Universidad
Politécnica
de Cartagena

Investigating partially supervised emotion classification from speech, utilizing self-training methods and fuzzy SVMs.

Diploma thesis

Escuela Técnica Superior de Ingeniería de Telecomunicación
Universidad Politécnica de Cartagena

presented by: José Domingo Esparza García

2011

Supervisor:	Dr. Jorge Larrey
Supervisor:	Dr. Stefan Scherer
Day of submission:	Day. Month 2011

Abstract

Human-machine interaction relies on many different communication channels in order obtain a good performance in terms of information comprehension. Audio and video channels are most informative for humans due to the large amount of information available. As for machines, however, there exist many context-dependent variants that make the task extremely more complex, as it might be the speaker's current emotion. Emotions can largely affect the speaker's expressions and should, therefore, be considered as such. The work conducted for this thesis is aimed at studying automatic emotion classification from speech data using fuzzy support vector machines.

Furthermore, traditional automatic classification approaches rely on large training sets that allow the learning of the patterns behind the generative distribution of the data. Since data labeling can represent an extremely expensive task, different techniques have been proposed in the literature for reducing the cost of it. This study also comprises evaluation of different partially supervised training methods, from which conclusions can be extended to other classification problems, considering that they are standard procedures applicable to any field of study.

References to original texts and publications are provided on every section of this thesis, being each of them a recommended way to acquire deeper knowledge on any of the fields of study to which this work is related.

Acknowledgements

Thanking all the people who contributed in some measure to this work is, indeed, a very difficult task.

First I would like to thank my friend and work supervisor Stefan Scherer. He gave me the opportunity to come to Ulm to do my thesis when he knew me no more than just a few hours and always had time to help me, even for the most insignificant things. I sincerely appreciate this blind trust and really hope that he enjoyed our working together as much as I did. This time would not have been the same without him and, definitely, I would not be able to cook candy were it not for his practical lessons.

I thank also Dr. Jorge Larrey, for supervising my thesis at the UPCT. There are many reasons why I esteem him and fully trust his advice. His lectures were always inspiring and, without doubt, influenced largely my career which is only starting now. His trust in me for this work is extremely appreciated and, hopefully, soon enough we will be able to have 餃子 together again.

Further, very special thanks to Dr. Friedhelm Schwenker, who had to deal with endless problems to regularise my situation at the Ulm University. Even more, he was a great source of ideas every time my work seemed to reach dead ends. Equally important to me was the help received from the always charming International Relations office staff at the UPCT. Not only did they help me to make my stay in Ulm possible but also made Japan a reality for me.

There are many people who did not participate directly in the work conducted for this thesis, but did make it possible with their support and good advice. Therefore, I would like to take the opportunity to thank them here, too.

If there are two people who should be thanked above any other, these are definitely my parents. I have requested help from them a countless number of times and they have not yet a single time declined to provide it. My sisters also should have a special mention here, as well as my two nieces. Unfortunately I do not get to see any of them very

often, mainly due to my continuous eagerness to keep myself away from home. They, however, keep supporting me in the distance (God bless Skype). Still now, I can not find better memories than those of the moments spent at home with them.

I consider myself a very lucky person if I think of the large amount of people that I could call my friends. This, unfortunately, makes it impossible to name them all here. There is, however, some of them that could, in no case, be skipped.

In the first place, my friends from Fuente Alamo. Always there, no matter how far I go or how long it takes me to come back, you are always there ready for it.

Secondly, all the people that I met in Cartagena. They definitely made my time in the city unforgettable. Special mention should be made of Alfredo Calvete and his family, who were my second family for five years, both in good and bad times. There are only good things that I could say about them. Also David Ardid and José Ángel García deserve to be thanked, for dealing with my grumpiness and bad humour so many times.

To all the people that I met in Japan, from all over the world, thanks for making that year so special, 本当にありがとうございます.

Last but not least, all my new friends in Germany, who made my stay in Ulm really entertaining. Vielen dank.

Contents

1 Introduction

Emotion classification is performed by humans in all kinds of situations but, even for us, it is not always an easy task. Literature shows that perception tests conducted on humans do not always produce error-free results (Wendt, 2007). On the contrary, there exist cross-class confusion patterns that can be observed with a certain frequency. Therefore, it is crucial to find features, and classification approaches comparable to the human perception, because otherwise it might happen that the machine recognizes expressions based on artifacts and not on actual modulation caused by humans' affective state.

In this work, we aim at emulating human perception capabilities to show that by means of choosing appropriate feature sets and exhaustive training, similar accuracies and confusions may be obtained by using large enough training sets. The experiments are based on multi-classifier multi-class support vector machines, combining eight separate feature sets, based on standard datasets. Joint use of the feature sets requires some kind of fusion approach to optimise the amount of information that they can provide. A basic fusion is proposed and used in the experiments, providing good results. The training process, however, implies a tedious labelling process conducted by an expert which, in general, may represent a very expensive and time consuming labour. Nevertheless, obtaining unlabelled samples does not necessarily incur in high costs. For this reason, there is continuous research being conducted with the aim of using unlabelled data for training. Within this basic idea, there exist different approaches and research lines, each of them focusing on different properties of the training process. There is previous research conducted, for example, on semi-supervised learning, where both labelled and unlabelled data are used for model training (Druck et al., 2008; Zhu, 2005; Blum and Mitchell, 1998), unsupervised learning, where only unlabelled data is used (eg. Clustering algorithms - Duda et al. (2001)) or active learning, where the system is allowed to choose its training data from a pool of samples (Lomasky et al., 2007).

Within the partially-supervised learning framework, in this work we also study both a semi-supervised approach based on k-nearest neighbor algorithm and an active learning approach. In order to better analyse the wellness of the proposed methods, a confidence measure for artificially generated labels is proposed, providing an estimate of how correct or informative an artificial label is, depending on the case.

There exist several applications in the real world that could make use of the emotion classification results presented in this work, for example, as feedback for a telephone-based service or, moreover, model adaption for a particular human-machine interaction. As for the partially supervised experiments, the application of the results extends not only to speech-based tasks, but to any classification task that requires a learning process, being able to cut down the cost of it by reducing the required number of labels.

Therefore, the objectives of this thesis may be summarised as:

- Find a good combination of feature sets that can resemble the human perceptive capabilities.
- Develop, if necessary, new feature sets to improve cross-class confusions.
- Create a multi-class classifier for the task of emotion recognition.
- Define a fusion to combine the different decisions obtained in every different feature domain.
- Study partially supervised techniques within the proposed system architecture.

The structure of this thesis is organized as follows: Chapter 2 introduces a description of the datasets utilized for this work, Chapter 3 gives a theorical and implementational overview of all the used methods. The experimental setup is described and the results are reported in Chapter 4. Obtained results and comparison of the automatic classification performances with the human perception are discussed in Section 4.3. Chapter 5 presents a summary of the conducted work and results that concludes this thesis.

2 Datasets

Discussion about the type of data to use arose in initial steps of this study. Emotion classification could easily comprise the use of different data modalities such as speech, video or biometric measures (Keltner and Ekman, 2003; Keltner et al., 2003). Previous research proves that speech data alone can provide good results for this task without incurring in difficult synchronisation procedures. Speech datasets utilized in this work are described in this chapter. They provide good reference sets of emotions, comprising the most representative ones with a good audio quality. However, there are only acted emotions leading to a scenario not highly realistic. For this reason, human perception tests were conducted on both sets. In this case they are used for comparison with the results of the experiments.

2.1 WaSeP Corpus

The speech dataset used for this work is the ``Corpus of spoken words for studies of auditory speech and emotional prosody processing" (WaSeP©) Wendt and Scheich (2002). The dataset is based on the German language and is structured in the following way: A first part of the corpus contains standard German nouns. A second part of it, contains phonetically balanced pseudo words that represent the German phonetical rules. Only the pseudo words subset has been used for this work. It consists of 222 words, spoken by a male and female speakers imitating six natural human emotions: neutral, joy, sadness, anger, fear and disgust. The speech was not recorded on real emotional environments but rather on an acted basis within an acoustic chamber. In order to evaluate the validity of the acted emotions, a group of 74 native German speakers were asked to listen to the recordings and emit a decision for each of the samples based on their perception. An average accuracy of 78.53% was achieved in this case. The results of this tests can be seen in Table 2.1 in the form of confusion matrix, representing each row the amount of data from a given emotion that is recognised as belonging o each of the defined

6 classes. Further explanation of how to interpret this table is given in Chapter 4. This table provides a good representation of the average accuracy of humans recognizing the emotions in the database as well as the most common confusions for those misclassified recordings. The original set was sampled at 44.1 kHz and quantified with 16 bits. For our experiments, the data was resampled to 16 kHz. An example of the raw audio data with its corresponding spectrogram is shown in Figures 2.1 and 2.2, respectively.

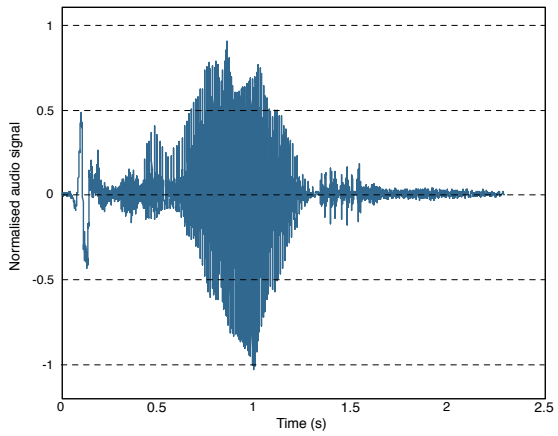

Figure 2.1: Example of one of the audio signals used from the WaSeP corpus: normalized raw data resampled to 16kHz.

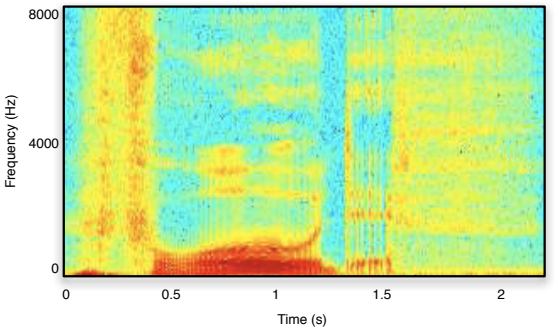

Figure 2.2: Example of one of the audio signals used from the WaSeP corpus: spectrogram of the audio signal resampled to 16kHz.

Table 2.1: Confusion matrix of the human performance test generated from the available labels for each of the utterances listed in the WaSeP database. Wendt (2007).

	F	D	H	N	S	A
Fear	**.77**	.01	.08	.03	.10	.01
Disgust	.05	**.72**	.06	.03	.07	.07
Happiness	.01	.00	**.75**	.22	.02	.00
Neutral	.01	.02	.05	**.79**	.00	.13
Sadness	.05	.01	.04	.13	**.76**	.01
Anger	.01	.03	.00	.01	.01	**.94**

Table 2.2: Confusion matrix of the human performance test generated from the available labels for each of the utterances listed in the Database of German Emotional Speech.

	F	D	H	N	S	A	B
Fear	**.85**	.04	.03	.03	.01	.04	.00
Disgust	.03	**.79**	.01	.04	.08	.02	.02
Happiness	.02	.02	**.83**	.06	.01	.05	.06
Neutral	.00	.00	.01	**.87**	.04	.02	.06
Sadness	.06	.02	.00	.06	**.78**	.00	.08
Anger	.01	.01	.01	.01	.00	**.96**	.00
Boredom	.00	.01	.00	.00	.11	.03	**.85**

2.2 Berlin Database of Emotional Speech (EmoDB)

To compare the obtained results with a more widely known dataset, the ``Database of German Emotional Speech (EmoDB)'' has been used as reference approach in this work. The speech data in this Database includes recordings from ten actors, both male and female. Both short and long sentences are utilized and they account for seven different emotions (the same as in the WaSeP corpus plus boredom). EmoDB was recorded in an anechoic chamber at the Technische Universität Berlin, Technical Acoustics Department. The audio was recorded at 48 kHz using a Sennheiser MKH40 P48 microphone and a Tascam DA-P1 portable DAT recorder and later down-sampled to 16 kHz Burkhardt et al. (2005). This database has been the basis of many analysis Scherer et al. (2008, 2007); Vlasenko et al. (2007); Wagner et al. (2007). An example of the raw audio data with its corresponding spectrogram is shown in Figures 2.3 and 2.4.

As for the WaSeP dataset, human perception tests were conducted with this corpus and the confusion matrix is shown in Table 2.2.

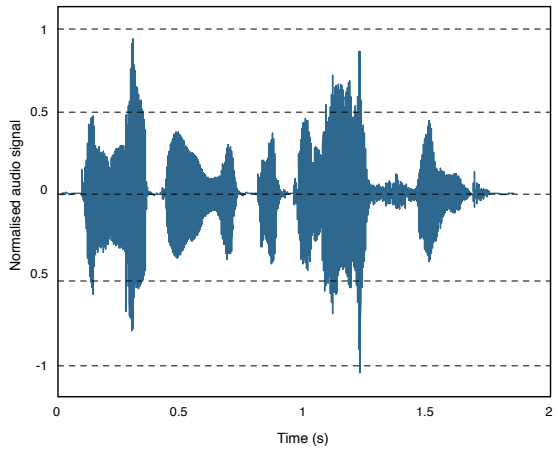

Figure 2.3: Example of one of the audio signals used from the EmoDB corpus: normalized raw data resampled to 16kHz.

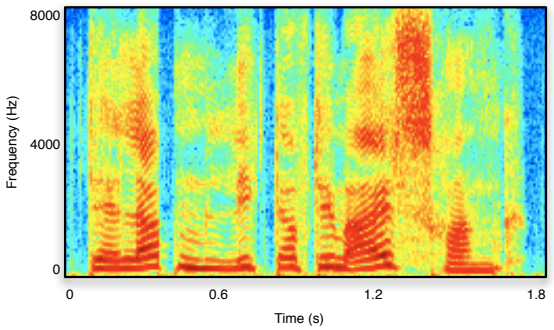

Figure 2.4: Example of one of the audio signals used from the EmoDB corpus: spectrogram of the audio signal resampled to 16kHz.

As already mentioned, the described datasets represent acted emotions and, therefore, differences with real emotional situations are to be expected. Nevertheless, these sets are largely utilised by the research community so they provide a good reference for this study.

3 Methods

Due to the differences in the nature of the problems faced in this work, a number of features, algorithms and classification techniques are utilized in a combined style. This chapter provides a basic theoretical description of all methods and procedures used during the experiments with a more detailed explanation for those that represent a novelty in the field of study.

3.1 Features

Selection of appropriate features for each specific purpose is likely to improve results notably, if compared with those obtained with default featuresets that were designed for different purposes. In similar work to this, different combinations of audio features are said to perform well in classification of audio data (Li et al., 2001). Nevertheless, within the scope of this study a specific analysis of speech characteristics and its quantification has been conducted with the aim of better comprehending the human emotions nature as well as the demodulation artefacts used for recognizing them (Scherer et al., 2003; Banse and Scherer, 1996). Given the characteristics of the used datasets, the chosen features for this work are described in this section.

3.1.1 Mel-frequency cepstral coefficients (MFCC, ΔMFCC)

MFCC are extensively used within the Automatic Speech Recognition (ASR) community due to their good performance in applications of different nature (Fang et al., 2001; Logan, 2000). Their first order derivatives (ΔMFCC) are also frequently used since they are more robust against noise effects or biases. MFCCs are obtained from a representation of the power spectrum on the Mel scale. This is a scale of frequencies where the pitch distances are based on the perception of

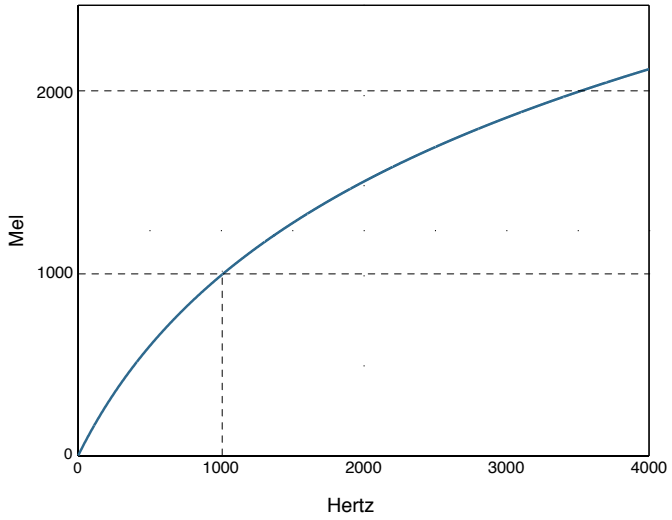

Figure 3.1: Representation of the Mel scale with respect to the Hertz scale.

humans, following a logarithmic function which is commonly assumed to be represented by the formula given in Eq. 3.1.

$$m = 2595 \log_{10}\left(1 + \frac{f}{700}\right) \tag{3.1}$$

As a reference, correspondence of 1000 Hertz to 1000 Mels is established. A representation of the relation Mel-Hertz can be seen in Figure 3.1.

The Mel-frequency cepstral cepstrum represents the energy of the signal in different frequency bands, uniformly distributed in the Mel scale. The coefficients (MFCC) are obtained by following these steps:

1. Calculate DFT of the windowed signal and obtain the power spectrum.
2. Apply a bank of triangular filters equally spaced in the Mel scale.
3. Obtain the Log-Energy of the output from every band-pass filter.
4. Compute Discrete Cosine Transformation (DCT).

The DCT used in this steps is a transformation based on a sum of cosine functions that is commonly used for data compression. There

exist different variants and implementations of it but, for this work, the expression in Eq. 3.2 is utilised:

$$MFCC_i = \sum_{k=1}^{K} X_k \cos\left(i\left(k-\frac{1}{2}\right)\frac{\pi}{K}\right), i = 1, ..., M \tag{3.2}$$

where M, K and X_k represent the number of MFCC coefficients, the number of bands considered and the Log-Energy of the k-th band, respectively. A representation of the MFCC features extraction process is represented in Figure 3.2.

