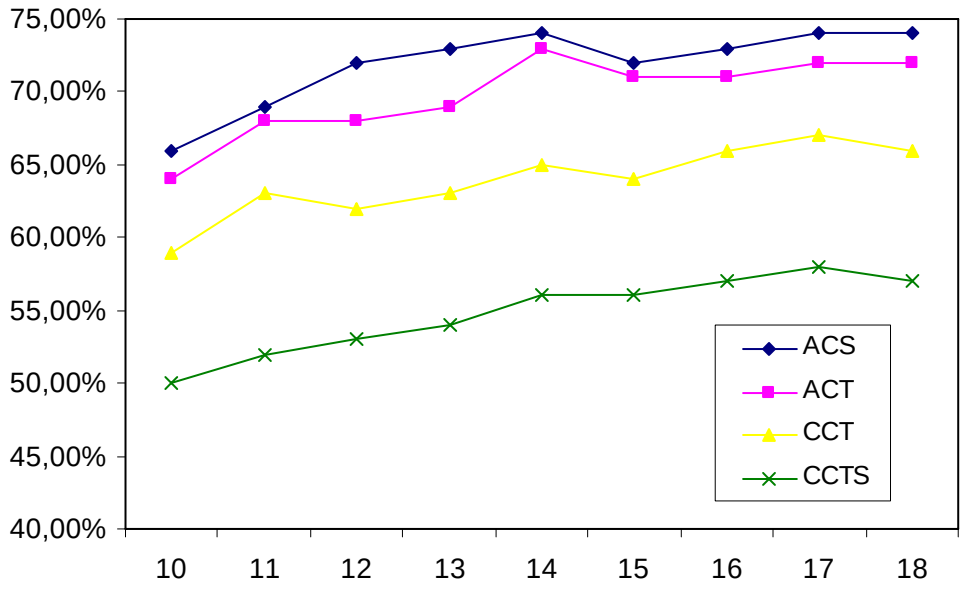


Figure 19.- NIST Database. Number of filters of MFCC

	ACS	ACT	CCT	CCTS
Baseline without 1st DCT coef.	74%	73%	65%	56%
Baseline with 1st DCT coef	70%	70%	62%	53%

Table 12.- NIST Database. Addition of the first coefficient of DCT

	ACS	ACT	CCT	CCTS
Utterance length 0,05 s	44%	43%	42%	39%
Utterance length 0,2 s	59%	58%	51%	48%
Utterance length 0,6 s	67%	65%	57%	51%
Utterance length 0,6 s	67%	65%	59%	53%
Utterance length 1 s	74%	73%	61%	56%
Utterance length 1,4 s	77%	70%	59%	51%

Table 13.- NIST Database. Different length of utterance

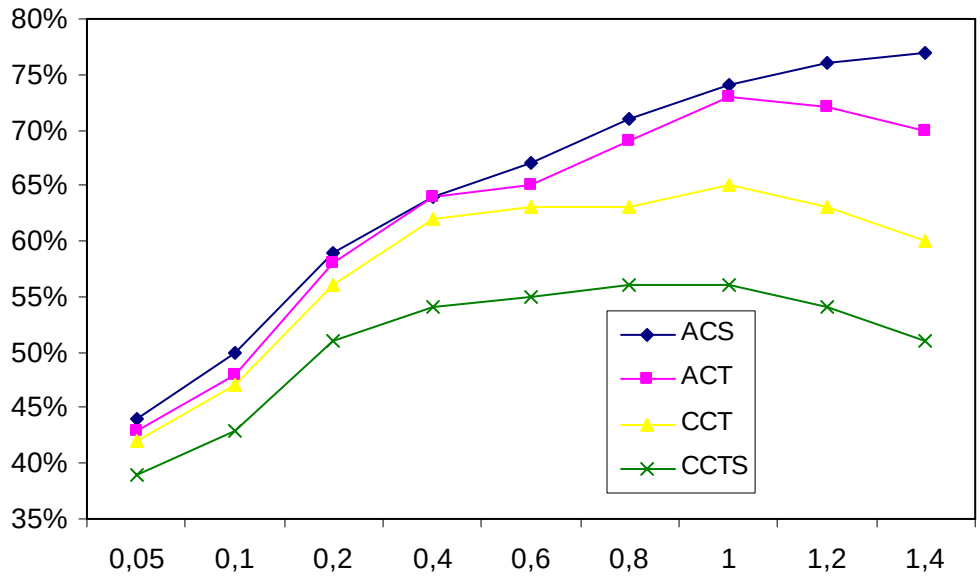


Figure 20.- NIST Database. Different length of utterance

	ACS	ACT	CCT	CCTS
8 Gaussians	28%	26%	26%	20%
16 Gaussians	45%	42%	40%	32%
24 Gaussians	72%	70%	62%	45%
32 Gaussians	73%	71%	65%	56%

Table 14.- NIST Database. Number of gaussians

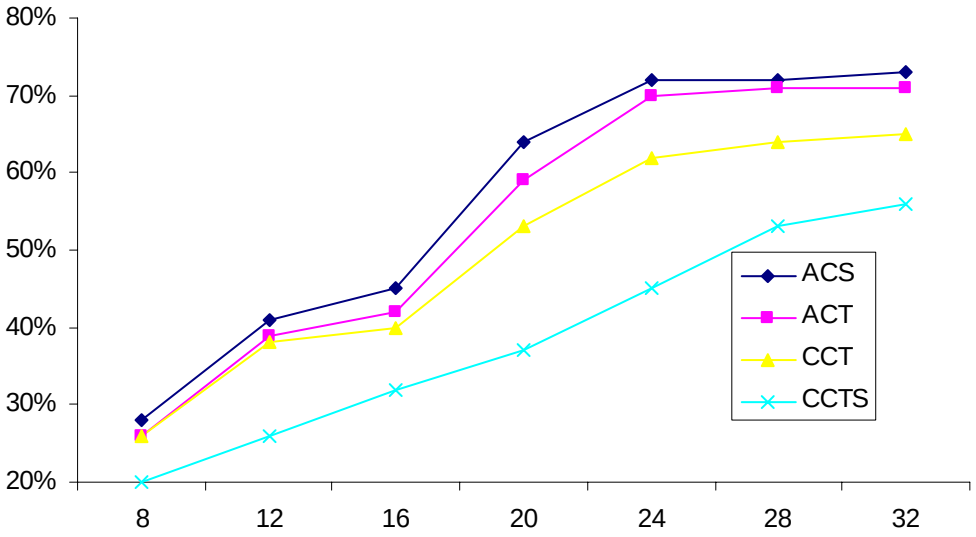


Figure 21.- NIST Database. Number of Gaussians

	ACS	ACT	CCT	CCTS
Using 1st ¼ o the speech	74%	73%	65%	56%
Using last ¼ of the speech	72%	71%	61%	55%
Using 1st 1/3 of the speech	79%	74%	64%	60%

Using last 1/3 of the speech	78%	75%	64%	60%

Table 15.- NIST Database. Training Situations

	ACS	ACT	CCT	CCTS
Without background model	74%	73%	65%	
With background model	71%	70%	60%	51%

Table 16.- NIST Database. Training the background model

REAL SCENARIO

Simulation 3.- Data from the real scenario.

Selecting the voice data from an array of 16 microphones was the first goal of this simulation. A trade-off exists between filtering noise in the room and maintaining the spectral info of each speaker intact. Preprocessing speech data is not considered erroneous, since the microphone array will create greater differences from one speaker to each other.

The data of the real scenario is also including 4 speakers, talking in turns, which were far more ordered than the speaker changes and overlappings on the NIST database. The parameters for the simulation are close to the ones selected in the previous simulations. One minute of training was taken for each speaker, during one minute; every speaker was talking to the array, sitting in the same place where he would stay all along the meeting. All speakers were men, and only one of them speaking Native American English.

The silences were not removed in this simulation, then the pieces of recorded data full of silence were not used for the results, and the pieces containing some speech info and some silence, were evaluated similarly to the ACT method.

	ACS	ACT	CCT	CCTS
Using one simple microphone	62%	60%	55%	50%
Addition of 2 microphones	64%	61%	57%	51%
Addition of 2 synchronized microphones	62%	60%	56%	51%
Addition of 5 microphones	65%	62%	57%	56%
Addition of 5 synchronized microphones	64%	61%	57%	55%
Addition of 10 microphones	65%	62%	60%	57%
Addition of 10 synchronized microphones	65%	63%	60%	57%

Table 17.- Room Data. Different microphone info

The previous filters used to emphasize the signal after the microphones are not mentioned here and are considered part of the microphone array system, though the author collaborated in the design and programming.

Simulation 4.- Training with clean data. Real scenario.

The training data used in the previous was extracted from the microphone array. It was supposed to capture some channel dependant filtering, which would help to the classification whenever the speaker would not move from its position. That's a non-desired constraint of the meeting room, and then we are in this simulation training the models with special data independent from the scenario.

One minute of speaking in front of a single microphone for each participant was recorded and used for training.

	ACS	**ACT**	**CCT**	**CCTS**
Result	64%	60%	56%	53%

Table 18.- Room Data. Training with clean data

The previous filters used to emphasize the signal after the microphones are not mentioned here and are considered part of the microphone array system. They were a key point to get the system working.

ARTICULATORY FEATURES

Simulation 5.- Baseline with articulatory features.

The MLP trained in the Simulation 12, in the Robust Feature Extraction section, will be used in this simulation to see how articulatory features perform when classifying speakers. In the discussion sections, the problems and coherence when using these outputs are commented, here we will go straight to the explanation of the simulation and its results.

This simulation will be based on GMM models; the input vector comes from the various previous MLP and pitch extractor, as they will be presented in simulations 10 and 11. The amount of input variables are: 27 outputs from MLPs, 3 outputs from the pitch extractor, including 1^{st} and 2^{nd} derivatives, and 2 outputs of energy, including 1^{st} derivative: A total of 32 inputs for the system. The outputs from the MLP were extremely abrupt variables when plotting their

PDF. The use of these features as inputs is not recommended, so a whitener transformation was calculated for each input vector. To accomplish the portability of this whitener transformation, the shape was calculated by using the densities from the TIMIT database, which was used to design the articulatory extractor.

The system was widely studied by trying different training percentages, different number of Gaussians, and all the parameters in simulation 1 mentioned. The best adjust of it all responded to 32 Gaussians, 30% of training. The speaking used for the simulation was created by mixing the utterances of the Switchboard database, where 16 speakers, 8 male and 8 female were summed in pairs and speaker recognition based in 2 speaker conversations was analyzed. 15 conversations were created by selecting 3 male and 3 female and mixing them in pairs. The articulatory extractors were trained with the TIMIT database, which could mean it being not adapted to the current speakers for several reasons. Several simulations were ran to analyze the correction of the algorithm in the current database, manually marking some utterances and checking the results; the performance was very similar to the ones showed in simulations 10, 11. The segment length used to classify speakers and the length of the MFCC used by MLP was another issue of segmentation solved with ML criteria.

The observed success was close to a 70%, but the intermediate results were more relevant. In the pairs made with male and female, the performance was better than the performance achieved in the male-male or female-female pairs. There were other particular cases, were a pair of male-male had a good performance. The results with the different pair types were:

	ACS	**ACT**	**CCT**	**CCTS**
Male-female pairs (9)	85%	83%	81%	79%
Male-male pairs (3)	63%	59%	56%	54%
Female-female pairs (3)	62%	60%	56%	55%
Total	76%	74%	71%	69%

Table 19.- Articulatory baseline. All pairs

	ACS	**ACT**	**CCT**	**CCTS**
Lucky pairs (5)	89%	86%	85%	81%
Other pairs (10)	69%	68%	64%	63%

Table 20.- Articulatory baseline. Lucky pairs

Simulation 6.- Dynamic model .with articulatory feature

In the discussion of simulations 10 and 11, the importance of the temporal behavior of the articulatory events is remarked in terms of feature extraction. For sure its importance would come to the speaker recognition issue if the dynamics of the measurements are speaker-dependent. Further discussions will note the inconsistency of the dynamic catch among a large set of speakers for the articulatory recognition issue and the particular dynamic study to search for peculiarities in individuals.

The current simulation applied HMM models instead of GMM to the same previous system. The input vector had a size of 32, and GMMs were based on 32 Gaussians also. The idea was to create 3 state models for phoneme transitions; each state having the ability to catch the utterance statistics the same way a 32 Gaussian with 32 inputs was capable of it, but further applying a temporal constraint of steps along the utterance execution. It was known the temporal parameters of the HMM transitions would not be speaker-dependent, especially when mixing the articulatory features with the pitch and the energy.
The system was trained to achieve the recognition of events at utterance time level, other dynamic events at pitch and energy magnitudes were discarded. The results when classifying speakers were again better for the same lucky pairs of speakers, and again the results are presented related to the pairs they belonged to:

	ACS	**ACT**	**CCT**	**CCTS**
Male-female pairs (9)	88%	85%	82%	80%
Male-male pairs (3)	62%	59%	55%	54%
Female-female pairs (3)	60%	59%	54%	53%
Lucky pairs (5)	90%	88%	87%	84%
Other pairs (10)	70%	68%	63%	61%
Total	77%	74%	71%	69%

Table 21.- Dynamic articulatory classification

Simulation 7.- Further simulations about the baseline for articulatory.

Since previous simulation resulted in very low performances, some fast simulations were run in order to analyze the results.

In the previous simulation, the whole group of features was used to create the models. It's well known the speaker dependency of energy and pitch, and that forced to consider the most of the classification success was due to the addition of those two specific features. A fast simulation was executed with only pitch and energy to analyze that possibility. The static models didn't perform as successful as dynamic ones. Trying the best of static models including first and second derivative, a better result was achieved, but resulting performance was very low compared to HMM using the pitch and energy contours.

A very simple model of 16 Gaussians and 3 states for the HMM transitions was used and showed to be as effective as more complicated models. The results with these models and similar utterance lengths and overlapping used in previous simulations, resulted in the next performances:

	ACS	**ACT**	**CCT**	**CCTS**
Male-female pairs (9)	81%	81%	74%	73%
Male-male pairs (3)	60%	59%	54%	53%
Female-female pairs (3)	59%	58%	53%	52%
Lucky pairs (5)	86%	86%	80%	78%
Other pairs (10)	66%	65%	57%	57%
Total	73%	72%	65%	64%

Table 22.- Pair classification using pitch and energy only

Note please the capability of the HMM to catch very small turns, by turning the ACS and ACT measures almost equal. The lucky pairs mentioned here remain the same than the lucky ones in the previous simulations.

The idea of analyzing the ability of the other articulatory features without the pitch and energy started to be interesting, and the exact same algorithms applied in simulations 5 and 6 were used for the same input vectors by discarding the pitch and energy from them. The results in the next table are extracted from the algorithm in simulation 6 applied to the commented input vectors. All those results, in this particular case, outperformed the results with the algorithm of simulation 5, showing up the small relevance of the dynamics that seemed not to be that important before:

	ACS	ACT	CCT	CCTS
Male-female pairs (9)	67%	66%	58%	56%
Male-male pairs (3)	58%	58%	54%	53%
Female-female pairs (3)	58%	57%	54%	53%
Total	63%	63%	56%	55%

Table 23.- Pair classification excluding pitch and energy from articulatory features

As the comments will note later in the discussion section, the results are very likely to be completely discouraging and discardable, but they will be maintained here, since simulation 14 in the Multimodal recognition section will try to reach at semantic level, similar successes than the ones obtained in the previous simulations, by mixing these two previous classifiers.

3.2. Robust feature extraction

Speech signal

Simulation 8.- Voice source features

Some fast simulations were run in order to answer some of the questions of the current issue.

The first one tried to achieve the speaker recognition by using the pitch.

The next one applied second and third order adaptive filter models to the speech generation processes and used the filter parameters extracted during training as a speaker dependant feature, then GMM models using those parameters could classify speakers.

Last one modified the Mel-frequency scaling factor, to achieve better resolutions in high-frequencies.

None of these simulations reported good results, but since they were considered on some pieces of the current work, here they are explained;

The pitch-based recognition is achieved in simulation 7, some new thoughts based on catching not only pitch, either the pitch energy, second and third peaks in the pitch search over the spectral domain and its respective energies were there tried, using the methods analyzed in simulation 10. The results showed up to be very similar to the ones achieved with only pitch value. The

capability of the pitch extractor when finding secondary peaks was slightly poor, and the energy values had a great variation among a same conversation and even the same speaker.
The variation of the MFCC extraction algorithm specially based on the idea of robustness against noise is developed in the next section, and the spectral varied resolution was there applied to test its performance. Again the results were not good enough to mention them.

The second order adaptive filters were used only in the current section, they belonged to a personal intuition of the author about the possibility of second, third and fourth order components in the speech voice. The adaptive filters with 10 coefficients were calculated by minimizing the likelihood, the same way than wiener filters minimize ML in MLP filters. They performed well for automatically generated noises, and gave apparently logical results over speech signal, where the voiced and unvoiced segments could even be analyzed from its output, but there was no speaker dependency found on that, neither for speaker pairs that were absolutely successful on all other simulations. The particular theoretical objective of the current simulation was discarded without a clear reason explaining the failure.

Increasing the spectral density of filters in the high frequencies of the filterbank was unsuccessful again. The performance of the system was exactly the same then using the traditional MFFB resolution.

Noise on MFCC

Since the first theoretical analysis of the noise distribution over MFCC resulted in a very confusing conclusion, next simulations will probably ensure the ability of the cepstral mean subtraction to achieve the best possible results.

Simulation 9.- Discard high frequencies on MFFB

The theoretical study commented the idea of the next simulation. It's based on the higher presence of noise in the high order frequencies. When the noise is powerful enough, the higher filters in the MF Filter Bank are very contaminated, because of the bandwidth of the filter. So, using figure 11 scheme, next plots show how discarding some filter outputs, the behavior against noise can be modified:

The added noise is white and Gaussian, generated using MATLAB, and the MFCC+GMM is adjusted by using the parameters of simulations 1 and 2.

