Design and realisation of an audiovisual speech activity detector

van Bree, K.C.

Award date:
2006

TU/e technische universiteit eindhoven

Faculty of Electrical Engineering
Section Design Technology For Electronic Systems (ICS/ES)
ICS-ES 867

Master's Thesis

DESIGN AND REALISATION OF AN AUDIOVISUAL SPEECH ACTIVITY DETECTOR

K.C. van Bree

Supervisor:	prof.dr.ir. G. de Haan
Coach:	dr.ir. H.J.W. Belt
Date:	February 2006

The Faculty of Electrical Engineering of the Eindhoven University of Technology does not accept any responsibility regarding the contents of Master's Theses

Title: Design and realisation of an audiovisual speech activity detector

Author(s): K.C. van Bree

Keywords: speech activity detection, audiovisual, lip detection, mouth feature extraction,

Abstract: For many speech telecommunication technologies a robust speech activity detector is important. An audio-only speech detector will give false positives when the interfering signal is speech or has speech characteristics. The modality video is suitable to solve this problem. In this report the approach to and implementation of a decision-based audiovisual speech detector is given. Acoustic and visual features of speech are first separately investigated. Firstly, a common method for speech detection based on audio has been built. Secondly, from the video data the mouth features have been extracted with the implementation of an own idea. The visual features were used to create a conservative visual non-speech detector. The low false detection rate makes the visual non-speech detector suitable to rule out some false speech detections of an audio only solution. Finally, the combination of the audio detector and the video detector leads to an audiovisual speech detector which uses basic mouth features and a common acoustical speech detection method to outperform an audio-only solution.

Conclusions: An audiovisual speech detection algorithm has been designed and implemented in a Matlab environment. For the audio-based detection an own algorithm has been designed based on general knowledge available in the literature. A conservative visual non-speech detection algorithm was designed and implemented. The fusion of the two modalities was done on decision level. This resulted in an audiovisual speech detection algorithm, of which the performance is better than that of the audio-only speech detector. Good results were achieved by the low-cost basic mouth feature extraction algorithm.

Preface

The research documented in this report is the graduation project of the author concluding his Master of science studies electrical engineering at the 'Eindhoven University of Technology'. The research was conducted in the video processing group of Philips research at the High Tech Campus in Eindhoven under the supervision of Prof. dr. ir. G. de Haan and dr. ir. H.J.W. Belt.

Contents

Preface **v**

1 **Introduction** **1**

1.1 Developments on audiovisual speech analysis 2

1.2 Assignment 4

1.3 Problem description 4

2 **Acoustical speech detection** **6**

2.1 Speech production 6

2.2 Speech activity detection 7

2.3 Speech Classification 12

2.3.1 Linear Prediction 12

2.3.2 Vowel groups 17

2.4 Experiments 18

2.4.1 Pitch based speech detection 18

2.4.2 Vowel detection 19

3 **Visual speech detection** **23**

3.1 General visual speech 23

3.2 Face finding 24

3.3 Visual speech feature extraction 25

3.3.1 Lip finding 25

3.3.2 Mouth finding fall-back procedure 38

3.4 Visual speech detection 40

3.4.1 Visual non-speech detection 40

3.4.2 Visual vowel detection 41

3.5 Experiments 44

3.5.1 Mouth Movement Dynamics 44

3.5.2 Lip and Skin Colour adaptation 44

3.5.3 Visual non-speech detection tests 46

3.5.4 Visual vowel detection tests 48

4 Audiovisual Speech Detection 52

4.1 Modality fusion 52

4.2 Integration of audiovisual speech detection 53

4.3 Audiovisual speech detection test 53

5 Conclusions and Recommendations 58

Chapter 1

Introduction

It is known that in normal conversation visual cues are important in conveying information. Especially face expression and mouth movement contribute to the intelligibility of speech. This has been known for a long time already [1]. Video telephony has been researched over 40 years and video conference systems exist for many years already [2]. But still it is not being used on large scale. Limited quality of the visual data, due to bandwidth limitations, restricts the full exploitation of audiovisual intelligibility of the speaker.
Suitable camera's, sensors and displays in both domestic and mobile devices, capable of conveying the important visual cues, are affordable and widely available now. Video conference systems, which vary from meeting room set-ups to mobile applications, are emerging [3]. Audiovisual speech detection is a valuable technique to improve the audiovisual quality of speakers in those video conference systems.
Part of the information in this document is based on the literature and preliminary research at the video processing group and a part is the author's own approach and research. At the beginning of each chapter a summary of the author's contribution will be given. This chapter will give an overview of the development on audiovisual speech analysis. It is followed by the assignment description. In chapter 2, general information about human speech production followed by the description and implementation of an available and standard method of acoustical speech detection. The implementation and explanation of our own method of distinct vowel grouping (DVG), based on general knowledge of speech is given in section 2.3 followed by the audio experiments in the last section of that chapter. Visual speech detection will be discussed in chapter 3. This chapter discusses general visual speech and the choice of the used face detector in the first two paragraphs. The third paragraph of chapter 3 gives an explanation for the chosen approach for visual speech detection. It also gives an elaborate explanation of the design and implementation of our luminance based mouth localisation algorithm and our luminance corrected colour-transformation-based algorithm for lip edge detection. Our proposal of visual speech detection with the extracted basic mouth features is discussed in paragraph 3.4. The last paragraph of chapter 3 shows the result of the experiments in the video domain. In chapter 4 the two modalities audio and video are fused in a framework that was especially written in a Matlab environment for this assignment by the author. The result of the audiovisual speech detector is given in the last paragraph of that chapter. This report ends with general conclusions and recommendations for future work in chapter 5.

1.1 Developments on audiovisual speech analysis

The importance of visibility of the speaker in verbal communication was first investigated by Sumby and Pollack (1954) [1]. Their results indicated visual cues to contribute substantially to speech comprehension at low acoustic signal to noise ratio. The illusory "McGurk" effect [4] clearly demonstrates the importance of optical information in the process of speech perception. Their research made clear that perception of speech is at least a bi-modal process. When the visual and acoustical modality contradict, a fusion of the two cues in the brain is the result. This effect is demonstrated by a sequence in which a visual 'ga' can be seen an acoustical 'ba' can be heard. The contradicting cues are then perceived as 'da'. Multi-modal speech perception has nowadays become a topic of considerable interest in the speech community. A key concept, central in visual speech, is the "viseme". The word viseme was introduced by C. Fisher [5], which is a compound word of "visual" and "phoneme". A viseme is often described as a class of phonemes, the visual constituents of which are equivalent, or identical, based on their perceptual confusability. The 'd' and 'n' for instance sound different but look the same and are therefore part of the same viseme class. Among researchers there is often discussion about different viseme classifications, but they seem to agree there is no viseme system that accurately describes the visual characteristics of all phonemes for all speech.
In the field of audio and visual speech perception, it has become clear that audio-only and video-only perception are extraordinary complementary. Especially visually distinctive consonants are acoustically easily confusable and visa versa [6]. For example "mi" and "ni" which are acoustically confusable are easy to distinguish visually. The exploitation of the complementary distinctiveness for audiovisual speech recognition has been extensively researched and has widely been reported to outperform audio-only speech perception or recognition (e.g. [7]). Audio-visual speech *detection* is a subject on which less research has been done. The article of Ross Cutler, [8], appears to be the first report on the exploitation of audio visual correlation of speech to detect a speaking person. It reports on the training of a time delayed neural network (TDNN) to detect a stationary person saying a pre-defined word. The work in this report is not focussed on detecting predefined speech segments, but rather to person-independently detect speech. It is focussed on finding elementary links between acoustical and visual features of presence of speech. We believe that the use of the elementary links can result in a more robust and person-independent speech detector.

Acoustical speech
Voice activity detectors can be classified as, either time-domain approaches or frequency-domain approaches. Implementation of time-domain algorithms is computationally simple but better quality of speech detection is usually obtained with the frequency-domain algorithms.
A standard frequency-domain approach to detect voiced speech is described by Peter Chu [9]. The presence of pitch is detected by calculating the periodicity in the magnitude-squared components of the fast fourier transform (FFT). The stationary noise is generally removed with noise subtraction as described in [10].
In the field of speech recognition much work has been done in describing vowels. Gordon Peterson and Harold Barney gave a detailed relation of American English vowels and their relation to the first three formant frequencies [11]. Their paper also describes the relation

between formants and tongue position. Formants will be discussed in section 2.1
A popular spectral analysis of audio is cepstrum analysis, because it has the desirable property of separating source (glottal) and filter characteristics (vocal tract) [12]. The cepstrum of a waveform is the Fourier transform of the log of the magnitude spectrum of that waveform. After cepstral transformation a convolution becomes an addition, and the (constant) filter characteristics can be separated from the varying signal by DC-filtering.
An improvement to any spectral estimation method results from using multiple resolutions, which comes down to use bigger or smaller time windows to calculate the spectrum. Each new analysis resolution means more computations, which triggered the usage of faster dedicated multi-resolution transforms like the wavelet transform.

Visual speech
The visual aspects of speech recognition can be categorised into geometric and non-geometric analysis. Geometric analysis focusses on the shape of the region of interest like mouth and lip features. The non-geometric analysis makes use of transformation to other domains to represent certain aspects of the region of interest. The discrete cosine transform (DCT) in combination with a neural network is generally reported to perform well in audio visual speech recognition.
Geometric mouth feature extraction started with contour detection by means of snakes [13], followed by deformable templates associated with an energy function that measures how well the model matches a particular object [14]. Both categories, however, still rely on an accurate region of interest location. To locate the region of interest, a face detection algorithm is generally applied. An overview of face detection algorithms is given in [15] and [16]. The trained cascade of Haar-like filters based on the AdaBoost algorithm described in reference [17] in combination with skin-colour pre-filtering, outperforms other techniques in terms of speed and accuracy.