3.1.2 Modulation Spectral Features

Based on the MFCC extraction architecture, these parameters represent the demodulated signal spectrum at different frequencies in order to measure how fast and in which amount the spectrum changes over time. Different bands of the spectrum may contain information about different speech properties and, therefore, separate analysis and consideration of them is likely to lead to a feature set with a large amount of information, as studied in Scherer et al. (2003). A time series of band energies is calculated and a second level Fourier analysis is conducted over these, leading to band coefficients that meassure the amount of band-energy changes with respect to the total energy oscillations. Figure 3.3 explains in detail how these features are obtained following these steps:

1. Short-Term Fourier Transformation (STFT) of the audio signal.
2. Band-filter the spectrum of each frame with filters equally spaced over the Mel frequency.
3. Log-Energy computation of each of the band-pass spectrums obtained for each frame.
4. Frame aggregation of the band Energies.
5. FFT computation of each Energy sequence.
6. Log-Energy computation of the sequence.
7. Ratio band energy over total produces a coefficient for each of the defined bands.

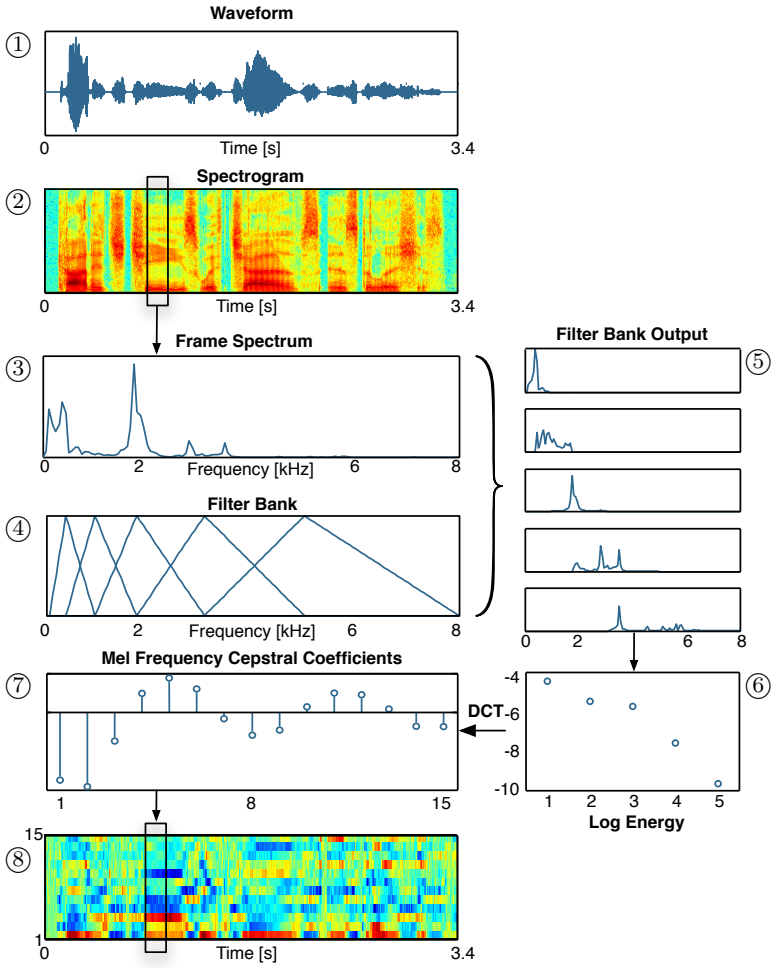

Figure 3.2: MFCC feature extraction algorithm. From the speech signal ① the short-time Fourier transformation (STFT) is calculated ②. The spectrum of each frame ③ is taken through a bank of triangular filters ④ equally spaced in the Mel frequency scale (see Figure 3.1). For each filtered signal ⑤, the log-energy is calculated ⑥ and the discrete cosine transformation (DCT) of these values represents the MFCC coefficients of the given frame ⑦. The aggregation of MFCCs over all frames then forms the MFCC features of the speech sample ⑧.

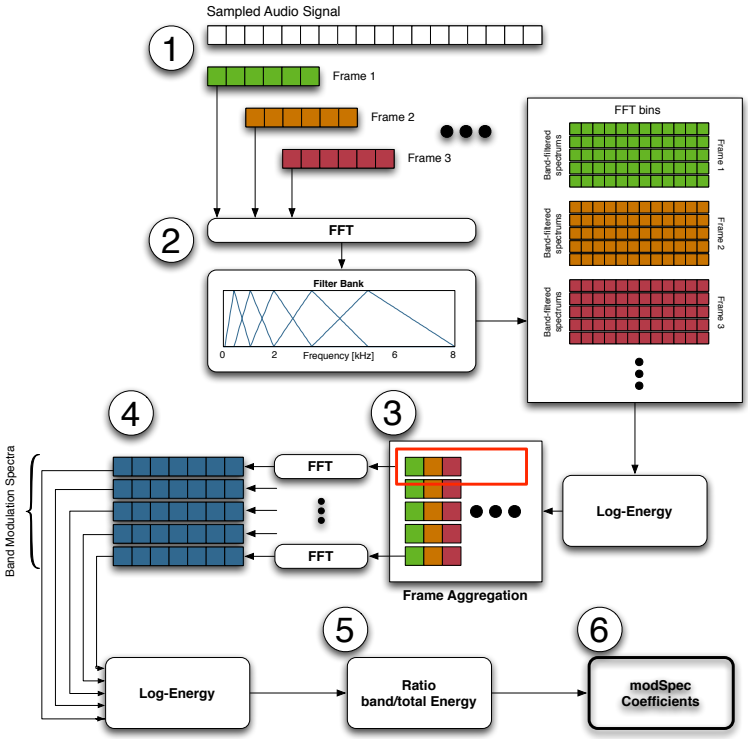

Figure 3.3: Modulation spectral feature extraction algorithm. From the sampled speech signal ① the fast Fourier transformation (FFT) is calculated at every frame. The spectrum of each frame is taken through a bank of triangular filters ② equally spaced in the Mel frequency scale (see Figure 3.1). For each frame, band-pass log-energies are calculated. Frame aggregation is performed to obtain sequential band-pass log-energies ③. For each frequency band, a new FFT is computed ④. Log-energy is once again calculated at each band together the ratio of all of them with respect to the total ⑤. The ratios are directly considered the modulation spectral coefficients ⑥.

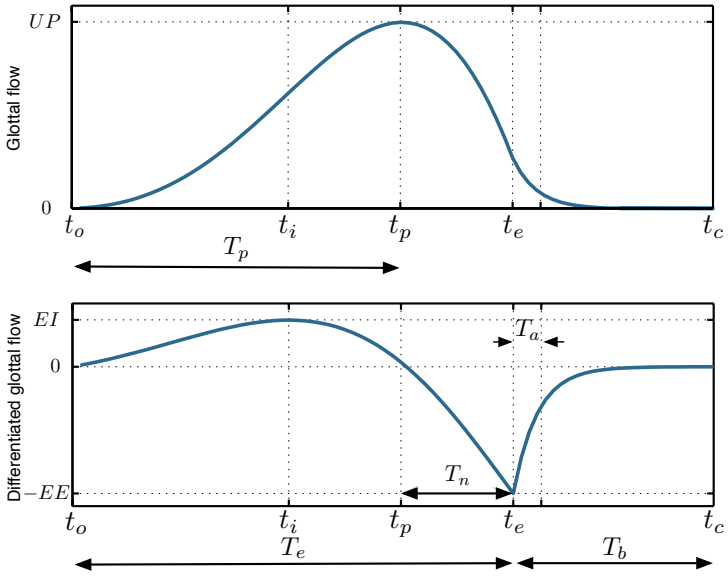

Figure 3.4: Example of a glottal flow (top) and differentiated glottal flow (bottom) of a Liljencrants-Fant (LF) model pulse.

3.1.3 Fundamental Frequency, f_0

It is possible to obtain different values of fundamental frequency (f_0) on each time frame. f_0 stands for the frequency value that holds most of the spectral energy within the considered frame. From the f_0 trail, different statistics are calculated: mean, standard deviation, maximum and quartile values, forming all these together the feature set. Extraction of f_0 is performed making use of the f_0 tracker which is available in the ESPS/waves+ software package.

3.1.4 Voice Quality

Voice Quality features are a group of parameters commonly assumed to be responsible for the different speaking styles (Kane et al., under review, 2010; Gobl et al., 2002; Yanushevskaya et al., 2008). These features can be estimated from the glottal source signal, modelled often with the Liljencrants-Fant (LF) model, as represented in Figure 3.4.

Full description of the LF model is out of the scope of this thesis but definition of the parameters used to comprise the feature set is

provided in the following. Further description of the features is given in detail in Scherer et al. (under review).

- EE: maximal glottal closure speed.
- T_p: elapsed time since the beginning of the glottal opening until the maximal aperture is reached.
- f_0: fundamental frequency, as described in 3.1.3.
- T_e: elapsed time since the beginning of the glottal opening until the moment when the maximal glottal closure speed is reached.
- T_a: fraction $\frac{1}{e}$ of the time interval since the maximal closure speed is reached until its complete closure.

Defined these model parameters, the values to be used as features can be calculated following Eqs. 3.3, 3.4 and 3.5:

$$R_g = \frac{1}{2T_p \cdot f_0} \tag{3.3}$$

$$R_k = \frac{T_e - T_p}{T_p} \tag{3.4}$$

$$R_a = T_a \cdot f_0 \tag{3.5}$$

The feature set will be the combination of EE, R_g, R_k and R_a.

3.1.5 Energy

The frame average energy is calculated using a window size of 32 ms with an overlap of 16 ms. For each frame n, the following formula is used to calculate the frame energy:

$$E_n = \frac{1}{W} \sum_{w=1}^{W} x^2[w] \tag{3.6}$$

where $x[w]$ and W represent the signal and frame duration respectively. Similar statistics to those of f_0 are used for this featureset.

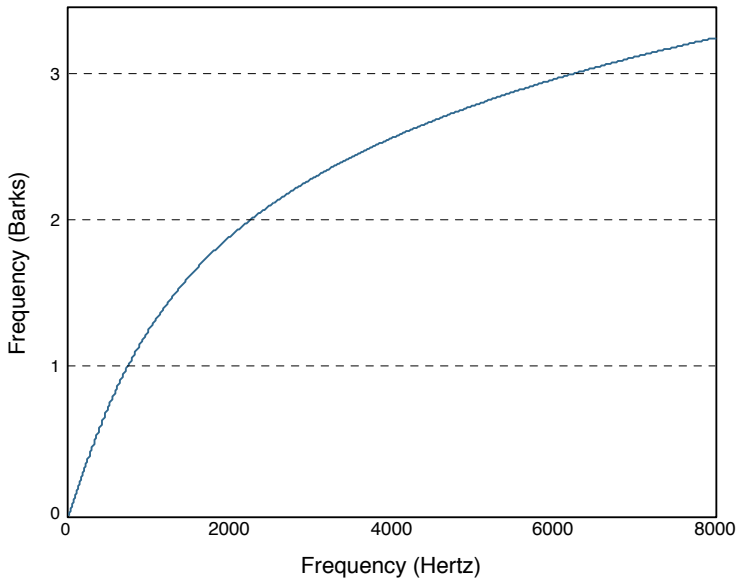

Figure 3.5: Relation Barks - Hertz, as given by Eq. 3.7.

3.1.6 Perceptual Linear Predictive Analysis, PLP

This set of features is based on the autoregressive all-pole model, traditionally used among the automatic speech recognition community for short-time power spectrum estimation. This model, obtained by linear predictive analysis, is capable of estimating the spectrum in frequencies with a high energy, commonly corresponding to the vocal tract formants. Nevertheless, the power spectrum is approximated with a similar accuracy over all frequency bands, which does not correspond to the human auditory capabilities. PLP features were designed with the intention of better representing the perceptive amplitude levels and spectral resolution over all frequencies, as described in Hermansky (1990), Hermansky and Morgan (1994). In a similar way to the MFCC feature extraction, a bank of filters is utilized for band analysis. These filters however, are not equally spaced in the Mel frequency, but in the Bark frequency, which is given by the Equation 3.7 and a representation is given in Figure 3.5.

$$b = 6 \cdot \text{arcsinh}\left(\frac{f}{600}\right) \tag{3.7}$$

A representation of the PLP feature extraction process can be seen in Figure 3.6.

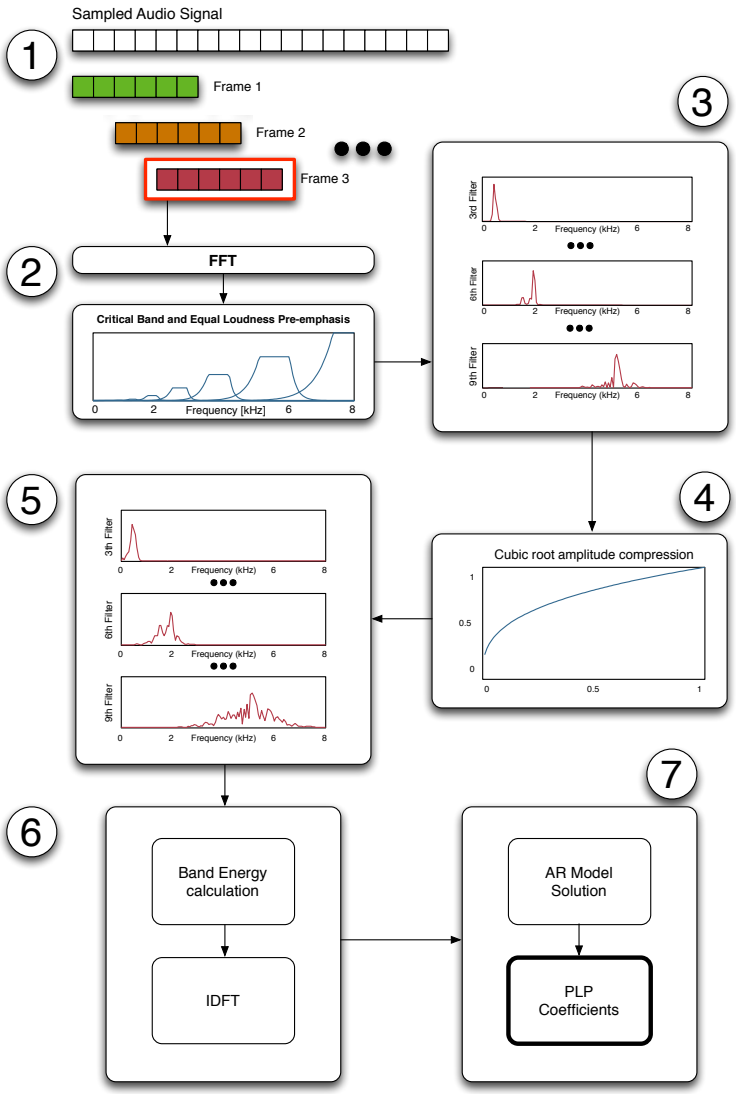

Figure 3.6: PLP features extraction algorithm. From the sampled speech signal ① the fast Fourier transformation (FFT) is calculated at every frame. The spectrum of each frame is taken through a bank of filters ② equally spaced in the Bark frequency scale (see Figure 3.5). Each band-pass signal ③ is compressed in amplitude following a cubic root function ④. From the amplitude-compressed spectrums ⑤, log-energy for is calculated for every band and inverse discrete Fourier transformation (IDFT) is computed ⑥. Solution of the autoregressive (AR) model is performed, being the obtained values the PLP feature coefficients ⑦.

The following, are the steps required for extraction of these features (as explained in Hermansky (1990)):

1. Short-Term Fourier Transformation (STFT) of the audio signal.
2. Critical band analysis in the Bark frequency.
3. Equal loudness pre-emphasis.
4. Intensity-Loudness conversion by cubic root amplitude compression.
5. Inverse discrete Fourier transformation.
6. Solution for autoregressive coefficients.

3.1.7 Periodicity

Previous analysis of the utilized dataset showed that some of the emotional expressions can be discriminated by using the segment lengths as a feature (Wendt (2007)). Since our aim is to find features that can be obtained in any context, the use of this property would result in unfair improvement of our results. Therefore, we developed a new feature set that can partly resemble that information.

Considering the number of syllables per second we do not incur in the use of unfair measures as it can be estimated from the signal directly and therefore in future applications. Syllable detectors may be implemented in different ways as shown in Pfitzinger et al. (1996); Crystal and House (1990); Cedergren and Perreault (1994). The utilized approach in this study is described in the following:

Let us assume that each syllable contains at least one vowel. If we consider the high periodicity that characterizes vowels in contrast to consonants, detecting speech segments with a high periodicity would give us markers of the syllables. A straight forward approach for obtaining a periodicity value is to compute the auto-covariance function in smaller sub-segments of the original speech. Once this step is completed, the sub-segments can be grouped according to their periodicity score as periodic or non-periodic. In order to achieve this, we designed a double-threshold system to resemble a hysteresis cycle (Mayergoyz (2003)), as can be seen in Figure 3.7. This system marks the beginning of a periodic zone when a value over 80% of the maximum is found. In a similar way, the start of a non-periodic zone will be detected by the presence of a lower value than 30% of the maximum.

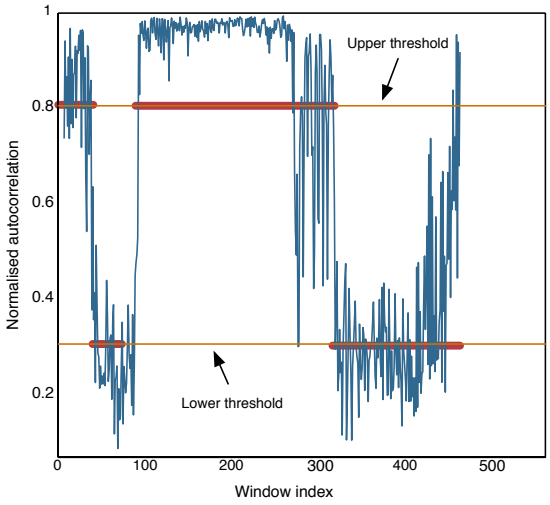

Figure 3.7: Periodicity featureset. Blue: autocorrelation function over consecutive time windows (each of 5ms). Orange: Upper and lower thresholds for identifying binary states. Red: States detection, high and low levels represent periodic and no-periodic respectively.

Additionally, for detecting syllables we consider energy variations over time. While assuming lower energy segments within the syllable boundaries, one may use an envelope detector and, once again, the double-threshold system to spot the syllables, as shown in Figure 3.8.