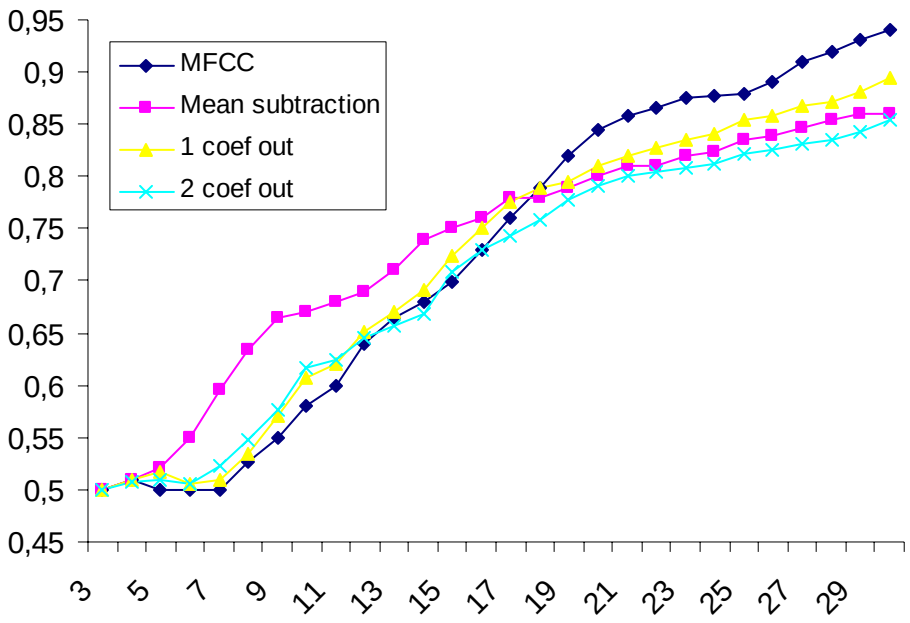


Figure 22.- Error Vs SNR discarding high frequencies in MFFB

The plain MFCC behavior against different SNRs is plotted, so are the behaviors of using cepstral mean subtraction, and discarding one or two high frequency coefficients of the filter bank. Taking a look at it, there are a couple of points where the coefficient discarding outperforms the plain MFCC method.

Articulatory features

Simulation 10.- Baseline to extract Pitch

Four different methods are here analyzed. Autocorrelation peak search; maximum likelihood, spectrum based product or the YIN method. All these algorithms are applied to windowed speech signal utterances from 0,2 to 0,4 seconds with a window overlapping of 0,1 seconds. These simple methods were satisfying enough, and no other post-processing algorithms were required for any of them. In the discussions section, this thread of thought is widely explained.

The autocorrelation method is very simple; a previous de-emphasizer filter is used to avoid pitch harmonics being more powerful than pitch itself, later the maximum peak is searched over the signal vector autocorrelation to calculate its related frequency.

The spectrum based product is an extension of the autocorrelation, because the product in the spectrum space means the correlation in the time domain. The spectrum based approach is used when the sampling frequency is low enough than changing the found correlation peak delay by one simple sample can bring up a big frequency error.

The likelihood is used to show a probabilistic approach to the problem, and the YIN method is based in correlation, but using a special window which showed better performances.

The goals section mentioned how we could use different evaluation methods for the results: SE and CSE were used, but only one is plotted here, because the comparisons between different methods using the different evaluation functions were completely equivalent, and the absolute value of the error is not important at this moment.

Since the plot of the comparison was slightly poor, the simulations for the different methods were expanded to include how the error performs when SNR grows. Then; for every algorithm mentioned, we calculated the performance for

the voice signals of 8 different speakers, 4 male and 4 female, swept by the addition of noises of growing power.

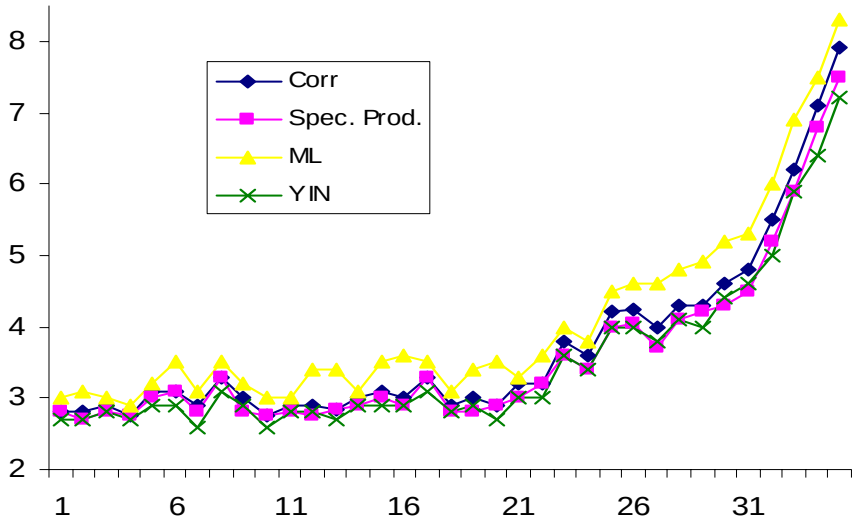


Figure 23.- Pitch extraction accuracy Vs SNR

Simulation 11.- Baseline to extract articulatory features

A Multi-Layer Perceptron will be used as baseline to extract the other articulatory features. The input features for the Multi-layer perceptron are the MFCC extracted from the audio; with a speech utterance length of 1 second, and without overlapping. The discussion section suggests using other extraction methods not based on MF Coefficients or MFCC, which has not been simulated but discussed, so has been the usage of computer learning techniques for it.

The simulations trained MLPs with three layers of 39, 16 and 8 nodes respectively in all of the cases except for the articulation point, which used 39, 32 and 16 nodes. The function shapes in the nodes were as plain as possible, trying to extract at the output of the classifiers not only the classification results, either the distance measure between different phonemes. The silence was included in several simulations. Including it in the training signal meant lower

performances. Then the considered results are the ones with silences discarded.

The input of the system is a vector of 39 values; 13 MFCC coefficients, including energies, and its 1st and 2nd derivatives, with some overlapping in the analyzing window. As the output of the described system is giving us more than one result during the length of a phoneme, systems to segment those outputs were considered. A very simple median window of 200 ms was used, because the visual analysis of the results showed that errors to be solved by segmentation were only 2%, and that method solved it fairly ok

The Manner network had a particularity; it was duplicated to analyze vowels and consonants, so it had 2 outputs. In the consonants output, a vowel was considered to be well-classified if fell into the Vowel class, known by the output of other networks, and equivalently in the vowel classifier. The overall performance of the manner extraction worked better in this separate situation, instead of creating a huge neural network with all the features at its output. Although creating a unique network using this conclusion was easy, it was discarded because of the coding effort it needed, and two different networks were used. The problem detected when using a big network for classification was assigned to the back-propagation algorithm being unable to find a good minimum in the error function.

The next plots show the confusion matrix of the articulatory features, using the trained MLPs:

84% accuracy	**Labial**	**Alveolar**	**Velar**	**Dental**	**Glottal**	**Vowel**	**RESULT**
Labial	362	18	29	22	27	10	77%
Alveolar	16	288	21	27	20	2	77%
Velar	18	33	344	25	15	5	78%
Dental	18	23	12	377	25	9	81%
Glottal	32	22	16	17	345	6	78%
Vowel	36	31	28	26	45	1454	90%

Table 24.- Confusion matrix. Manner I

82% accuracy	**Back**	**Central**	**Front**	**-**	**RESULT**
Back	556	87	75	36	74%
Central	66	397	59	57	69%
Frontal	47	53	595	41	81%
-	42	70	62	1561	90%

Table 25.- Confusion matrix. Manner II

85% accuracy	Stop	Flap	Fric	Nasal	-	RESULTS
Stop	454	16	27	20	25	83%
Flap	4	180	9	15	8	84%
Fricative	22	37	421	29	19	80%
Nasal	17	22	11	465	24	86%
-	32	22	16	17	1892	96%

Table 26.- Confusion matrix. Place

90% accuracy	Voiced	Voiceless	RESULTS
Voiced	1757	167	91%
Voiceless	186	1694	90%

Table 27.- Confusion matrix. Voiced-voiceless

82% accuracy	Tense	Relaxed	-	RESULTS
Tense	867	160	18	83%
Relaxed	170	852	20	82%
-	108	102	1507	88%

Table 28.- Confusion matrix. Vowel

88% accuracy	Low	Middle	High	-	RESULTS
Low	603	33	43	11	87%
Middle	75	499	81	18	74%
High	21	47	613	30	86%
-	18	23	42	1647	95%

Table 29.- Confusion matrix. Height

89% accuracy	Rounded	No rounded	-	RESULTS
Rounded	752	97	58	83%
No rounded	133	1167	60	86%
-	28	42	1467	95%

Table 30.- Confusion matrix. Round

Since next simulations are meant to be compared with this one, the next table will compact the data in the previous confusion matrices:

Feature	Accuracy
Manner I	84%
Manner II	82%
Plac	85%
Voicidity	90%
Vowel	82%
Height	88%
Rounding	89%
Total	85,7%

Table 31.- Articulatory baseline performance

Simulation 12.- Improving articulatory features baseline

The reader was encouraged to believe the previous results in the baseline, while the adjusting of the system was completely uncommented. That's why the next simulations were performed, to show how the parameters chosen in the 10th simulation were the best effort of the author and, though the title of this section includes 'improving', the general performance of the baseline will not be defeated.

Found in the discussion sections, there are a new couple of questions that can be considered when the previous results are analyzed.
- Is the network too small for the extraction?
- Is the 12 MFCC + energy+ derivatives the best input vector for a system of this nature?

These are the simulations that helped to choose the network complexity and the input vector size:

Network complexity	Accuracy
39 + 16 + 8 MLP	83%
39 + 32 + 16 + 8 MLP	85%
39 + 12 MLP	74%
39 + 16 + 8 MLP (but 39+32+16 art)	85%

Table 32.- MLP complexity analysis for articulatory baseline

Input vector	Accuracy
32 MFCC	72%
16 MFCC	76%
13 MFCC	79%
32 MFCC + 1st der	76%
16 MFCC + 1st der	79%
13 MFCC + 1st der	83%
32 MFCC + 1st 2nd der	78%
16 MFCC + 1st 2nd der	82%
13 MFCC + 1st 2nd der	85%

Table 33.- MPL input vector analysis for articulatory baseline

The results show how including 1st and 2nd derivatives, and using a smaller number of MFCC coefficients resulted in better results. Other simulation not mentioned showed how less than 13 MFCC started to decease the performance, either the use of 3rd derivatives.

We should mention than the number of MFCC coefficients used in the simulation sweeping the MLP complexity was 13 + 1st and 2nd derivatives. The MLP complexity used when trying to choose the amount of input features to include in the system, was particularly chosen for every simulated situation. Both premises are tied to result in the best pair of chosen parameters for input vector size and layer complexity.

Simulation 13.- Dynamic models for articulatory features

The baseline for the classification of articulatory features was based on static models. It's particularly interesting how the articulatory event behave along time. That's why the usage of dynamic models instead of static is the obvious way to go.
Using the nature of the previous simulations, the easiest way to move forward was to apply HMM to the feature extractor instead of MLP. That would mean the capability of the new models to catch the time-dependency of the vocal tract. But the comparison between the MLP and the HMM is not on equal conditions. Another simulation will be executed to analyze the importance of time variation, and it will use GMM instead of MLP.

The GMM simulation is adjusted for using time periods of 200 ms, overlapped 50 ms, 32 Gaussians, and the 30% of the samples were used for training. No posterior segmentation algorithm used.
The HMM simulation was based also in the same conditions than GMM, 3 state models, and the Waum-Welch algorithm was used for training.

The results are plotted here briefly, without including the confusion matrices:

Feature	GMM	HMM	MLP
Manner I	76%	81%	84%
Manner II	75%	80%	82%
Plac	76%	84%	85%
Voicidity	79%	86%	90%
Vowel	74%	79%	82%
Height	80%	87%	88%
Rounding	79%	88%	89%
Total	77%	83,6%	85,7%

Table 34.- Articulatory baseline performance

3.3. Theoretical multimodality studies

Result 1.- Simple example results.

To see the stated question about where the theoretical best semantic level solution is located between product rule and Bayes rule, the next plot shows probabilities of error of the final decision for different values of noises' correlations:

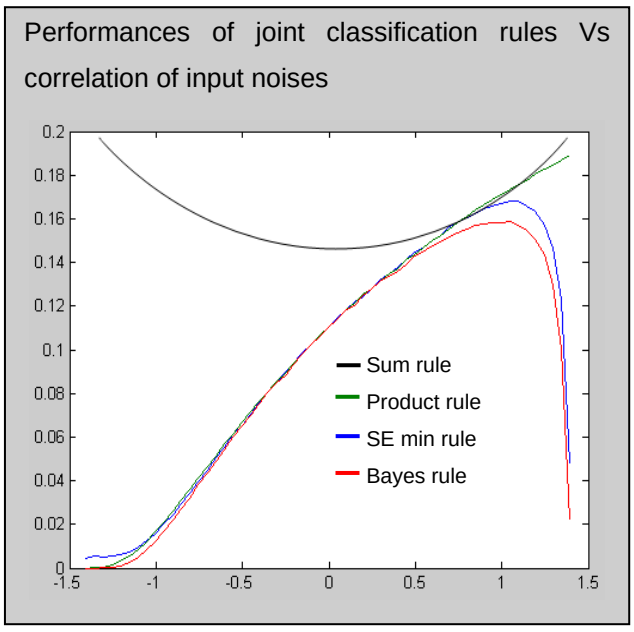


Figure 24.- Performances of Bayes, sum, product and SE

These plots express probability of error against cross-correlation values, because it's on correlated input conditions where SL solutions work inefficiently. Green solution shows the performance of product rule, black one is sum rule, blue for the best semantic decision based on SE minimization, and red for the Bayes decision.

Result 2.- Previous performances importance

The choice of a classification method is not easy and is for sure critical for the performance of the system. The next plot shows the accuracy improvement as a function of previous classifiers' performances in our simple example. Superior corner on the left represents the improvement for previous classifiers which probabilities of error are 0.5, and lower corner on the right the improvement when previous classifiers have no errors.

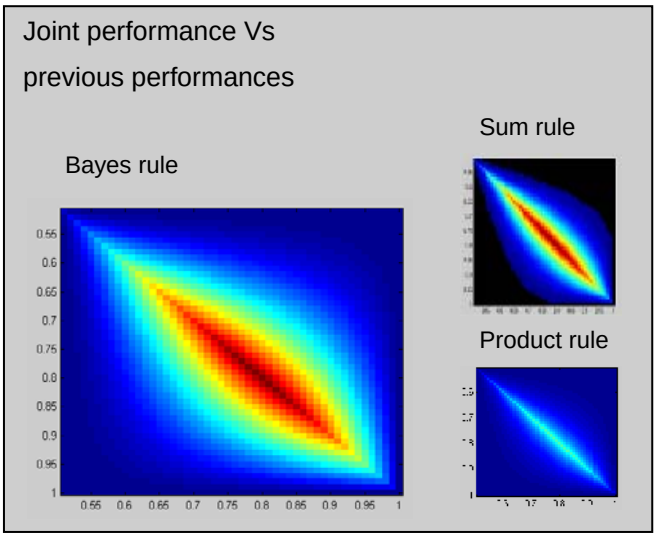


Figure 25.- Multimodal performance against previous performances

The big plot refers to Bayes improvement, the blue-colored one to product rule applied on dependent noises, and the other one to sum rule.

The blue color in the plots means 0 improvements, and the red is the higher improvement. The exact value of the higher point is not important, since the aim of the simulation is to have an idea of the areas where it works better. The best location in the graph for the multimodal issue is the line of equal performances, the one linking the upper-left corner with the lower-right. It's easy to get an improvement in this area, higher or lower depending on the amount of new

information added by the new source. It decreases when we move from this line. This rule is accomplished whatever used method.

Black areas represent the zones where the classification is worst than the best of the previous sources.

3.4. Multimodal speaker recognition

Simulation 14.- Mixing pitch+energy with articulatory classifiers at semantic level.

Simulations 5 to 7 showed how the articulatory features performed when classifying speakers. The timescale difference between pitch and energy compared to the other articulatory features was considered the most important reason of the poor results. In the current simulation, they are mixed at semantic level instead of doing it at feature level.

Last runs in simulation 7 already tried to classify by only using articulatory features or energy and pitch. The different timescales are about 6 times greater for the pitch+energy classifier. The HMM had to be trained with period steps of that order to accomplish the best results, that is giving us speaker turns every 500-600ms, while articulatory features are segmenting the time on 100 ms periods.

The way to join those previous classifiers was selected from a large set of tests, where all the methods approached in the Theoretical multimodality studies were applied. Product rule resulted again the best choice. The temporal difference between the previous semantic classifiers forced the system to multiply the pitch classifier per all the articulatory classifiers under the same time period; so each segment classification was using 2 priors from the pitch+energy and 12 from the articulatory, resulting in 1,2 seconds periods, with steps of 0,6 sections when throwing results.

	ACS	ACT	CCT	CCTS
Articulatory classifier	63%	63%	56%	55%
Pitch+energy classifier	73%	72%	65%	64%
Product rule applied	74%	72%	67%	65
Semantic level mixture	77%	74%	71%	69%

Table 35.- Articulatory baseline performance

Result 3.- Number and positions of people in the room

The solution for computing the total speakers in the room will be based on a stationary state assumption, where the current number of people in the room is known, and hypotheses that a new speaker is coming into the room, and that an old speaker is moving out from it are continuously analyzed. The veracity of these hypotheses will be tested in order to add or delete people from the room, creating an on-line method.
Then the new problems are the tracking and re-estimation of people location, and the formulation of the given hypotheses.

Each set of vectors from the video is received each 100 ms. Under the assumption that the position of a person is constant during half a second, his coordinates can be estimated from a sliding window 5 frames long. Before the estimation process, it's necessary to select which 5 samples inside the window correspond to each of the speakers. Every selection has an associated cost, and the selection with lower cost for each person will be assigned to him/her.