Audiovisual speech
The problem that still remains is to find a good fusion of the acoustical and visual data. Two methods of mode fusion are often described; *feature*-level fusion and *decision*-level fusion (e.g. [18]). When the data is fused at feature level, feature parameters from different modalities are joined to form a vector. This fusion method can for instance be used when the data is accurate, but the features alone cannot make a proper distinction. Linear discriminant analysis (LDA) is a useful method to make the distinction in the multi-dimensional audio video domain as reported in [18]). Fusion on decision level is the fusion at a more advanced stage of the processing. Each modality first classifies an event independently. It can for instance be used for detection of false positives in one of the modalities. In the research described in this document, the decision level fusion will be used to visually detect false positives of the audio-only speech detector.

1.2 Assignment

The assignment is the *design and realisation of an audiovisual speech activity detector*. The context is a scenario in which one user is in the camera's field of view and more people are possibly talking outside this field of view. An audio-only detector would react on the people outside the camera's field of view, while the audiovisual detector should perform better in this scenario. The distance of the user to the camera is small, so that the user's face is recorded with a sufficient number of pixels to capture face and lip features. Only one microphone is available. An example of a scenario is a mobile phone with a camera facing the speaker.

1.3 Problem description

Audio-only speech activity detectors have been developed long ago and work fine as long as the signal to noise ratio is high. In the context of this report, the user can be situated in a noisy environment. If the noise is speech from people outside the field of view, or has speech characteristics, the audio-only solution is certain to give false positives. The combination with the modality video could be suitable to mitigate the problem of false positive detections. An audio visual solution is therefore investigated.

Applications

The information given by an audiovisual speech activity detector can be used for the following applications:

- Speech enhancement; For speech enhancement algorithms (e.g. noise suppression and acoustic echo cancelation) it is useful to know when the near-end speaker talks.
- Speech compression, For bandwidth saving, speech detection is useful to make use of optimal speech compression during speech segments.

For audiovisual speech activity detection, the location on the video image of mouth of the speaking person is also found. This additional information can be used for:

- Visual enhancement; Important visual speech areas can be encoded with less compression to maintain important visual cues for intelligibility, when sent through a bandwidth limited channel.
- Extra features; Speaking persons can for instance be marked on the screen.

The enhancement algorithms are however beyond the scope of this report. The context of the assignment can be extended to a scenario with multiple people in the camera's field of view.

The audiovisual speech detector can then be used for localisation of the speaking person. This would be an audiovisual alternative to the audio-only solution with multiple microphones as considered in [16].

Chapter 2

Acoustical speech detection

The human speech production system is a complex sound generator. In this chapter, general information about acoustic speech and acoustic speech detection will be given. Also the author's approach for acoustic speech detection will be explained.
Section 2.1 will briefly introduce the physical aspects of the human speech production system and the aspects of voiced speech and vowels. Section 2.2 describes a general audio-only approach for speech activity detection. In section 2.3, our distinct-vowel-grouping (DVG) approach, based on common knowledge of speech, is explained. Acoustic experiments will be described in the final section of this chapter.

2.1 Speech production

The production of speech is widely described as a two-level process. In the first stage the sound is initiated and in the second stage it is filtered [19].

1. The glottal source
Vocal fold vibration for voicing is achieved by passing air through the glottis, which is the gap between the vocal cords. The spectrum of the periodic glottal waveform is a line spectrum of harmonics at multiples of the fundamental (pitch) frequency. For unvoiced speech also air is passed through the glottis but it is highly uncorrelated which makes its spectrum noise-like. The vocal cords do not vibrate during the production of unvoiced speech.

2. The vocal tract filter
The vocal tract is a time-varying acoustic filter which modifies the excitation from the glottal depending on the position of the articulators. The articulators are the parts of the vocal tract that can be moved and determine its resonant, or formant frequencies. The relationship between these resonances is known as the formant structure, which provides means to distinguish sounds. The most important articulators are the tongue, the lips, the jaw and the soft-palate,

which is the movable fold that can close off the nasal cavity from the oral cavity.

Voiced speech
In case of voiced speech, the source sound produced by the glottal source has a repetitive character. The frequency F_{pitch} is the distance between the repetitive frequency peaks. For males the pitch or fundamental frequency is generally between 60 Hz and 200 Hz and typically around 120 Hz. For females and children the pitch frequency is generally in the range 120 Hz to 300 Hz. It is typically around 230 Hz for females and around 260 Hz for children. Figure 2.1 shows a waveform and spectrum of an example of voiced speech. The formant structure can be seen as the envelope of the spectrum. The peaks in this envelope are the formants.

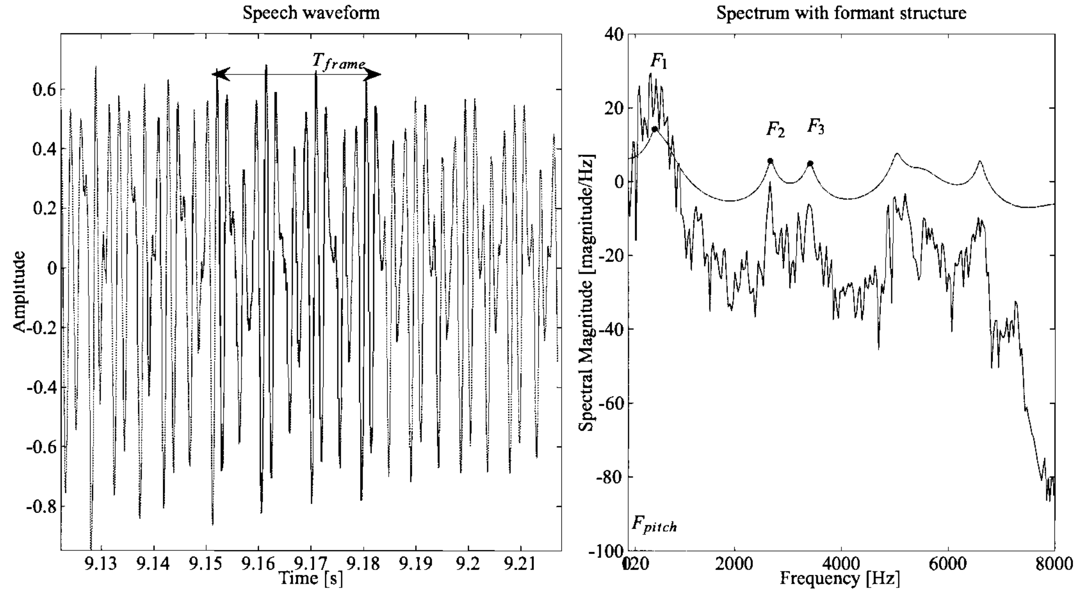


Figure 2.1: The waveform and spectrum of an example of voiced speech

Vowels
The strong relation between the first 3 formants (F1, F2, F3) and vowels is a common method to classify speech, which will be used in this report. Figure 2.2 shows the relation between vowels and the first 2 formants as was published by Peterson and Barney [11]. Points for each vowel lie in isolated areas. There are variations between the classes men, women and children which gives rise to overlap between indistinct vowels in the F1-F2 plane. Formant F3 is less important for recognition of vowels by humans.

2.2 Speech activity detection

Many voice activity detectors are based on the detection of presence of pitch. A C implementation of this approach based on [9] was available at the start of this project. The detector

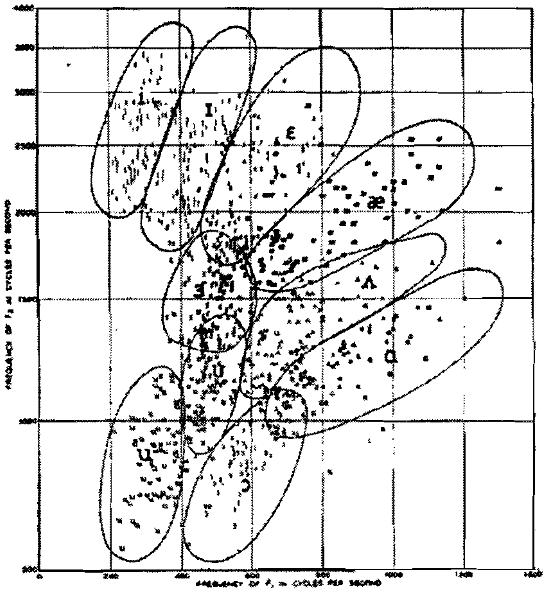


FIG. 8. Frequency of second formant *versus* frequency of first formant for ten vowels by 76 speakers.

Figure 2.2: Ten vowels by 76 speakers on F1-F2 plane according to Peterson and Barney [11]

output has three levels. The first level implies silence or acoustic activity below a certain threshold. The second level implies acoustical activity above that threshold. The third level implies acoustic activity and presence of pitch. The speech activity detector was programmed in a Matlab environment. The Matlab environment was chosen for processing of audio and video data simultaneously. A framework for the processing of audio and video was not yet available and therefore also programmed. The implementation that was made is described in this section.

Audio frames

Acoustical activity and pitch are detected in audio blocks not longer than 32ms. This length counts as a rule of thumb for the stationarity interval of speech. In this research, it is assumed that audio and video are in sync, meaning that speech and lip motion are synchronised. Unsynchronised audio and video is not a subject of this thesis. The difference between the available implementation of the speech activity detector and the implementation described in this report, was in the choice of the audio frames. The audio frames in the audio-only speech detector are consecutive blocks of audio samples. The choice for audio frames in the audiovisual framework is constrained by the frame rate of the video data.