With the initial speech segment divided into periodic and non-periodic subparts, as well as, low and high energy parts, we calculate: the lengths of the parts, the largest difference in width, and the energy ratio of the parts to the total length. The combination of those features resembles the full periodicity feature set.

3.2 Sequential Data

Analysis of time series is a complex task due to the sequential properties of the data. Due to the time dependency, the observations cannot be considered as statistically independent and, therefore, a complex model capable of dealing with this kind of data seems necessary.

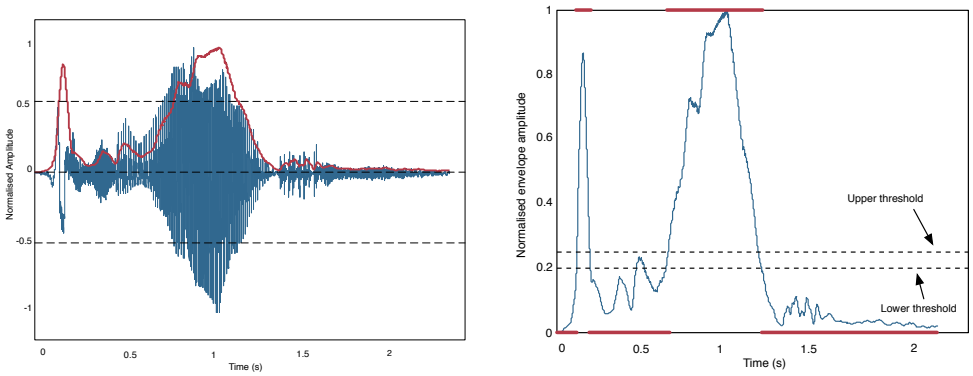

Figure 3.8: On the left: normalised raw data (in blue) and detected envelope (in red). On the right: Detected envelope (in blue), threshold values for identifying binary states (dashed line), and high - low segments detected (in red).

3.2.1 Hidden Markov models

A commonly used statistical model for sequential data are the hidden Markov models (HMM). These are assumed to be Markov models with non observed states. Markov models are stochastic models based on the assumption of the memory-less property. According to this property, future states depend only on the current state and not on the history of the system, as described in Eqs. 3.8, 3.9.

$$p(x_n \mid x_1, ..., x_{n-1}) = p(x_n \mid x_{n-1}) \tag{3.8}$$

$$p(x_1, ..., x_N) = \prod_{n=1}^{N} p(x_n \mid x_1, ..., x_{n-1}) = p(x_1) \prod_{n=2}^{N} p(x_n \mid x_{n-1}) \tag{3.9}$$

where x_n represents the model state in time n. There also exist higher order extensions to the Markov models, where not only the current state is considered to predict the future states, but also the $m-1$ previous ones, where m represents the model order. This extension, however, is not relevant for out work and will therefore be omitted. In standard Markov models the states are visible, being the only parameters of the system the transition probabilities. In the case of HMMs, the states are only partially observed. Observations are related to the state of the system but are not sufficient to specify it. HMMs can be defined in terms of the following elements:

1. Number of states, N: represents the number of hidden states assumed for the HMM.
2. State transition probability distribution, A: there exist a set of probabilities associated with each of the states that defines the state transitions on every instant.
3. Observation symbol probability distribution functions (pdf), b: represents the output pdf of each of the states. Given a particular state i, this is the distribution that generates the observations. It can be considered a discrete or continuous distribution depending on the observations to be modelled.
4. Initial state probability distribution π: for the case of no previous state.

In the literature, there exist three typical problems for use of HMM, depending on the task that is required to be performed with them.

- Problem 1: Given the observation sequence O and the HMM λ, compute the likelihood $P(O \mid \lambda)$.
- Problem 2: Given the observation sequence O and the HMM λ, find which is the state sequence Q that better explains the observations.
- Problem 3: Given the observation sequence O, adjust the model parameters λ to maximize the likelihood $P(O \mid \lambda)$.

Due to the interest of this work, we will only focus on Problems 1 and 3, also known as evaluation and training respectively.

3.2.1.1 Problem of evaluation

The likelihood $P(O \mid \lambda)$ is defined as the probability of the observation sequence O given the model parameters λ. It provides a way to measure how well the model parameters explain the observations. Log-likelihood is commonly used since it is an equally monotonous function and its calculation is simplified by converting multiplications into additions. Since the state sequence is not observed there exist many different possible sequences that could generate the given observations. Brute force calculation of all the possible state sequences of length T implies an extremely large number of computations (on the order of $2T \cdot N^T$). Therefore, a more practical method must be considered. The used procedure for this purpose is the so-called forward procedure.

Let us assume an observation sequence $O = \{O_1, O_2, ..., O_T\}$ of length T and a given HMM λ. It is possible to define the forward variable $\alpha_t(i) = P(O_1, O_2, ..., O_t, q_t = S_i \mid \lambda)$ as the joint probability of the observation until time t and being in state S_i at time t, given λ. $P(O \mid \lambda)$ can then be obtained by following these recursive steps:

- Step 1: $\alpha_1(i) = \pi_i b_i(O_1) \quad \forall i \in 1, ..., N.$
- Step 2: $\alpha_{t+1}(j) = \left(\sum_{i=1}^{N} \alpha_t(i) a_{ij} \right) b_j(O_{t+1}), \quad \forall t \in 1, ..., T-1, \quad \forall j \in 1, ..., N.$
- Step 3: $P(O \mid \lambda) = \sum_{i=1}^{N} \alpha_T(i).$

where a_{ij} corresponds to the (i, j)-th element of A and represents the transition probability from state S_i to state S_j in any instant of time. $b_j(O_t)$ represents the value of the j-th pdf, given the obsevation symbol O_t.

3.2.1.2 Problem of training

There is no analytical way to calculate the optimal model parameters of the HMMs. There are, however, iterative algorithms that can estimate them in a locally optimal way. The most used is the Baum-Welch method, which is equivalent to Expectation-Maximization (EM) approach in this case.

In the first place, forward and backward variables must be taken into consideration:

$\alpha_t(i)$: as defined in Section 3.2.1.1.

$\beta_t(i)$: similarly to $\alpha_t(i)$, it is possible to define a backward variable representing the joint probability of the observation from time t until its final instant T as:

$$\beta_t(i) = P(O_{t+1}, ..., O_T \mid q_t = S_i, \lambda) \tag{3.10}$$

Induction can be conducted once again to obtain the values of β for all time instants:

- Step 1: $\beta_T(i) = 1, \quad \forall i \in 1, ..., N$

- Step 2: $\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(O_{t+1}) \beta_{t+1}(j), \quad \forall t \in 1, ..., T-1; \quad \forall i \in 1, ..., N$

From the definitions of $\alpha_t(i)$ and $\beta_t(i)$, new variables can be defined to perform the Baum-Welch procedure:

$\xi_t(i,j)$: represents the probability of being in state S_i at time t and in state S_j at time $t+1$, given the observation sequence O and the model parameters λ:

$$\xi_t(i,j) = P(q_t = S_i, q_{t+1} = S_j \mid O, \lambda) \tag{3.11}$$

$\gamma_t(i)$: probability of being in state S_i at time t, given the observation sequence O and the model parameters λ:

$$\gamma_t(i) = P(q_t = S_i \mid O, \lambda) \tag{3.12}$$

Using the previous definitions, it follows now to express these variables in terms of the known parameters:

$$\begin{aligned}
\xi_t(i,j) &= \frac{\alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)}{P(O \mid \lambda)} \\
&= \frac{\alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)}{\sum_{i=1}^{N} \alpha_t(i) \beta_t(i)} \\
&= \frac{\alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)}{\sum_{i=1}^{N} \alpha_t(i) \left[\sum_{j=1}^{N} a_{ij} b_j(O_{t+1}) \beta_{t+1}(j) \right]} \\
&= \frac{\alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)}{\sum_{i=1}^{N} \sum_{j=1}^{N} \alpha_t(i) a_{ij} b_j(O_{t+1}) \beta_{t+1}(j)}
\end{aligned} \tag{3.13}$$

$$\gamma_t(i) = \sum_{j=1}^{N} \xi_t(i,j) \tag{3.14}$$

Combination of the obtained expressions can then be utilised for the estimation of certain model probabilities. Let us define, for an observation of length T, τ_i as the number of transitions from state S_i and τ_{ij} as the number of transitions from state S_i to state S_j. Then it follows:

Expected number of transitions from state S_i:

$$E[\tau_i] = \sum_{t=1}^{T-1} \gamma_t(i) \tag{3.15}$$

And expected number of transitions from state S_i to S_j:

$$E[\tau_{ij}] = \sum_{t=1}^{T-1} \xi_t(i,j) \tag{3.16}$$

Finally, expressions for the new model parameters can be obtained:

$$\bar{\pi}_i = \gamma_1(i) \tag{3.17}$$

$$\bar{a}_{ij} = \frac{\sum_{t=1}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1} \gamma_t(i)} \tag{3.18}$$

$$\bar{b}_i(k) = \frac{\sum_{t=1 \cap O_t = v_k}^{T} \gamma_t(i)}{\sum_{t=1}^{T} \gamma_t(i)} \tag{3.19}$$

where v_k represents the discrete output symbol generated in time t by the discrete distribution $b_j(k)$. The reestimation expressions (Eqs. 3.17, 3.18 and 3.19) correspond identically to the Expectation-Maximization (EM) algorithm solution for this particular problem.

Up to this point, output distributions have been considered discrete. This, however, is not an assumption that holds for many applications where the observations are not discrete but continuous. For these situations, modifications in the formulae must be introduced in order to model the observations as continuous mixture distributions.

The underlying state transition probabilities are not affected in their definition by this extension, but the output distribution should now be considered as:

$$b_j(O) = \sum_{m=1}^{M} c_{jm} \Psi(O, \mu_{jm}, U_{jm}), 1 \leq j \leq N \tag{3.20}$$

where O represents the variable being modelled, c_{jm} is the mixing coefficient of the m-th mixture component of the j-th state and Ψ stands for any density, but will be considered as Gaussian distribution for the purpose of this work since it is a well known distribution which can be used to approximate any continuous density function. μ_{jm} and U_{jm} represent the mean and covariance matrix for the corresponding components respectively.

The mixture coefficients hold the constraints:

$$\sum_{m=1}^{M} c_{jm} = 1, \quad 1 \leq j \leq N \tag{3.21}$$

$$c_{jm} \geq 0, \quad 1 \leq j \leq N, \quad 1 \leq m \leq M \tag{3.22}$$

It is not complicated to obtain the reestimation formulas for the distribution parameters. The expected number of times in state j with the active component k over the expected number of times in state j is used to estimate the mixture coefficients:

$$\bar{c}_{jk} = \frac{\sum_{t=1}^{T} \gamma_t(j,k)}{\sum_{t=1}^{T} \sum_{k=1}^{M} \gamma_t(j,k)} \tag{3.23}$$

Similarly to Eq 3.23, but weighting each numerator term by the observation to give an estimation of the value produced by the k-th component can be used as reestimation of the mean value μ_{jk}:

$$\bar{\mu}_{jk} = \frac{\sum_{t=1}^{T} \gamma_t(j,k) O_t}{\sum_{t=1}^{T} \gamma_t(j,k)} \tag{3.24}$$

Once again, for the reestimation of the covariance matrix U_{jk} a similar expression can be found:

$$\bar{U}_{jk} = \frac{\sum_{t=1}^{T} \gamma_t(j,k)(O_t - \mu_{jk})(O_t - \mu_{jk})'}{\sum_{t=1}^{T} \gamma_t(j,k)} \tag{3.25}$$

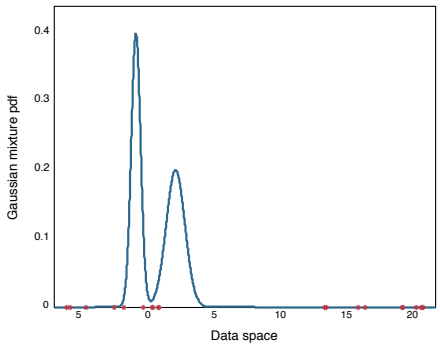

Figure 3.9: Example of data points and initial Gaussian mixture model (GMM). The data points are created following the distribution $x \sim U[12:21] + U[-7:1]$. The mixture model is composed of two Gaussians with parameters $\mu_1 = -1, \sigma_1 = 0.5$ and $\mu_2 = 2, \sigma_2 = 1$ respectively. It can be observed that the random initialisation of the Gaussians is too far away from the real data distribution, so adaptation is likely not to perform well due to very bad log-likelihood scores.

where

$$\gamma_t(j,k) = \left[\frac{\alpha_t(j)\beta_t(j)}{\sum_{j=1}^{N}\alpha_t(j)\beta_t(j)}\right]\left[\frac{c_{jk}\Psi(O_t,\mu_{jk},U_{jk})}{\sum_{m=1}^{M}c_{jm}\Psi(O_t,\mu_{jm},U_{jm})}\right] \tag{3.26}$$

The obtained expressions allow an iterative update of the HMM parameters, until a certain convergence level is reached.

3.2.2 Data normalisation

Since the HMM training is performed on a local maximisation problem assumption, and given that the feature spaces are usually very heterogeneous, the obtained model is very susceptible to different initialisations. Therefore, a straight forward use of features for training HMMs might become a difficult task if the initial model parameters are not chosen appropriately. The effect that might happen in this situation is shown in Figure 3.9. The gaussians are too steep for the data space and the training algorithm is likely not to perform well.

In our experiments, a normalisation process is conducted prior to the HMM training in order to avoid this effect. The normalisation is carried

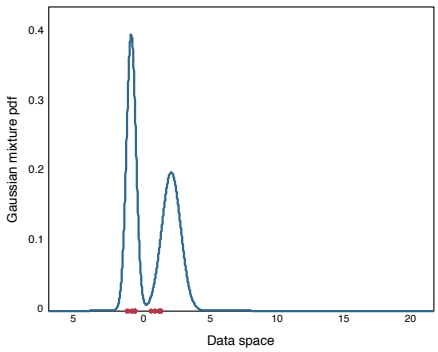

Figure 3.10: Normalised data points and initial GMM. As can be seen, normalisation of the data allows the Gaussian mixture model to produce better log-likelihood scores which will allow a good adaptation process from the first steps.

out by use of Equation 3.27, where x represents the data to normalise and μ, σ its mean and standard deviation, respectively. During the training of the system, mean and standard deviation (μ_{train} and σ_{train}) are calculated in each feature domain and for each class, prior to the HMM training. To remove the effect of outliers, all values above and below the 95% and 5% percentiles, respectively, are discarded. With the normalized data, the HMM are trained and the same normalization values (μ_{train} and σ_{train}) are later used to normalize the unseen data in the test step, before calculating their likelihood values. The effect of normalisation can be observed in Figure 3.10.

$$x_{norm} = \frac{x - \mu}{\sigma} \tag{3.27}$$

Once the training data is normalised, the initial HMM parameters are more likely to be correctly adapted to the real distribution that generated the data. An example of a well adapted model can be seen in Figure 3.11.

3.2.3 Data Alignment

Hidden Markov models are commonly used for modelling sequential data. Nevertheless, since this work is aimed to study data and its characteristics for each different emotion, a representation space where these can be compared seems necessary.

Emotion classification from speech data proves to be a challenging problem due to the sequential nature of the data. Therefore,

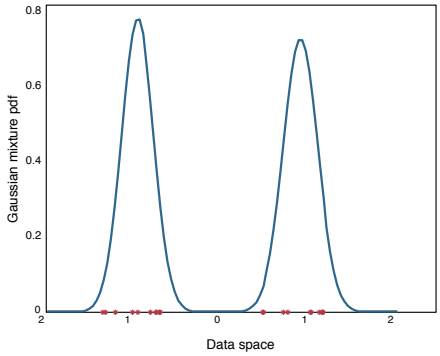

Figure 3.11: Normalised data points and adapted GMM. After the adaptation process, the GMM represents the data distribution well.

dynamic features extracted on short segments of speech (usually around 32ms windows) are useful for the classification of expressive clips. However, in order to be able to compare these sequential features with static features it is necessary to encode them into vectors of a fixed length. There exist different approaches for dealing with this type of situations. In this work, the HMMs, as in Bicego et al. (2003), are used to encode the sequential data to a new representation space, where every sequence can be represented in terms of a fixed number of dimensions. The feature sets considered as sequential and, therefore, aligned with this procedure are $MFCC$, $\Delta MFCC$, $ModulationSpectral$ and $VoiceQuality$. The rest of the feature sets are considered static and, therefore, no alignment is required.

Let us assume a reference set of sequential observations $\mathcal{R} = \{O_1, ..., O_R\}$, where O_i, $\forall i = 1, ..., R$, is an observation without restriction of length. It is possible to train, for each of the reference observations, an HMM that best represents the model that produced it. Therefore, we can easily obtain a set $\lambda = \{\lambda_1, ..., \lambda_R\}$ of HMM where λ_i, $\forall i = 1, ..., R$, represents the HMM trained from the observed sequence O_i. Given an unseen sequential observation $\tilde{O}$, its representation as a single vector in the R-dimensional encoding space is obtained by calculating its log-likelihood with respect to the trained reference HMMs:

$$D_R(\tilde{O}) = \frac{1}{T} \times \begin{pmatrix} \log P(\tilde{O} \mid \lambda_1) \\ \log P(\tilde{O} \mid \lambda_2) \\ \vdots \\ \log P(\tilde{O} \mid \lambda_R) \end{pmatrix}, \tag{3.28}$$

where T represents the length of the given sequence and $P(\tilde{O} \mid \lambda_i)$ is the likelihood of the i-th HMM, given the observation $\tilde{O}$. With this

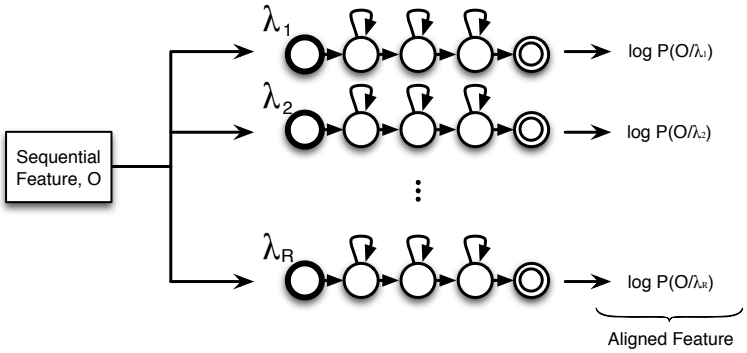

Figure 3.12: Feature alignment scheme. Observation O from a sequential feature set is used to calculate the log-likelihood of each one of the R trained HMMs. The scores obtained from each of them are aggregated and considered a single vector of dimension R, comprising in this way a single vector of fixed length for every observation.

transformation, we create a new space of dimension R, where every observed sequence is represented as one single vector. In this Euclidean space, any standard techniques for classification (supervised learning, unsupervised learning, clustering, etc.) may be applied. A representation of the alignment system is shown on Figure 3.12.