Being o a set of states, with associated set of vectors, then the decision for the speaker j is based on:

$$O_j^t = \arg\max_O \left\{ \prod_{\substack{\tau=t-4 \\ i \subset O}}^{t} L\left(\overline{v_i^\tau} \mid S_j\right) \right\} = \arg\max_O \left\{ \prod_{\substack{\tau=t-4 \\ i \subset O}}^{t} \mathrm{N}(\overline{v_i^\tau} \mid \overline{\mu_j}^t, \overline{\overline{\Sigma_j}}) \right\}$$

Equivalently;

$$O_j^t = \arg\min_O \left\{ \sum_{\substack{\tau=t-4 \\ i \subset O}}^{t} d\left(\overline{v_i^\tau}, \overline{\mu_j}^t\right) \right\}$$

This set of states will be used to estimate current position:

$$\overline{\mu_j}^{t+1} = \sum_{\substack{\tau=t-4 \\ i \subset O_j^t}}^{t} \overline{\nu_i^\tau}$$

To avoid the choice of error samples for estimation in cases of misdetection, a new row of states corresponding to no detection will be added to the current state-graph (figure 17):

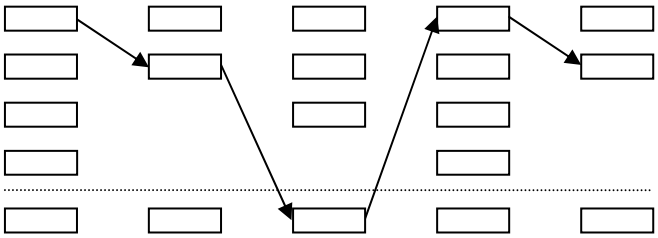


Figure 26.- State-transition graph used in video

To choose the most likely path for a speaker, it is necessary to compare the facts:

- All the samples are most likely to belong to misdetections of the current speaker:

$$P\left(nd \mid \overline{\nu_1^t}, \overline{\nu_2^t} \ldots \overline{\nu_n^t}\right) = \frac{P\left(\overline{\nu_1^t}, \overline{\nu_2^t} \ldots \overline{\nu_n^t} \mid nd\right) P(nd)}{P\left(\overline{\nu_1^t}, \overline{\nu_2^t} \ldots \overline{\nu_n^t}\right)}$$

The samples are likely to be speaker detection

$$P\left(d \mid \overline{\nu_1^t}, \overline{\nu_2^t} \ldots \overline{\nu_n^t}\right) = \sum_{i=1}^{n} P\left(d_i \mid \overline{\nu_1^t}, \overline{\nu_2^t} \ldots \overline{\nu_n^t}\right) = \sum_{i=1}^{n} P\left(d_i \mid \overline{\nu_i^t}\right) =$$

$$= \sum_{i=1}^{n} \frac{P\left(\overline{\nu_i^t} \mid d_i\right) P(d)}{P\left(\overline{\nu_i^t}\right)} = P(d) \sum_{i=1}^{n} \frac{P\left(\overline{\nu_i^t} \mid d_i\right)}{P\left(\overline{\nu_i^t}\right)}$$

Then comparison is

$$P(nd) \frac{\prod_{i=1}^{n} P\left(\overline{\nu_i^t} \mid nd\right)}{\prod_{i=1}^{n} P\left(\overline{\nu_i^t}\right)} \leftrightarrow P(d) \sum_{i=1}^{n} \frac{P\left(\overline{\nu_i^t} \mid d_i\right)}{P\left(\overline{\nu_i^t}\right)}$$

There are too many computations in this formula, and there are some suitable simplifications.

For simplicity, the first model for the distribution of no-detection case will be a uniform distribution along all the room space, and if the distribution of samples is assumed uniform also, the same computation mode based on distances can be used:

$$O_j^t = \arg\min_{O} \left\{ \sum_{\substack{\tau=t-4 \\ i \subset O}}^{t} d\left(\overline{\nu_i^\tau}, \overline{\mu_j}^t\right) \right\}$$

In this case, the distance to a state corresponding to not-detected is set to a constant d_{nd}.

These assumptions are too constraining but result in good performances in simple cases.

For a completeness of the no-detection model, a more accurate distribution can be assigned to misdetection. The distribution is the same than the given for a sample except for the Gaussian component related to the current speaker:

$$L\left(\bar{\nu} \mid nd\right) = \left(1 - P_{fa}\right) \frac{1}{\sum_{\substack{i=0 \\ i \neq j}}^{N} \alpha_i} \left[\sum_{\substack{i=0 \\ i \neq j}}^{N} \alpha_i \mathrm{N}(\bar{\nu} \mid \overline{\mu_i}, \overline{\overline{\Sigma_i}}) \right] + P_{fa} U\left(\bar{\nu}\right)$$

$$L\left(\bar{\nu} \mid d\right) = \mathrm{N}(\bar{\nu} \mid \overline{\mu_j}, \overline{\overline{\Sigma_j}})$$

But this computation is demanding, because of the number of different pdfs created during the algorithm. The algorithm can compute the activations of each of the Gaussians, and for each case use those that are necessaries.

There is another suitable improvement on the state-search. Up to now, weights have been promoting some state-selection, but state-transition weights can add dynamic considerations to the system. Considering the probabilities of the state conditioned to the previous state, probabilities of detection or misdetection can assign weights to the transitions:

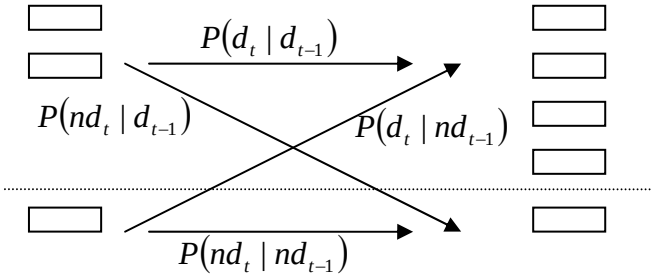


Figure 27.- State-transition graph used in video

These state-transition probabilities help the tracking of people in the room. In the first approaches, there is a threshold up to which tracking is lost. Using transitions, this threshold increases when number of misdetections increase.

These methods update the position of the speakers in such a way of filtering fast movements and committing errors just when the jump between two detections is too long.

The method is highly configurable. The weight of the states can be more accurately modeled, making them conditioned to the number of currently received vectors, the number of people in the room and the values of the samples. Either the probability false alarm is better defined with these priors. Finally a state-transition weight tied to the statistics of appearance of false alarms could bring up to a viterbi-search over the states. These features were discarded because of the reliability of the video system and the performance of the first model.

The subtraction of a speaker from the room will be based on how long the speaker has been misdetected. If the state selection means that the speaker has been not detected for the last 5 frames, he/she will be considered out of the room. The probability of committing an error is the addition of the probability of being a speaker not detected during 5 frames and the probability of the speaker samples being noisy enough during 5 samples:

$$Pe = \left(P_{nd} + P(d\left(\overline{v_i}, \overline{\mu_j}^t\right) > d_{nd})\right)^5 =$$
$$= 0.2^5 + \left[erfc\left(\frac{d_{nd}}{\sqrt{2 \cdot 0.0232}}\right) erfc\left(\frac{d_{nd}}{\sqrt{2 \cdot 0.0246}}\right) erfc\left(\frac{d_{nd}}{\sqrt{2 \cdot 0.0330}}\right)\right]^5 =$$
$$= 0.00032 + \varepsilon$$

The distance of not detection must be higher of 0.1 m in order to achieve an error lower than 0.00032. The distance of speaker misdetection; d_{nd} can be derived for each of the configurations of the algorithm.

The proposed method to add a person into the room is based on a study of the statistic of false alarms. False alarms will be compared with the statistical model suggested in order to know if they hide the samples of a real speaker.
A Gaussian model will be adjusted to the statistic of the input false alarms. From each input frame, the closer sample to the mean of this Gaussian will be used to train the model parameters. In case of false alarm absence, the parameters of the model will be adjusted with a standard deviation and means, equivalent to the real false alarms properties.
Then the update of the parameters, assuming a diagonal covariance matrix for the Gaussian, will be:

$$\overline{\mu_{fa}^{t+1}} = (1-\alpha) \cdot \overline{\mu_{fa}^{t}} + \alpha \overline{v_c^t}$$

$$\overline{\Sigma_{fa}^{t+1}} = \begin{pmatrix} \sigma_{xx}^{t+1} & 0 & 0 \\ 0 & \sigma_{yy}^{t+1} & 0 \\ 0 & 0 & \sigma_{zz}^{t+1} \end{pmatrix} \qquad \sigma_{xx}^{t+1} = \sqrt{(1-\alpha)\sigma_{xx}^{t} + \alpha\left(\overline{v_{cx}^t} - \overline{\mu_{fa_x}^t}\right)^2}$$

when a false alarm is found and $c = \arg\min_i \left(d\left(\overline{v_i}, \overline{\mu_{fa}^t}\right)\right)$

$$\overline{\mu_{fa}^{t+1}} = (1-\beta) \cdot \overline{\mu_{fa}^{t}} + \beta \overline{\mu_0}$$

$$\overline{\Sigma_{fa}^{t+1}} = \begin{pmatrix} \sigma_{xx}^{t+1} & 0 & 0 \\ 0 & \sigma_{yy}^{t+1} & 0 \\ 0 & 0 & \sigma_{zz}^{t+1} \end{pmatrix} \qquad \sigma_{xx}^{t+1} = \sqrt{(1-\beta)\sigma_{xx}^{t} + \beta \sigma_{0x}{}^2}$$

when there's no false alarm

To analyze if the Gaussian converges to the expected values let's see the cases of an existing speaker and noise.

Being a new static speaker in the room detected each frame, mean of false alarms will converge to the mean of the speaker as

$$\overline{\mu_{fa}^{t}} = (1-\alpha)^{n}\left(\overline{\mu_0} - \overline{\nu_s}(1-\alpha)\right) + \overline{\nu_s}$$

And the variance of the false alarms as:

$$\sigma_{fa}^{t\ 2} = (1-\alpha)^{n}\sigma_0^2 + \left(\sigma_s - \nu_{sx} - \mu_{fa}^{t}\right)^2\left(1+(1-\alpha)^{n+1}\right)$$

$$\begin{aligned}\sigma_{fa}^{t\ 2} = {} & \sigma_s + (1-\alpha)^{3n+1}\left(\mu_0 + \nu_{sx}(1-\alpha)\right)^2 + \\ & + (1-\alpha)^{2n}\left[\left(\mu_0 + \nu_{sx}(1-\alpha)\right)^2 - 2\sigma_s\left(\mu_0 + \nu_{sx}(1-\alpha)\right)(1-\alpha)\right] + \\ & + (1-\alpha)^{n}\left[\sigma_0^2 + \sigma_s(1-\alpha) - 2\sigma_s\left(\mu_0 + \nu_{sx}(1-\alpha)\right)\right]\end{aligned}$$

Then a value for alpha can determine the time used by the false alarms system to estimate a variance lower than a threshold.

$$\overline{\mu_{fa}^{t+1}} = P_d\left[(1-\alpha)\cdot\overline{\mu_{fa}^{t}} + \alpha\overline{\nu_c^{t}}\right] + P_{nd}\left[(1-\beta)\cdot\overline{\mu_{fa}^{t}} + \beta\overline{\mu_0}\right]$$

And so on. Parameters can be adjusted for a determined catching time. Using two parameters instead of one is not necessary. False alarm probabilities would set the opposite bound for the parameters; too much fast catching algorithm could take false alarms thinking they are people. This problem is not present in our data, so it was not considered in the study.

Result 4.- Classification over spatial domain

Once the speaker locations are estimated with the video information, and noise centre is known in the calibration procedure, parameters of MA pdfs are complete. The video results are used as the centers of the distributions. Prior probabilities can be extracted from the generated models and sent to the multimodal module.

The next plots show: the probability density functions and the decision boundaries created by the mentioned procedures respectively.

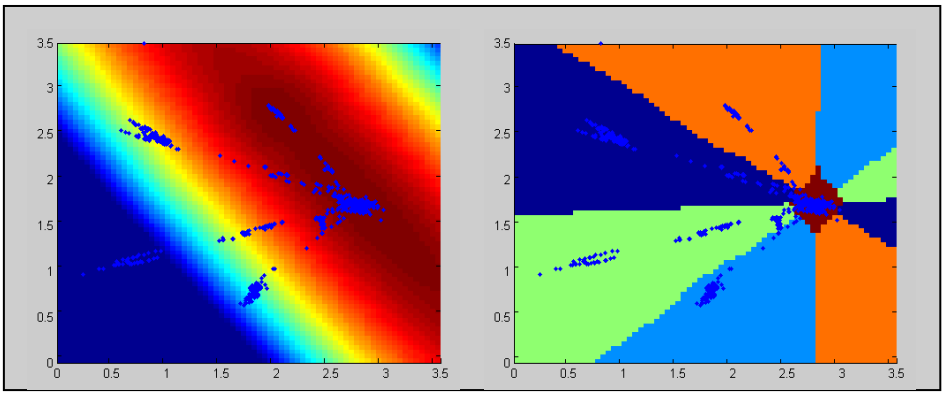


Figure 28.- Model pdf and classification spaces

The model for the silence is missing in the goals section of this work, which has been added as white Gaussian, where center and covariances can be estimated from the training data.

No further specifications are needed for the MA classification. Considering the possibility of receiving incorrect positions from the video classifier, a fast clustering algorithm was run to analyze, only from audio data, the number and positions of the speakers in the room. Results were satisfying enough for offline purposes, but spent a long time to converge when used online, because of misdetections in the microphone array. More ideas in the Discussion section.

Result 5.- Joining classification spaces

It's time to review the study made in a previous section of this work, based on how to combine prior probabilities from two classifiers:

$$P(x_a, x_{MA} \mid \theta_j) = f\left(P(x_a \mid \theta_j), P(x_{MA} \mid \theta_j, \hat{\mu})\right)$$

Mu represents the estimated positions of the people in the room.

But previous to the choice of the function, there is a new problem generated by the independence of speaker models in the spatial domain. Each of the positions must be related to each of the acoustic models. There's no prior knowledge about where each person is located in the room. The only information relating each model to its position is the temporal similarity. Prior

probabilities can be compared in terms of temporal correlation. If the alone performances are good, the probabilities of the models must be correlated. At least correlation between the priors of the same speaker must be greater than crossed correlations.

Having the signals:

$$a_j[n] = \frac{P(x_a[n]|\theta_j) - E\{P(x_a[n]|\theta_j)\}}{E\left\{P(x_a[n]|\theta_j)^2\right\}} \quad with\ \ j = 1..N$$

$$MA_j[n] = \frac{P(x_{MA}[n]|\theta_j, \hat{\mu}) - E\{P(x_{MA}[n]|\theta_j, \hat{\mu})\}}{E\left\{P(x_{MA}[n]|\theta_j, \hat{\mu})^2\right\}} \quad with\ \ j = 1..N$$

The correlations matrix R can be computed as

$$R = \begin{pmatrix} R_{a_1 MA_1} & \dots & R_{a_1 MA_N} \\ \dots & \dots & \dots \\ R_{a_1 MA_N} & \dots & R_{a_N MA_N} \end{pmatrix} + R_0$$

Mean subtraction and normalization are applied to probabilities vectors, and an offset is added to the matrix R to get it being positive-definite

Once the matrix is constructed, a mathematical procedure is applied. Singular value decomposition of the matrix results in matrix U, V and S, where S is a diagonal matrix with singular values in the diagonal.

$$R = USV^T$$

Substitution of S with a identity matrix, and computing R' again,

$$R' = US'V^T$$

Matrix R' has two important mathematical properties. It keeps the relation between weights given by each coordinate of the matrix. And it's constructed in a way that, searching through the columns of R', if a maximum is found in the row i, then no other columns are having a maximum in the row i. This property will help us with searching the right pairs of classes.

Simulation 15.- Real data from the meeting room

All the results from 3 to 5 mentioned in the current section were applied to real registered data in the meeting room scenario. The data was recorded in the

CommVision lab at USC, with the help of the CommVision project integrators, all of them present in the Acknowledgements section. Two conversations of 5 minutes each were first recorded for the speaker recognition algorithm set up. Another 4 conversations between 4 interlocutors resulting on a total of 20 minutes of recorded meetings with all the data necessary for speaker and spatial classification.

About people's position estimation, by only using the video data, the system performed perfect with the first algorithm approached in result 3. The time used to converge and detect the 4 present speakers in the room was about 3 seconds averaged from the 4 conversations. The time ratio given by the periods where at least one speaker was undetected during the conversations resulted in 0,025%, which is a perfectly acceptable result, considering than duration lost was about 3 seconds long, the most of the time spent by the algorithm to converge to a new speaker model. All these results were extracted by the best selection of the algorithm parameters.

Applying these noisy people localization to the speaker localization algorithm, a first speaker classification performance was achieved. About 78% of samples incoming from the MA were correctly classified. Considering than 2,5% of those errors are resilient in the Video people's localization, and the other 20% of errors are missed localizations of the microphone array, at this point we can consider the video errors as less important.

Finally, that classification can be joined with the speaker recognition algorithm. This time, only the GMM approach based in MFCC is used, by applying the SONIC program online or the MATLAB algorithms simulating the online procedure. It resulted slightly better than the simulations using the first real data recordings. 68% is the accuracy for these recorded conversations, to be joined with the 78% given by the MA.

The final system was able to achieve an 81% of performance using the product rule, which is poor better result than the 78% given by only the MA, but much better than the 77% given by the sum rules or other tried approaches.

	ACS	ACT	CCT	CCTS
Video error rate	-2,5%	-	-	-
Microphone array + video online classification	78%	76%	72%	69%
MFCC+GMM classification	68%	66%	65%	64
Multimodal integration of previous	81%	78%	73%	71%

Table 36.- Multimodal integration at CommVision room

4. Discussion

4.1. Speaker Recognition

A big number of simulations have been performed for the speaker recognition issue. This section reviews the results of each one.

Simulation 1.- Selecting microphones, training and simulation data.

This simulation was executed to analyze differences on the microphones present in the baseline database. Main answered question was the usability of the lapel/head microphones for model training, and then the noisy environment is avoided, at least in the training step. Another conclusion is the possibility to reduce the noise with the sum of microphones, which was expected to need some delayed sum or some similar channel equalization, but the system would also work without it.

The choice of a microphone was easy. The cleaner omni-directional one was giving the best results.
Taking different microphones for training and testing didn't work properly; neither when using cepstral mean subtraction to avoid the channel effect. The author tried to train models by using the lapel microphones, which had a better SNR. That could be caused for different reasons; the non-linearity of the channels in the lapel seems to be the most important. That conclusion came up when using cepstral mean subtraction for those microphones, it caused the system to decrease in performance. The cepstral mean was supposed to eliminate those constant filters all along the meeting, and also those multiplicative noises. One should take into account the cepstral mean is calculated with all the meeting information, where there's more silence info than speech. Then if it causes decease of accuracy, it means that the constant multiplicative side of the data was belonging mostly to the speaker than to the channel or environment. Otherwise, listening to the microphones, it is clear the

channel and the environment effects are really important, that rejects the normal situations where mean subtraction is incorrect because of good SNRs. Further corporas demonstrate more flexibility when using different microphones, so we accuse the recording of the database of the performance decease when using different microphones for training and testing.