Let $s(i_t)$ be the waveform sampled at sampling frequency F_s where $i_t \epsilon (-\infty, \infty)$ is the sample number. For each video frame with index n, two acoustic vectors $\overline{a}(n)$ and $\overline{b}(n)$ are defined. $\overline{a}(n)$ is a vector of length B with elements $a_i(n)$ and $\overline{b}(n)$ is a vector of length B with elements $b_i(n)$. The block size B of these acoustic vectors depends on the ratio of the frame rate F_v

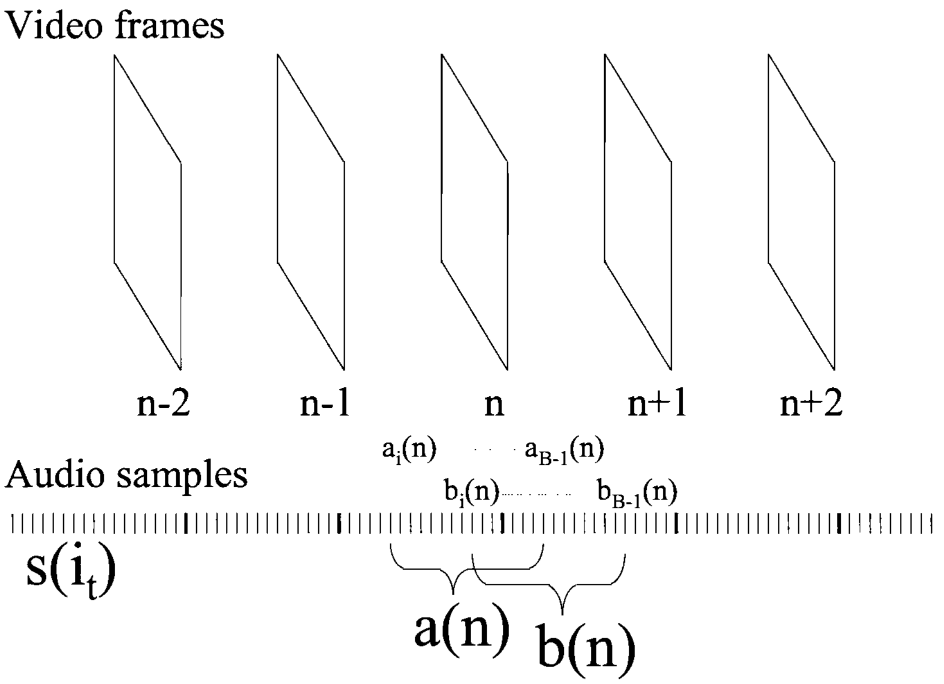


Figure 2.3: Acoustic vectors $a(n)$ and $b(n)$ consisting of samples $a_i(n)$ to $a_{B-1}(n)$ and $b_i(n)$ to $b_{B-1}(n)$.

and the sampling frequency F_s, according to: 2.1.

$$B = \mathtt{min}\Big(\mathtt{max}\big(128 \cdot 2^{\lfloor \log_2\left(\frac{F_s}{128F_v}\right)\rfloor}, 128\big), \lfloor 0.032F_s \rfloor\Big) \tag{2.1}$$

The upper bound to the block size is necessary to prevent the acoustic frames from becoming too long in time. This ensures that all audio samples in the blocks are close in time to the corresponding video frame. The elements $a_i(n)$ and $b_i(n)$ of these vectors are subsequent partially overlapping elements from waveform $s(i)$ according to:

$$a_i(n) = s\big(\lfloor n\frac{F_s}{F_v} - \frac{3}{4}B \rfloor + i\big) \quad \text{, for i = 0,...,B-1} \tag{2.2a}$$

$$b_i(n) = s\big(\lfloor n\frac{F_s}{F_v} - \frac{1}{4}B \rfloor + i\big) \quad \text{, for i = 0,...,B-1 .} \tag{2.2b}$$

An illustration of the acoustic vectors is shown in figure 2.3 The sampling frequency F_s is typically 16kHz and the frame rate F_v is typically 25 or 30 frames per second. The block size B will then be 512 samples.

Windowing

Each acoustic vector is smoothly windowed with a Hanning window to prevent artifacts from discontinuities introduced at the beginning and end of the frame. The elements w_i of Hanning window vector $\overline{w}$ are defined by:

$$w_i = 0.5 - 0.5\cos\left(\frac{2\pi i}{B-1}\right) \quad \text{, for i = 0,...,B-1} \tag{2.3}$$

$\overline{a}_w(n)$ and $\overline{b}_w(n)$ are the two windowed acoustic vectors of length B of video frame n with elements $a_{w,i}(n)$ and $b_{w,i}$. The elements $a_{w,i}(n)$ and $b_{w,i}$ are defined by:

$$a_{w,i}(n) = w_i a_i(n) \quad \text{, for i = 0,...,B-1} \tag{2.4a}$$

$$b_{w,i}(n) = w_i b_i(n) \quad \text{, for i = 0,...,B-1 ,} \tag{2.4b}$$

Pitch detection

The repetition frequency of the peaks in the spectrum of voiced speech is equal to the pitch frequency F_{pitch}. Autocorrelation of a sequence is the common method for detection and estimating periodicities in a signal. The autocorrelation, which is equivalent to the convolution of a sequence with a time-reversed version of itself, is equivalent to multiplying the Fourier transform of the sequence with the complex conjugate of the same Fourier transform and then taking the inverse Fourier transform [9]. Let $\mathtt{DFT_M}(\bullet)$ be the M-points discrete Fourier transformation. Let column vector $\overline{F}$ be the discrete Fourier transformation of column vector $\overline{f}$ defined by:

$$\overline{F} = \begin{pmatrix} F(0) \\ \vdots \\ F(M-1) \end{pmatrix} \qquad \overline{f} = \begin{pmatrix} f(0) \\ \vdots \\ f(B-1) \end{pmatrix} \tag{2.5}$$

The matrix multiplication of column vector $\overline{f}$ by $M \times B$-matrix $\mathbf{W}$ then yields column vector $\overline{F}$.

$$\overline{F} = \mathbf{W} \cdot \overline{f} \tag{2.6}$$

This multiplication is the matrix form of the $\mathtt{DFT_M}$ where $\mathbf{W}$ is the twiddle matrix ([20] p.149) with at its $(p+1)^{th}$ row and $(q+1)^{th}$ column the value W_M^{pq} as defined in equation 2.7

$$\mathbf{W} = \begin{pmatrix} W_m^{0\cdot 0} & W_m^{0\cdot 1} & \cdots & W_m^{0\cdot(B-1)} \\ W_m^{1\cdot 1} & W_m^{2\cdot 2} & \cdots & W_m^{2\cdot(B-1)} \\ W_m^{\vdots} & \vdots & \ddots & \vdots \\ W_m^{M-1\cdot 1} & W_m^{M-1\cdot 2} & \cdots & W_m^{M-1\cdot(B-1)} \end{pmatrix}, \tag{2.7}$$

where $W_M = e^{-j\frac{2\pi}{M}}$. Let $\overline{A}_s(n)$ be the vector of M complex frequency components $A_{s,0}(n)$ to $A_{s,M}(n)$ calculated from acoustic vector $\overline{a}_w(n)$ of length B as defined in equation 2.8. The following operations are shown only for block $\overline{a}_w(n)$ as the operations are the same for both acoustic vectors $\overline{a}_w(n)$ and $\overline{b}_w(n)$.

$$\overline{A}_s(n) = \mathbf{W} \cdot \overline{a}_w(n) \tag{2.8}$$

The self multiplication operation of the Fourier transform ($A_{s,k}(n) \cdot A^*_{s,k}(n)$, for $k = 0, ..., M$ where * denoted complex conjugation) is equivalent to squaring the magnitude of each frequency component. As the time signal is real, the spectrum is symmetrical. The squaring of each frequency component needs only be done for the first half of the spectrum from the DC element to the Nyquist element:

$$P_{s,k}(n) = real^2(A_{s,k}(n)) + imag^2(A_{s,k}(n)) \quad \text{, for } k = 0, 1, ..., M/2 \text{ .} \tag{2.9}$$

Where $P_{s,k}(n)$ are $((M/2)+1)$ magnitude-squared components of $\overline{P}_s(n)$. Before commencing with pitch detection, noise is removed from the signal and an acoustic event detection is performed to determine if pitch detection is useful.

Stationary noise estimation

Stationary noise can be recognised by spectral peaks that are present at the same locations for consecutive spectra in time. The magnitude of the stationary noise spectrum is defined as a vector $\overline{P}_{\tilde{n}}(n)$ with elements $P_{\tilde{n},k}(n)$ for $k = 0, 1, ..., M/2$. The stationary noise spectrum $\overline{P}_{\tilde{n}}(n)$ is estimated with a minimum statistics approach [10].

Stationary noise subtraction

The magnitude squared components of the noise vector $\overline{P}_{\tilde{n}}(n)$ are subtracted form the magnitude squared signal components $\overline{P}_s(n)$ as shown in equation 2.10. This yields $\mathbf{M}(k)$ in which the noise contribution in each frequency component is removed.

$$M_{s,k}(n) = P_{s,k}(n) - P_{\tilde{n},k}(n) \quad \text{, for } k = 0, 2, ..., M/2 \tag{2.10}$$

The elements $M_{s,k}(n)$ must be non-negative. If $M_{s,k}(n) < 0$ element $M_{s,k}(n)$ will be set to zero.

Acoustic event detection

The acoustic sound event is detected after subtracting the estimated noise power from the total signal power. The difference of the sum of the stationary noise spectrum over all frequencies and the sum of the signal spectrum across all frequencies is compared to a fixed acoustic event threshold value th_{ac} (equation 2.11).

$$\text{Acoustic event} = \begin{cases} 2, & \text{when } \sum_{k=0}^{M/2} M_{s,k}(n) > th_{ac} \\ 1, & \text{otherwise} \end{cases} \tag{2.11}$$

Only when threshold th_{ac} is exceeded, the detector goes to level 2 and pitch detection is started.

Spectral peak emphasis

To enhance pitch detection, the spectral peaks will be emphasised by feeding the magnitude squared components through a high pass filter with filter kernel h_{hp}.

$$Mf_{s,k}(n) = \begin{cases} M_{s,k}(n) - 0.5\big(M_{s,(k-2)}(n) + M_{s,(k+2)}(n)\big) & \text{, for k = 2,3,...M/2-2} \\ 0 & \text{, for k = 0, 1, M/2-1,M/2} \end{cases} \tag{2.12}$$

This pre-processing step before performing the autocorrelation has been found useful in pitch estimation [9]. The inverse discrete Fourier transformation (IDFT) will then produce an approximation of the autocorrelation of the audio signal. The peak at delay zero, which is equal to the sum of the element wise multiplication of audio signal times the same audio signal with a delay of zero, is in essence the energy of the processed frame.