For the experiments conducted in this work, a subset was randomly chosen from the initial set of audio segments to train the HMMs. For each different class c, $\forall c = 1, ..., C$, where C represents the number of classes, 2 HMMs were trained. Since the used data set contains 6 different classes, the final number of HMM is $R = 12$. Every HMM was initialised with 2 states and 2 GMM per state. Each of the models was trained with 5 different audio segments for a number of iterations no longer than 30. Experiments with a higher number of states and mixture components proved not to give better results and, on the contrary, required much more computation time.

3.3 Principal Component Analysis

The use of a large set of trained HMMs for the data alignment leads to a high dimensionality scenario where data inspection is not a trivial task. There exist different techniques for dimensionality reduction that allow transformation of the data into a space with lower dimensions. In this work, the utilized technique is principal component analysis (PCA), first incroduced in Pearson (1901). PCA is a procedure

that converts correlated variables into uncorrelated variables, the so-called principal components (PC). The principal components have the property that each of them is orthogonal with respect to the previous component, being the first one a variable that has maximal variance. The total number of principal components is less or equal to the dimensions of the original data. The built-in implementation used in these experiments is based on eigenvalue decomposition, providing a transformation matrix. Representation of the first PC versus the second PC is used in some graphs in this thesis (eg. Figures 4.4, 4.5). An example of the transformation that takes place by the PCA process is shown in Figures 3.13 and 3.14. In this figures, a 2-dimensional gaussian distribution with mean $\mu = (3, 2)^T$ and covariance matrix $\sigma = \begin{pmatrix} 2 & -1.5 \\ -1.5 & 2 \end{pmatrix}$ is shown. By PCA, the transformation matrix

$$\begin{pmatrix} 0.7087 & 0.7055 \\ -0.7055 & 0.7087 \end{pmatrix} \tag{3.29}$$

is obtained. The PC directions can also be seen in the graph. As it can be observed, the obtained transformation vectors represent the maximum variance directions and are, at the same time, orthogonal to each other.

3.4 Support Vector Machines

Support vector machines (SVMs) are one the most commonly used methods in all kinds of classification problems. Assuming two linearly separable classes, the aim is to find a hyperplane that can separate them maximizing the gap (margin) between the nodes supporting it, the support vectors, as shown in Figure 3.15. If the assumption of the classes being linearly separable fails, transformation to a space of higher dimensions is possible by the use of a kernel function, making the search of the hyperplane an easier task. There exist extended versions of the SVMs for the situations where more than two classes need to be separated. For the purpose of this work, one-against-one SVMs for a multi-class problem will be considered (Kahsay et al., 2005).

Traditional implementation of the SVMs were based on a crisp-input crisp-output assumption. Nevertheless, there are situations in which crisp input labels might be difficult to obtain, due to subjective perception of the annotator. For this kind of situations fuzzy SVMs (FSVM) were designed to produce a crisp output from a fuzzy input. Furthermore, a later approach considering fuzzy-input fuzzy-output SVMs (F^2SVM) has also been developed (Thiel et al., 2007).

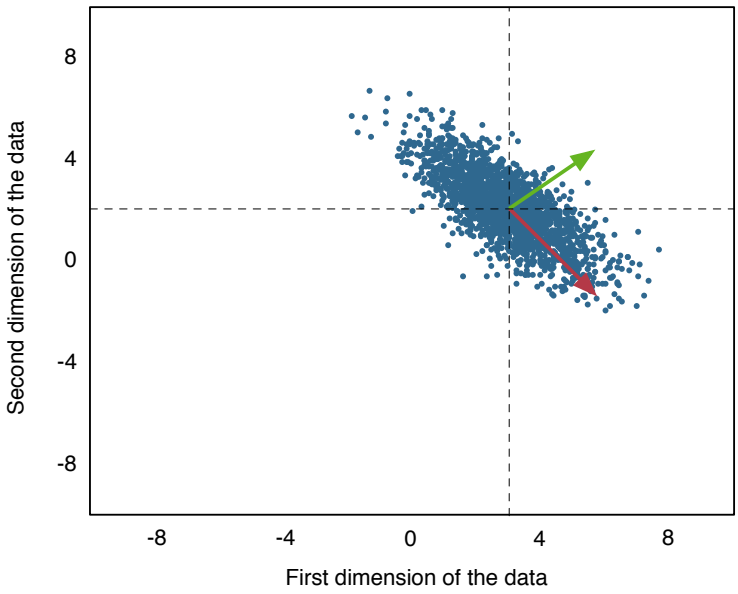

Figure 3.13: Principal component analysis example. In blue: 2-dimensional Gaussian distribution. In red: First PC coefficient represents the direction of the variable with a highest variance. In green: Second PC coefficient represents the orthogonal direction to the first PC with the highest possible variance.

In this section, both classical SVMs and F^{2}SVM approaches are explained.

3.4.1 Crisp-Input Crisp-Output SVMs

An initial two class training set M can be defined as:

$$M = \{(x_\mu, l_\mu) \mid x_\mu \in \mathbb{R}^n, \quad l_\mu \in \{-1, +1\}, \quad \forall \mu = 1, ..., |M|\} \tag{3.30}$$

where x_μ represents the μ-th data vector and l_μ its corresponding label. It is possible to define a hyperplane characterized by its surface normal vector w and a bias value b and that satisfies the constraint:

$$l_\mu(w^T x_\mu + b) \geq 1, \quad \forall \mu = 1, ..., |M| \tag{3.31}$$

After maximization of the margin at least two samples must fulfil the equality constraint in Eq. 3.31. Let us name these two samples x_ν

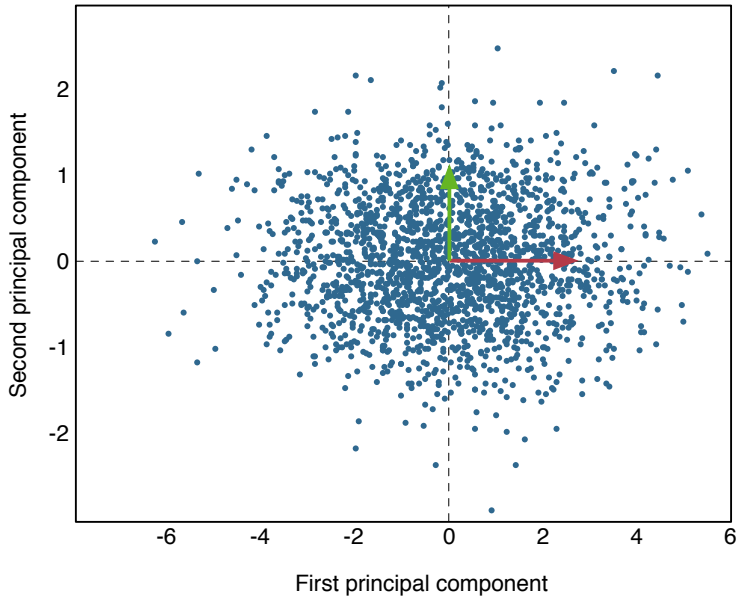

Figure 3.14: Principal component analysis example. In blue: 2-dimensional gaussian distribution normalised and transformed to the PC space. X and Y axis represent the first and second principal components respectively, which are also represented by the red and green arrows.

and x_λ, belonging to the subgroups labelled as positive and negative respectively. Then, the width of the margin area can be expressed as:

$$\frac{w^T}{||w||}(x_\nu - x_\lambda) = \frac{2}{||w||} \tag{3.32}$$

As explained in Bishop (2006), maximization of Eq. 3.32 is equivalent to minimization of:

$$\theta(w) = \frac{||w||^2}{2} \tag{3.33}$$

Solving of this problem requires the use of Lagrange multipliers $\alpha_\mu \geq 0$, obtaining the function:

$$L(w, b, \alpha) = \frac{||w||^2}{2} - \sum_{\mu=1}^{|M|} \alpha_\mu \{l_\mu(w^T x_\mu + b) - 1\} \tag{3.34}$$

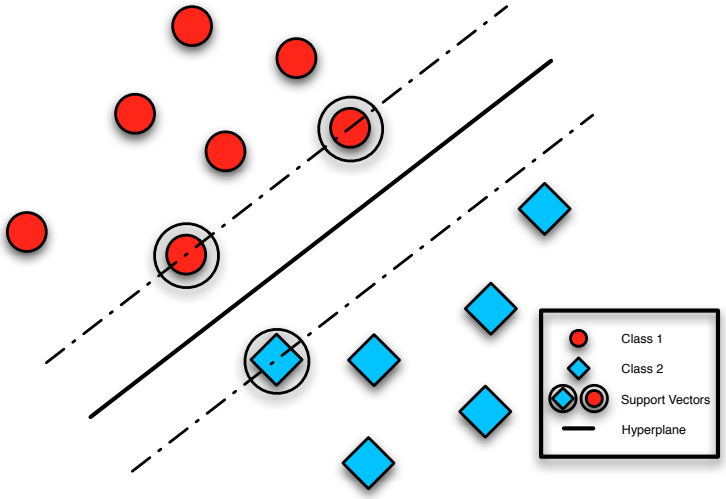

Figure 3.15: Example of Support Vector Machine. Two classes of data are represented (in blue and red, respectively). The separation hyperplane which maximises the distance between them is also shown. In this case, the support vectors are the three samples circled in black coincident with the dashed lines.

If $\frac{\partial L}{\partial w} = 0$ and $\frac{\partial L}{\partial b} = 0$ restrictions are introduced, Equations 3.35 and 3.36 can be encountered.

$$w = \sum_{\mu=1}^{|M|} \alpha_\mu l_\mu x_\mu \tag{3.35}$$

$$0 = \sum_{\mu=1}^{|M|} \alpha_\mu l_\mu \tag{3.36}$$

Considering this result, w and b can be obviated from Equation 3.34, leading to the maximization problem of:

$$\tilde{L}(\alpha) = \sum_{\mu=1}^{|M|} \alpha_\mu - \frac{1}{2} \sum_{\nu=1}^{|M|} \sum_{\mu=1}^{|M|} \alpha_\nu \alpha_\mu l_\nu l_\mu x_\nu^T x_\mu \tag{3.37}$$

subject to the restrictions:

$$\alpha_\mu \geq 0, \quad \forall \mu = 1, ..., |M| \tag{3.38}$$

$$\sum_{\mu=1}^{|M|} \alpha_\mu l_\mu = 0 \tag{3.39}$$

Assuming an optimal solution that maximizes the margin, also the Karush-Kuhn-Tucker constraints are met (Bishop (2006)):

$$\alpha_\mu \{l_\mu (w^T x_\mu + b) - 1\} = 0, \quad \forall \mu = 1, ..., |M| \tag{3.40}$$

following that for all $\alpha_\mu \neq 0$ it is true that:

$$l_\mu (w^T x_\mu + b) = 1 \tag{3.41}$$

being x_μ a support vector. Then, for a new point x, the classification into one class or the other can be calculated as:

$$y(x) = sign \left(\sum_{\mu=1}^{|M|} \alpha_\mu l_\mu x^T x_\mu + b \right) \tag{3.42}$$

Determining the bias b can be done with any support vector x_ν as:

$$b = \frac{1}{l_\nu} - w^T x_\nu \tag{3.43}$$

Until this point it has been assumed that there exist a separation hyperplane. However, in a more realistic situation this assumption would quite likely not hold and, therefore, some reformulation is necessary. Slack variables ξ_μ can be introduced to allow datapoints to be between the hyperplane and the support vectors $(0 \leq \xi_\mu < 1)$ or even on the wrong side of the hyperplane $(\xi_\mu > 1)$. The new optimization problem would now be:

$$\theta(w, \xi) = \frac{||w||^2}{2} + C \sum_{\mu=1}^{|M|} \xi_\mu \tag{3.44}$$

with the constraints:

$$l_\mu (w^T x_\mu + b) \geq 1 - \xi_\mu, \quad \xi_\mu \geq 0, \quad \forall \mu = 1, ..., |M| \tag{3.45}$$

The free parameter $C > 0$ tunes the number of points allowed to be out of their corresponding subset area. A higher value of this parameter implies a lower number of allowed errors.

3.4.2 Fuzzy-Input Fuzzy-Output SVMs

This section describes the extension of the classical SVM to the more recent approach F²SVM, first introduced in Thiel et al. (2007). Membership values m_μ^+ and m_μ^- for the positive and negative classes respectively must be defined and included in the optimization problem:

$$\theta(w, \xi^+, \xi^-) = \frac{||w||^2}{2} + C \sum_{\mu=1}^{|M|} (\xi_\mu^+ m_\mu^+ + \xi_\mu^- m_\mu^-) \tag{3.46}$$

with the new constraints:

$$w^T x_\mu + b \geq 1 - \xi_\mu^+, \text{ with } \xi_\mu^+ \geq 0, \quad \forall \mu = 1, ..., |w| \tag{3.47}$$

$$w^T x_\mu + b \geq -1 + \xi_\mu^-, \text{ with } \xi_\mu^- \geq 0, \quad \forall \mu = 1, ..., |w| \tag{3.48}$$

The new four constraints also require introduction of the Lagrangian multipliers $\alpha^+, \alpha^-, \beta^+ \beta^-$:

$$\begin{aligned} L(w, b, \xi^+, \xi^-, \alpha^+, \alpha^-, \beta^+, \beta^-) &= \frac{||w||^2}{2} + C \sum_{\mu=1}^{|M|} (\xi_\mu^+ m_\mu^+ + \xi_\mu^- m_\mu^-) \\ &- \sum_{\mu=1}^{|M|} \alpha_\mu^+ ((w^T x_\mu + b) - 1 + \xi_\mu^+) \\ &+ \sum_{\mu=1}^{|M|} \alpha_\mu^- ((w^T x_\mu + b) + 1 - \xi_\mu^-) \\ &- \sum_{\mu=1}^{|M|} \beta_\mu^+ \xi_\mu^+ - \sum_{\mu=1}^{|M|} \beta_\mu^- \xi_\mu^- \end{aligned} \tag{3.49}$$

Similar expressions to 3.35 and 3.36 can be now obtained by inserting the restrictions $\frac{\partial L}{\partial w} = 0$, $\frac{\partial L}{\partial b} = 0$, $\frac{\partial L}{\partial \xi_\mu^+} = 0$ and $\frac{\partial L}{\partial \xi_{mu}^-} = 0$:

$$\tilde{L}(\alpha) = \sum_{\mu=1}^{|M|} \alpha_\mu^+ + \sum_{\mu=1}^{|M|} \alpha_\mu^- - \frac{1}{2} \sum_{\nu=1}^{|M|} \sum_{\mu=1}^{|M|} (\alpha_\nu^+ - \alpha_\nu^-)(\alpha_\mu^+ - \alpha_\mu^-) x_\nu^T x_\mu \tag{3.50}$$

having that $\alpha_\mu^+, \alpha_\mu^- > 0, \quad \forall \mu = 1, ..., |M|$ and subject to:

$$\sum_{\mu=1}^{|M|}(\alpha_{\nu}^{+}-\alpha_{\nu}^{-})=0 \tag{3.51}$$

$$0\leq\alpha_{\mu}^{+}\leq Cm_{\mu}^{+},0\leq\alpha_{\mu}^{-}\leq Cm_{\mu}^{-} \tag{3.52}$$

Next, the Karush-Kuhn-Tucker conditions can be obtained:

$$\alpha_{\mu}^{+}((w^{T}x_{\mu}+b)-1+\xi_{\mu}^{+})=0 \tag{3.53}$$

$$\alpha_{\mu}^{-}((w^{T}x_{\mu}+b)+1-\xi_{\mu}^{-})=0 \tag{3.54}$$

Finally, those samples x_{μ} that verify $(\alpha_{\mu}^{+}-\alpha_{\mu}^{-})\neq 0$ are the so-called support vectors and they define the decision function:

$$y(x)=sign\left(\sum_{\mu=1}^{|M|}(\alpha_{\mu}^{+}-\alpha_{\mu}^{-})x^{T}x_{\mu}+b\right) \tag{3.55}$$

For the multi-class extension with a set of samples

$$M=\{(x_{\mu},l_{\mu})\mid x_{\mu}\in\mathbb{R}^{n},l_{\mu}\in\mathbb{R}^{k},with\sum_{j=1}^{k}l_{\mu,k}=1,\quad\forall\mu=1,...,|M|\} \tag{3.56}$$

where l_{μ} represents the fuzzy label or the probability of belonging to each of the k classes. With the described extensions, the crisp classification problem is a mere particular case of the F^{2}SVM approach, where $l_{\mu,j}=0$ for all classes except for one ($l_{\mu,j*}=1$). Also, to solve the multi-class problem, $\frac{k(k-1)}{2}$ F^{2}SVMs (represented as $S_{i,j}$) in the one-against-one configuration must be trained for each pair of classes i and j and every feature space. To train each of the F^{2}SVMs, the sample set is defined as:

$$M_{i,j}=\{(x_{\mu},m_{\mu,i}^{+})\mid m_{\mu,i}^{+}=l_{\mu,i}\}\cup\{(x_{\mu},m_{\mu,j}^{-})\mid m_{\mu,j}^{-}=l_{\mu,j}\} \tag{3.57}$$

Once all F^{2}SVMs are trained, the steps to generate a fuzzy output vector are as follows:

1. For every unseen sample $z\in\mathbb{R}^{n}$ and trained F^{2}SVM $S_{i,j},\quad\forall i,j=1,...,k\mid i\neq j$.

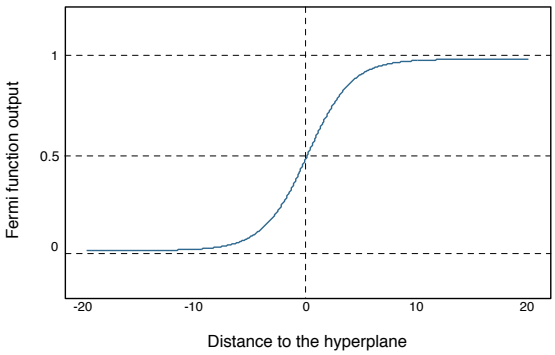

Figure 3.16: Fermi function used to limit the distance $d_{i,j}(z)$ to the range $[0:1]$. This representation is for a parameter $A = 0.5$.