About the arrays present in the room; the best way to use them for recognition has been summing them delayed; inter-correlations were used to find the exact delays. The improvement is about 2%. This result throws up an important conclusion. We could consider summing the microphones directly would cause the GMM models to catch the channel effect, and use it as an added feature in order to better distinguish the speakers in the signal. But it showed up it doesn't. The reasons can be several; the GMM is not capable to save that information; or maybe there's no great difference between speakers' delays. The spectral shape of the sum should be a clear multiplicative valley in the spectrum, which GMM should catch if MFFB have enough accuracy in that area, then it is right to mostly accuse the no-difference between speaker delays than the lack of ability of the GMMs. We could consider then using new multimodal features linked in a higher level to the spectrum, but the use of the channel effect in the signal would fall again out of our freedom constraint; we don't want the users of the system to stay in the same position during all the meetings to use that effect.

Table 7 shows the results with different training lengths. We had a strange behavior moving from ¼ to 1/3 for training, but we'll consider that a casualty of the current corpora.

Simulation 2.- Selecting speech analysis parameters

Since this simulation was focusing the parameter selection; let's analyze how every parameter affects the final performance:

The NUMBER OF MFCC FILTERS had to be higher than 12 to reach good results. This selection felt on a bad classification when lower, and maintained

good levels when higher. It is directly related with the shapes and spectral characteristics that GMM are trying to model. Since we are catching the characteristics shapes of the speaker's voice, we need to maintain the info of those shapes in every step of the signal information extraction for GMM. It seems we loose that particular info with only 10 MF filters.

The avoidance of the FIRST DCT COEFFICIENT resulted in loss of performance. Other pieces of this study explain this behavior; when the noise is low, the 1st coefficient is speaker-dependant, and in high noise conditions, it depends on the channel.

The UTTERANCE LENGTH is the clue of all this work. The final aim of the global project in which this work is included was to accomplish real-time systems where the speaker recognition is necessary. By using speaker utterances of 1.4 seconds for example, we can miss speaker turns or try to classify utterances where several people interfere.
The longer the utterance length, the better the performance is. But CCT/CCTS is not growing as fast as ACS does. That is logic; when the utterances are short, they hardly fall into speech periods shared between several users, if we consider those shared periods always misclassified, they are only a few wrong classified periods when the decision period is short, but a bunch of them when is large.
In that line of results, there's a misclassification step between 1s and 1.4s. The ACS still grows, but the other evaluation methods go back and result in erroneous speech segments. That's why 1s of utterance is the selected length as a conclusion of this whole work. Classifying the speech signal in 1s segments or using half a second segments of overlapped windows of 1s length showed up very similar results, but the evaluation logic is confusing.

Selecting the NUMBER OF GAUSSIANS for the models was a simple choice. All performance measures grew to reach a top limit as number of Gaussians grew. Since the speaker models were close to the limit when using 24 Gaussians without cross-correlations, the model for the silence/background still needs some more Gaussians to be accurate. But 32 Gaussians is a good

choice for it, because moving to higher numbers revealed no great improvement.

When BACKGROUND MODEL is used, unfortunately new errors appear in the classification. The results given are plotted using 32 Gaussians. As previous results showed, less Gaussians do not accurately model the background, and then differences between having and not the silence model are higher.

Then we can state these general conclusions for the MFCC baseline; we'll use 12 MFCC coefficients or higher, we'll extract or not the 1st cepstral depending on the amount of noise, we'll use 1s of non overlapped utterances and 32 or 24 Gaussians for simulations using or not using silence respectively.

Simulation 3.- Data from the real scenario

Changing the data meant results changes. They are still comparable, because the number of speakers is the same. Conclusions already mentioned in the previous simulation are not reported. We just found interesting the selection of signal source.

The scenario had a microphone array in the middle of the central table, where the meeting was developed. The method followed in simulation 1 was used again, and cross-correlations calculated to sum the different microphone sources. Apart from the chosen method to sum the microphones, summing clearly showed better results than using a single microphone. That was expected, since the sum of the same signal contaminated with uncorrelated noise means the decease of noise power against signal power. Noise was not considered uncorrelated before this simulation, and surely we have different sources of noise, some of them could be considered correlated, but we stated here there is a part of the noise which can be removed by summing, then it is uncorrelated.

Although the synchronization showed up better results in the previous simulations, it doesn't here, and it reinforces the idea mentioned then. The directly summed microphones, when they are not synchronized, helped the GMMs. A clear multiplicative valley in the spectrum is a good shape to force the GMMs to adjust to it. We have here a good difference between speakers that improves the classification. If we synchronize the microphones, we loose a 2% of performance. The only reason for that difference is an incorrect corpora in the previous simulations, or a better positioning of spectrum valleys in this corpora thanks to different spatial separation of microphones. Note that speakers are not moving during the meeting, which could be also a reason.

Finally, we could state a conclusion similar to the one realized with 1st cepstral coefficient; we should synchronize microphones when speakers have no great difference in the inter-delays, and we want to maintain the speaker's spectral info. But we can sum them directly if we want to take advantage of that info, which we would like to note again; violates our freedom constraint for the participants.

Simulation 4.- Training with clean data. Real Scenario

The result for this simulation is slightly discouraging. The idea behind it was to create better trained models, in order to demonstrate than bad training could be the problem of the system. But results showed up than training the models with data extracted from the meeting itself was the best choice. The performance difference is about 2-5%, which is not a big step.

The only explanation for that behavior is the channel effect. It was supposed to be no-included in the models, to accomplish the freedom constraint of letting the users move around the meeting room. We also realized in previous conclusions than discarding the first cepstral coefficient was extracting the channel effect and some continuous speaker speech characteristics, being more or less important depending on the scenario. The current simulation was executed with and without the 1st cepstral coefficient, throwing similar results. Having a clean and different info for training will make 1st coefficient in the training data be very different that 1st coefficient in the testing environment

Simulation 5.- Baseline with articulatory features

This simulation was run with the articulatory features extracted as in simulation 10th and 11th. There are some comments about those simulations related to the speaker recognition issue that are not mentioned in the corresponding discussion section because of the specific pure extraction purpose of those methods. So this is the place for its discussion.

The differences found in the articulatory features between speakers will surely be installed in one of these characteristics:

- The articulatory feature distribution:
Either single feature common distribution, either joint probability density distribution is suspected to be different among speakers, because the articulatory events are commonly executed in different ways depending on the speaker, and the ability to perfectly pronounce our phonemes is widely different. A specific speaker can be good at differencing voiced from voiceless phonemes, and another one can be particular in mixing rounded and unrounded sounds.

- The articulatory temporal behavior:
Every speaker tends to execute the sounds in a dynamic specific way. Moving from voiced to voiceless sounds (for example) can be done in different steps, each of one being characteristically repeated in every speaker, either the single feature execution can be different among them. The next simulations are trying to catch those movements and see how they collaborate with the speaker recognition issue.

From the feature extraction point of view, there's a matter of different objectives compared with the purpose of this section. Extraction has enough with a line/plane in the space to know where every input vector is to be outputted. Statistic models for speaker recognition instead, need wide ranged values, to

catch statistically repeated patterns. That issue is solved in the extraction algorithm by forcing the trained MLP to use functions in the node without abrupt shapes. Although those lines were soft, the output of a classifier tends to force the distances between the different groups, being that its particular purpose.
That's why the classification performance of the classifiers is not a measure of how good the feature will be in the speaker recognition algorithm. In a general scope, what we're achieving in the whole system is to apply a non-linear transformation to the MFCC features, moving the input vector from a first space to a second one, and later try again to classify them. The more important constraint to be achieved in this process is to move the first vector into a space where the speaker models can be more easily created and differentiated.
The performance of an articulatory feature classification system is a measure of the ability of the transformation to group the specific articulatory event in the new space. For the next step of the whole system, that's not the main reason for the transformation, as we mentioned. Then, although knowing the feature extraction algorithm is probably far from our best effort, we are using it in the current simulation.
To avoid one of these mentioned effects, the pre-whitener transformation is used.

The results of the simulations were showing how several pairs of speakers had specific differences in their articulatory behavior, enough to differentiate one from another. On the other hand, other pairs had no significant differences to permit the GMM models a neat classification. The pairs were studied, and there was no particular speaker being repeated among the more successful simulations, neither other specific notable patterns.
Note than the results are really poor, specially for some pairs, and because of the fact that we're right now working with pairs instead of groups of 4 people, which was the main thread followed all along this document. Small simulations were ran for larger groups, and results were completely discouraging. The revealing results of this simulation brought up new question to be solved, such as:
- Which are the particular articulatory features performing the best?

- Is the current system architecture the responsible of the bad performance? Meaning, would another different system catch speaker differences?
- Is the articulatory event a fact among all speaker or does it depend on particular ones?

As usual, these questions are not easily solved, some simulations can throw some light on some of them, but there is no strict analysis to assign the culpability of the unsuccessful accuracy. The best way to continue would be to try different methods than this one with the same overall objective, match similar patterns and suggest again such questions.

Simulation 6.- Dynamic model .with articulatory feature

The current simulation was to probe the importance of the articulatory event dynamics in the speaker recognition paradigm, applying HMM instead of GMMs to the articulatory meters.
HMMs are already successfully used in the feature extractor approached in simulation 13. If a HMM is good on finding articulatory features among a huge set of speakers, it means the temporal behavior is very common, at least there is a specific common dynamism in all the people in the training set. How much of the dynamics belongs to a specific speaker and how much to a common routine will give us the ability of these models on identifying speakers. Following this thought and noting than the results of simulation 13 showed a great importance of the timings in the feature modeling, the bad results of the current simulation could be predicted.

As mentioned, the temporal behavior of the articulatory events is related to a common pattern that all the people needs to obey in order to pronounce utterances correctly than to individual particularities. That is clearly showed up by the results of this section, at least when talking about utterance level events. The performance of the speaker classifiers created for the previous simulations are not better by using the HMM temporal capabilities. The lucky pairs, that had great performances by using the GMM models, increased modestly their successes, but the other pairs got a slight worst result. That result is considered

good, since seems to show up than HMM fit better models than GMM. About the results for the other pairs, they demonstrate than the classification function being minimized in the training algorithms, has non-clear minimums, and both minimization algorithms stop in local minimums, since the number of calculations and parameters in the HMM is greater than the ones in GMM, the number of possible local minimums is greater, then the search for a general minimum is more complicated, and the possibility of falling in a localized one is bigger.

Simulation 7.- Further simulations about the baseline for articulatory.

About the last simulation ran for the articulatory paradigm; let's say it obeyed to a personal thought of the author. The bibliography about speaker recognition showed great results in the pitch and energy usage, especially in terms of statistic distribution and time contours. There was no exact explanation about the usage of the commented techniques neither about the evaluation method used. Although the bibliography assigns to the pitch and energy features a success of about 82%, the overall achievement of the current work, was close to the 70% for the best algorithm, and the ACT evaluation method. Again the culpability of the result is considered because of the similarities of specific speaker pairs. The results over the lucky pairs are close or even better than the known results in the bibliography. In any case, there is another group of speaker pairs unable to reach good classification results.
The author accuses the similarities between the speakers in this corpora, and the lack of rigor in the bibliography when talking about the achieved results to be guilty of those differences.

Once assuming the validity of those simulations, and the reality than the showed results are the best ones achievable by the proposed algorithms, one could analyze how the different information among different extracted features collaborated in the previous classifiers. The pitch+energy classifier is unable to success the way the whole articulatory system did, and the same happens to the alone articulatory classifier. Both together are a successful team, separating them we find than the pitch+energy is the stronger player, but the articulatory

fills some lacks of the first. That firmly demonstrates the improvement given by the articulatory features, showing it to be less important, but significant.

Using articulatory features, we can achieve another classification results than the ones based on the previous MFCC+GMM classifiers. The pitch and energy are important, but other articulatory are bringing up with some different information. The real question from this point is; are this three groups of classifiers; the ones based in MFCC+GMM, the pitch+energy, and the articulatory features coming up with new classification algorithms, including each of them new information to the main objective of the project, and will they all together perform in a better way than the stand-alone systems? All the questions are tried to be solved in the simulations below inside the topic Multimodal speaker recognition. In the Theoretical multimodality studies are approached some methods to perform the mixtures, and the Robust feature extraction will explain how the features of the last three simulations were extracted, how could they be approached in a more strict theory, and what's the finally chosen extraction method.

4.2. Robust feature extraction

Speech signal

Simulation 8.- Voice source features

The current topic ended in the simulation section with completely unsuccessful results, specially due to incorrect premises. The idea of finding a speaker-dependent feature completely unrelated to the MFCC mandatory features fell into several unsuccessful simulations

The idea under the usage of several peaks for the pitch extractor tried to extract the more of the info possible from the speech voice, related to the source piece of the speech model. The vibration of the vocal chords is a key piece of the voice generation information outside the spectral shape modeling, so the extension of the number of features of its extraction was considered a good start point. That start point is still correct, the failure is the feature selection; the

next peaks and its values. Those features were considered to analyze the effect of a different manner when forcing the vocal chords to vibrate, but results showed they caught in a better way the frequencies of the vowel formants, instead of the harmonics of the source vibration. The temporal variation of the energy in the pitch peak, caused the energy to be very speaker-independent, not only for the first peak, either for the next ones.

When focusing on second order models, the classifiers behavior seemed to work over correct data, meaning, there was no apparent phenomena indicating the dysfunction of the feature extractor. But the results were completely noisy, which probably would mean the inexistence of second order filters in the human voice, and the ability of the proposed extractor to produce some kind of noise, or even better: the independency among different speakers of the mentioned behavior.

Increasing the resolution of the filter bank at high frequencies was done to differentiate the speaker and the speech recognition problems. The result was discouraging again, but conclusion is: there's no speaker dependency on high frequencies.

Noise on MFCC

Simulation 9.- Discard high frequencies on MFFB

The result of this simulation, although being unusable, is considered very interesting. The pair of points where the MFCC line is crossed by the graphs of 1 coef discarded and 2 coefs discarded are a particular theoretical point. The power of noise is increasing as lines go to the left side of the graph. Although the main power of the noise is not comparable to the main power of the speech until the left border of the graph, since the distribution of noise energy is more important in the high frequencies, where the voice is particularly weak, there exists a SNR ratio where the last filter in the filterbank has a localized SNR close to 0. That crossing point showed up to be close to the area where mean

subtraction starts to outperform MFCC. That is considered a pure casualty, and author is pretty sure than applying the same simulation to different corpora will surely result in similar shapes but those points would not coincide.
The same happens with the 2 discarded coefficients line. The overall SNR must be lower to get a completely contaminated 2nd coefficient, but at that point it was better to discard it, being this argument the one we wanted to demonstrate.

Unfortunately, there was no point where the filter discarding worked better than the MFCC or the mean subtraction. That is surely because of the nature of noise, being in a main part constant along the time of a meeting. The out performance could only happen in the case of being some part of the noise also variable and contaminating also the mean subtraction. That seemed not to happen from the results, and finally mean subtraction is considered from now on the best approach when using MFCC.

Articulatory features

Simulation 10.- Baseline to extract pitch

The results extracted from the simulation 10 showed up a very reliable articulatory feature: the pitch. Since this section was only focused on the trade-off between the accuracy of the extraction and how it performs when noise is present, one could say the simulations were completely satisfactory. Later the pitch showed not to be as good for recognizing speakers, but that's in the focus of the previous sections.
After the pitch extraction algorithms, there was no need for filtering or grouping the results. A simple median filter was tried in the posterior processing, which resulted in a great performance where noise was hard enough to contaminate the pitch. But that was not the ideal scenario for the corrected pitch extraction in the speaker recognition area, because the classification result was so contaminated by errors, that solving a 3% of pitch errors didn't help to the whole system.

All the methods mentioned were equivalent. One could just view a very slight difference between them in the graph. That difference was mainly caused by the time resolution of the different methods. Some of them were working in the time domain, then they assigned a value to the peak delay, which was calculated in terms of samples. That meant the pitch resolution is constraint to the sample resolution and obviously to the sample frequency. Other methods worked in the spectral domain. A slightly better resolution can be achieved, theoretically in terms of interpolation, in the spectral domain, but the FFT function must be chosen carefully to achieve it. The ML method, against prognostics, gave the best resolution in time domain, that's caused because its dereference against time-samples and floating point calculation.

Simulation 11.- Baseline to extract Articulatory features

The results of the 11th simulation were encouraging enough. The performance of static models based on MFCC features to classify articulatory events, was more satisfactory than expected. The first thought of the author was that static models should never catch the vocal tract events successfully enough, but results showed to be not that bad. The classification result was considered the best effort achievable in these terms, and better ratios were left for dynamic models or classifiers not based on MFCC. The posterior processing of the feature extraction, based on median filter was an important step for the speech segmentation success.

The immediate conclusion about these results was to accept MFCC catch the most of the necessary information in the speech voice to classify the articulatory events. That's good in terms of speech recognition, but doesn't mean a good behavior for speaker identification.

The silence issue was especially interesting. It had the constraint to force our MLP extractor to be redesigned in terms of complexity. The ability of the network to catch it was good, but only when a certain complexity was achieved. That could bring us to the conclusion that the speech model for silence is a complex statistical group of features needing higher complexities than simple phoneme characteristics. It is shocking when silence detectors are easy to

build. Then the reasonable explanation is that an articulatory feature extraction algorithm need all of the MLP network complexity to its purpose. Then using the same network to classify the silence, it complains about the need of a separate/parallel group of perceptron nodes to categorize it.

Simulation 11 is working on the idea of dynamic models for articulatory events; at this moment 1st and 2nd derivatives are included. Although that is a poor consideration of the temporal behavior of the input variables, that's taken into account. Next simulations, particularly simulation 13 is using HMM to tie the result in a major way to the dynamic characteristics.

Simulation 12.- Improving articulatory features baseline

There are a couple of important conclusions in the current simulation; which are the unnecessarily complexity of large MLPs, and the maximum capability of the MFCC to catch the articulatory effects when having 13 coefficients. The MLPs complexities are widely discussed in literature, and there's no need to go deeply into it. But MFCC complexity is very interesting.