Peak search
The peak of the autocorrelation is searched for lags appropriate to the range of human pitch. If the ratio of the peak value to the autocorrelation peak at lag 0 is sufficiently high, pitch is declared for that frame. The ratio threshold is typically 0.75.

Audio-only speech declaration
If pitch is detected in the last three frames and the standard deviation in pitch value of the last three frames is small enough, level 3 of the speech detector has been reached. This means that voiced speech is detected. A typical value for the allowed standard deviation of the pitch frequency is about 3Hz. A C implementation of this algorithm was available. For integration with video signals it was re-implemented in Matlab environment.

2.3 Speech Classification

To improve the basic speech detector discussed in the previous chapter, it was decided to perform a reduced vowel recognition. The vowel classification that was described in subsection 2.1 is exploited to recognise three well distinguishable groups of vowels. The groups are illustrated in figure 2.4, where formant frequencies $F1$ and $F2$ are the involved classification parameters. From subsection 3.4.2, which describes the visual features of speech, it will become clear why the author chose to make these three vowel groups. The method to use ellipses as described in 2.3.2 is the author's own idea to regroup the ten vowels that are shown in 2.2.
A vowel utterance is always voiced speech. The classification of speech will therefore only be initiated when pitch is detected.

2.3.1 Linear Prediction

To classify the frames of 32ms to one of the vowel groups in figure 2.4, the first 2 formants need to be extracted. Prior to extracting the formant structure. The -6dB roll off in the spectrum is compensated with a first-order high-pass filter as illustrated in figure 2.5. To retrieve the formant structure and extract the formant frequencies F_1, F_2 an estimation of the envelope spectrum of the speech signal is required. Autoregressive (AR) spectral estimation is a popular time series modeling approach because the AR parameters can be found by solving a set of linear equations. AR spectral estimation is also known as linear prediction spectral estimation ([21] p.153).
Linear prediction and AR modeling are two different methods of which the numerical results can be the same. Both methods have the goal to determine the parameters of a linear filter. The use of that filter $F(z)$ however, is different for both cases as shown in figure 2.6. The

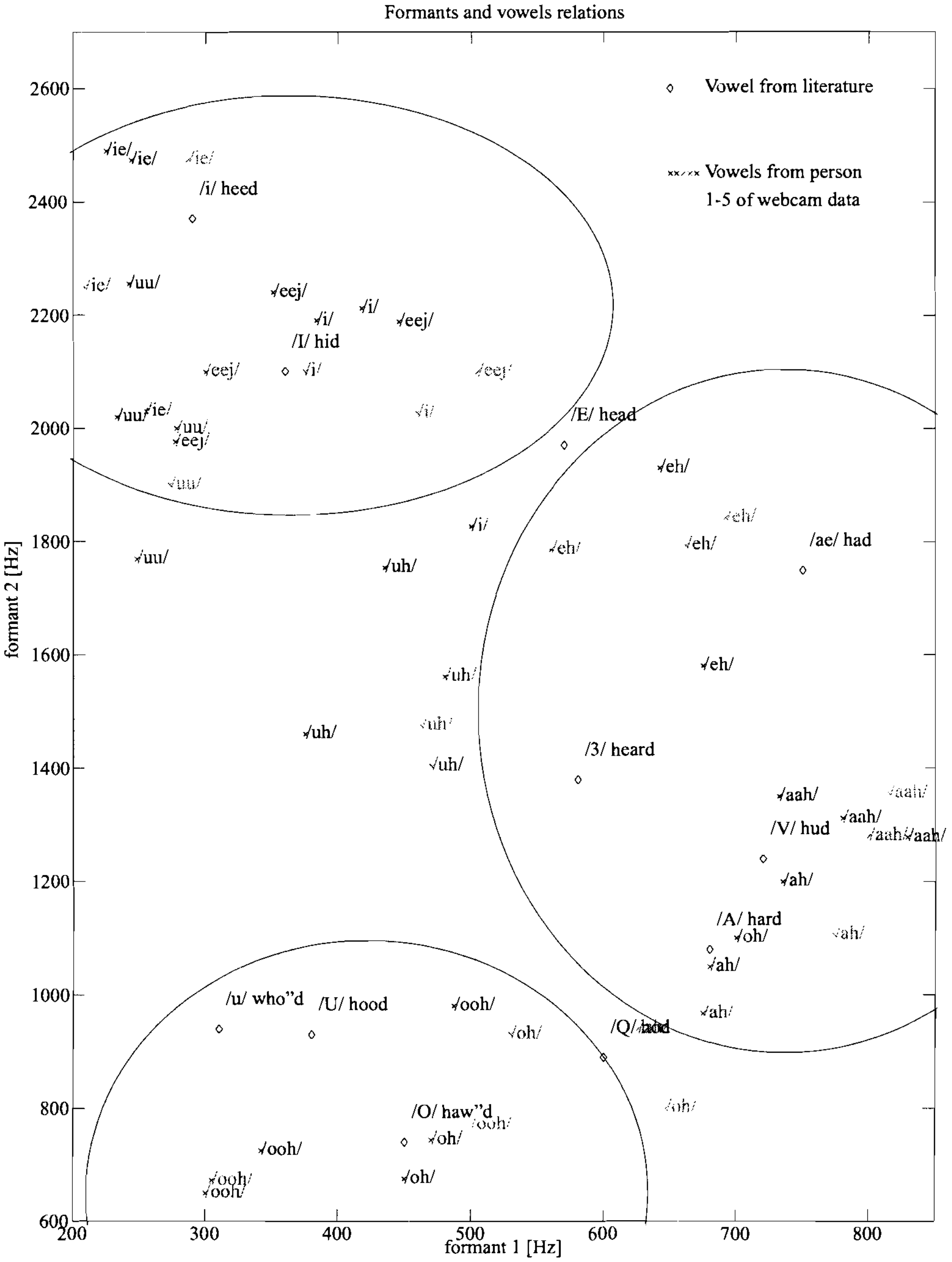


Figure 2.4: Vowels and vowel groups in formant1-formant2 plane. The vowels marked with black diamonds are written in International Phonetic Alphabet (IPA) notation and retrieved from literature. The vowels marked with x are retrieved from measurements. They are common Dutch vowels as in the following Dutch words: /ie/ as in 'hier', /i/ as in 'wit', /eej/ as in 'wees', /eh/ as in 'met', /uh/ as in 'muts', /ah/ as in 'bak', /aah/ as in 'baard', /oh/ as in 'bord', /ooh/ as in 'hoofd' and /uu/ as in 'muur'.

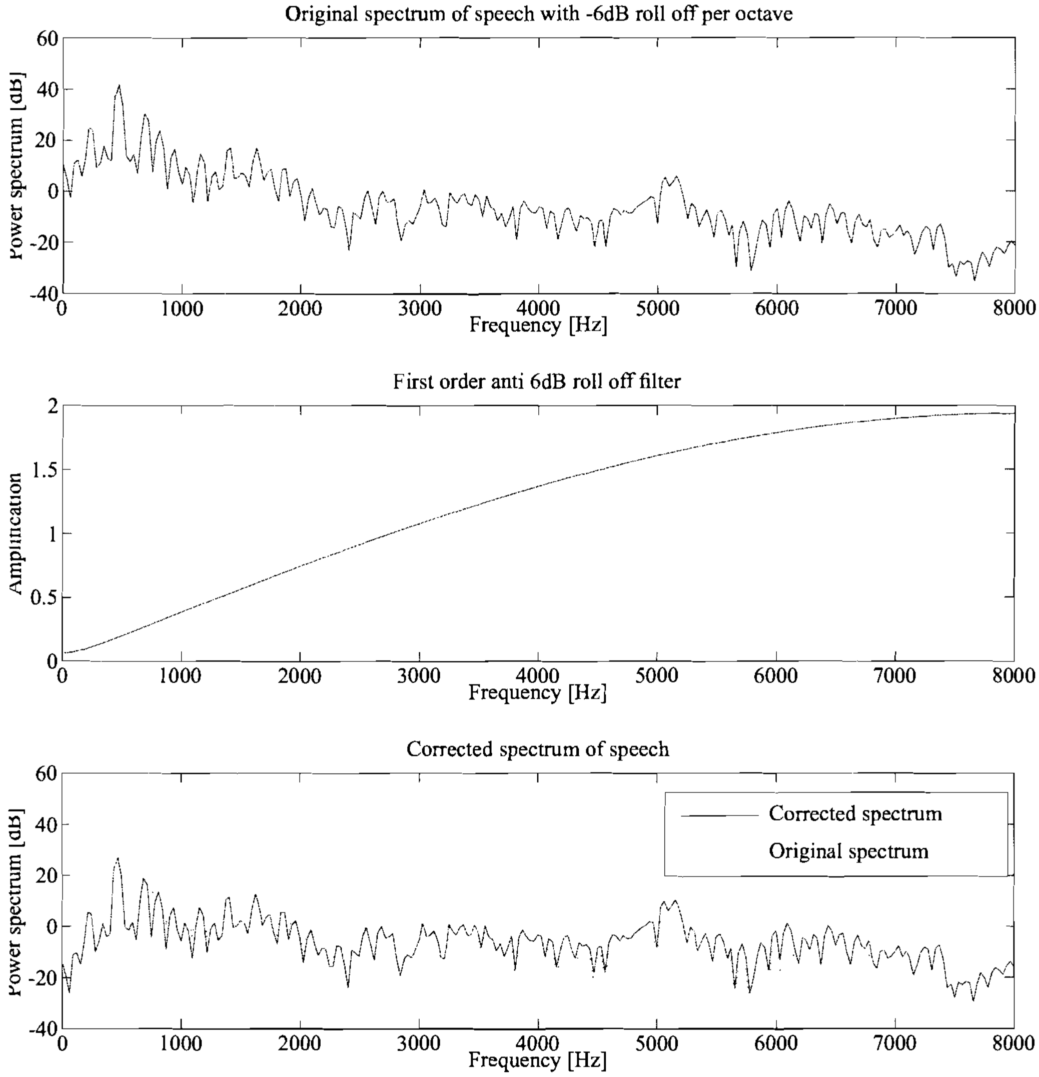


Figure 2.5: The effect of the anti 6dB roll off filter on the spectrum of a segment of speech.