2. Calculate the distance $d_{i,j}(z) \in \mathbb{R}$ to the hyperplane corresponding to $S_{i,j}$.
3. Transform the distances $d_{i,j}(z)$ using the fermi function $r_{i,j}(d_{i,j}) = \frac{1}{1+exp(-Ad_{i,j})}$, with A subject to optimization if required. Representation of the fermi function can be observed in Figure 3.16.
4. Fuzzy output label $\tilde{y}(z)$ for all k classes is obtained as in Thiel et al. (2009) and Thiel (2009):
   I. Estimate of p_i by averaging: $\hat{p}_i = \frac{\sum_{j \neq i} r_{ij}}{\frac{1}{2}k(k-1)}$.
   II. Update the pairwise probability estimates: $\hat{\mu}_{ij} = \frac{\hat{p}_i}{\hat{p}_i + \hat{p}_j}$.
   III. Correct the single class probability estimates: $\hat{p}_i = \hat{p}_i \frac{\sum_{j \neq i} r_{ij}}{\sum_{j \neq i} \hat{\mu}_{ij}}$.
   IV. Normalise the single class probabilities: $\hat{p}_i = \frac{\hat{p}_i}{\sum_{\forall j} \hat{p}_j}$.
   V. Loop until convergence: If change of $\hat{p}_i > threshold$ then go to II.
5. Normalised class probabilities estimation: $\tilde{y}(z) = (\hat{p}_1, ..., \hat{p}_k)$

The steps followed to obtain $\tilde{y}(z)$ are represented in Figure 3.17.

3.5 Data Fusion

As explained in Section 3.4, the SVMs are trained in a one-against-one configuration. This means, that for each of the 8 features sets, a number of 15 SVMs is trained, making a total of 120 evaluations

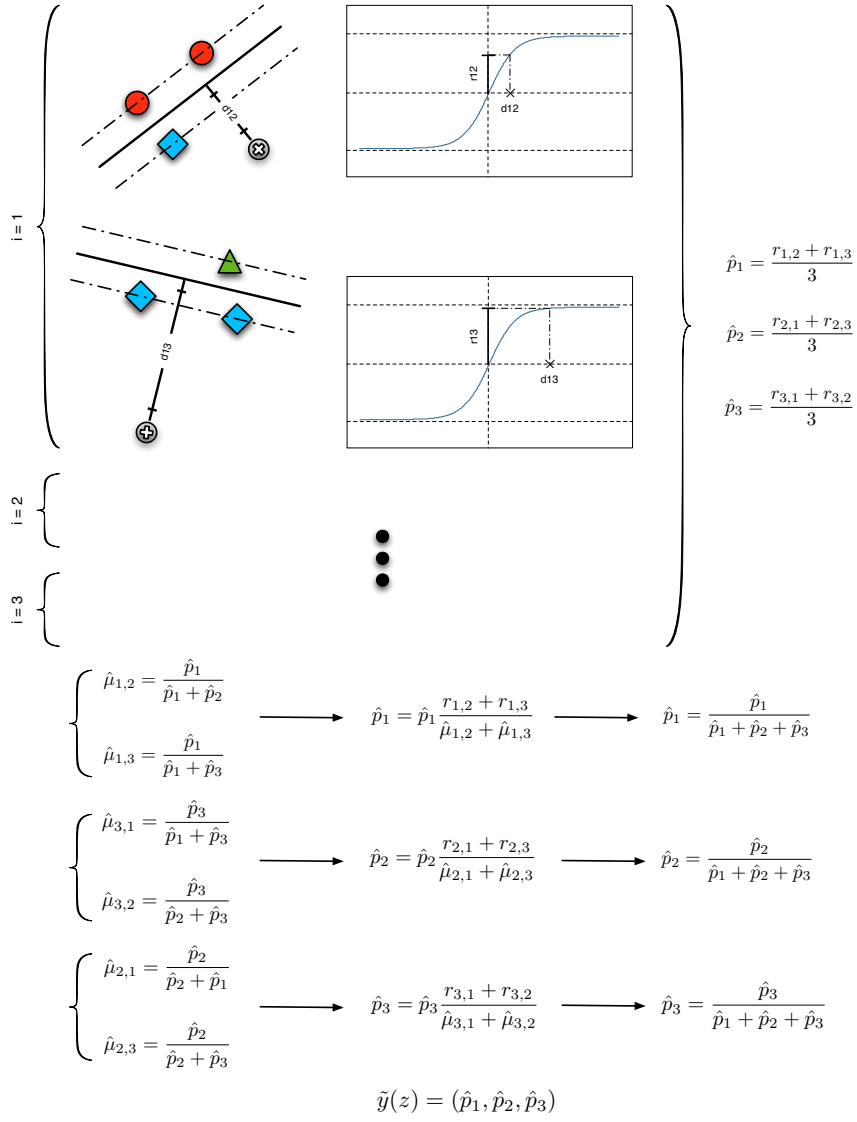

Figure 3.17: Fuzzy output generation for a 3-class problem. This process is conducted for each different feature set, before their fusion is considered. First, distances from a new sample to all the hyperplanes are calculated. The distances are transformed to the range $[0:1]$ by use of a Fermi function. With these values, normalisations and corrections described in steps I-IV are conducted. The final fuzzy output for the given feature set is obtained grouping the normalised probabilities of all 3 classes.

for each new sample. For each feature set, the corresponding 15 SVMs are used together to produce a single fuzzy output for each new point. But there is still the need to define a fusion that can combine all 8 outputs into a single one. There exist several ways to define a fusion strategy (Kuncheva, 2001; Kuncheva et al., 2001; Kuncheva, 2004). The utilised approach is based on a simple multiplication of the fuzzy labels obtained from each different feature set and normalisation. During the conducted experiments, it was observed that this basic fusion was able to produce notable improvements over all the single feature outputs, therefore, no further research of more complex implementations was done. This approach is described in Figure 3.18.

3.6 Partially Supervised Learning

Despite being in possession of all labels to the dataset, for this part of the experiments it is assumed that not all of them are available, so it is possible to study the system performance with respect to the amount of work put into labelling new samples.

3.6.1 Semi-supervised Learning

Semi-supervised learning approaches are used when not all the data labels are available. In this situation, the available labels can be used as a reference for the system to automatically generate artificial ones for the rest of the data. Although there exist different techniques that can accomplish this task, only k-nearest neighbour (k-NN) is described and utilised in this work. k-NN is a classification algorithm based on a distance measure to the reference samples. Different results can be achieved by modification of the parameter k. This parameter controls the amount of training points that are used to emit a decision for each new unseen datapoint. In the most simple case, 1-NN, the label of the nearest point would just have to be copied. In the conducted experiments, however, a value of $k = 5$ was used.

Prior to the self-learning steps, description of the new labels wished to obtain is necessary. Since a total of $J = 6$ classes are used for our experiments, the fuzzy labels must contain 6 different fields, p_j that represent the degree of membership to each of the J classes:

$$l = \{p_j\}, \quad \forall j \in 1, ..., J \tag{3.58}$$

where l represents the fuzzy label of any sample, and is subject to the constraint

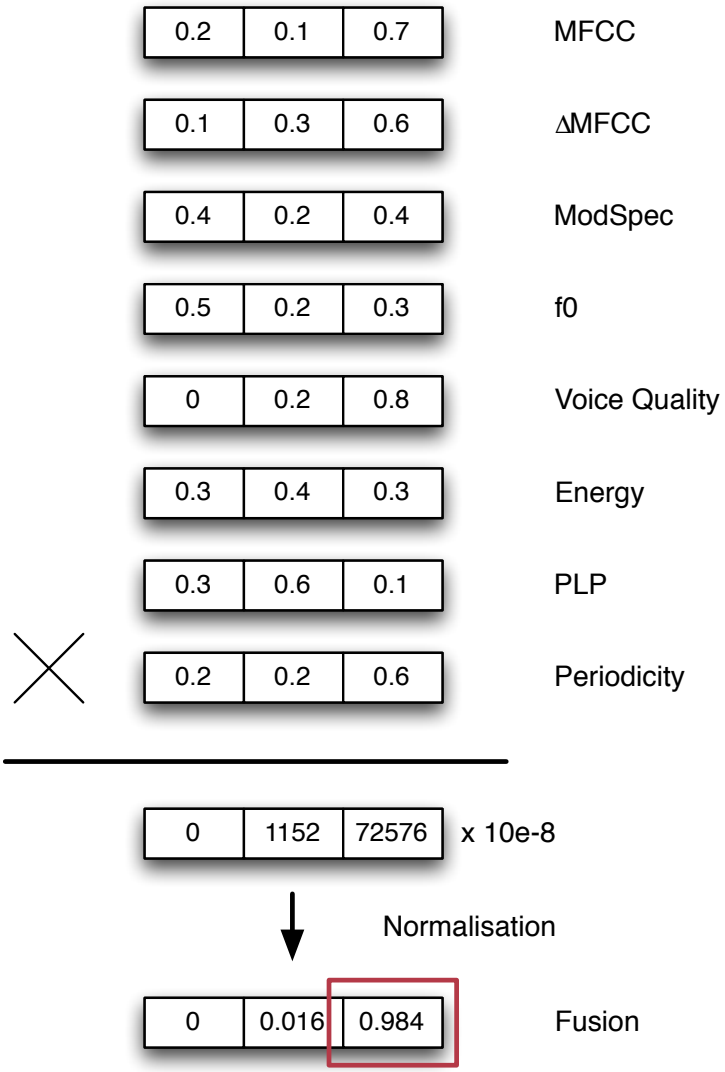

Figure 3.18: Classifiers' fuzzy outputs fusion. For each different feature set there is a classifier defined which produces a fuzzy output. The outputs from all classifiers are combined to emit a single output by defining a fusion approach. In this figure a 3-class problem with the 8 feature sets described in Section 3.1 is represented. From each feature set classifier, a probability to all 3 classes is obtained, as represented in Figure 3.17. The outputs from each classifier are multiplied and normalised, obtaining a single vector that sums up to 1. The class with to which the highest probability corresponds is the final decision of the system.

$$\sum_{j=1}^{J} p_j = 1 \tag{3.59}$$

The first step for the generation of new fuzzy labels for the unseen data consists of relabelling the reference set. This is a simple procedure since the crisp label is available.

$$p_j = \begin{cases} 1 & : j = r \\ 0 & : j \neq r \end{cases}$$

where $r \in \{1, ..., J\}$ represents the real class which the sample belongs to. Once this step is concluded, an iterative labelling process can be conducted. It is considered iterative since every new label that is generated is also utilized for generating the ones for the successive samples.

When a new point is considered, euclidean distance to all the reference points is calculated. Of all the calculated distances, only the k smallest ones are kept. With the labels of the k-nearest neighbours l_n, the new label can be calculated as:

$$l = \frac{1}{k} \sum_{n=1}^{k} l_n \tag{3.60}$$

The newly generated label is then included into the reference set and considered as correct for all the samples still to come. The iterations are repeated for all the unseen datapoints. An outline of the whole process is shown here:

1. Extend the reference labels to fuzzy labels.
2. For each unlabeled sample:
   I. Calculate the euclidean distance from this point to all the points in the reference set.
   II. Keep only the label of the nearest k neighbours.
   III. Create the new fuzzy label by averaging the k nearest labels as:
   $$l = \frac{1}{k} \sum_{n=1}^{k} l_n$$
   IV. Extend the validation set with the newly calculated label.

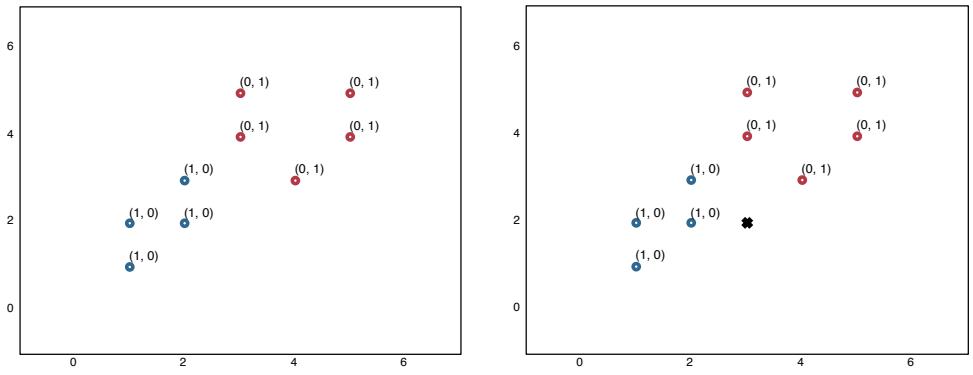

Figure 3.19: On the left: Reference data points from two different classes (blue and red, respectively) with their real labels extended to a fuzzy representation. On the right: Reference data points and an observation (black) without label.

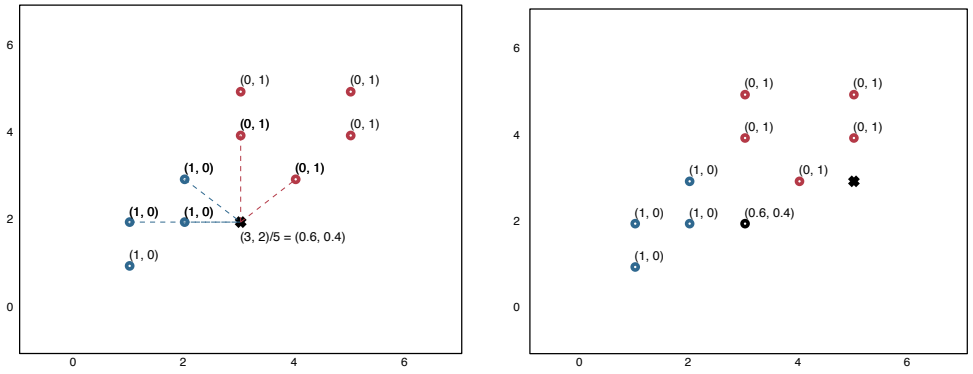

Figure 3.20: On the left: Distance to the $k = 5$ nearest neighbours and new fuzzy label for the new point. On the right: The new point and its label are included in the training set. A second unlabelled observation (black cross) comes into the system.

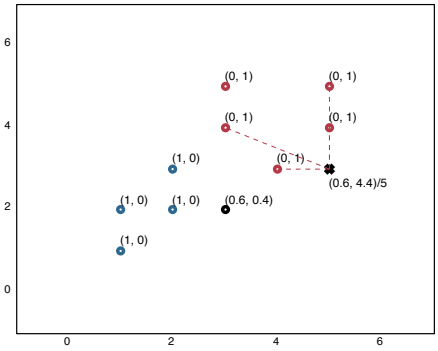

Figure 3.21: 5-NN distances for the new point and resulting new fuzzy label. The k-NN label of the previous unlabelled points is also considered as a reference for the current iteration.

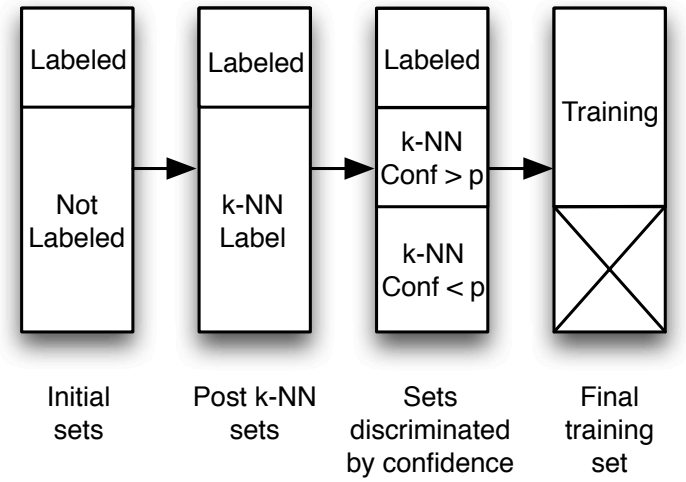

Figure 3.22: Evolution of training sets in the semi-supervised training approach. The initial sets are formed by labelled and unlabelled data. By k-NN, the unknown labels are generated. Those labels with a confidence higher than a discriminatory parameter $p \in [0, 1]$ are used for training and the rest are discarded.

A representation of the algorithm can be seen in the Figures 3.19, 3.20, 3.21. Evolution of the sets described in this process over the iterations is also shown in Figure 3.22.

Once the artificial labels are generated, it seems necessary to have a confidence measure that contains information about the level of correction that they have. Since no expert supervises the process, certain amount of error will be introduced into the labels. The confidence measure will allow discrimination of the labels that contain too much of this error to avoid extra penalisation in training. The proposed measure to use in these experiments is the value that represents the highest degree of membership to any of the classes. From the definition of fuzzy label given in Equation 3.58, the confidence of the label l can be obtained as:

$$c = \max_j \{p_j\} \quad \forall j \in 1, ..., J \tag{3.61}$$

where p_j represents the degree of membership of the sample to the j-th class. A flow chart of the whole experiment setup, both for automatic labeling and test evaluation is represented in Figure 3.23.

A discrimination parameter p is included in the system to decide which automatic labels are considered valid and which are not, based on the confidence measure. A high value of p (within the range $[0, 1]$) means using samples with a high confidence while reducing the amount of them that can be used for training.