The major differences in the articulatory events in the speech signal are tied to spectral shapes, dynamic behaviors, temporal glitches… There is a lot of information in the speech signal usable for analyzing, and the MFCC is one of the best known information extraction methods for the voice. Varying the number of coefficients in the MFCC extraction method, should only mean to extract a different vector size from the same input signal. If we are talking about features completely clear to identify when looking at spectral shape, we were also completely confident to identify them in the MFCC domain. The use of a greater number of coefficients would mean a greater definition in the spectral/cepstral domain, but the expected shapes would be the same while changing the vector size.

That means that the length of the input vector is the same than the number of filters in the MF filter bank. That's not colliding with the previous assertion, where changing that number will only mean having a better resolution, but can

throw up a new idea. The position of the filters in the filter bank can achieve the particular separation of several spectral specific characteristics, though filters are contiguous, because they are completely overlapped, they can favor a specific spectral shape when changing the resolution.

The results plotted in the previous sections, were showing how 13 coefficients result the best choose. The performance when increasing/decreasing that number was not plotted, but it was lower and noisy, meaning some particular number of filters behaved surprisingly better than other numbers. The clear conclusion of this effect is to say that MF filter bank and respectively, MFCC are missing some features valuable for articulatory feature classification, and only in special cases it's able to catch them in a successful way.

Simulation 13.- Dynamic models for articulatory features

The results of this simulation are odd, because they reveal the need of temporal behavior catching, and also reveal the outperformance of a static model against a dynamic one.
The GMM model is very poor in classification results. The bad results are assigned to the training algorithm, which is unable to find a good minimum, and it stops in localized ones. The noisy results as the system was trained several times indicated it. But those results will still be a good comparison objective since HMM and GMM are very similar algorithms.
The results of the HMM are surprising. Similar models than the previously used were able to improve the classification results in an 8% by applying the state change weights. This result is taken as unequivocally showing the relevance of temporal behavior on the articulatory events. That is another argument for future extractors that can avoid MFC coefficients, and obviously another relevant feature for the speaker recognition issue.

About the difference between the machine learning technique and the other models, one could think than applying time-dependency to a model such successful could achieve still more accurate results. But that's one of the

oddities of the machine learning techniques, the resulting systems are not easily analyzed neither can be modified to work on different scenarios or premises. It's important to remember than the MLP simulation included the 1st and 2nd derivatives of the MFCC, that's the first step into dynamism, and probably it's enough for the current system.

As it will be seen later, the global conclusion of the articulatory feature extraction ends up with the need of different extractor architecture, starting from the signal samples, and ending on the feature measure, by using meters rather than classifiers, and avoiding a common start point for all features such as MFCC. Having this conclusion in mind, the behavior of the MLP is given as good enough as baseline, and will not be discussed how to make it dynamic.

4.3. Theoretical multimodality studies

Result 1.- Simple example results.

There are some conclusions from the plots stated in Result 1:

- From the given formulas, for null cross-correlation between noises, product, semantic and Bayes solutions are the same. But as correlation increases, the new bound is higher than the Bayes bound, and sets a new limit to be reached by SL solutions.
- Multiple sources combined at SL are useful when the errors at its outputs are uncorrelated, because the classifications measures use to be affected equally and that becomes to equal errors after combination. But the correlation of the noise in the source signal doesn't necessary mean correlated classification errors in the SL. Two sources with uncorrelated added noises ensure uncorrelated output errors, but correlated error sources aren't necessarily correlated errors at the output of the joint classification. In the last part of the curves, as the noises get more and more correlated, the performance increases.

And some new questions:

-Is the SE minimization the optimal SL method? The final performance of a classifier depends on the decision areas and bounds. SE method penalizes deviation from real PDF along all the area of the function, and gives no special importance to areas closer to the decision bounds. There are algorithms based specially on the creation of a good decision plane, and they sometimes show up as better classifiers, being completely unaware of the PDF shape far from the decision bound.

General solution

From the found solution, it's relevant how hard is to select a function fitting the joint PDF. An easy solution for the problem would be a parametric joint probability function, and it would need to adjust the parameters to best fit the joint density. To adjust those parameters, a function to be minimized is needed, for example the SE between PDFs or Minimum Error at the output. Note that a wide training data would be necessary. It's known than lineal functions are the easiest ones for convenient parameter estimation. The problem is that the function we found for the simple example is not lineal at all. Neither is possible to find analytical solutions for SE in the most of the cases. Then it's necessary to study families of functions and to find a technique able to train its parameters. Machine learning techniques have been using this kind of parameter training in different applications, based in these operations: They already solve the problem of being the resulting PDF more important in the areas closer to decision boundaries, as long as minimization is usually performed in terms of output errors. The methods performed by machine learning are similar to:
-Create a binary posterior PDF such as 0 in areas where the current class is not the correct decision, and >0 where it is.

$$\hat{P}\left(x_0, x_1 \mid \theta_j\right) = \begin{cases} p & P\left(x_0, x_1 \mid \theta_j\right) = \max_k \left\{P\left(x_0, x_1 \mid \theta_k\right)\right\} \\ 0 & elsewhere \end{cases}$$

-Use a trainable function to approximate these PDFs.
-Apply error minimization using a defined distance function.

Although there is some nonlinearity used in the process; in the probability function to fit or on the distance measures, training methods are well studied and stated. These methods have some advantages in simple classification spaces, fitting the decision boundaries to the real bounds. Either it has disadvantages with variable PDFs, when estimated error is usually not representative of the performance of the classifier, but each of the machine learning techniques are scalable and its performance Vs escalation discussed.

Neural networks and back-propagation or support vector machines are two techniques focused on fitting decision boundaries. They are used in bibliography with higher or lower success.

Validation of the stated model

The stated model tried to be the simplest problem to be approached. Its simplicity would surely be a problem in order to extract conclusions for higher order systems.

However, the extracted conclusions take this order-dependent problem into account so the author gives full validity to the model.

A word about product rule

Since other solutions such as sum, min or max rules assume errors on previous classification spaces, product rule is simply based on independency. The assumption can be shown not far from reality, and the results are good enough for the most of the applications.
Before considering independency of previous classifications, let’s assume that a new classifier is added into a system because it’s expected to solve the problem we work on where the first classifier doesn’t do it. Then a new feature is added into a set of features when its classification result is uncorrelated with the

previous ones. If the addition of a new algorithm means falling into the same errors, the solution may be discarded.
One could say that uncorrelated input vectors become uncorrelated previous classifiers. And it will usually hold, but inputs are not necessarily uncorrelated when output errors are.
In cases when multimodality is used for noise robustness, we cannot assume independency. The new features are not added because of its independency, they are maybe correlated with the previous ones but more robust against noise. As seen in previous figure, dependency doesn't mean decreasing of performance but it invalidates product rule assumption.

Result 2.- Previous performances importance

The problem represented by the dark areas in this experiment reflected a bad choice of the combination function. The conclusion to extract from this experiment should be the danger of an incorrectly chosen function. And again in favor of product rule; it showed up to never be worst than simple classifiers.

But there are other areas that, not being black, they mean no significant improvement with the second signal source. Generally, adding two classifications where one of them has a good performance but the other one has a bad one, revealed a hard to reach improvement.

The idea of moving these conclusions to higher order problems is hard to justify. In high level approaches of the problem, one could consider a very simple model for the previous classifiers; clustering systems contaminated with noise. The mixture of good and bad classifiers then would mean to group noisy and clear systems. They are for sure throwing less uncourageous results than the mixture of similar ones, that's proved with formulas very similar to the ones used in the simple problem.
But modeling a whole multimodal system is meant to be more accurate than joining contaminated sources. The better the model, the best it will classify. But all the simulations executed in this work showed up that behavior, which,

though we demonstrated it is using a non-general model, we will take it as definitive.

4.4. Multimodal speaker recognition

The simulations 14 and 15 are run in this section, results 3 to 5 are trying to theoretically analyze the methods suggested and applied in simulation 15.

Simulation 14.- Mixing pitch+energy with articulatory classifiers at semantic level.

In the results table of this simulation, we can see how the product rule outperforms the previous classifiers, but it is not as good as semantic level combination. That confirms the general idea about the choice of semantic or feature level combination, where dependent features, like these ones are better combined at feature level. Although it is not a great conclusion, this result has a special interest because of the pair of features we are talking about. They are temporally correlated and apparently throwing similar results -except for the errors- but are scaled differently in time, meaning the best of their previous classifiers are having very different time periods. That seemed to direct the author to the semantic approach, but it didn't work out.

The semantic combination so, had here another bad result. So the basic ideas about choosing the type of multimodal combination should be reconsidered, and give more importance to the feature-level combination. Probably the different time scales of the features is the problem when trying to integrate them at semantic level

Result 3.- Number and positions of people in the room

The first method suggested in the current result got 2,25% of errors. The method based on state change weights performed equal than the simpler method. It is considered better than the first method, but the first approach is good enough.

In case source would throw faster and noisier results, the second method would perform better than the one used.

It is important to note than a sliding window of 5 samples is the right choice to filter the errors and combine the samples incoming from the video system. The general conclusion about the current result is the correctness of the video solutions. Except for the noted limitations of the system, which is very sensible to movements and light changes, the performance for people who are not constantly moving and the light ambience is not violently modified, is very good. The system is not complicated but has a great result.

Result 4.- Classification over spatial domain

The extracted results when mixing the video information and the MA output are pretty good. The observed errors are evidently misdetections on the MA system. There are only a little group of samples close to the noise center, where some samples can be considered ambiguously related to a speaker or to the noise. The author tried to force the sample border by manually modifying the variance of the noise model. The results showed than the estimation given by the noise samples was the best choice for the noise variation.

The final result is about the 78% for the MA samples. There exists the confidence of being extracting the best from the given data. Since the output of this system will not only be based on the decided model, otherwise the likelihoods of every model will be used. The values of these likelihoods were plotted and observed, they were slightly deviated when the MA sample was ambiguous, but they were widely separated when the sample was clearly incoming from a specific model. The possibility of whitening these samples was considered, but discarded, in order to follow a strict probabilistic approach.

Result 5.- Joining classification spaces

The offline nature of the current linkage of classification spaces is the worst conclusion in the current system. As mentioned in the results, a perfect match

was achieved in all simulations between the models belonging to spatial or speech spaces. Author considers there is no other way to link the different models, because of the completely different spaces and natures of the information.

A pair of simulations was run also in order to analyze the online performance of the current system. Although the system is always converging to the right decision, the only difference between the online and the offline methods is the convergence time, analyzed as the time while the decision is incorrect until it is fixed. The time was close to 10 seconds. Note than once the decision has converged, the previous incorrect results can be fixed, meaning the online decision being incorrect in that period, but the final result is correct.

Simulation 15.- Real data from the meeting room

Mentioned in the previous sections; results 3 to 5, the mix of the video an MA spaces, and the model matching between this space and the speech spaces result in good performances. The final decision though, was chosen by using simulation 15. It used the product rule to decide the more probable speaker, discarding sum rule or any other mixing methods. A time period of 1 second was integrated to extract the currently speaking person in the room.

Results were not as good as expected. Previous classifiers with 78% and 63% of accuracy resulted in a mixed performance of about 81%. The use of the product rule is the best choice inside the group of traditional probabilistic approaches for multimodal semantic integration. The results in the Theoretical multimodality studies are oriented to analyze the particularity and difficulty of the best multimodal integrators.

The theoretical approaches to the semantic combination are only improved by the knowledge of pdfs of previous classifiers, by knowing them; some minimization function could be applied. There was no approachable mathematically simplification possible, so it was theoretically abandoned. Other

empirical observations and mixtures were used, based on the idea of creating functions applied to the likelihoods of the spatial and speech classifiers.

A fast experiment using a neural network was ran, trained and applied to the current semantic combination, as the conclusions of theoretical studies suggested. The results were very close to the product rule application, they were slightly better than the first one, but not enough to mention it as a success.

5. Conclusions

There have been three main purposes in this work; improving the speech analysis for speaker recognition, mixing the speaker classification from different classifiers, and applying it to the meeting room scenario.

The idea of finding information in the speech different than the traditional approach based on MFCC is a good start point, and it is still considered worth working on it, though the results achieved here were not satisfactory. They are failing because of the different performances between the traditional MFCC method and the newly suggested articulatory classification. A theoretical study showed how classifiers with very different performances can't achieve big improvements; furthermore they can increase the number of errors if the multimodal integration is not correctly designed. So the general conclusions about the speech only classification is the correctness of the method followed, but the need to achieve better results in the alternatives to MFCC. Articulatory feature seems to extract important information for the speaker recognition issue, so it is considered to be a correct choice. The lack of accuracy when working stand-alone is the drawback, and is considered improvable by using other articulatory extraction methods. When high SNR appears and MFCC looses accuracy, the methods are more equivalent because articulatory features are less affected by noise, so there are remarkable achievements in that situation.

Methods to integrate different classifiers are mentioned all along this work. The traditional literature was only based on the probabilistic assumptions resulting in the sum or product rules. The probabilistic approach is mentioned in this work, applied to the PDFs just before the previous classifiers. The need of knowing those PDFs, and the application of some error function and its minimization, brings up an unapproachable problem, it is mathematically solved for a very simple example, but seems not to be reachable for more complicated scenarios. The idea is considered to be right, and is suggested as a good method for other scenarios, but it seems not to be solvable in most of them. The conclusions show the little difference between the mentioned solution and the product rule in most of the cases. In other cases it can be mathematically very sophisticated.

That kind of complicated mathematical functions and the approaching of unaffordable problems seem to match with the situations usually solved with machine learning techniques such as the neural networks or SVM.

The meeting room scenario has been the place to apply all the mentioned algorithms. The performance in the room is not good enough at the current moment, but the author considers it to be ok as a baseline. The major problem incomes from the inability of the traditional speaker recognition based on speech and MFCC, and it was expected to be solved with the articulatory extraction. The MA has a very good performance, and the empirical experience reveals the system to be more valuable. A better adjust for the MA can bring up with a much better performance, and it is achievable with some more work done in the configuration. Otherwise, the speech classification is not considered to be easily improved, and the video system seems to be weak when dealing with moving people. The method mentioned for joining all the systems is considered to be working fine for the current previous performances, but it is also subject to improvements.

The major objectives of the current work have been achieved: the study of the speech signal and its relations to the speaker dependencies, the study of mixing new information into a multimodal system, and the application of these novelties to a real scenario. The need of updating the real scenario simulations and the work on the meeting room was occupying more than the expected time for it in the whole work schedule, but it was satisfying given the bad results on the research side of the work.

There are no successful results in the current work, which was pretty discouraging. Some parts of the work are even trying to analyze why the improvements are hard to achieve. Finalization of the work and presentation was done before the typing of this document. It was finalized later, by mixing a lot of documents and handwritten types related to all mentioned simulations. The creation of the document was delayed in part trying to achieve good results; but the author decided bad results were as good and notable as good

results, and finalized the current work with the relation of all the unsuccessful but important simulations; which resulted the common routine of this work.

6. Further work

The extraction of articulatory features in the current work was mainly based on MFCC, even when using a wider set of filters, there must be a better way of extracting articulatory features from voice segments, and the methods have been enumerated in the discussion section of this work.

Though the usability of temporal-dependencies of articulatory features was demonstrated, the exact importance of its behavior is to be studied. The periods while these features are relevant, the differences in time-level between them or even which of them are less time-oriented are good questions to be analyzed in the future. The articulatory features were included in GMM/HMM models in this work, using other models should be analyzed. Particularly, domain transformations could be applied to the features before fitting them to the model.

Approaching the multimodal integration, the theoretical study showed how the shapes of models and decision boundaries were based on PDF shapes, though they were not necessarily the same. Then studying the performances of different functions families in the integration step would be a nice simulation to be run in the future.

Application of supervised learning and monte-carlo methods to the decision step is a conclusion extracted from the starting bibliography than the author of the project is willing to revisit and apply to the last step of our classification process.

When thinking about the CommVision project, and mixing the theoretical experience acquired in the current work, the sampling of the ML source localization function with the estimated people location would be a nice start point to try to improve the system’s performance.

An accurate people tracking system could be as well useful in the CommVision project, since the people-track established in the latter simulations was very simplistic, and maintained because of the nice performance in the main database. Addition of spatial information in statistical models for a better spatial classification would improve the overall performance as well.

7. References

[1] Content analysis for audio classification and segmentation

Lu, L.; Zhang, H.-J.;Jiang, H.

Speech and Audio Processing, IEEE Transactions on , Volume: 10 , No: 7 , 2002

Pages:504 – 516

[2] Robust text-independent speaker identification using Gaussian mixture speaker models

Reynolds, D.A.; Rose, R.C.;

Speech and Audio Processing, IEEE Transactions on , Volume: 3 , Issue: 1 , Jan. 1995

Pages:72 – 83

[3] Speaker verification using adapted mixture models

Reynolds, D.A.; Dunn, R.B.; Quatieri, T.F.;

Digital signal processing, IEEE Transaction on, Volume: 10 , 2000

Pages:I-181 - I-202 vol.10

[4] Modelling prosodic dynamics for speaker recognition

Adami, A.G.; Mihaescu, R.; Reynolds, D.A.; Godfrey, J.J.;

Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03). 2003 IEEE International Conference on , Volume: 4 , 6-10 April 2003

Pages:IV - 788-91 vol.4

[5] The SuperSID project: exploiting high-level information for high-accuracy speaker recognition

Reynolds, D.; Andrews, W.; Campbell, J.; Navratil, J.; Peskin, B.; Adami, A.; Qin Jin; Klusacek, D.; Abramson, J.; Mihaescu, R.; Godfrey, J.; Jones, D.; Bing Xiang;

Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03). 2003 IEEE International Conference on , Volume: 4 , 6-10 April 2003

Pages:IV - 784-7 vol.4

[6] A study of generic models for unsupervised on-line speaker iindexing

Kwon, A.; Narayanan, S.;

[7] Modeling dynamic prosodic variation for speaker verification.