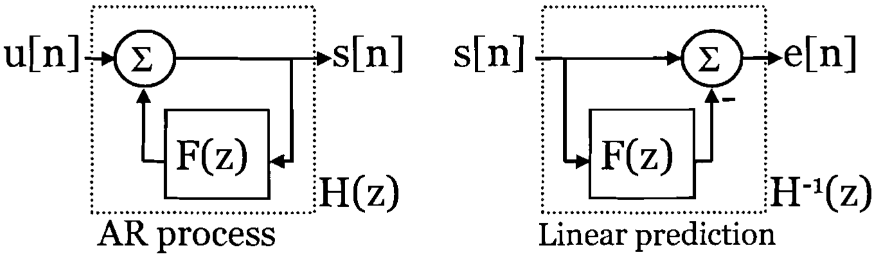


Figure 2.6: Linear filter $F(z)$ in a AR process and linear prediction

intention of linear prediction is to determine a FIR (finite impulse response) filter $F(z)$ that optimally predicts future samples with a linear combination of past samples. The difference between the actual signal and the predicted signal is called the prediction error $e[n]$ as shown in the linear prediction diagram in figure 2.6.
The intention of AR modeling is to determine an all-pole IIR (infinity impuls response) filter

$$H(z) = \frac{1}{1 - F(z)}, \tag{2.13}$$

that when excited with white noise produces a signal with the same statistics as the process that is modeled. The process in this case is the vocal tract filter that filters the exitation from the vocal cords to produce speech.
The linear prediction coefficients (LPC) are generally used to model the time varying vocal

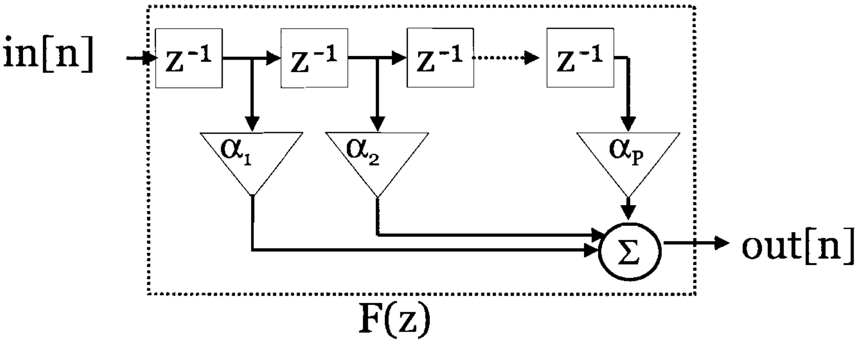


Figure 2.7: Linear prediction filter

tract filter. For every 32ms of speech the vocal tract filter remains approximately constant. The vocal tract filter $H(z)$ for each frame is given by equation 2.14 in which $\alpha_1, \alpha_2, ..., \alpha_P$ are the LPC parameters.[1,2]

$$H(z) = \frac{1}{1 - (\alpha_1 z^{-1} + \alpha_2 z^{-2} + ... + \alpha_P z^{-P})} \tag{2.14}$$

The vocal tract transfer function $H(z)$ can also be written as equation 2.15 in which p_1 to p_P are the roots of the polynome in the denominator of equation 2.14.

$$H(z) = \frac{z^P}{(z - p_1)(z - p_2)(z - p_3)...(z - p_P)} \tag{2.15}$$

The order of the linear prediction filter is determined by the number of parameters. The number of parameters has to be between 1 and B-1, where B is the number of samples of an acoustic vector. The order P of the approximation algorithm is typically 18 for the length and sampling frequency of the speech waveform. The parameters $\alpha_1, \alpha_2, ..\alpha_P$ can be determined

[1] Index (n) for frame number and index a and b for block label can be added to $H(z)$ and all coefficients a and p, but are left out here for readability reason

[2] Note that (n) between round brackets is a video frame number and [n] between straight brackets is a sample number of a digital filter.

by minimising a mean square error criterion defined by:

$$\sum_{n=0}^{B-1}(e^2[n]) \tag{2.16}$$

with

$$e^2[n] = \left(s[n] - \sum_{k=1}^{P}\alpha_k s[n-k]\right)^2 \tag{2.17}$$

Minimisation is achieved by taking derivatives with respect to parameters α_0 to α_P and set them to zero. This results in P linear equations:

$$\begin{gathered} \frac{\delta}{\delta\alpha_1}\sum_{n=0}^{B-1}(e^2[n]) = 0 \\ \frac{\delta}{\delta\alpha_2}\sum_{n=0}^{B-1}(e^2[n]) = 0 \\ \vdots \\ \frac{\delta}{\delta\alpha_P}\sum_{n=0}^{B-1}(e^2[n]) = 0 \end{gathered} \tag{2.18}$$

Which results in matrix equation

$$\mathbf{R}\overline{a} = \overline{r}, \tag{2.19}$$

where

$$\begin{gathered} \mathbf{R} = \begin{bmatrix} R(0) & R(1) & \cdots & R(P) \\ R(1) & R(0) & \cdots & R(P-1) \\ \vdots & \vdots & \ddots & \vdots \\ R(P) & R(P-1) & \cdots & R(0) \end{bmatrix} \\ \overline{\alpha} = [\alpha_1, \alpha_2, ..., \alpha_P] \\ \overline{r} = [R(1), R(2), ..., R(P)]. \end{gathered} \tag{2.20}$$

The elements of matrix **R** are autocorrelation terms with different lag of s[n]:

$$R(k) = \sum_{n=0}^{B-k} s[n]s[n+k] \tag{2.21}$$

Matrix equation 2.19 resulting from the LPC analysis can be solved by:

- Any matrix inversion method
- The Gaussian elimination method
- The Levinson-Durbin recursion

The Levinson-Durbin algorithm [22] is generally applied for its good numerical properties. Gaussian elimination and matrix inversion methods for instance do not exploit the fact that

Table 2.1: Formant frequencies for vowels for men (M) women (W) and children (Ch). (taken from [11] table II) The vowels are written in International Phonetic Alphabet (IPA) notation.

		i	I	ε	æ	a	ɔ	U	u	Λ	ɝ
Fundamental	M	136	135	130	127	124	129	137	141	130	133
frequencies(cps)	W	235	232	223	210	212	216	232	231	221	218
F_0	Ch	272	269	260	251	256	263	276	274	261	261
Formant											
frequencies(cps)	M	270	390	530	660	730	570	440	300	640	490
F_1	W	310	430	610	860	850	590	470	370	760	500
	Ch	370	530	690	1010	1030	680	560	430	850	560
	M	2290	1990	1840	1720	1090	840	1020	870	1190	1350
F_2	W	2790	2480	2330	2050	1220	920	1160	950	1400	1640
	Ch	3200	2730	2610	2320	1370	1060	1410	1170	1590	1820
	M	3010	2550	2480	2410	2440	2410	2240	2240	2390	1690
F_3	W	3310	3070	2990	2850	2810	2710	2680	2670	2780	1960
	Ch	3730	3600	3570	3320	3170	3180	3310	3260	3360	2160

the matrix **R** is 'Toeplitz' [21]. The Levinson-Durbin recursion itself is not described in this document.

The formant frequencies F_1 and F_2 are derived from 2 poles of the set of 18 poles. Only the poles in the first quadrant of the z-plane, close to the unity circle are candidates to extract formant frequencies from. $F_{c.i}$ with $i = 1, 2...p$ are the candidate formant frequencies calculated from these poles calculated with equation (2.22).

$$F_{c,i} = F_s \frac{arg(p_i)}{2\pi} \tag{2.22}$$

The number of candidate formant frequencies for the speech signals is typically 3 or 4. The formant frequencies for speech are bounded by constraints. The three smallest candidate formant frequencies need to fall into the three boundaries set for the first three formants. The bounds are based on averages of formant frequencies of vowels ([11] table II). The data from that table is also shown in table 2.1.

2.3.2 Vowel groups

The three well-distinguishable groups of vowels are illustrated in figure 2.4. The choice to group these vowels is based on the observations on the distinctiveness of the shapes of the mouths in the sequences with isolated vowels, which is described in more detail in subsection 3.4.2. The boundaries of each of the three well-distinguishable groups in the F1-F2 plane are chosen to outline the center area of the intersection of the chosen vowels from figure 2.2. This result in a conservative acoustical vowel detection. The chosen boundaries are

Table 2.2: Foci and major axis length of vowel detection ellipses

	focus a (F1,F2)	focus b (F1,F2)	major axis length c
I	(340,2550)	(390,1960)	715
A	(590,2180)	(790,800)	1430
O	(180,2000)	(600,400)	1700

an approximation of the actual intersection area. Because it was chosen to detect only the vowels in the center of the intersection areas, the boundaries can be defined by equations that describe ellipses. Each ellips (I, A and O) is defined by two foci in the F1-F2 plane and the length of the major axis. The ellips in which I-like sounds are located has foci $\overline{f}_{Ia}$ and $\overline{f}_{Ib}$ and major axis length c_I. The chosen ellips for A-like sounds has foci $\overline{f}_{Aa}$ and $\overline{f}_{Ab}$ and major axis length c_A. Foci $\overline{f}_{Oa}$ and $\overline{f}_{Ob}$ and major axis length c_I define the ellips for O-like sounds.

$$\text{acoustic vowel} = \begin{cases} I, \text{if} & \mathtt{dist}(\overline{f}_a(n), f_{Ia}) + \mathtt{dist}(\overline{f}_a(n), f_{Ib}) < c_I \\ A, \text{if} & \mathtt{dist}(\overline{f}_a(n), f_{Aa}) + \mathtt{dist}(\overline{f}_a(n), f_{Ab}) < c_A \\ O, \text{if} & \mathtt{dist}(\overline{f}_a(n), f_{Oa}) + \mathtt{dist}(\overline{f}_a(n), f_{Ob}) < c_O \\ 0, \text{otherwise} & \end{cases} \tag{2.23}$$

where $\overline{f}_a(n)$ is the vector (F_1, F_2) in the F1-F2 plane and $\mathtt{dist}(\bullet, \bullet)$ is the operation to calculate the euclidian distance between the two points in the F1-F2 plane. The chosen numerical values of the foci and major axis are given in table 2.2.