3.6.2 Active Learning

Traditional machine learning approaches rely on a large amount of labelled data distributed over the feature spaces with as much information as possible concerning the underlying generative distribution. The larger the amount of data is, the more likely the system will be to learn the characteristics of the distribution (assuming that there is some) and emit a correct guess for unseen data. However, this type of approach may have some drawbacks, like the difficulty of labelling data (it can be an expensive task in many situations) which may lead to a scenario where not-so-large training sets are available. If it were possible, for example, to change the assumption "the more labelled data we have, the better we can train" to "the better our labelled data is, the better we can train" then it might be possible to achieve good model trainings with smaller datasets. And this is exactly what active learning's basic idea is all about. In a randomly chosen training set, probability of redundant or non representative data is quite high. Therefore, the learning system might be wasting some of the effort and cost put into labeling it, given that it is difficult to know a-priori

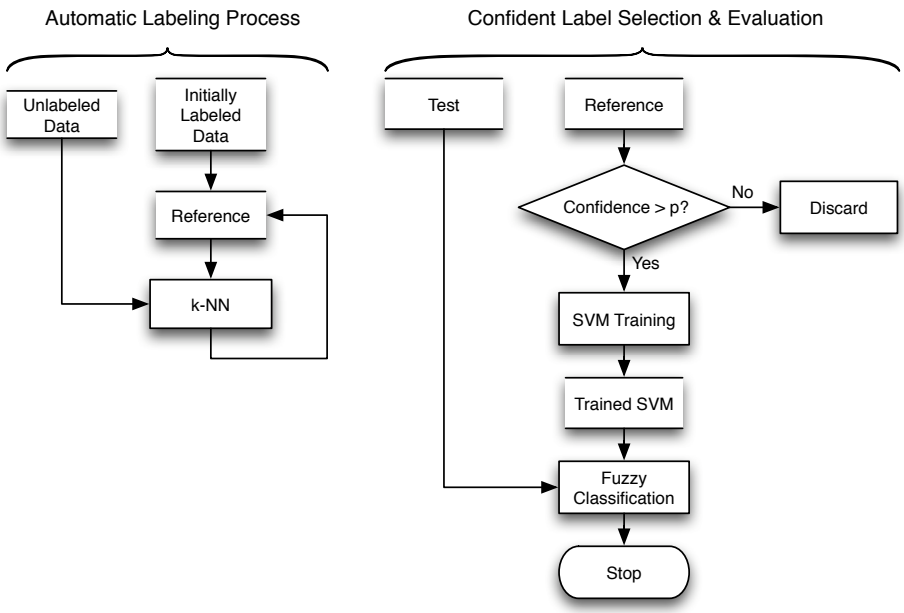

Figure 3.23: Flow chart that describes the semi-supervised approach. On the left: automatic labelling process; using a labelled reference set new labels are generated for the unlabelled data by the k-NN procedure and incorporated into the reference set. On the right: confident labels selection and evaluation; a discrimination process is conducted to decide which k-NN labels are good enough to be used for training the SVMs.

which data samples are more or less representative. If, however, the system is allowed to ask for the samples from which it wants to learn, it is more likely that it will choose the most useful, based on the proximity to the classifier boundaries. There are several ways in which this can be interpreted and implemented and a good review of them can be found in Settles (2009). The approach utilized in this work is explained in this section.

First of all, the whole available dataset with labels is divided in two groups: training and test. The test set remains unchanged during the whole process. The training set, on the contrary, represents the pool of available data from which the system will decide on every iteration which labels it wants to know and use for training. Evolution of the sets over the iterations is shown on Figure 3.24.

A small number of labels is initially used for the training, then evaluation is conducted on the unused training points. For each of these

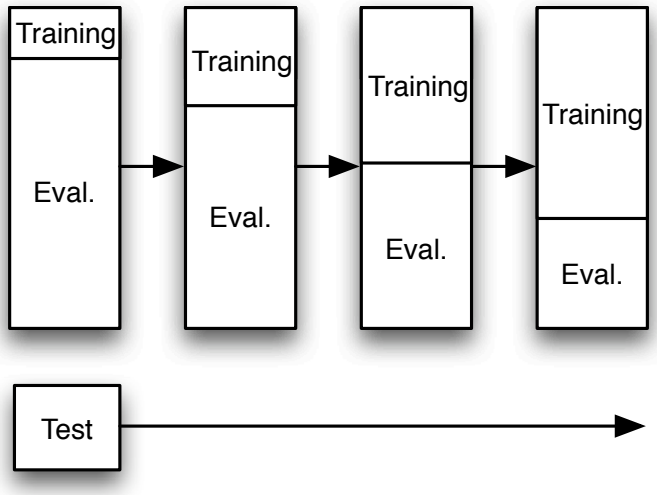

Figure 3.24: Evolution of the different sets of data used during the active learning approach. The evaluation set represents a pool of samples from which the system decides on every iteration which ones to label. The samples that get a label become part of the training set, while the test set stays untouched over during all the process.

points, the classifier produces a fuzzy output label that represents the degree of membership to all the classes. The accumulated membership to the classes must be equal to 1 and, therefore, considering the highest membership in one label also accounts for the most likely class. It is then possible to define the confidence of the label as the degree of membership to the most likely class, as defined in Eq. 3.61. Considering only the most likely class for each label can be assumed to provide a measure of how confident a label is. Under this assumption, it makes sense to believe that low confidence labels are the ones that the system has more trouble in classifying. If those samples are correctly labeled by an expert and then used for training, the system is likely to be learning information from areas in the data space where it does not prove to be very confident. These areas represent, in general, the decision boundaries, where overlap of classes is frequent and more information is required. On the other hand, for the output labels that show a good confidence, the system does not require more information since they represent an easy task for it. A flow chart of the active learning and evaluation processes is represented in Figure 3.25.

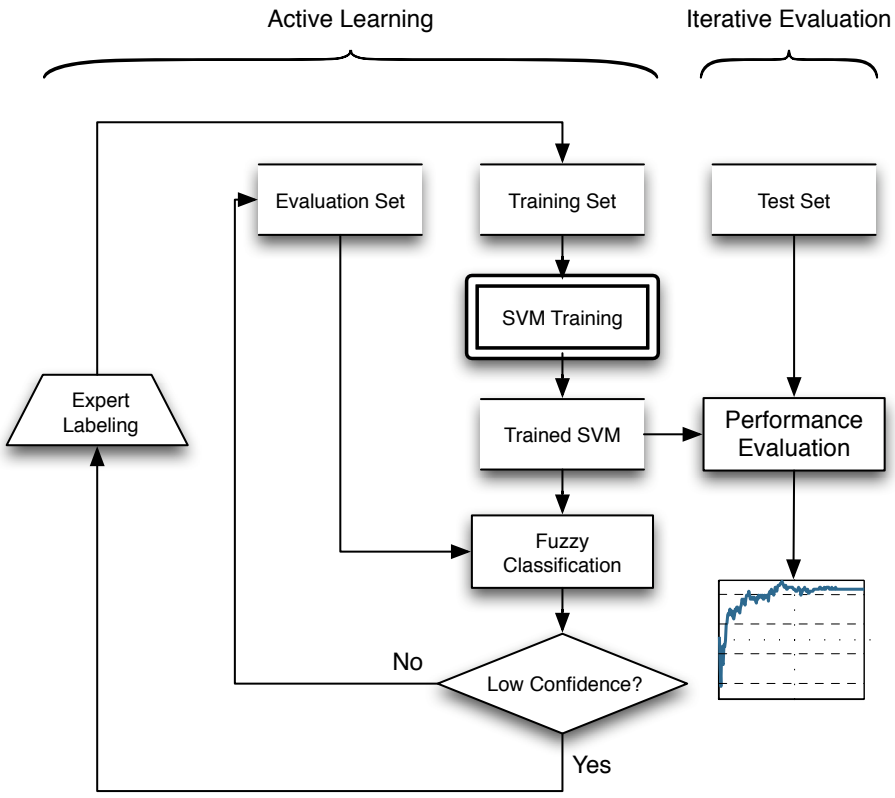

Figure 3.25: Flow chart of the active learning approach, comprising training and iterative evaluation parts. On the left, SVMs are initially trained with a reduced set. A validation set (pool of samples) is used as input to the classifier and decision confidence is computed. Those samples that produce a low confidence are labelled and used for training. The right part shows the evaluation process, carried out on every iteration with the same test set.

A graphical example of the active learning step is shown in Figure 3.26.

This chapter provided a basic theoretical background, sufficient to understand the fundamentals of the conducted study. However, more detailed and accurate description might be required by the reader. For this purpose, references to all original sources of texts and formulae are provided, for each of the techniques utilised.

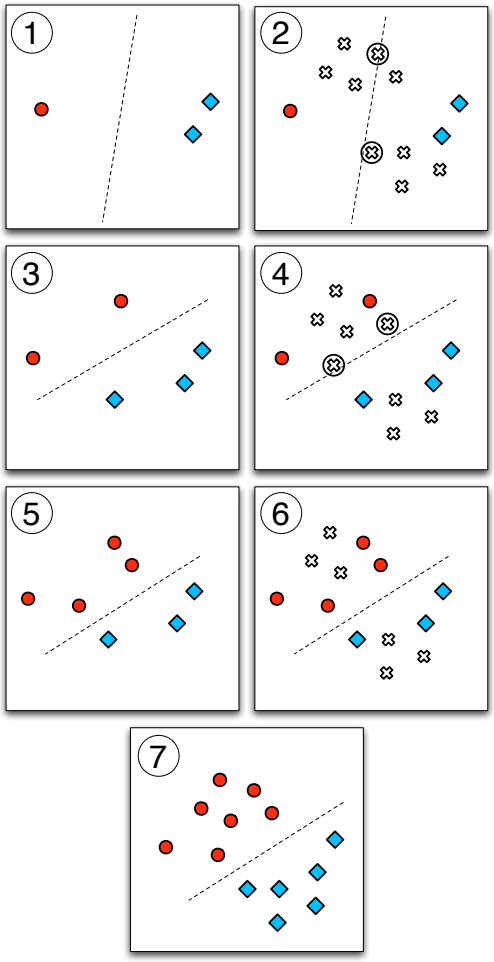

Figure 3.26: Active learning process. Data from two classes (red and blue) are considered. On a first step, 3 random samples are labelled and a classifier is trained with them ①. The pool of unlabeled data is evaluated with this classifier and the least confident samples (the ones closest to the decision boundaries) are selected ②. The expert labels the selected samples and they become part of the training set, defining a new classifier ③. The pool of samples goes once again through the classifier and more samples are chosen to be labeled ④. These steps are repeated ⑤, ⑥ until a good enough accuracy is reached. Labels of all samples are shown in ⑦. However, not all of them were necessary to build a good classifier. The defined one in step ④ is likely to perform already well, so no more labeling would be necessary in this particular case.

4 Experiments and Results

The experiments conducted in this work are designed within three different approaches. First of all there is the aim to study the human emotions and their characteristics based on speech. For evaluation of the proposed system a model is trained, for which supervised learning techniques are utilised. Then automatic recognition tests are conducted. In a second stage of the experiments, discussion arises about the effect of partially-supervised learning in our results. For this purpose, automatic label generation (for the semi-supervised learning) is studied, as well as active learning.

For evaluation of the results, confusion matrices are generated with the automatic recognition results and they are compared with those obtained by humans in the perception tests. On the matrices, every row sums up to one, showing how much data from one class is classified by the system as belonging to any of the possible ones. The columns (which do not necessarily sum up to one) show how much data from all classes is classified as part of a given one. The main diagonal of the confusion matrix represents, therefore, the average accuracy for each class (ie. what percentage of the samples from a given class are correctly classified). Average accuracy over all classes is calculated as the mean value of the matrix diagonal. Supposing a perfect classification, the expected result would be the identity matrix.

The architecture utilized for the system is shown in Figure 4.1.

4.1 Supervised Learning

In this part of the experiments crisp labels for the speech samples are available for all the training data. These labels are used to train the SVMs as explained in Section 3.4 in a crisp-input crisp-output configuration.

Features are extracted and aligned for all the available data as described in Section 3.1, accounting a total of 8 sets of features for

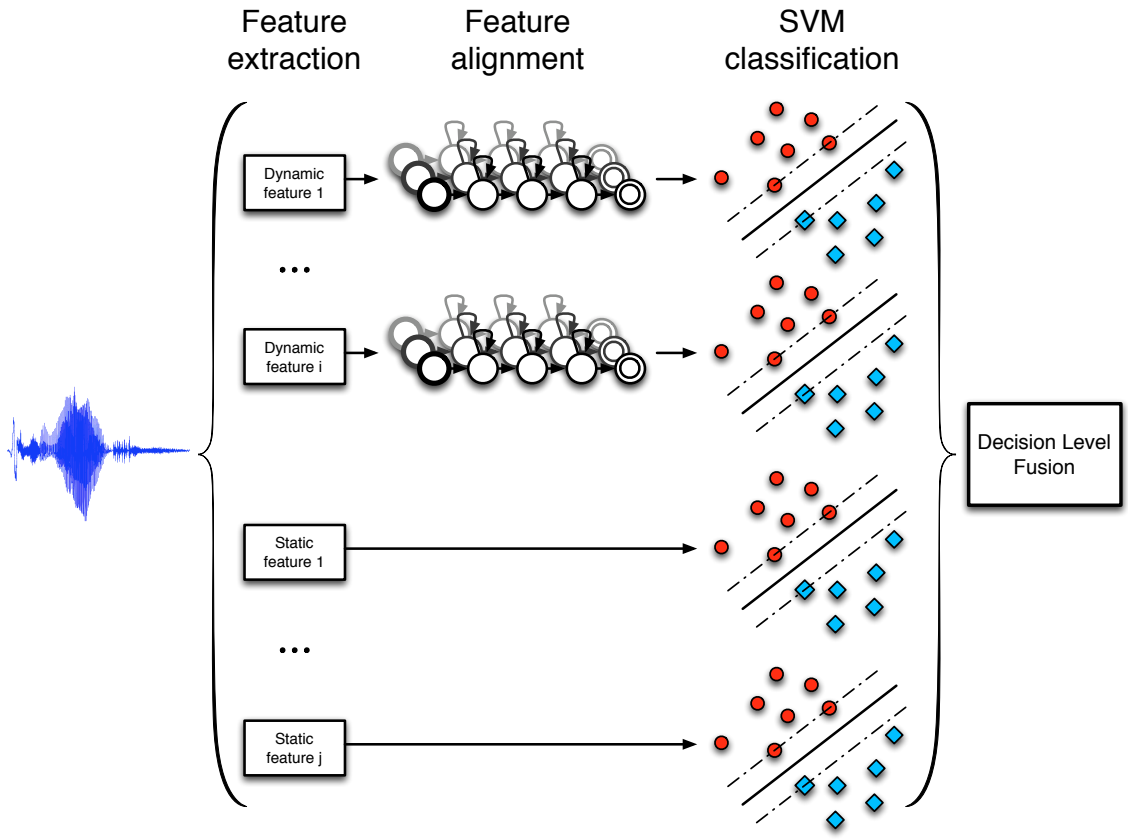

Figure 4.1: System architecture. Sequential features are taken through a feature alignment process based on HMM likelihood before being used for training or test, as described in Section 3.2.3. Static features do not need to go through the alignment process since they are already of a fixed length. After all the features are aligned, SVM classification is conducted and, finally, decisions are taken based on the fusion of all the feature sets.

Table 4.1: Confusion matrix for the male automatic classification experiments, conducted with the WaSeP dataset. Average accuracy 87%.

	F	D	H	N	S	A
Fear	**.87**	.01	.06	.01	.00	.05
Disgust	.01	**.92**	.06	.00	.00	.01
Happiness	.13	.07	**.67**	.09	.01	.04
Neutral	.00	.01	.06	**.93**	.00	.00
Sadness	.00	.00	.00	.00	**.99**	.00
Anger	.01	.07	.04	.02	.00	**.85**

Table 4.2: Confusion matrix for the female automatic classification experiments, conducted with the WaSeP dataset. Average accuracy 84%.

	F	D	H	N	S	A
Fear	**.74**	.04	.16	.01	.01	.05
Disgust	.02	**.90**	.04	.01	.01	.03
Happiness	.03	.02	**.77**	.13	.02	.03
Neutral	.00	.00	.09	**.87**	.04	.00
Sadness	.01	.00	.02	.05	**.91**	.00
Anger	.01	.07	.02	.02	.01	**.88**

Table 4.3: Confusion matrix for the gender-independent automatic classification experiments, conducted with the WaSeP dataset. Average accuracy 84%.

	F	D	H	N	S	A
Fear	**.80**	.03	.08	.01	.02	.06
Disgust	.01	**.88**	.05	.00	.04	.03
Happiness	.08	.02	**.71**	.12	.04	.03
Neutral	.00	.01	.16	**.82**	.01	.00
Sadness	.02	.00	.03	.01	**.95**	.00
Anger	.01	.07	.02	.03	.00	**.86**

each speech file. With the aligned features, two sets are created on a 90% and 10% distribution basis for training and testing of the SVMs, respectively in a 10 fold cross-validation style.

The confusion matrices obtained with these tests are shown in Tables 4.1 , 4.2 and 4.3 for the male speaker, female speaker and gender-independent cases respectively using the WaSeP corpus.

The average accuracy in these cases is 87% for the male, 85% for the female and 84% for the gender-independent case. In all experiments

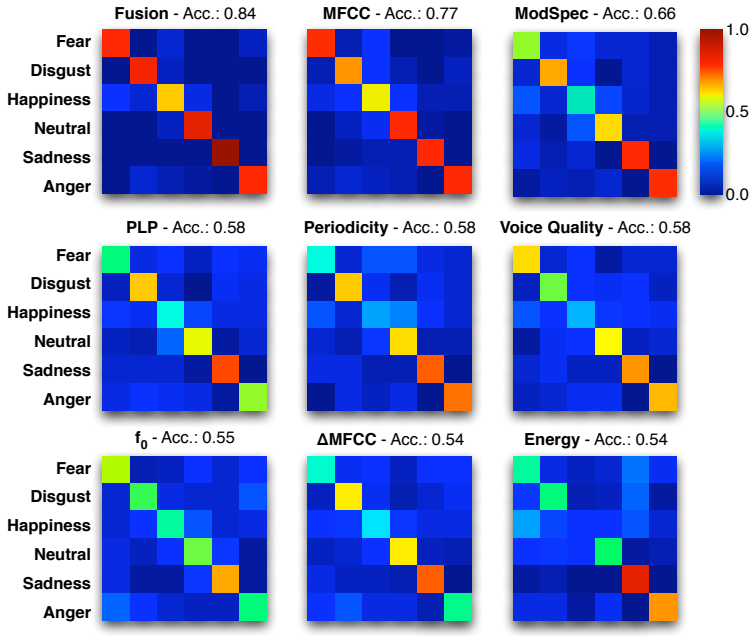

Figure 4.2: Confusion matrices of the automatic classification experiment with the WaSeP corpus. Matrices are displayed in a descending order of accuracy, starting with the fusion of all feature sets. Columns and rows are in the same order, representing the main diagonal the accuracy achieved for each class. Rows sum up to one since they represent the total number of labels considered for each class. Columns, however, do not necessarily sum up to one, as they represent the output of the classifier. Warm and cold colours represent high and low values, as coded in the colour bar to the right.

happiness is the emotion with a lowest accuracy, not being in any case over 80%. In opposition to this, there is sadness, which is in all cases well classified with an accuracy over 90% in all the experiments. In order to better observe the effect of the fusion, Figure 4.2 has been generated. This figure presents scaled images of the confusion matrices produced by each of the feature sets alone and their fusion.