Sonmez, K.; Shriberg, E.; Heck, L.; Weintraub, M.;
Proceedings. (ICLSP '03). Volume: 7 , 2003
Pages:IV - 3189-3192 vol.7
[8] Using prosodic and conversational features for high-performance speaker recognition: report from JHU WS'02
Peskin, B.; Navratil, J.; Abramson, J.; Jones, D.; Klusacek, D.; Reynolds, D.A.; Bing Xiang;
Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03). 2003 IEEE International Conference on , Volume: 4 , 6-10 April 2003
Pages:IV - 792-5 vol.4
[9] Stream-weighted HMM for audio-visual ASR: a study on connected digit recognition
Chan, M.T
3rd IEEE Workshop on Multimedia signal processing, 1999
[10] Look who's talking: speaker detection using video and audio correlation
Cutler, R.; Davis, L.
IEEE International Conference Multimedia and Expo, 2000
[11] Integration of multimodal features for video scene classification based on HMM
Huang, J.; Liu, Z.; Wang, I.; Cheng, Y.; Wong, E.K.
3rd IEEE Workshop on Multimedia signal processing, 1999.
[12] Boosted Learning in Dynamic Bayesian Networks for Multimodal Speaker Detection
Garg, A.; Pavlovic, V.; Rehg, J.M.
Proceedings of the IEEE, 2003.
[13] Multimodal Speaker Detection using Error Feedback Dynamic Bayesian Networks
Pavlovic, V.; Garg, A.; Huang T.S.; Rehg, J.M.
Proceedings from IEEE Conference on computer vision and pattern recognition, 2000.
[14] Mutual disambiguation of Recognition Errors in a Multimodal Arquitecture
Sharon L.Oviatt
Conference on Human Factors in Computing Systems, 1999.
[15] Multimodal integration – A Statistical View

Lizhong Wu, Sharon L.Oviatt and Phillip R. Cohen
IEEE Transaction on Multimedia, 1999
[16] Audio-visual integration in multimodal communication
Tsuhan Chen; Rao, R.R.
Proceedings of the IEEE, 1998
[17] Automatic analysis of multimodal group actions in Meetings
Iain McCowan; Samyie Bengio; Daniel Gatica-Perez; Guillaume Lathoud
IIDIAP Research Report, 2003
[18] Segmenting Multiple Concurrent Speakers Using Microphone Arrays
Guillaume Lathoud ; Iain McCowan; Darren C. Moore
Proceedings of EuroSpeech, September 2003
[19] Audiovisual Speaker Tracking with Importance Particle Filters
Daniel Gatica-Perez; Guillaume Lathoud; Iain McCowan; Jean Marc Odobez; Darren C. Moore
Proceedings of the IEEE, 2003

8. Appendices

8.1. ICAASP' 05 Paper

The paper presented at ICAASP 2006 in Toronto is appended to this work in the following pages

8.2. Presentation plots

Because of the nature of the evaluation process of this thesis, the images and plots used in the oral defense of the project are also appended.

SMART ROOM: PARTICIPANT AND SPEAKER LOCALIZATION AND IDENTIFICATION

Carlos Busso, Sergi Hernanz, Chi-Wei Chu, Soon-il Kwon, Sung Lee*,*
Panayiotis G. Georgiou, Isaac Cohen, Shrikanth Narayanan*

Integrated Media Systems Center, Department of Electrical Engineering,
*Department of Computer Science
Viterbi School of Engineering, University of Southern California, Los Angeles

ABSTRACT

Our long-term objective is to create Smart Room Technologies that are aware of the users presence and their behavior and can become an active, but not an intrusive, part of the interaction. In this work, we present a multimodal approach for estimating and tracking the location and identity of the participants including the active speaker. Our smart room design contains three user-monitoring systems: four CCD cameras, an omnidirectional camera and a 16 channel microphone array. The various sensory modalities are processed both individually and jointly and it is shown that the multimodal approach results in significantly improved performance in spatial localization, identification and speech activity detection of the participants.

1. INTRODUCTION

New developments in communications technologies have brought to light a number of exciting and challenging applications that promise to change the way people communicate and interact. An application that has recently gained significant attention in the literature is the development of multimodal, unobtrusive *Smart Room Technologies* (SRT): monitor and infer important clues about users in specific environments such as their spatial position, identities and behavior. This is a challenging multidisciplinary application that involves research in diverse topics including object tracking, speaker activity detection, speaker identification, human action recognition and user behavior modeling.

One of the well-studied areas in SRT is the *detection and tracking of user locations*. Two important sources of information are the visual and the acoustic modality. Within a multimodal framework, these two sources have been used to track a single active speaker using methods such as *Sequential Monte Carlo* [1] [2], *Kalman filtering* [3] and *Dynamic Bayesian Networks* (DBN) [4], taking advantage of the complementary information represented by these two modalities.

Recently, [5] and [6] extended these approaches to track multiple speakers using particle filtering, while at the same time achieving *active speaker detection*, which is another important aspect of smart room technologies. In [7], visual clues were used to track users and a microphone array to select the active speaker by computing the distance between the visual and acoustic results.

Another important aspect of SRT is *speaker identification* (SID), in which the identity of the user is detected. There are several additional possible biometric systems for smart room applications (*e.g.*, retina, fingerprint), although most of them are impractical due to their invasive nature. One feasible option is to classify the user according to acoustic speech features [8] or through face recognition.

In this paper, we propose a real time multimodal approach to determine the spatial position of the user, detect speaker activity, and additionally determine the speaker's identity aimed at applications such as remote video-conferencing and audio-video indexing and retrieval for tasks such as meetings.

Our conference room contains three user monitoring systems: four synchronized cameras located in the corners of the room, a full-circle 360 degree camera located at the center of the table, and an array of sixteen microphones located at the end of the table. The location of each user is computed based on (i) the 3D polygon surface model from 4 synchronized cameras and (ii) a face detection technique using a full-circle 360 degree camera. Subsequently a dynamic model, under the Gaussian distribution assumption, is used with a moving window to combine the above information and *localize the participants*. The *Speaker ID*, operating on far-field sound obtained from the microphone array, algorithm employs a standard Gaussian Mixture model based on MFCCs. Finally, the active speaker's *identity and location* is estimated by fusing all the information channels.

The long-term objective of this project is to create a system which is cognizant of the users and can become an active but non-intrusive member of the interaction. The specific goal of this paper is to present a smart room design suitable for real time multi-speaker remote video-conferencing, with augmented information channels containing speaker IDs and relative location of the participants and the active speaker. Moreover, the extracted information can be used in a number of other applications such as video indexing and retrieval, human posture inference [9], modeling of human behavior, and as the device technologies further mature, for applications such as audio-visual speech and emotion recognition [10].

2. THE SMART ROOM

The present initial design primarily comprises microphones and cameras for activity sensing. The microphone array consists of 16 omnidirectional microphones that process sound at 48kHz sampling frequency. Fourteen microphones are distributed on a square frame of 50×50cm and two microphones are raised in the middle of the frame to allow for vertical plane localization. The room is acoustically treated on three walls and has a full-wall glass window on the other side, and has ceiling panels and carpeting on the floor.

The 3D camera system consists of 4 firewire CCD cameras near the corners on the ceiling that overlook the meeting area around the main table and capture the image sequences of the meeting from multiple angles. Each camera provides 1024×768 images at 15 frames per second, but we scale them to 320×240 for real time processing. The room is lighted with halogen lights.

At the center of the meeting table, a full-circle omnidirectional (360°) camera captures the faces of all participants. The size of the original omnidirectional image is 1280×960.

The next subsection describes the algorithms used to process each of these raw information sources.

2.1. Microphone array

One modality of localization is the sound source localization using a microphone array. The principle of sound source localization is based on the *Time Difference of Arrival* (TDOA) of the sound to the various sensors and the geometric inference of the source location

Fig. 1. Four firewire CCD cameras

from this TDOA. In this microphone array implementation we first estimate the pair-wise delays [11] and then employ a least-squares estimation procedure for the source localization[12].

Georgiou *et al*[11] have demonstrated the impulsive nature of audio signals and introduced a time delay estimation approach to mitigate its effects. The algorithm called *Fractional Lower Order Statistics-Phase Transform Method* (FLOS-PHAT) is based on a signed-non linearity on the input signal that reduces the detrimental effects of outliers.

As is common practice, this implementation of the FLOS-PHAT algorithm employs memory in order to approximate the expectation in the lower order statistics, and additionally the memory varies as a function of time to mitigate temporal propagation of errors.

Subsequently, based on the TDOA estimates, a computationally simple algorithm presented by Huang *et al*[12], called *One Step Least Squares* (OSLS), can be used to spatially locate the source using these pairs of delays.

The resulting localization algorithm is quite robust, but as expected, not very accurate in range due to the small aperture of the array (aprox. 70%, see Fig 6)). We expect, however, that this shortcoming will be countered by the visual modalities, which have higher accuracy in the horizontal plane.

2.2. Speaker ID

Speaker identification was implemented by analyzing the short-time spectrum (through mel frequency cepstral coeeficients, MFCCs) of the spoken phrases. In speaker recognition, the *Gaussian Mixture Model* (GMM), a weighted sum of Gaussian distributions, has been found to be good to capture the speaker information in MFCCs, and hence a GMM with 16 mixtures was used as a speaker model. Model training was accomplished by the standard *Expectation-Maximization* (EM) algorithm. All frames were initially divided into 16 clusters. An initial model was obtained by parameter estimation for mean and covariance matrices, which were estimated from the vectors in each cluster. The prior weights of GMM can be simply set by the proportion of feature vectors in each cluster. Next, the feature vectors are clustered by the *Maximum Likelihood* (ML) method using the previously estimated model. This process is iteratively executed until the model parameters converged. Additionally, we have created a silence/background noise model.

The speech signal was obtained through beamforming from the microphone array (see Fig. 4). The result of the speaker identification was in terms of pairs, (S_i, P_i), where P_i refers to the probability of speech activity of speaker S_i given for all speakers i. This information is evaluated and transmitted to the fusion algorithm every 1 second.

We should note that the acoustic signal processed is a reverberant, far-field signal corrupted by noise, and so the performance of this method is expected to be lower compared to a case when clean signal from a close-talking microphone is to be used.

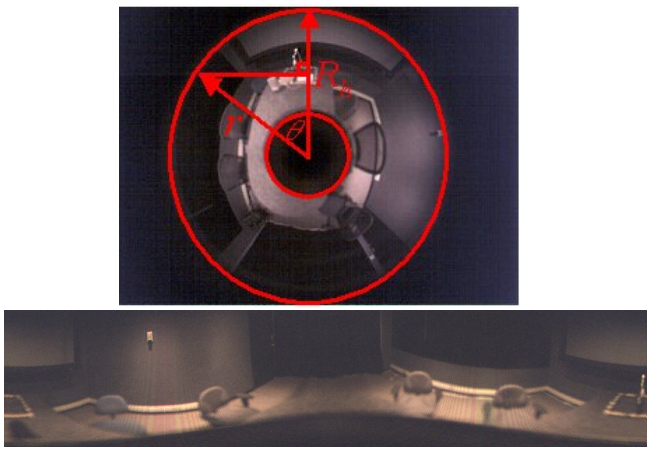


Fig. 2. Omnidirectional image from 360° camera and its panoramic transform

2.3. Video detection

The goal of visual tracking is to detect and track the 3D locations of the participants in the meeting room using video streams acquired by multiple synchronized cameras.

We use a Gaussian background-learning model to segment moving regions in the scene. When large variations from the learned Gaussian models are detected the foreground pixels are extracted. These pixel changes are then merged into regions. However, this method will segment actual people as well as their shadows and reflections. In our indoor setting, the shadow regions cast by diffused light do not have strong boundaries. We eliminate the shadows by combining the foreground pixels detection and the edge features detection [9] for segmenting into moving regions and corresponding cast shadows. The resulting regions are the silhouettes of the moving objects in the room.

The detected silhouettes across the views are integrated for inferring the 3D visual hulls of people in the room [13]. The silhouette contour is converted to a polygon approximation and a visual hull with polyhedral representation is then computed directly from these polygons [14]. This polygonal 3D approximation of the shapes is fast and is done in real-time. In detecting the locations of the people in the meeting room, we only need an estimation of general location of blobs of shapes instead of a precise reconstruction. Furthermore, we want the detection to cover an area as large as possible given a limited number of cameras. For this purpose we use a variation of the visual hull method proposed by [15]: the polyhedral visual hull is required to be the integration of only a subset (at least 3 out of 4) of the silhouettes instead of all of them. The resulting visual hull shape is less accurate, but the 3D shape of all people in the room can be approximated.

The computed visual hull is in a polygonal representation. We randomly sample points on the polygon surface and construct a height map of those points. This map assumes the XY plane in the Cartesian space is the meeting room floor and the Z coordinate represents the height. The local maxima of the height are then detected and considered as heads of the meeting participants. In this process some thresholds are applied to eliminate small regions such as moving chairs.

2.4. Full-circle 360-degree camera

We have added an omnidirectional (360°) camera on the meeting table to capture faces of all participants in order to get thumbnail representation of "who's talking". The image of the omnidirectional (360°) camera is the result of the projection of the surrounding scene into a hemisphere. We can unroll the captured original image and project it back onto a cylinder as in Fig. 2.

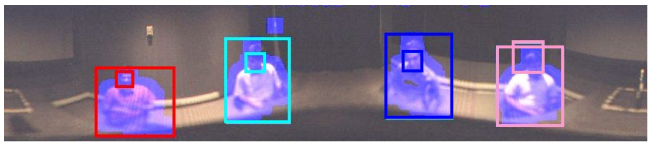


Fig. 3. Detection of participants' faces with the 360° camera

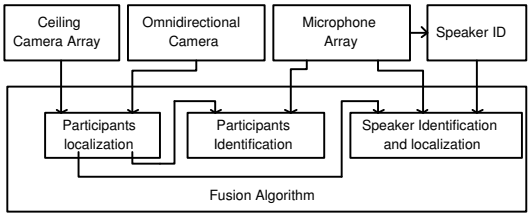


Fig. 4. The system is distributed running over TCP, with information exchange as depicted above.

To detect the foreground region, we use adaptive Gaussian background-learning model. All pixels in a new frame are compared to the current color distribution in order to detect moving blobs prior to capturing the faces. Morphological operators are used to group detected pixels into foreground regions, and small regions are eliminated. Pixel color distributions are updated in these regions for adapting the background model to slow variations. In these moving regions, we perform face detection. The face detector is based on Haar-like features and is implemented using Intel's open source computer vision library [16]. To accurately detect faces under low light level conditions, the color histogram of detected regions is normalized beforehand. Detected regions are then tracked using a graph-based tracking approach [17]. These regions correspond to the upper body of the meeting participants. Spatial and temporal information of tracking regions are combined as a graphical structure where nodes represent the detected moving regions and edges represent the relationship between two moving regions detected in two separate frames. Each newly processed frame generates a set of regions corresponding to the detected moving objects. The size of original omnidirectional image is 1280×960 and the panoramic image resolution is 848×180. The average size of detected faces is approximately 30×30. The faces are detected and tracked at approximately 13 FPS in a 2.8 GHz Pentium4 PC. In Fig. 3 we show an example of detection and tracking of the participants' faces during a meeting.

2.5. Synchronization

Each modality was initially processed independently and asynchronously. Therefore, the estimated 3D coordinates from the polygonal representation (X_v) and from the microphones array (X_{MA}), the angles of the faces detected (X_θ), and the speaker information from the acoustic analysis (S_i, P_i) are sent to the fusion algorithm for integration. Although the results are received in an asynchronous manner, they are transformed and processed in a synchronous fashion.

3. MULTIMODAL INTEGRATION

The various modalities are subsequently received and processed by a fusion algorithm for the purpose of finding and tracking the participants' spatial locations and identifying where and who the current active speaker is. Fig. 4 shows the information flow between the various modules, and what information is used for each decision.

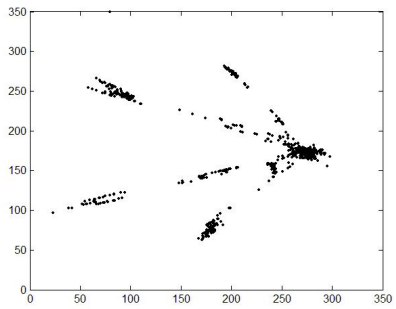


Fig. 5. Microphone Array Distribution

3.1. Participant localization

It is well known that visual tracking algorithms have better spatial resolution than acoustic localization techniques [7, 6]. Hence, our algorithm for localization of all the participants' location employs a dynamic visual approach that uses only information obtained by cameras $X = (X_v, X_\theta)$. Based on the distribution of the samples X, we model the position of each speaker as a multidimensional Gaussian distribution.

A single distribution with covariance K of a significant spread and mean M is initialized at the center of the room. As data are obtained, the variance and mean converge to the detected object's location. When information is received for a location scoring below a certain threshold of belonging to the existing distribution, a new multidimensional Gaussian is initialized at (M, K). The process continues sequentially until all the speakers are detected, with new data points either spawning new participant models or adapting the existing ones. In addition, temporal filtering ensures that false participant detections are identified and removed. This procedure allows us to determinate not only the spatial positions of the participants (X_P), but also the number of participants in the room (N_P).

3.2. Participant Identification

The spatial location of the current speaker (X_{MA+P}) as obtained from the microphone array (X_{MA}) and participants' location information (X_P), as well as the speaker ID from the GMM algorithm (S_i, P_i) are used to determine the identities of the participants. The goal is to detect who the participants are and also correlate their identity with their location in space (derive the "seating arrangement").

Fig. 5 shows a sample scatter plot of the raw microphone array localization X_{MA}, and as can be observed, the range information is highly noisy. For simplicity, we model $P(C_i|X_{MA})$, the probability that the acoustic source comes from cluster i given X_{MA}, as a multidimensional Gaussian distribution centered at the locations X_P and with a large variance in range and smaller variance in the other two dimensions.

Using (S, P), the probabilistic identity of the participant along with $P(C|X_{MA})$, the probabilistic location of the current speaker, over time and with physical constraints[1] we estimate the participants seating arrangement (L).

3.3. Speaker Identification and Localization

We compute activity speaker detection by employing all modalities: X_{MA+P}, which is derived from the visual modality and the microphone array, and (S, P) obtained from the acoustic analysis of the signal. The information is fused as described in (1), where r_{ij} is the correlation measure between the probabilities of the current speaker belonging in cluster j and being speaker i.