2.4 Experiments

The experiments described in this section are done on recordings with RME audio acquisition equipment for the PC [23]. The tests were done with sequences of speakers saying isolated vowels and speaking naturally.

2.4.1 Pitch based speech detection

Introduction

The goal of this experiment is to demonstrate the performance degradation of our pitch based speech detection algorithm in a noisy environment. [3]

Method

Speech segments in the sequences of speakers saying isolated vowels are marked with unity by the pitch based speech detector and other segments with zero. Manual inspection of the detection is done by multiplying the detection with the audio waveform and listening to the result. Also one minus the detection is multiplied with the audio waveform. This reveals any false positives when fragments of speech are still audible in the multiplied signal.

The data retrieved from the sequence without addition of noise is regarded as the truth. The speech detection algorithm is now done on the same sequences with addition of a continuously

[3] The audiovisual speech detector performance will be tested in a similar way.

interfering speech signal. This is done several times with increasing intensity of the interfering continuous speech signal. The interfering speech signal is first normalised to the acoustic signal of the test sequence. The RMS value of the interfering signal is equalised to the RMS value of the audio waveform of the test sequence. The interfering signal is then added to the test signal with increasing multiplication factors in each test.

Analysis

Figure 2.8 shows an example sequence of the true detection and a few plots of the detection for increasing power of interfering speech. A negative value means a missed detection. The integral of the false positives and the absolute integral of the missed detections are shown for different factors of interfering speech in figure 2.9.

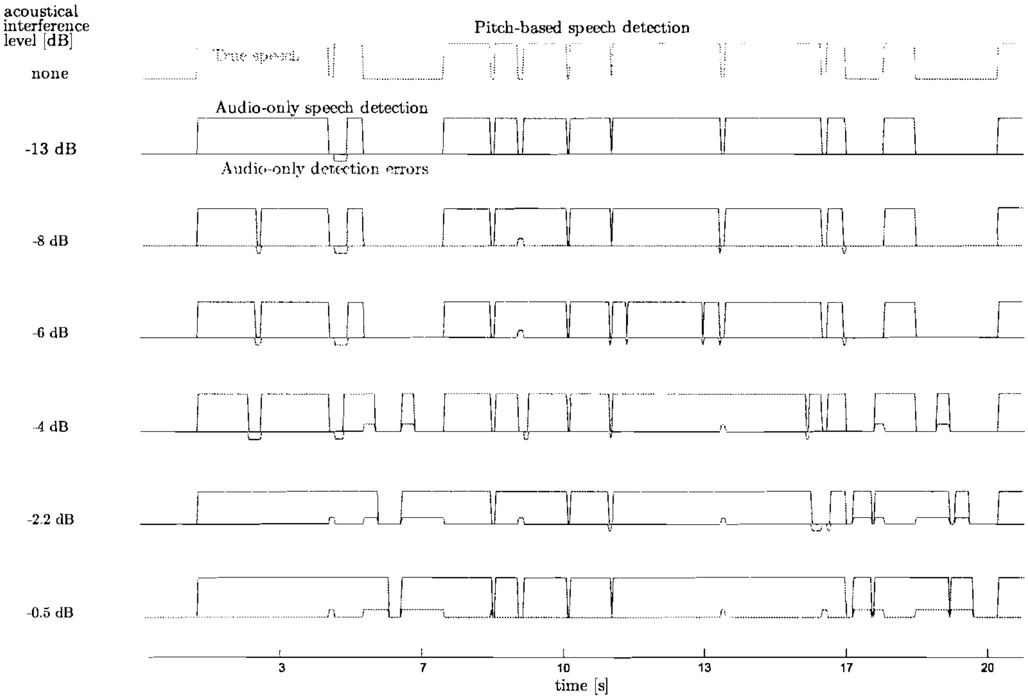


Figure 2.8: Example sequences of audio-only speech detection for increasing acoustical interference

Conclusion

As expected, the performance of the audio-only pitch-based speech detector degrades when interfering speech is increases in level. The detection errors mostly consist of false positives.

2.4.2 Vowel detection

Introduction

The vowel detection with the implementation described in this document is done for audio wave segments of approximately 32 milliseconds. A perception test by humans on all segment should be done on all these segments to classify how the vowel detector should identify each segment. There was no time available for this perception test on the test data. Published

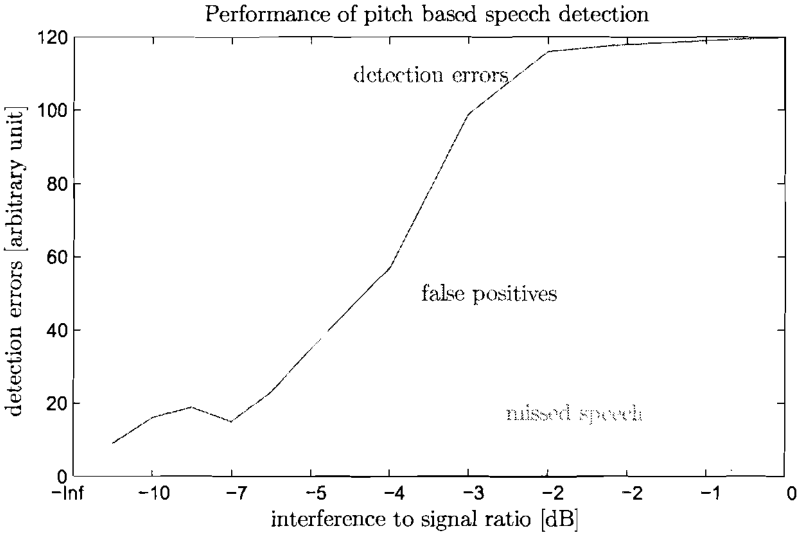


Figure 2.9: Performance of audio-only speech detection for increasing power of acoustical interference

Table 2.3: This table gives the detection rate of the audio-only vowel detector. This is the result of the training set consisting of 50 isolated dutch vowels

	acoustically recognised as [%]			
	I	A	O	none
eej, i, iee (15)	**93**	7	0	0
eh, ah, aah (15)	0.00	**100**	0	0
oh, ooh, uh, uu (20)	10	0	**85**	5
false recognition occurrences				
none	**2**	0	0	

perception tests (e.g. [11]) show that recognition of vowels by humans is not unambiguous.

Method

The author therefore marked audio segments, which in his perception should definitely be detected by the vowel detector. The noise can be, either an interfering speaker, or environmental noise. It is expected that voice interference has the biggest influence on the vowel detection performance and is therefore chosen as the interference source. The same method of addition of interfering speech, as described in subsection 2.4.1, is used.

Analysis

A positive detection is counted when at least one correct marking by the acoustic vowel detector during the presence of a vowel was given. The detection rate was determined for sequences with well-articulated isolated vowels. Table 2.4.2 shows the detection rate of the acoustical vowel detector on test sequences of different people saying isolated vowels. Figure 2.10 shows in red the increasing number of errors in vowel detection when an interfering voice signal is added to the test sequence of one person saying isolated vowels. The integral of these errors is shown in figure 2.11 for increasing interference level.