For comparison of the results, tests are also carried out with the EmoDB dataset. Evaluation of the proposed system on this dataset shows a 77% accuracy, slightly lower than the 84% obtained for the WaSeP dataset. The confusion matrix for this test can be seen in Table 4.4 and the corresponding scaled image is shown in Figure 4.3.

Table 4.4: Confusion matrix of the gender-independent automatic classification experiments, conducted with the EmoDB dataset.

	F	D	H	N	S	A	B
Fear	**.77**	.01	.10	.01	.06	.00	.05
Disgust	.10	**.69**	.04	.04	.07	.01	.05
Happiness	.08	.02	**.53**	.00	.03	.00	.33
Neutral	.00	.01	.00	**.80**	.16	.03	.00
Sadness	.01	.01	.00	.06	**.92**	.00	.00
Anger	.00	.00	.00	.03	.07	**.89**	.00
Boredom	.04	.01	.14	.00	.00	.00	**.81**

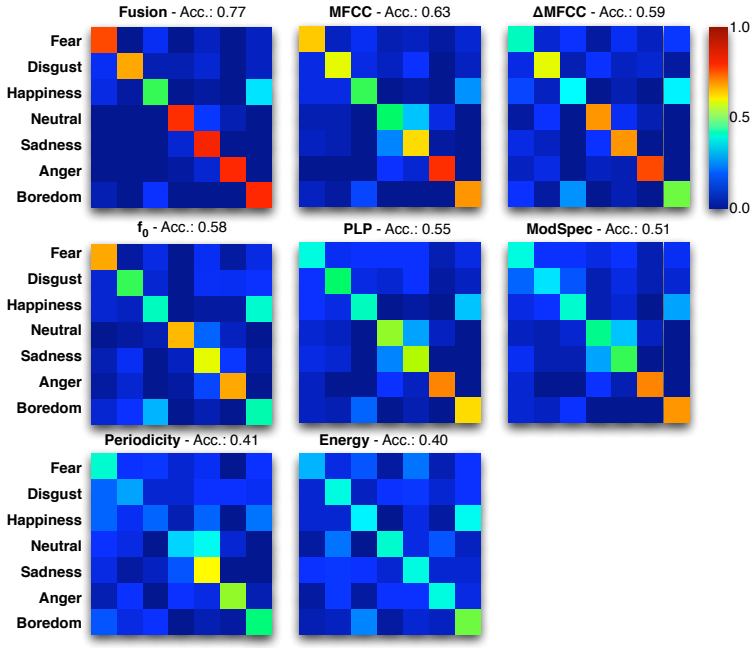

Figure 4.3: Confusion matrices of the automatic classification experiment with the EmoDB corpus. Matrices are displayed in a descending order of accuracy, starting with the fusion of all feature sets. Columns and rows are in the same order, representing the main diagonal the accuracy achieved for each class. Rows sum up to one since they represent the total number of labels considered for each class. Columns, however, do not necessarily sum up to one, as they represent the output of the classifier. Warm and cold colours represent high and low values, as coded in the colour bar to the right. In this case, Voice Quality features are not used, since their extraction for long sequences is complex and phoneme level annotation might have been necessary.

4.2 Partially Supervised Learning

Experiments have also been conducted within the partially supervised learning framework. First, on a semi-supervised learning approach, where the system is capable of automatically generating labels for the data based on only a few reference datapoints labelled by an expert. In the second approach (active learning), it is the expert who labels the data but this time only those datapoints chosen by the system itself will be labelled.

4.2.1 Semi-supervised learning

In this section there are partial results shown in order to permit a better comprehension of the techniques described in Section 3.6, and their level of accuracy for a value of the parameter $k = 5$.

On a first experiment, fuzzy labels are automatically generated for 600 unlabelled samples, following the procedure described in Figures 3.19, 3.20 and 3.21. As a initial reference set, 50 labelled samples per class are used. Figure 4.4 shows a representation of the reference set and the test set. For the latter, both real and artificial labels are shown. Every different colour represents one of the six different classes considered. Since the automatically generated k-NN labels are fuzzy, only the class with highest degree of membership is represented. Nevertheless, it is still possible to observe the close similarity obtained with the artificial labels compared with the real ones. In order to represent larger amount of the information contained in the fuzzy labels, Figure 4.5 is generated. This one shows the amount or correct and wrong labels generated by the k-NN procedure considering not only the class with a highest degree of membership, but also the second and third ones.

The rate of correct labels in the three different situations are: 52%, 77%, and 89%, for the cases of 1, 2 and 3 best scoring classes respectively.

As it has been observed, when there are enough reference points, k-NN produces considerably good results for a fuzzy label generation process. Nevertheless, our experiments are aimed studying the results when the reference set is not large. Several experiments have been conducted within this approach with the aim to produce a significant improvement in the classification performance when the system is trained with a reduced set of crisp labels, extended with a large number of fuzzy automatically generated labels. The baseline in this experiment has been lowered to resemble a situation with small amounts of data available. This baseline provides an average accuracy of 73% for

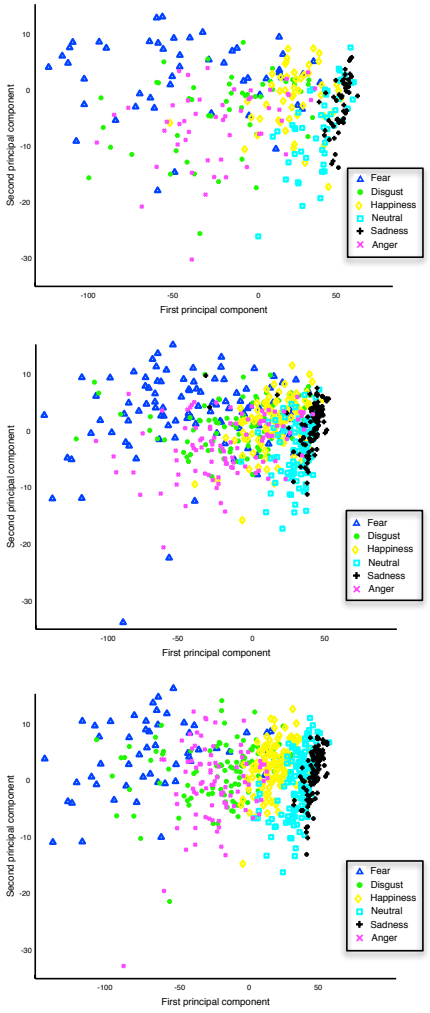

Figure 4.4: All three figures correspond to the first two principal components of the mfcc feature set, obtained from a female speaker in the WaSeP corpus. Top figure: Reference points, annotated by an expert (50 per class). Center figure: The plot corresponds to the real labels of the test set. Bottom figure: Representation of the classes with a highest degree of membership, according to the labels automatically generated by use of the k-NN algorithm with a parameter value $k = 5$. Under the assumption of perfect automatic labeling, the center and bottom figures should look exactly the same.

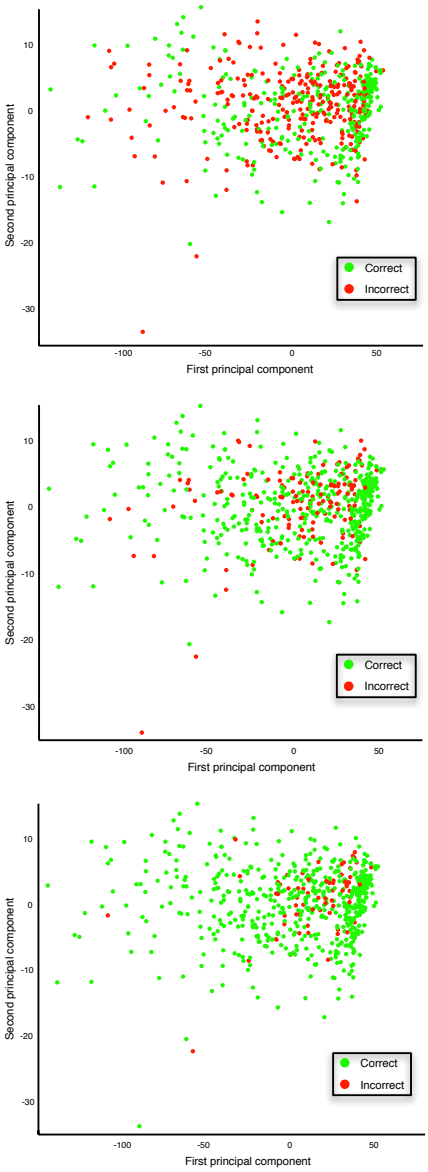

Figure 4.5: Representation of correct (green) and incorrect (red) fuzzy labels generated by k-NN with $k = 5$ and using as a reference set 50 samples per class. Top figure shows the correct labels, considering only the class with a highest membership. Figure in the center uses not only the first, but also the second highest membership degree class to compare with the real label. Bottom figure considers also the third highest membership.

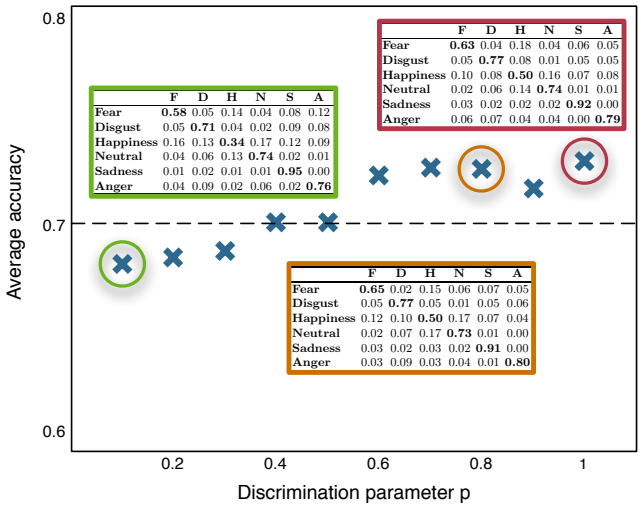

Figure 4.6: Average accuracy obtained for different values of the discrimination parameter p. Confusion matrices are shown for values of p equal to 0.1 and 0.8 as well as the baseline, represented in the graph as p = 1.

Table 4.5: Confusion matrix of the gender-independent semi-supervised learning experiments conducted with the WaSeP dataset, for a distrimination parameter $p = 0.1$. Average accuracy = 68%.

	F	D	H	N	S	A
Fear	**.58**	.05	.14	.04	.08	.12
Disgust	.05	**.71**	.04	.02	.09	.08
Happiness	.16	.13	**.34**	.17	.12	.09
Neutral	.04	.06	.13	**.74**	.02	.01
Sadness	.01	.02	.01	.01	**.95**	.00
Anger	.04	.09	.02	.06	.02	**.76**

the gender-independent case with the WaSeP corpus. A sweep analysis over the parameter p shows that the maximum is found at $p = 0.8$, achieving also an average of 73%. Graphical representation of this analysis is shown in Figure 4.6. Gender-independent results obtained with this approach are shown in Tables 4.5, 4.6 and 4.7 for values of the discrimination parameter p equal to 0.1, 0.8 and 1 respectively.

Another experiment has been carried out to check the effect of the discrimination parameter p over the performance when the k-NN labels are generated with a large reference set. Training of the SVMs is then conducted with the k-NN labels together with a very reduced

Table 4.6: Confusion matrix of the gender-independent semi-supervised learning experiments conducted with the WaSeP dataset, for a distrimination parameter $p = 0.8$. Average accuracy = 73%.

	F	D	H	N	S	A
Fear	**.65**	.02	.15	.06	.07	.05
Disgust	.05	**.77**	.05	.01	.05	.06
Happiness	.12	.10	**.50**	.17	.07	.04
Neutral	.02	.07	.17	**.73**	.01	.00
Sadness	.03	.02	.03	.02	**.91**	.00
Anger	.03	.09	.03	.04	.01	**.80**

Table 4.7: Confusion matrix of the gender-independent semi-supervised learning experiments conducted with the WaSeP dataset, for a distrimination parameter $p = 1$ which corresponds to the new baseline since no k-NN labels are considered as they cannot have a confidence higher than 1. Average accuracy = 73%.

	F	D	H	N	S	A
Fear	**.63**	.04	.18	.04	.06	.05
Disgust	.05	**.77**	.08	.01	.05	.05
Happiness	.10	.08	**.50**	.16	.07	.08
Neutral	.02	.06	.14	**.74**	.01	.01
Sadness	.03	.02	.02	.02	**.92**	.00
Anger	.06	.07	.04	.04	.00	**.79**

part of the reference set. The way in which the sets are utilised is described in Figure 4.7.

With this approach it is possible to observe the effect of the discriminative parameter p over the system accuracy, as shown in Figure 4.8.

Not only this, it is also observed that an increase of 10% in accuracy can be achieved by extending the reduced training set with automatically generated labels. This proves that the automatic labels can be used to train the system, but the amount of error introduced should be kept as low as possible.

4.2.2 Active Learning

These experiments are designed with the intention of evaluating how the system performs when the training data is selected by the system itself. The flow chart of the the system is the represented in Figure 3.25. In the iterative training and evaluation process, each iteration represents an increase of 10 samples in the training set. For evaluation of the results obtained in this section, Figure 4.9 has been

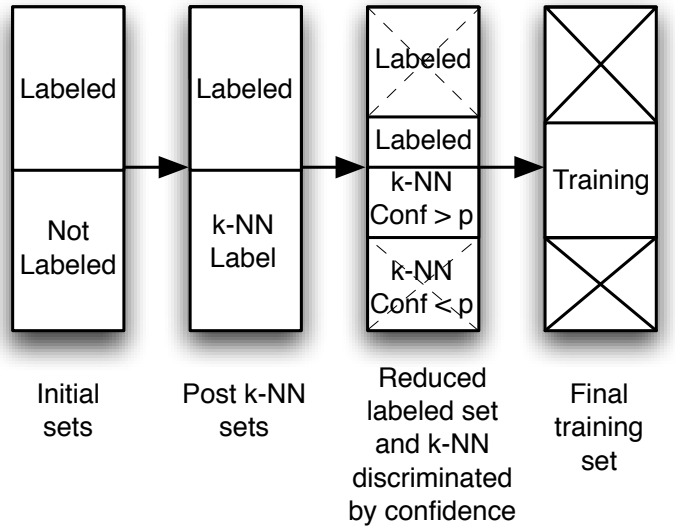

Figure 4.7: Evolution of the training sets over time for validation of the proposed confidence measure. k-NN labels are automatically generated from an extended reference set to reduce the error in them. To train the SVMs, not all the reference set is used, but only a small fraction of it, together with the artificially labeled samples of confidence higher than the discrimination parameter p.

generated. This figure shows the average accuracy of the trained system for each step of the iterative process in the gender independent experiments. Tables 4.8, 4.9 and 4.10 show the confusion matrices of the active learning experiment after the last iteration (with all the available training data used). Average accuracy in this case, 88% for the gender-independent case, is higher than the 84% obtained in Section 4.1 due to a larger training set utilized for a better representation of the effect produced by the active learning. It can be observed that an 80% accuracy level is reached with only 40 iterations. This means that by using only one third of the training data used in previous experiments, a similar level of accuracy is achieved. Further, it should be noted that after only a few iterations some sort of saturation point is reached, that comprises only a small portion of the available data for training.

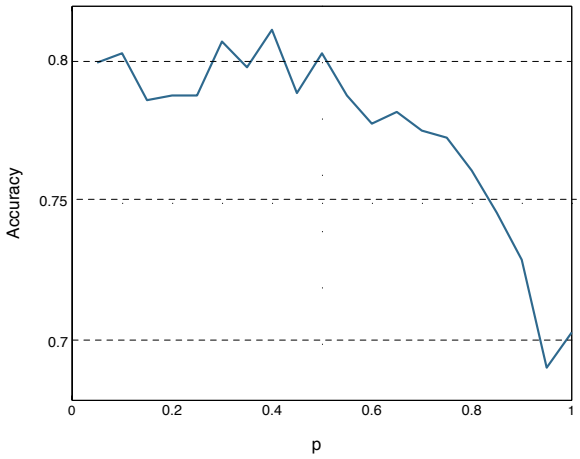

Figure 4.8: Classification accuracy for different values of p using k-NN labels generated with a large reference set. The reference set was later reduced and only a 10% of it was used to train the SVMs together with the new labels.

Table 4.8: Confusion matrix of the male active learning experiments conducted with the WaSeP dataset. Average accuracy = 90%.

	F	D	H	N	S	A
Fear	**.85**	.01	.12	.00	.00	.02
Disgust	.00	**.92**	.07	.00	.00	.01
Happiness	.09	.02	**.80**	.06	.00	.03
Neutral	.00	.00	.06	**.93**	.00	.00
Sadness	.00	.00	.01	.00	**.99**	.00
Anger	.01	.02	.03	.01	.00	**.93**

Table 4.9: Confusion matrix of the female active learning experiments conducted with the WaSeP dataset. Average accuracy = 85%.

	F	D	H	N	S	A
Fear	**.72**	.04	.17	.01	.01	.05
Disgust	.02	**.86**	.06	.01	.01	.04
Happiness	.02	.02	**.82**	.10	.02	.02
Neutral	.01	.00	.12	**.85**	.02	.00
Sadness	.01	.00	.02	.03	**.94**	.00
Anger	.01	.03	.02	.01	.00	**.93**

Table 4.10: Confusion matrix of the gender-independent active learning experiments conducted with the WaSeP dataset. Average accuracy = 88%.