[1] Such that a participant can only be at one point in space at a time, and one position can only be occupied by one participant at a time

		Session	Strong Decision	Weak Decision	Decision
A	Speaker ID (GMM based)	1	58.30%	60.70%	ID
		2	56.70%	58.40%	ID
B	Microphone Array + Video	1	68.10%	69.50%	Loc
		2	71.00%	72.00%	Loc
C	Microphone Array + Video + Speaker ID (Assumes known seating arrangement L)	1	73.20%	76.50%	Loc+ID
		2	77.90%	79.50%	Loc+ID
D	Microphone Array + Video + Speaker ID (Participant location (L) learned through data)	1	73.80%	77.60%	Loc+ID
		2	74.90%	76.70%	Loc+ID
E	Speaker-location learned through data (L)	1	93.30%		L
		2	94.90%		L

Fig. 6. All of the above results are obtained in real time, and include the whole length of the meeting, with *no* time given for initial convergence. **A**: Speaker ID as obtained purely from the speech signal using a GMM; **B**: Localization obtained by the two visual information channels and the microphone array; **C**: Speaker Identification & Localization based on all information channels. Assumes perfect knowledge of L, the seating arrangement of the participants; **D**: As C, but the mapping of speaker-location, L, is continuously estimated from the data; **E**: Speaker Location mapping, L.

$$P(S_i) = P_i \cdot \sum_{i}^{n} r_{ij} \cdot P(C_j|X_{MA}) \quad (1)$$

4. RESULTS & DISCUSSION

The experiments were performed using two meetings (each 5 minute long) with four participants, processed in real time. Off-line computations were also performed later for comparison purposes. The conversation in the meetings was casual with many interuptions, overlaps and short utternaces, making this an extremely challenging task for both the microphone array and the Speaker ID. We used two criteria: strong decision, in which the detection was considered correct if the speaker was active at least 50% of the time interval, and weak decision, in which the detection is considered correct if the speaker was active in any part of the time interval.

The participants localization algorithm takes about 3 seconds per participant to converge during the start of the meeting. As can be observed from the results in Fig. 6 (rows C & D), the speaker identification and localization based on all the modalities is fairly robust, achieving about 70% performance. This is a significant improvement of about 30% compared to the speaker ID based purely on the speech signal as shown in row A, which suffers from the far field and noisy nature of the data.

Similarly, there is a significant improvement in the accuracy of localization (row B) as contrasted to the performance based purely on the microphone array. The microphone array as a single modality is very unreliable when it comes to the range of the speaker (as can be observed from Fig. 5), both due to the noisy environment as well as the range errors due to the aperture of the array. The multimodal localization accuracy is also further improved by the acoustic speaker ID modality, as the correlation between active speaker and sound source location is providing additional information. This results in about 10% improvement when comparing all modalities (row C & D) versus the visual and microphone array only (row B).

Finally, the identification of the participants' spatial arrangement (row E) is extremely accurate, a fact that explains the very close results observed in rows C & D.

5. CONCLUSION

In this paper, we have presented the first results from the smart room that we are developing at USC. We have demonstrated that complementary modalities can increase the general participant identification and localization (without any prior knowledge of the number of participants) including the active speaker identification and localization.

Our goal of increasing the system's awareness of the users in the space has many more challenges ahead. In our future work we propose to investigate further integrated recognition technologies including face recognition, gesture recognition and head pose estimation. Additionally we plan to collect and share with the research community a multimodal data corpus from this testbed.

6. REFERENCES

[1] J. Vermaak, M. Gangnet, A. Blake, and P. Perez, "Sequential monte carlo fusion of sound and vision for speaker tracking," in *International Conference on Computer Vision*, 2001, vol. I, pp. 741–46.

[2] D. Zotkin, R. Duraiswami, and L.S. Davis, "Multimodal 3-d tracking and event detection via the particle filter," in *IEEE Workshop on Detection and Recognition of Events in Video*, 2001, pp. 20 – 27.

[3] S. Spors, R.Rabenstein, and N.Strobe, "Joint audio-video object localization and tracking," *IEEE Signal Processing Magazine*, vol. 18, no. 1, pp. 22 31, Jan 2001.

[4] V. Pavlovic, A. Garg, J.M. Rehg, and T.S. Huang, "Multimodal speaker detection using error feedback dynamic bayesian networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2000, vol. II, pp. 34 – 41.

[5] N. Checka, K. Wilson, M. Siracusa, and T. Darrell, "Multiple person and speaker activity tracking with a particle filter," in *International Conference on Acoustics, Speech, and Signal Processing*, 2004, vol. V, pp. 881–84.

[6] D. Gatica-Perez, G. Lathoud, I. McCowan, J.-M.Odobez, and D. Moore, "Audio-visual speaker tracking with importance particle filters," in *International Conference on Image Processing*, 2003, vol. III, pp. 25–8.

[7] G. Pingali, G. Tunali, and I. Carlbom, "Audio-visual tracking for natural interactivity," in *Proceedings of the seventh ACM international conference on Multimedia*, Orlando, Fl, 1999, p. 373 382.

[8] S. Kwon and S. Narayanan, "A study of generic models for unsupervised on-line speaker indexing," in *Proceedings of IEEE Automatic Speech Recognition and Understanding Workshop*, 2003, pp. 423–28.

[9] I. Cohen and H. Li, "Inference of human postures by classification of 3d human body shape," in *IEEE International Workshop on Analysis and Modeling of Faces and Gestures*, 2003, pp. 74–81.

[10] C. Busso, Z. Deng, S. Yildirim, M. Bulut, C.M. Lee, A. Kazemzadeh, S. Lee, U. Neumann, and S. Narayanan, "Analysis of emotion recognition using facial expressions, speech and multimodal information," in *Sixth International Conference on Multimodal Interfaces ICMI 2004*, State College, PA, 2004.

[11] P. G. Georgiou, P. Tsakalides, and C. Kyriakakis, "Alpha-stable modeling of noise and robust time-delay estimation in the presence of impulsive noise," *IEEE Transactions on Multimedia*, vol. 1, no. 3, pp. 291–301, September 1999.

[12] Y. Huang, J. Benest, and G. Elko, "Adaptive eigenvalue decomposition algorithm for realtime acoustic source localization system," in *International Conference on Acoustics, Speech, and Signal Processing*, Phoenix, AZ, 1999, vol. II, pp. 937–40.

[13] A. Laurentini, "The visual hull concept for silhouette-based image understanding," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 16, no. 2, pp. 150162, Feb 1994.

[14] W. Matusik, C. Buehler, and L. McMillan, "Polyhedral visual hulls for real-time rendering," in *In Proceedings of Eurographics Workshop on Rendering*, 2001.

[15] E. Boyer and J.-S. Franco, "A hybrid approach for computing visual hulls of complex objects," in *Computer Vision and Pattern Recognition (CVPR'03)*, 2003, vol. I, pp. 695–701.

[16] A. Kuranov, R. Leinhart, and V. Pisarevsky, "An empirical analysis of boosting algorithms for rapid objects with an extended set of haar-like features," in *Intel Technical Report MRL-TR-July02-01*, 2002.

[17] I. Cohen and G. Medioni, "Detecting and tracking moving objects for video surveillance," in *Computer Vision and Pattern Recognition (CVPR'99)*, 1999, vol. II, pp. 319–325.

Robust Feature Extraction for Multimodal Speaker Recognition

Sergi Hernanz Nogueras
Shrikanth S. Narayanan

Overview

- New scenarios for speaker recognition
- Meeting rooms
  - Meetings indexation
  - Remote access to a meeting
- New constraints
  - Low SNR
  - Fast speaker changes
- Old algorithms are not enough for these new problems

Current Solutions

- Feature-Based approach
  - A set of features can define all properties of speech
- Multimodal approach
  - New sources can add new information to the classification problem

    - Both solutions are based on a posterior parallel classification.

Joint classification

- Problem appears when new features are added for classification
- A set of measures can join the same classification space with the previous ones, or create a different classification space.
  - Feature-level combination
  - Semantic-level combination
- A few questions to solve
  - What performance can be reached with semantic or feature-level combination?
  - Which methods and algorithms can be applied to joint classification?

Bayes bound

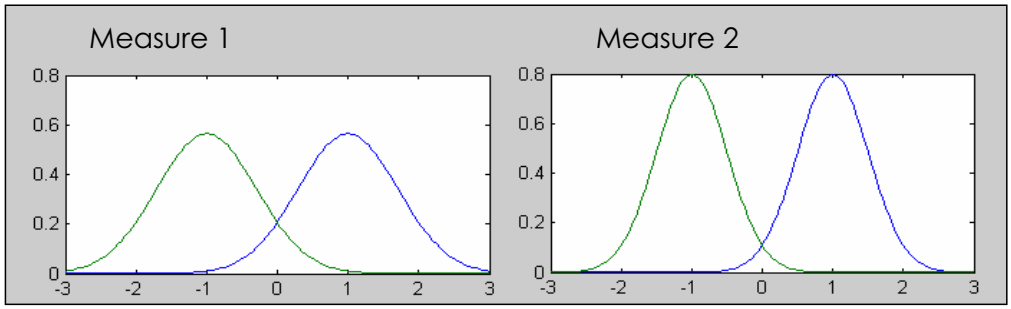
- Bayes classification; optimal statistical solution

$$\underset{j}{\operatorname{argmax}}\left\{P\left(\theta_j \mid \bar{x}_1, \bar{x}_2 \ldots \bar{x}_n\right)\right\} = \underset{j}{\operatorname{argmax}}\left\{\frac{P\left(\bar{x}_1, \bar{x}_2 \ldots \bar{x}_n \mid \theta_j\right)P\left(\theta_j\right)}{P\left(\bar{x}_1, \bar{x}_2 \ldots \bar{x}_n\right)}\right\} = \underset{j}{\operatorname{argmax}}\left\{P\left(\bar{x}_1, \bar{x}_2 \ldots \bar{x}_n \mid \theta_j\right)\right\}$$

- A simple example

$$L_{X_1}\left(x_1 \mid \theta_0\right) = \frac{1}{\sqrt{2 \cdot \pi \cdot \sigma_{11}^2}} \exp\left(-\frac{1}{2}\frac{x_1 - 1}{\sigma_{11}^2}\right) \qquad L_{X_2}\left(x_2 \mid \theta_0\right) = \frac{1}{\sqrt{2 \cdot \pi \cdot \sigma_{22}^2}} \exp\left(-\frac{1}{2}\frac{x_2 - 1}{\sigma_{22}^2}\right)$$

$$L_{X_1}\left(x_1 \mid \theta_1\right) = \frac{1}{\sqrt{2 \cdot \pi \cdot \sigma_{11}^2}} \exp\left(-\frac{1}{2}\frac{x_1 + 1}{\sigma_{11}^2}\right) \qquad L_{X_2}\left(x_2 \mid \theta_1\right) = \frac{1}{\sqrt{2 \cdot \pi \cdot \sigma_{22}^2}} \exp\left(-\frac{1}{2}\frac{x_2 + 1}{\sigma_{22}^2}\right)$$



Semantic bound

- Having access just to posterior probabilities, the classification can not be as accurate as feature-level classification
- Solution is a function of prior probabilities given by previous classifiers

$$P\left(\bar{x}_1, \bar{x}_2 \ldots \bar{x}_n \mid \theta\right) = f\left(P\left(\bar{x}_1 \mid \theta\right), P\left(\bar{x}_2 \mid \theta\right), \ldots, P\left(\bar{x}_n \mid \theta_j\right)\right)$$

- There's a need to analyze how accurate the solution is

$$SE = \iint\left[P\left(\bar{x}_1, \bar{x}_2 \ldots \bar{x}_n \mid \theta\right) - f\left(P\left(x_1 \mid \theta\right), \ldots, P\left(x_n \mid \theta\right)\right)\right]^2 \partial x_1 \partial x_2 \ldots \partial x_n$$

Simple example (I)

- To evaluate bounds, the example is taken
- Bayes bound is set assuming joint Gaussian distribution for both noises.

$$L_{X_1,X_2}(x_1,x_2\mid\theta_0)=\frac{1}{2\cdot\pi\sqrt{\sigma_{11}\sigma_{22}-\sigma_{12}^2}}\exp\left(-\frac{1}{2}\begin{pmatrix}x_1-1\\x_2-1\end{pmatrix}\begin{pmatrix}\sigma_{11}&\sigma_{12}\\\sigma_{12}&\sigma_{22}\end{pmatrix}^{-1}\begin{pmatrix}x_1-1\\x_2-1\end{pmatrix}\right)$$

- Semantic bound can be analytically found.

$$f(x_1,x_2\mid\theta_0)=f(-x_1,x_2\mid\theta_0)=f(x_1,-x_2\mid\theta_0)$$

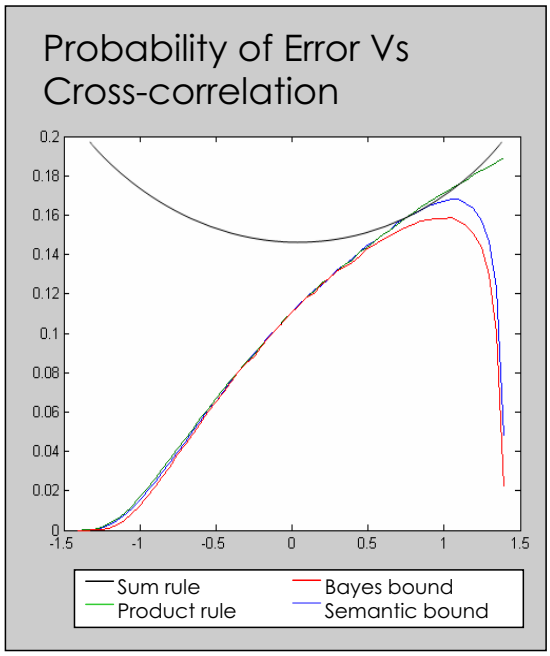
$$SE=\iint\left[P_{even-even}(x_1,x_2\mid\theta_0)-f(P(x_1\mid\theta_0),P(x_2\mid\theta_0))\right]^2\partial x_1\partial x_2$$

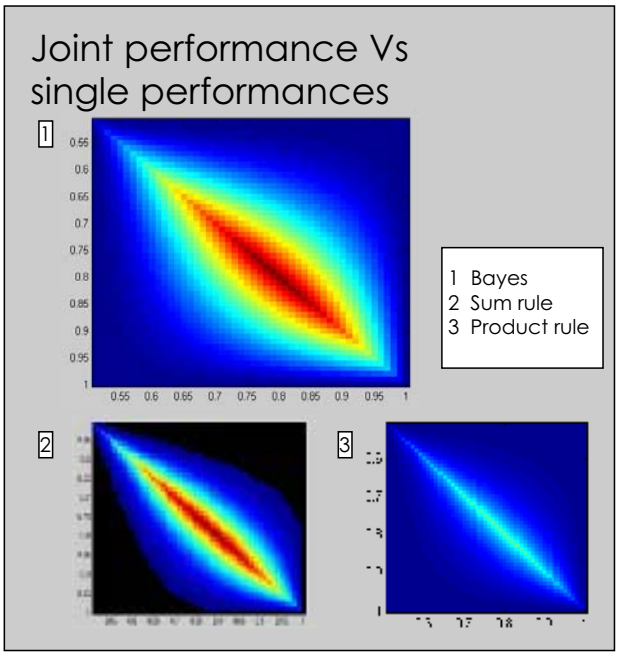
$$P_{ee}(x_1,x_2\mid\theta_0)=\frac{1}{4\pi\sqrt{\sigma_{X_1X_1}\sigma_{X_2X_2}-\sigma_{X_1X_2}{}^2}}\exp\left(\frac{\sigma_{X_2X_2}x_1{}^2+\sigma_{X_1X_1}x_2{}^2}{2(\sigma_{X_1X_1}\sigma_{X_2X_2}-\sigma_{X_1X_2}{}^2)}\right)\left[\exp\left(-\frac{\sigma_{X_1X_2}x_1x_2}{\sigma_{X_1X_1}\sigma_{X_2X_2}-\sigma_{X_1X_2}{}^2}\right)+\exp\left(\frac{\sigma_{X_1X_2}x_1x_2}{\sigma_{X_1X_1}\sigma_{X_2X_2}-\sigma_{X_1X_2}{}^2}\right)\right]$$

$$x_1=\sqrt{-2\sigma_{X_1X_1}\ln\left(\sqrt{2\pi\sigma_{X_1X_1}}P(x_1\mid\theta)\right)}$$

$$x_2=\sqrt{-2\sigma_{X_2X_2}\ln\left(\sqrt{2\pi\sigma_{X_2X_2}}P(x_2\mid\theta)\right)}$$

Simple example (II)



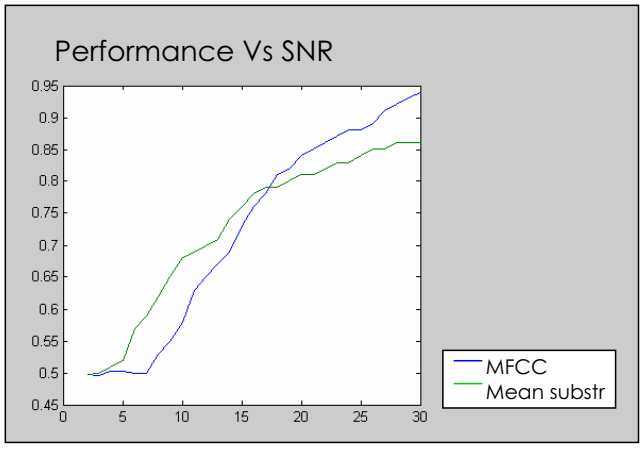


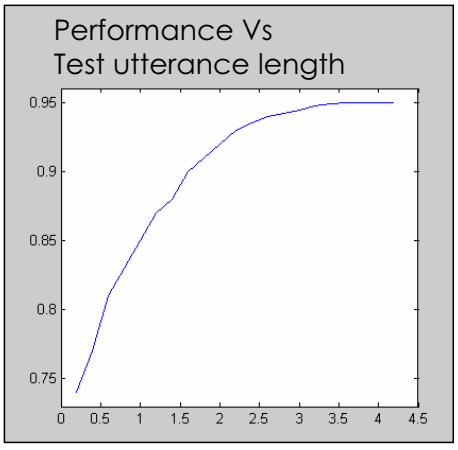
Conclusions

- Correlation of input noise sources doesn't necessary mean correlation of errors
- Hard to deal at semantic level with
  - Dependencies between input data
  - Classifiers with disperse performances
- Product rule is not that bad
  - Ensures better performance than alone classifiers
  - Incorrelation in multimodality is usual
- Proposed method: supervised learning
  - The searched f is a high order function
  - A parametric function f can be trained in order to fit joint pdf
  - Because of the lack of the pdf, usually output of training data is used to compute error function

Feature-based approach

- MFCC are the basic features from speech
- Noise affects to features. It has a strong effect over classification performance
- Speaker dependant statistics need long-term analysis. Length of test frame





Addition of features

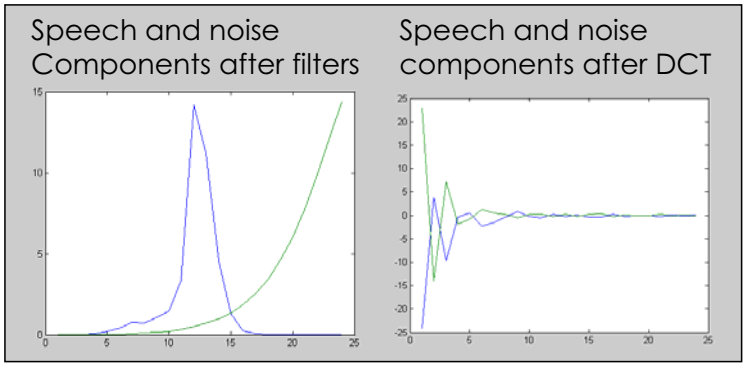
- Basically there are two reasons to add new features to a system
  - Addition of new information
  - Robustness against noise
- Procedure to evaluate a new set of features
  - Intra-speaker dependency. Utility for classification
  - Mutual information. Joint classification
  - Robustness to noise
- Questions to solve
  - Does exist a set of features completely representative of the speech signal?
  - Which representations or methods to extract these features are less affected by noise?