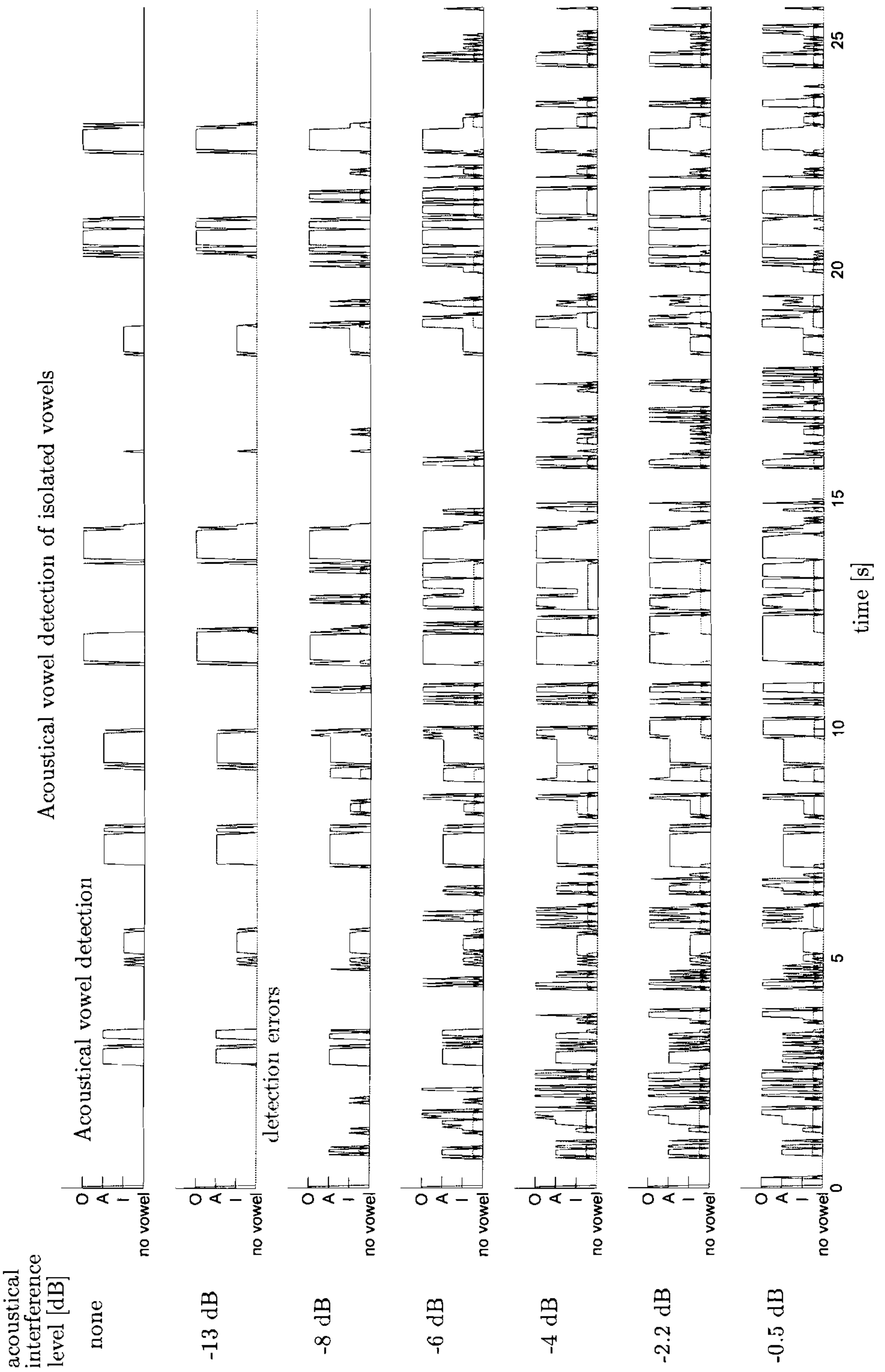


Figure 2.10: Test sequences of audio-only vowel detection for increasing acoustical interference.

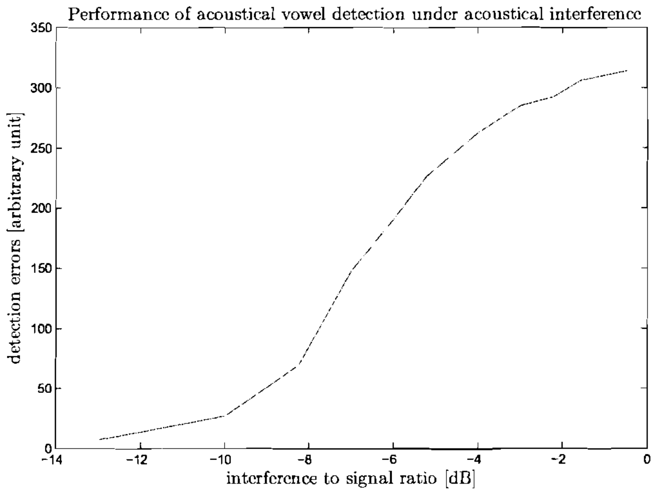


Figure 2.11: Performance of the acoustic vowel detector for increasing interfering voice signal.

Conclusion

The vowel recognition rate for isolated vowels is high. When noise is added to the acoustical signals, the performance drops. A complete analysis of interference in the audio domain deserves great attention. A complete analysis is beyond the scope of this research. It therefore suffices to show the performance drop of the vowel detector when acoustic interference is added in this experiment.

Chapter 3

Visual speech detection

In the previous chapter, acoustic speech is described. In this chapter the approach to detect visual speech features will be explained. Also the relation between visemes and phonemes and an explanation of basic vowel classification from video data is given. Section 3.1 will briefly introduce visual features of speech and the approach of the research that this document reports about.
Section 3.2 gives an explanation of the used face detector followed by a detailed explanation of the implementation of our colour transformation based method to extract visual speech features in section 3.3. The last section of this chapter describes the experiments designed to create and test the face feature extraction algorithm.

3.1 General visual speech

Visual speech recognition has been widely reported on, but there is no visual-only speech recognition system. The reason for this is that only a small portion of the speech production system is visible, as explained in chapter 2. This is why some phonemes are grouped together to one viseme class. The grouping to viseme classes is done by the visual confusability of the phonemes that form a viseme class. For example the spoken phonemes /p /b and /m are indistinguishable without sound and therefore make up one viseme class. Throughout history visual speech features have been clustered in to various distinct viseme groups. In 1988 P.L. Jackson claimed that "there is no one viseme system that accurately describes the visual characteristics of all phonemes for all talkers" [24].
The approach in this research is to detect a few well-distinguishable viseme classes with basic mouth features. The choice of the classes is based on distinguishability in the acoustic as well as in the visual domain. The result will then be used for an audiovisual speech activity detector.

3.2 Face finding

In order to extract mouth features, first faces are located in the video data. Face detection is a subject which is widely reported on. From preliminary research [16, 25] a "Viola and Jones" face detector [17] was available, which outperformed other face detection algorithms on speed and accuracy.

The Viola and Jones face detection algorithm returns squares in which a face is detected. The detection is done for multiple scales. Multiple squares around the same face location are therefore often returned. Viola and Jones face finding and tracking is not a subject of this report. More details can be found in [17, 16, 25]. The intersection of these squares yields one rectangle for each face with the author's own implementation of a straight forward grouping algorithm explained next. First, the detection squares are ordered from top left to bottom right. All squares that have their top left corner within the distance of half the first square's width from the top left corner of the first square (in horizontal and vertical direction) are grouped to one face object. The same operation is performed for the remaining squares until all squares are added to a face object. For each face object the intersection of all squares forms the detection rectangle. The minimum scales for which the Viola and Jones detector returns squares is limited to one quarter of the frame width. An illustrative example is given in figure 3.1. The blue squares are returned by the Viola and Jones face detection algorithm.

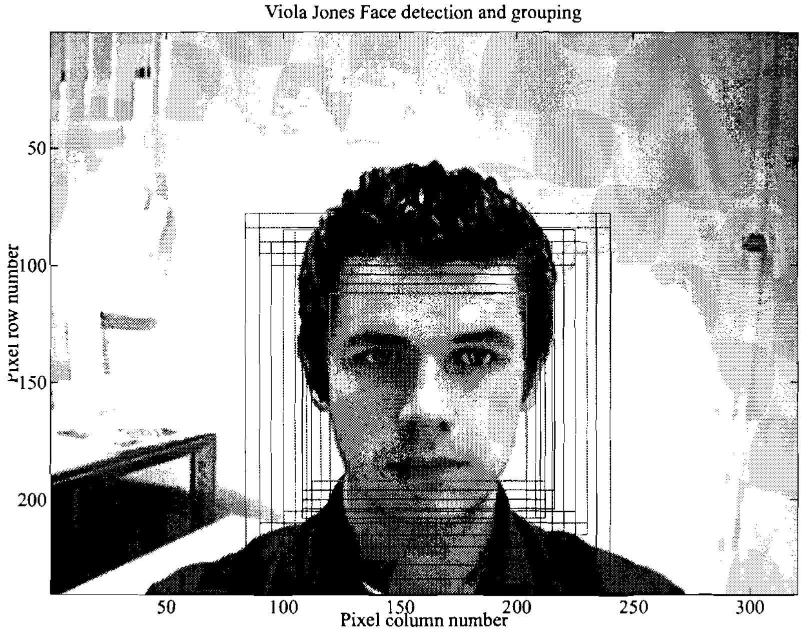


Figure 3.1: Grouping of Viola Jones face detection squares

The green rectangle is the result of the grouping of these squares. The grouping algorithm is suitable to return the locations of multiple non-overlapping persons in the image.

3.3 Visual speech feature extraction

In the research area of audiovisual speech recognition there are reports claiming a better performance of methods based on the discrete cosine transform (DCT) over geometrical approaches. From the DCT coefficients however, it is not possible to easily detect a closed mouth. Furthermore, a DCT is not shift invariant which makes its performance depend on exact mouth tracking [26]. In the context of this report it is assumed that the video data has a high resolution, which makes accurate lip extraction easier than for low resolution video [27]. For these reasons, we chose to base our lip feature extraction algorithm that is described in this document on geometriccal features and not on the DCT coefficients.
Relevant visual features for speech recognition are vertical mouth opening, horizontal mouth opening and the first derivatives in time of horizontal and vertical opening and closing of the mouth [7, 28, 29]. Also, roundness of mouth opening and angle of the lip corners are useful features for recognition of certain visemes.
Literature widely reports about the use of deformable templates (e.g. [30]) and snakes [13] to match mouth shapes. A snake is a combination of points that exert force on each other but are also associated with an energy function, which measures how well the model matches for instance the curves of an object. The object can be a mouth. A deformable template can be a polynomial model (with constraints that make it resemble a mouth shape), which is also associated with an energy function that measures how well the model matches the mouth curves. Both methods are iterative processes. Silveira [28] reports on applying gray scale thresholding and an erosion operation to extract the lip contour. It can be noted that most methods that were found in the literature were based on luminance information only. Using colour information for feature extraction has been proposed, but colour information is more sensitive to lighting conditions and camera properties.
The lip finding and mouth feature extraction that is discussed in the rest of this chapter was initially inspired by the colour transformations for face detection described in [31]. That publication describes transformations in the U and V color space to emphasise eyes and mouth. These emphasised areas were then used to locate faces in images. The algorithm presented in the rest of this chapter is our luminance-based mouth localisation algorithm and our colour-transformation-based method for locating the lip edges and extracting the mouth shape and mouth features.

3.3.1 Lip finding

To locate the lips, they have to be distinguished from the darker part inside the mouth, the teeth and the skin surrounding the lips. For all experiments in this report, I have used data that I retrieved from the "AR faces database" [32] and sequences recorded with "Philips ToUcam II" at the resolutions 360x240 and 720x480. The frame rate of the speech sequences was chosen 30 frames per second (fps). A brief experiment was done to determine the minimum frame rate to record visual speech. This experiment is described in paragraph 3.5.1. From the images of the recordings and the AR database the author deduced that lips should be well-distinguishable from skin using colour information. The lips are however less distinguishable from the inner mouth and teeth based on colour alone.