	F	D	H	N	S	A
Fear	**.83**	.01	.11	.01	.01	.02
Disgust	.01	**.89**	.05	.01	.02	.03
Happiness	.05	.02	**.79**	.10	.02	.02
Neutral	.01	.00	.10	**.88**	.00	.01
Sadness	.01	.00	.02	.01	**.97**	.00
Anger	.01	.02	.02	.01	.00	**.93**

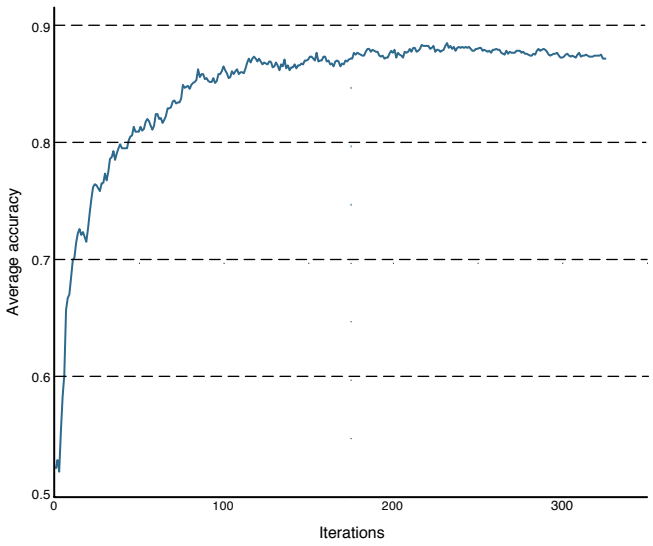

Figure 4.9: Active learning accuracy over iterations, for the gender-independent case conducted with the WaSeP dataset. Each iteration represents 10 new labels used for training.

4.3 Discussion

The confusion matrices in Section 4.1 provide a good basis for the comparison of human and machine capabilities and errors. A first glance at them shows that human and machine performances are quite similar on an overall scale. With the WaSeP dataset, the 84% accuracy rate obtained is exactly the same as that from humans in average. In the case of EmoDB, a 77% accuracy rate performed by automatic recognition compares to the 84.7% achieved by humans. The average scores for both datasets are very similar, however, there are a few patterns that seem to diverge quite strongly. First of all, with respect to the WaSeP dataset (compare tables 2.1 and 4.3) a lot of human perception errors are due to votes for the class neutral, which leads to the assumption that humans tend to vote for a class that does not provide clear evidence for the intended emotion. The machine does not vote for neutral that frequently, partly of course due to the fact that the word neutral does not bear any meaning to it.

Further, it is seen that happiness is badly recognized in both automatic classification experiments. For the human perception tests this does not hold. Even though, there are evidences in the literature that happiness is often difficult to recognize as reported in Scherer et al. (2001), where it only reached an accuracy of 48%. In Wendt (2007) it is also reported that the recognition performance of humans with respect to the male recordings of happiness is at 66%, with the main confusions towards neutral speech, which is confirmed by the automatic classification in Table 4.3.

Anger is in both human perception experiments the best recognized emotion, whereas the automatic classification does not reach such high levels of accuracy. The classifier on the other hand reaches great recognition performances for the sad expressions, which humans seem to confuse quite frequently for both datasets. In Wendt (2007) once more a gender difference is reported: the female sad expressions are only recognized at an accuracy of 65% and again mostly confused with neutral.

With the lowest human accuracy in the WaSeP experiments, there is disgust, which is in accordance to Van Bezooyen (1984); Scherer et al. (1991). It is also among the worst in EmoDB, only second to happiness. This effect was initially present in our experiments with the standard features. The design of a new feature set, as explained in Section 3.1.7, with a high accuracy for disgust permits an improvement on the fusion rates of about 20%.

It must be noticed also in Figures 4.2 and 4.3 that the accuracy obtained with the fusion of the features outperforms any of them in-

dividually. This proves that the proposed fusion technique produces good results despite being a very simple approach.

As for the semi-supervised experiments with the data labelled by the k-NN algorithm, analysis of the obtained results (see Figure 4.6) shows that the use of unlabelled data in the training process does not improve the baseline. The baseline in this experiment has been lowered to resemble a situation with small amounts of data available (i.e. only 20 samples per category). This baseline provides an average accuracy of 73% for the gender-independent case. As already commented in section 4.2.1, a sweep simulation over different values of p was conducted, finding its maximum at $p = 0.8$. This, however, does not represent an improvement with respect to the baseline. Given the large amount of automatically generated labels used in the training, it is wise to think that the style in which the experiments were designed was not optimal. There might be different reasons for this, like a bad selection of the confidence measure or the excessive amount of error present in the self-labelled samples. Assuming that the chosen confidence measure is correct, better results are to be expected if the artificial labels are more accurately generated. To prove this assumption, a second experiment has been conducted where the aim was to reduce the error artificially put into the k-NN labels. The set of labelled data used for training the SVMs is now reduced in order to be able to measure the accuracy with most of the training data obtained by the k-NN process. As seen in Figure 4.8, if a small amount of error is introduced artificially, good performance improvements can be achieved. It makes sense to believe that with an automatically labelling algorithm that inserts less error than k-NN it may be possible to use semi-supervised learning with good accuracy results. The utilised confidence measure proved to give good results when the artificial labels contain more correct information.

In opposition to the poor results encountered with the semi-supervised approach, active learning proved to be a very good approach for reducing the amount of labelled data required. It can be seen that after approximately 60 iterations (100 training samples per class) the accuracy already reaches a similar level of performance to that of the supervised learning approach, using twice as much data. This means a large reduction of the required amount of labelled data, proving that the approach works and produces good results. In Figure 4.9 it is observed that after a certain iteration (around the 100-th), the addition of new labelled data does not lead to an accuracy increase. We can, therefore, affirm that the active learning works well and can significantly reduce the required amount of data without penalising the obtained results.

The obtained results are comparable to the reference studies cited in this work, outperforming most previously proposed solutions for the problem of automatic emotion classification.

5 Summary and Conclusions

5.1 Summary

This study presents a new approach for the task of automatic emotion classification based on speech data. A combination of standard sets of features were described and a new feature set was developed to better resemble the human performance. A feature alignment process based on HMM likelihood was also introduced in the work, which represents a novelty for emotion classification. With this technique, feature sequences of different lengths can be encoded into vectors of fixed dimensions, allowing comparison among them. Futher, this will also allow the comparison of static and dynamic features. Multi-classifier multi-class SVMs were trained in a supervised style and used for evaluation of accuracy. Comparison of the obtained results with the human perception tests were conducted showing that similar accuracies can be achieved presenting also similar cross-class confusions. This fact proves that the proposed combination of features is capable of representing the human perception capabilities well.

There are also contributions in the field of partially supervised learning, where semi-supervised and active learning techniques have been subject to study within the classification scheme in this work. A confidence meassure was proposed for both cases, representing the level of correction in the first case and the distance to the decision boundaries in the latter. For semi-supervised experiments, it is interesting to know the amount of error introduced, being important in this case the samples with a highest confidence. On the other hand, for active learning, the confidence measure is used to find the most informative samples, represented by a low confidence value. The confidence measure definition was a key factor since the partially supervised experiments require such a parameter to emit decisions on what to do with each particular available sample. In the semi-supervised experiments, results proved that for very reduced reference sets k-NN introduces too much error causing the system to decrease its accuracy. Nevertheless, a second experiment with reduced k-NN error proved that

the system accuracy can be increased with artificial labels. For this purpose, a procedure better than k-NN must be found, in order to reduce the error introduced artificially. On the contrary, active learning provided great results and places itself as a very good option to cut down the cost of labeling by reducing the amount of data required in training steps. The proposed confidence measure ensures a continuous increase in the system accuracy, starting from a very reduced labeled set and asking the expert to label only the least confident samples on every iteration.

In general, good results have been achieved, outperforming previous studies on emotion classification and also regarding partially supervised learning. Nevertheless, there are still open issues that may allow improvements in different aspects of the proposed approach.

5.2 Open Issues and Future Work

As already explained in Chapter 2, the datasets utilised for this work are standard datasets extensively used in the literature. However, they both have the drawback of being characterised only by acted emotions, which adds a non realistic component to our experiments. Development of a real-situation emotional dataset would be desirable in order to more accurately represent affective behaviour.

During the study of the different standard feature sets commonly utilised for speech recognition and other tasks based on speech data, it was observed that all of them are based on the signal energy. Some on energy distribution over different frames and others on the energy distribution within a single frame. This, however, implies that quite a lot of information is being discarded, since the representation of a signal in the Fourier domain does not only hold energy, but also phase properties. Tests have been conducted where the Fourier transformation of a speech signal has been modified to contain either only energy or only phase information. Results showed that the reconstruction obtained only with energy information is incomprehensible, while phase-based reconstruction proves to be understandable by the human ear (Hayes et al., 1980). Given these results, it might be interesting in future work the development of new feature sets that hold information of both phase and energy in a combined style.

As a fusion, a simple multiplication and normalisation of the outputs provided by different classifiers was considered, providing improvements with respect to all the feature separatedly. Nevertheless, there may be more complex approaches that can achieve better accuracies considering the characteristics of each feature set individually. For example, MFCC features perform very well in general (see Figures

4.2 and 4.3) but energy not so well. Therefore, these conclusions could be used to define different ways of weighting each classifier's output.

The confidence meassure proposed in Section 3.6.1 proved to give good results while its computation is straight forward. Nevertheless, more robust measures could probably be implemented considering the results shown in 4.5. Since the labels that our system utilises are fuzzy, it would make sense to consider not only the class with highest member of degree, but the combination with other classes, too.

Regarding the semi-supervised approach studied in this thesis, it was concluded that k-NN alone does not produce confident enough labels when the reference set is very small. To solve this problem, it might be possible to use co-training techniques, where an iterative cooperation among the k-NN process and the SVMs can be defined and used to improve the automatically generated labels. Also, it should be possible to find different algorithms that could perform better in this particular problem like Naive Bayes (NB), Fuzzy Assignmemt Procedure for Nominal Sorting (PROAFTN for its acronym in French) or even another early stage of SVMs trained with the reference set.

List of Tables

List of Figures

Bibliography

Banse, R., Scherer, K. R., 1996. Acoustic profiles in vocal emotion expression. Journal of Personality and Social Psychology 70 (3), 614--636.

Bicego, M., Murino, V., Figueiredo, M., 2003. Similarity-based clustering of sequences using hidden markov models. In: Perner, P., Rosenfeld, A. (Eds.), Machine Learning and Data Mining in Pattern Recognition. Vol. 2734. Springer, pp. 95--104.

Bishop, C. M., October 2006. Pattern Recognition and Machine Learning (Information Science and Statistics), 1st Edition. Springer.

Blum, A., Mitchell, T., 1998. Combining labeled and unlabeled data with co-training. In: Proceedings of the eleventh annual conference on Computational learning theory. COLT' 98. ACM, New York, NY, USA, pp. 92--100.

Burkhardt, F., Paeschke, A., Rolfes, M., Sendlmeier, W., Weiss, B., 2005. A database of german emotional speech. In: Proceedings of Interspeech 2005. ISCA, pp. 1517--1520.

Cedergren, H. J., Perreault, H., 1994. Speech rate and syllable timing in spontaneous speech. In: Third International Conference on Spoken Language Processing (ICSLP 94). IEEE, pp. 1087--1090.

Crystal, T. H., House, A. S., 1990. Articulation rate and the duration of syllables and stress groups in connected speech. Journal of The Acoustical Society of America 88, 101--112.

Druck, G., Mann, G., McCallum, A., 2008. Learning from labeled features using generalized expectation criteria. In: Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval. SIGIR '08. ACM, New York, NY, USA, pp. 595--602.

Duda, R. O., Hart, P. E., Stork, D. G., 2001. Pattern Classification, 2nd Edition. Wiley, New York.

Fang, Z., Guoliang, Z., Zhanjiang, S., 2001. Comparison of different implementations of mfcc. J. Comput. Sci. Technol. 16 (6), 582--589.

Gobl, C., Bennett, E., Chasaide, A. N., Sept. 2002. Expressive synthesis: how crucial is voice quality? In: IEEE Workshop on Speech Synthesis, 2002. IEEE, pp. 91--94.

Hayes, M., Lim, J., Oppenheim, A., dec 1980. Signal reconstruction from phase or magnitude. Acoustics, Speech and Signal Processing, IEEE Transactions on 28 (6), 672 -- 680.

Hermansky, H., Apr. 1990. Perceptual linear predictive (PLP) analysis of speech. Acoustical Society of America Journal 87, 1738--1752.

Hermansky, H., Morgan, N., 1994. Rasta processing of speech. IEEE Transactions on Speech and Audio Processing, special issue on Robust Speech Recognition 2, 578--589.

Kahsay, L., Schwenker, F., Palm, G., 2005. Comparison of multiclass svm decomposition schemes for visual object recognition. In: Kropatsch, W. G., Sablatnig, R., Hanbury, A. (Eds.), Pattern Recognition. Vol. 3663 of Lecture Notes in Computer Science. Springer Berlin / Heidelberg, pp. 334--341.

Kane, J., Kane, M., Gobl, C., 2010. A spectral lf model based approach to voice source parameterisation. In: Proceedings of Interspeech 2010. ISCA, pp. 2606--2609.

Kane, J., Scherer, S., Gobl, C., under review. Glottal source parameterisation in the frequency domain based on the lf model spectrum. Speech Communication: Special Issue on Advanced Voice Function Assessment.

Keltner, D., Ekman, P., 2003. Handbook of Affective Sciences - Introduction: Expression of Emotion. Affective Science. Oxford University Press, Ch. 21, pp. 411--414.

Keltner, D., Ekman, P., Gonzaga, G. C., Beer, J., 2003. Handbook of Affective Sciences - Facial expression of emotion. Affective Science. Oxford University Press, Ch. 22, pp. 415--432.

Kuncheva, L. I., 2001. Using measures of similarity and inclusion for multiple classifier fusion by decision templates. Fuzzy Sets and Systems 122 (3), 401--407.

Kuncheva, L. I., 2004. Combining pattern classifiers: methods and algorithms. Wiley.

Kuncheva, L. I., Bezdek, J. C., Duin, R. P. W., 2001. Decision templates for multiple classifier fusion: an experimental comparison. Pattern Recognition 34 (2), 299--314.

Li, D., Sethi, I. K., Dimitrova, N., McGee, T., 2001. Classification of general audio data for content-based retrieval. Pattern Recognition Letters 22 (5), 533 -- 544.

Logan, B., 2000. Mel frequency cepstral coefficients for music modeling. In: In International Symposium on Music Information Retrieval.

Lomasky, R., Brodley, C. E., Aernecke, M., Walt, D., Friedl, M. A., 2007. Active class selection. In: ECML'07. pp. 640--647.

Mayergoyz, I., 2003. Mathematical Models of Hysteresis and their Applications. Springer.

Pearson, K., 1901. On lines and planes of closest fit to systems of points in space, 559–572.

Pfitzinger, H., Burger, S., Heid, S., 1996. Syllable detection in read and spontaneous speech. In: Proceedings of The 4th International Conference on Spoken Language Processing (ICSLP'96). Vol. 2. Philadelphia, pp. 1261--1264.

Scherer, K. R., Banse, R., Wallbott, H. G., 2001. Emotion inferences from vocal expression correlate across languages and cultures. Journal of Cross-Cultural Psychology 32, 76--92.

Scherer, K. R., Banse, R., Wallbott, H. G., Goldbeck, T., 1991. Vocal cues in emotion encoding and decoding. Motivation and Emotion 15, 123--148.

Scherer, K. R., Johnstone, T., Klasmeyer, G., 2003. Handbook of Affective Sciences - Vocal expression of emotion. Affective Science. Oxford University Press, Ch. 23, pp. 433--456.

Scherer, S., Kane, J., Gobl, C., Schwenker, F., under review. Investigating fuzzy-input fuzzy-output support vector machines for robust voice quality classification. IEEE Transactions on Audio, Speech and Language Processing.

Scherer, S., Oubbati, M., Schwenker, F., Palm, G., 2008. Real-time emotion recognition from speech using echo state networks. In: Proceedings of the 3rd IAPR workshop on Artificial Neural Networks in Pattern Recognition (ANNPR'08). Springer, Berlin, Heidelberg, pp. 205--216.

Scherer, S., Schwenker, F., Palm, G., Sept. 2007. Classifier fusion for emotion recognition from speech. In: 3rd IET International Conference on Intelligent Environments 2007 (IE07). IEEE, pp. 152--155.

Settles, B., 2009. Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin--Madison.

Thiel, C., 2009. Multiple classifier systems incorporating uncertainty. Ph.D. thesis, Ulm University.

Thiel, C., Giacco, F., Schwenker, F., Palm, G., 2009. Comparison of neural classification algorithms applied to land cover mapping. In: Proceeding of the 2009 conference on New Directions in Neural Networks. IOS Press, Amsterdam, The Netherlands, The Netherlands, pp. 254--263.

Thiel, C., Scherer, S., Schwenker, F., 2007. Fuzzy-input fuzzy-output one-against-all support vector machines. In: 11th International Conference on Knowledge-Based and Intelligent Information and Engineering Systems (KES 2007). Vol. 3 of Lecture Notes in Artificial Intelligence. Springer, pp. 156--165.

Van Bezooyen, R., 1984. Characteristics and Recognizability of Vocal Expressions of Emotion. Foris Pubns USA.

Vlasenko, B., Schuller, B., Wendemuth, A., Rigoll, G., 2007. Frame vs. turn-level: Emotion recognition from speech considering static and dynamic processing. In: Proceedings of the 2nd international conference on Affective Computing and Intelligent Interaction (ACII'07). Springer-Verlag, Berlin, Heidelberg, pp. 139--147.

Wagner, J., Vogt, T., André, E., 2007. A systematic comparison of different hmm designs for emotion recognition from acted and spontaneous speech. In: Proceedings of the 2nd international conference on Affective Computing and Intelligent Interaction (ACII'07). Springer-Verlag, Berlin, Heidelberg, pp. 114--125.

Wendt, B., 2007. Analysen Emotionaler Prosodie. Vol. 20 of Hallesche Schriften zur Sprechwissenschaft und Phonetik. Peter Lang Internationaler Verlag der Wissenschaften.

Wendt, B., Scheich, H., 2002. The "Magdeburger Prosodie Korpus" - a spoken language corpus for fMRI-studies. In: Speech Prosody 2002. SProSIG, pp. 699--701.

Yanushevskaya, I., Gobl, C., Ní Chasaide, A., 2008. Voice quality and loudness in affect perception. In: Speech Prosody 2008. Campinas, Brazil.

Zhu, X., 2005. Semi-supervised learning literature survey. Tech. Rep. 1530, Computer Sciences, University of Wisconsin-Madison.