Possible speech features

- Classical speech production model is based on a sound source and a vocal tract filtering
  - Filter properties are well represented by MFCC
  - Source properties have been modeled and used for speaker recognition. Pitch and energy dynamics
- Other features have been applied to speaker ID. Most of them used text transcription
  - Phonetic, lexical and conversational statistics
- Articulatory features has not been used for speaker recognition
  - They are expected to add robustness to acoustic-based current systems

Noise on MFCC

- AWGN added to speech contaminates MFCC
- SNR is equal along MFCCs



- DCT concentrates energy of speech and voice shapes in the same coefficients
  - Selective spectral-shape information compaction could separate robust and noisy features

MFCC Baseline

- NIST Database
  - Noisy meeting
  - 30% for training, 70% testing
- Acoustical statistics
  - GMM with 16 mixtures
  - MFCC 24 filters
- Evaluation
  - Assuming ideal silence detector 73%
  - Modeling silence with GMM 61%

Articulatory features (I)

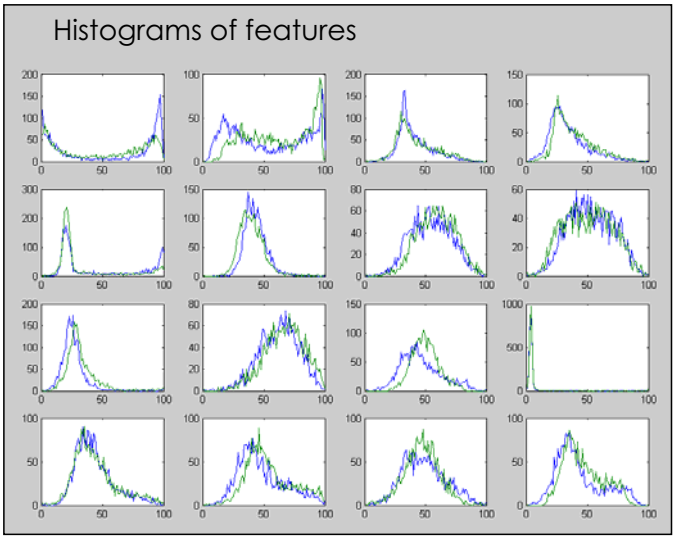
- Features extracted
  - Voicing: voiced-voiceless
  - Place: labial-dental-velar-glottal-alveolar-back-central-frontal
  - Manner: approximant-stop-fricative-affricate-nasal-vowel
  - Vowels high: frontal, central, back
  - Vowels open, closed
  - Vowels rounding: rounded, unrounded
- TIMIT database for training detectors.
  - 32 speakers: 16 male and 16 females
  - 10 utterances each speaker
- Methods
  - MLP: 3-layer NNs applied to 39 coefficients(13 MFCCs including energies and its 1st and 2nd derivatives)
  - HMM: 3-state and 24 mixtures applied to 39 coefficients

Articulatory features (II)

- Switchboard database for Background model.
  - 16 speakers: 8 males and 8 females
  - 90 seconds of speech each speaker
- Performance over 2 speaker conversations
  - 10-15% of error on 30% of pairs
  - 40% of error on 70% of pairs
  - Overall performance is low
- Joint Classification with MFCC reveals no new information in articulatory features

Discussion

- The most of the classification performance comes from a few features
  - Voicing and rounding are the most important
- Classification is accurate in male-female pairs
- Derivatives doesn't improve the accuracy
- Long test utterance is better



Multimodal approach

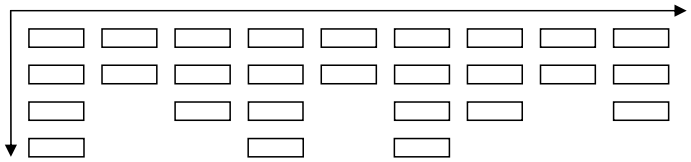
- Some scenarios can take advantage of information from other sources. Meeting Room
- New sources for speaker recognition
  - Source and people localization
  - Video information
- CommVision Lab
  - 16-channel microphone array
  - 4 cameras focusing the center of the room
  - One 360º camera in the center of the room

Front modules

- Acoustic-based speaker recognition
  - Received prior probabilities of each of the speaker models in the database and silence model (1Hz)

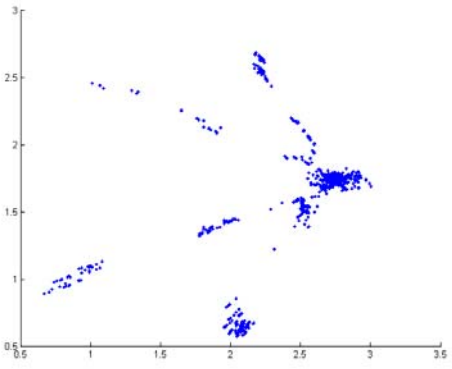
$$L(x_a(t) \mid S_i) \quad i = 1 \ldots N+1$$

- TDA-based source localization
  - Received most probable coordinates for incoming voice (12 Hz) $\bar{x}_{MA}(t)$
- People localization
  - Received a set of coordinate vectors each frame (15 Hz)

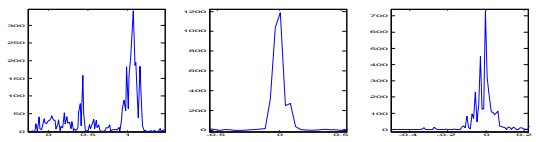


Statistical models - Source localization

- A parametric model is used for the received data from the microphone array
  - Position of the speaker and 'noise centre' help us to transform samples coordinates to new ones. Models assume them independent



$$\bar{s}_1 = \frac{\bar{v}_s - \bar{v}_0}{\left\| \bar{v}_s - \bar{v}_0 \right\|^2} \qquad \bar{s}_2 = \frac{\frac{\bar{s}_1}{\left\| \bar{s}_1 \right\|} \times \hat{z}}{\left\| \frac{\bar{s}_1}{\left\| \bar{s}_1 \right\|} \times \hat{z} \right\|} \qquad \bar{s}_3 = \frac{\bar{s}_1}{\left\| \bar{s}_1 \right\|} \times \bar{s}_2$$



$$L(s_1) = \alpha_1 \mathrm{N}(s_1 \mid 0, \sigma_{S10}) + \alpha_2 \mathrm{N}(s_1 \mid 1, \sigma_{S11})$$

$$L(s_2) = \mathrm{N}(s_2 \mid 0, \sigma_{S2})$$

$$L(s_3) = \mathrm{N}(s_3 \mid 0, \sigma_{S3})$$

Statistical models - People localization

- The way video data arrives is modeled

$$\bar{x}_{V1}(t) = \zeta \cdot \bar{v} + (1-\zeta)\bar{\chi}$$

ζ is a binary random variable being 0 for a false alarm with probability P_{fa}, and 1 otherwise.

$\bar{\chi}$ is a three coordinates uniform random variable distributed along all the room space

$$L(\bar{v}) = \sum_{i=0}^{N} \alpha_i \mathrm{N}(\bar{v} \mid \bar{\mu}_i, \bar{\bar{\Sigma}}_i)$$

N is the number of speakers in the room

α_i is the global probability of being speaking speaker i

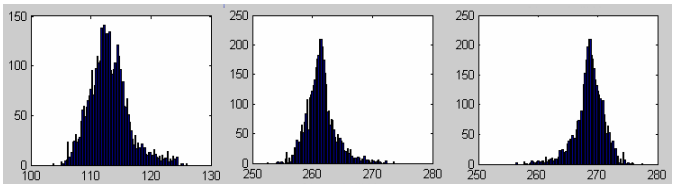
$\bar{\mu}_i$ is the position of speaker i

$\bar{\bar{\Sigma}}_i$ is a fixed covariance matrix analyzed from the error generated by the video system

P_{nd} is the probability of a present speaker being not detected in a frame

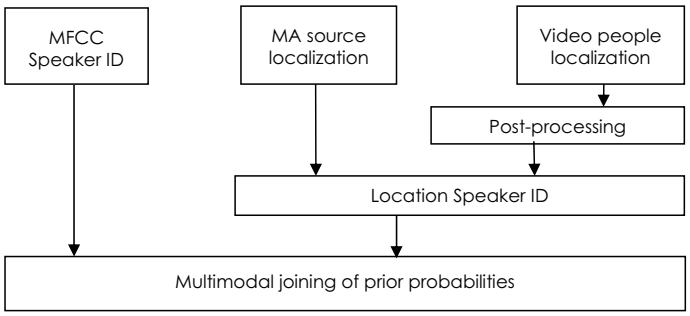
$$P_{fa} = 0{,}15 \qquad P_{nd} = 0{,}2$$

$$\bar{\bar{\Sigma}} = \begin{pmatrix} 0.0232 & 0 & 0 \\ 0 & 0.0246 & 0 \\ 0 & 0 & 0.0330 \end{pmatrix}$$



General scheme

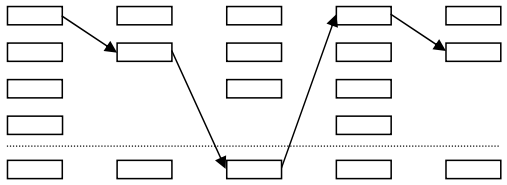
- Sound and physical space are two dimensions where a classification is possible
- Configuration of the modules forces a semantic level approach
- Position of people is needed as a first step. Reliable from video data



  - Noise can create dependencies between spaces, then the fusion module should deal with this

People localization (I)

- Steady state assumption
- Noise filtering and mean estimation
- State-search based, addition of misdetection state



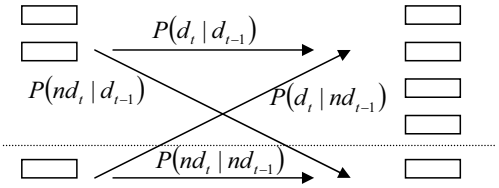
$$O_j^t = \arg\max_O \left\{ \prod_{\substack{\tau=t-4 \\ i \subset O}}^{t} L\left(\overline{v_i^\tau} \mid S_j\right) \right\} = \arg\max_O \left\{ \prod_{\substack{\tau=t-4 \\ i \subset O}}^{t} \mathrm{N}(\overline{v_i^\tau} \mid \overline{\mu_j}^t, \overline{\overline{\Sigma_j}}) \right\}$$

$$P\left(nd \mid \overline{v_1^t}, \overline{v_2^t} \ldots \overline{v_n^t}\right) = \frac{P\left(\overline{v_1^t}, \overline{v_2^t} \ldots \overline{v_n^t} \mid nd\right) P(nd)}{P\left(\overline{v_1^t}, \overline{v_2^t} \ldots \overline{v_n^t}\right)}$$

$$P\left(d \mid \overline{v_1^t}, \overline{v_2^t} \ldots \overline{v_n^t}\right) = P(d) \sum_{i=1}^{n} \frac{P\left(\overline{v_i^t} \mid d_i\right)}{P\left(\overline{v_i^t}\right)}$$

$$\overline{\mu_j}^{t+1} = \sum_{\substack{\tau=t-4 \\ i \subset O_j^t}}^{t} \overline{v_i^\tau}$$

- Uniform assumptions for simplicity $O_j^t = \arg\min_O \left\{ \sum_{\substack{\tau=t-4 \\ i \subset O}}^{t} d\left(\overline{v_i^\tau}, \overline{\mu_j}^t\right) \right\}$
- State-transition weighting
  - Allows tracking
  - Minimizes lost of speakers



People localization (II)

- Speaker moving out hypothesis
  - 5 continuous misdetections are assumed as a speaker lost $Pe < \left(P_{nd} + P(d(\overline{v_i}, \overline{\mu_j}^t) > d_{nd})\right)^5 = 0.00032 + \varepsilon$
- New speaker hypothesis
  - False alarms are used to train a spatial speaker model. In case it fits enough, it's added to the room
  - Gaussian model updating

Closer false alarm $\overline{\mu_{fa}^{t+1}} = (1-\alpha)\overline{\mu_{fa}^t} + \alpha\overline{v_c^t}$

$$\overline{\Sigma_{fa}^{t+1}} = \begin{pmatrix} \sigma_{xx}^{t+1} & 0 & 0 \\ 0 & \sigma_{yy}^{t+1} & 0 \\ 0 & 0 & \sigma_{zz}^{t+1} \end{pmatrix} \qquad \sigma_{xx}^{t+1} = \sqrt{(1-\alpha)\sigma_{xx}^t + \alpha\left(\overline{v_{cx}^t} - \overline{\mu_{fa_x}^t}\right)^2}$$

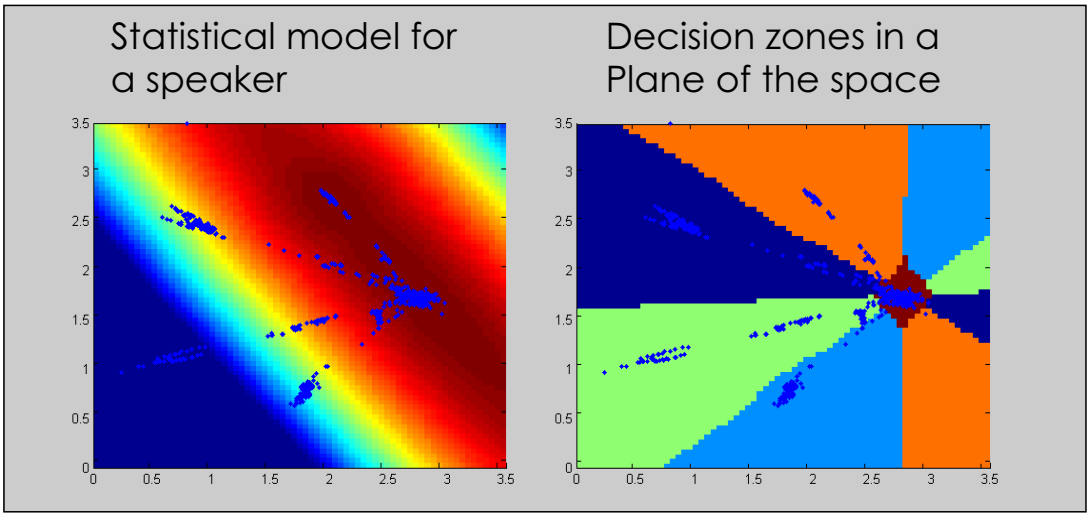
No false alarm $\overline{\mu_{fa}^{t+1}} = (1-\beta)\overline{\mu_{fa}^t} + \beta\overline{\mu_0}$

$$\overline{\Sigma_{fa}^{t+1}} = \begin{pmatrix} \sigma_{xx}^{t+1} & 0 & 0 \\ 0 & \sigma_{yy}^{t+1} & 0 \\ 0 & 0 & \sigma_{zz}^{t+1} \end{pmatrix} \qquad \sigma_{xx}^{t+1} = \sqrt{(1-\beta)\sigma_{xx}^t + \beta\sigma_{0x}{}^2}$$

  - Variance lower than a threshold = new speaker
  - Convergence conditions and speed of convergence set the parameters

Spatial speaker classification

- Locations estimated from video used as MA parameters
  - Decision based on comparison of prior probabilities
  - Gaussian model added for silence



Classes relation

- Classes in spatial domain are not related to spectral-shape domain classes
  - Need of a measure of correlation between them

$$a_j[n] = \frac{P(x_a[n] \mid \theta_j) - E\{P(x_a[n] \mid \theta_j)\}}{E\left\{P(x_a[n] \mid \theta_j)^2\right\}} \quad with\ j = 1..N$$

$$MA_j[n] = \frac{P(x_{MA}[n] \mid \theta_j, \hat{\mu}) - E\{P(x_{MA}[n] \mid \theta_j, \hat{\mu})\}}{E\left\{P(x_{MA}[n] \mid \theta_j, \hat{\mu})^2\right\}} \quad with\ j = 1..N$$

  - Search of correct pairs through correlation matrix

$$R = \begin{pmatrix} R_{a_1 MA_1} & \dots & R_{a_1 MA_N} \\ \dots & \dots & \dots \\ R_{a_N MA_1} & \dots & R_{a_N MA_N} \end{pmatrix} + R_0 \longrightarrow R'$$

Mathematical Transformation Based on SVD

Search over each column of new correlation matrix to create pairs

Joint classification

- Selection of function forced by dynamic range of the prior probabilities given by both systems

$$P(\bar{x}_a, \bar{x}_{MA} \mid \theta) = [P(\bar{x}_a \mid \theta)]^{\alpha} [P(\bar{x}_{MA} \mid \theta)]^{\beta}$$

- Sweep of parameter values for a parameter selection

Results

- Database
  - 2 conversations of 5 minutes to evaluate the total system
  - 4 conversations, total of 20 minutes for spatial speaker recognition evaluation
- People's position estimation
  - Initial time to converge 3s
  - Lost of speaker per minute 0,025
- Spatial speaker recognition
  - Performance 78%
- Joint classification
  - With previous classifiers working at 78% and 63%. Final accuracy of the system performed at 81%.

Future work

- Joint classification
  - Study of performances of different functions families
  - Application of supervised learning and monte-carlo methods
- Features from speech
  - New extraction methods for articulatory features
  - Evaluation of dynamics of articulatory features
- CommVision Project
  - Sampling of the ML source localization function with the estimated people location
  - Accurate people tracking system
  - Addition of spatial information in statistical models