In the coming paragraphs the lip finding algorithm will be explained that is based on both colour and luminance. In short the algorithm comprises the following ten steps:

Step 1: From the face detection algorithm output, the mouth region is selected based on face geometry. Statistical information of the mouth region of interest (MROI) is used to adaptively determine a luminance threshold [33]. This threshold is used to convert the MROI into a black and white image.
Step 2: Morphological operations are performed on the black and white image.
Step 3: The pixels are grouped to blobs with a 'grassfire' algorithm, which was described in a Philips restricted document [34].
Step 4: The most likely mouth blob is selected by classifying the blobs with weak classifiers based on blob dimensions.
Step 5: The mouth corners are extracted from the mouth blob.
Step 6: On a few vertical lines across the mouth, the edges of the lips are searched.
Step 7: The RGB values of each vertical line across the mouth are transformed to YUV values. The U and V values of each pixel are then converted into a scalar value for each pixel with which lips have a high distinction from the surrounding skin, teeth and inner mouth pixels.
Step 8: A luminance correction is done to further distinguish teeth and dark area inside the mouth from the lips.
Step 9: The two high reddish areas around the middle of each vertical mouth line are marked as lip.
Step 10: Finally the parabolic mouth model is fitted through the marked lip areas on the vertical lines with a weighting function.

Lip finding algorithm

Step 1: *Luminance threshold*
The face location is returned by the grouping algorithm. The mouth region of interest (MROI) rectangle is directly computed from the face rectangle based on the knowledge where the mouth is normally positioned. The red rectangle marks the contour of the MROI in figure 3.2. Figure 3.3 shows a luminance image and the luminance histogram of a typical MROI.

Let $F_{MROI}(x, y, n)$ be the luminance value of a pixel in the MROI at spatial coordinates (x, y) of frame index n. The mean $\mu_{MROI}(n)$ and standard deviation $\sigma_{MROI}(n)$ of the pixel luminance values in the MROI are determined by:

$$\mu_{MROI}(n) = \frac{1}{N}\sum_{x}\sum_{y} F_{MROI}(x, y, n) \tag{3.1}$$

$$\sigma_{MROI}(n) = \sqrt{\frac{1}{N-1}\sum_{x}\sum_{y}(F_{MROI}(x, y, n) - \mu_{MROI}(n))^2}, \tag{3.2}$$

From $\mu_{MROI}(n)$ and $\sigma_{MROI}(n)$ a luminance threshold $\theta_{MROI}(n)$ is calculated for the distinction between dark inner mouth pixels and other pixels according to:

$$\theta_{MROI}(n) = \alpha\mu_{MROI}(n) - \beta\sigma_{MROI}(n). \tag{3.3}$$

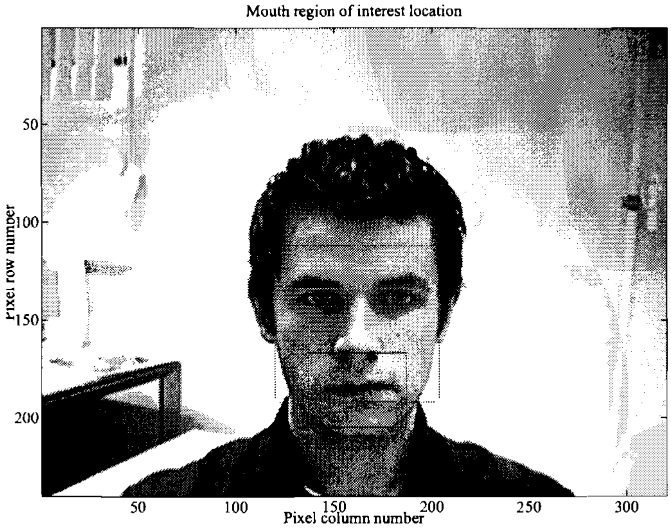


Figure 3.2: Mouth region of interest location

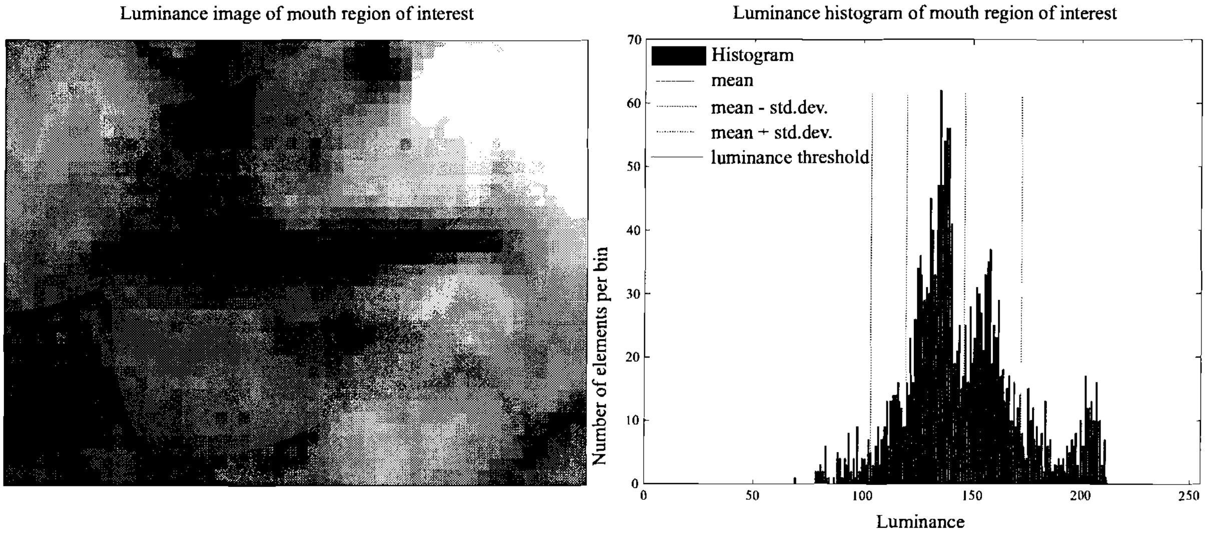


Figure 3.3: Luminance image of mouth region of interest and luminance histogram

The parameters α and β are determined with the least squares algorithm (LS) on a set of MROI's for which optimal thresholds θ_{MROI} were chosen manually. The LS algorithm minimises error function[1]:

$$\varepsilon(n) = \sum_{n=0}^{239 \text{ or } 247} \left\{\alpha\mu_{MROI}(n) - \beta\sigma_{MROI}(n) - \theta_{MROI}(n)\right\}^2 \tag{3.4}$$

to find:

$$\begin{pmatrix}\alpha_{opt}\\ \beta_{opt}\end{pmatrix} = arg \min_{\binom{\alpha}{\beta}} \varepsilon(n) \tag{3.5}$$

$$\tag{3.6}$$

The data set consisted of 240 "Philips ToUcam II" webcam images of 6 individuals and 248 images of 62 individuals from the AR faces database [32]. Any combination of α and β in the 'valleys' shown in de error functions in figure 3.4 results in an acceptable luminance threshold. The choice for the value of coefficients α_{opt} and β_{opt} however was not made arbitrarily.

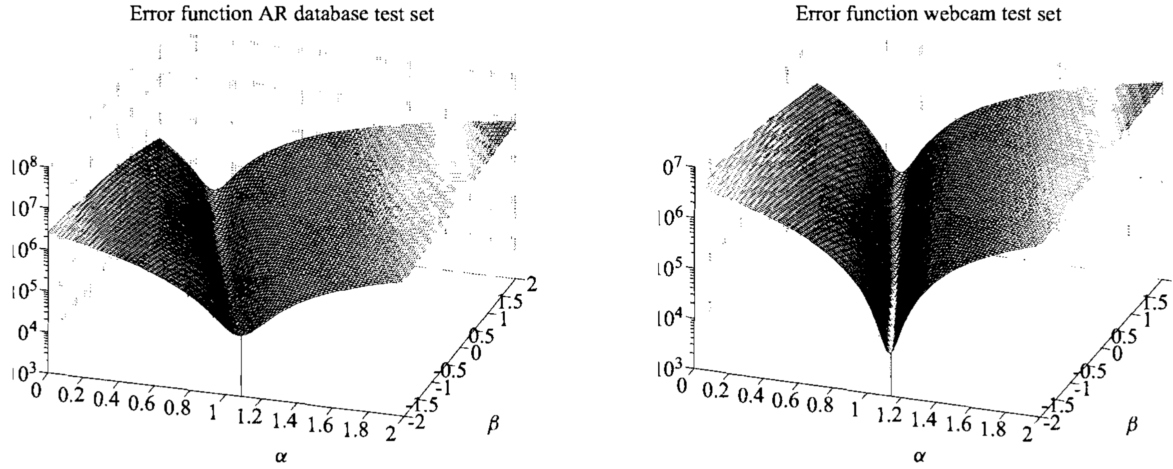


Figure 3.4: Error functions for determination of α and β for AR database set and the webcam data set

High coefficients make the determination of the luminance threshold more sensitive to noise. Especially the calculation of the standard deviation is more sensitive to noise pixels in the MROI. A lower value for β makes the determination of the luminance threshold more robust. The coefficients are chosen $\alpha_{opt} = 0.8$ and $\beta_{opt} = 0.50$ as this choice works for both the webcam data set and the AR faces database. The fact that acceptable thresholds are found with $\alpha \neq 1$ (while $\beta \neq 0$) means that the threshold value is not scalable[2] with the luminance of the MROI. This can be explained by the fact that the luminance value of dark pixels in the mouth do not linearly change when lighting conditions change. The dark pixels in essence tend to remain at low luminance value. The threshold is used to create a binary image $B(x, y, n)$ according to equation 3.7.

$$B\,(x, y, n) = \begin{cases} 0 & \text{if } F_{MROI}(x, y, n) \geq \theta_{MROI}(n) \\ 1 & \text{otherwise,} \end{cases} \tag{3.7}$$

[1] The LS algorithms was done separately on two different data sets

[2] With scalable it is meant that if all luminance values in the MROI would be multiplied by a constant, that the threshold found for that image would divide the pixels of the image identically

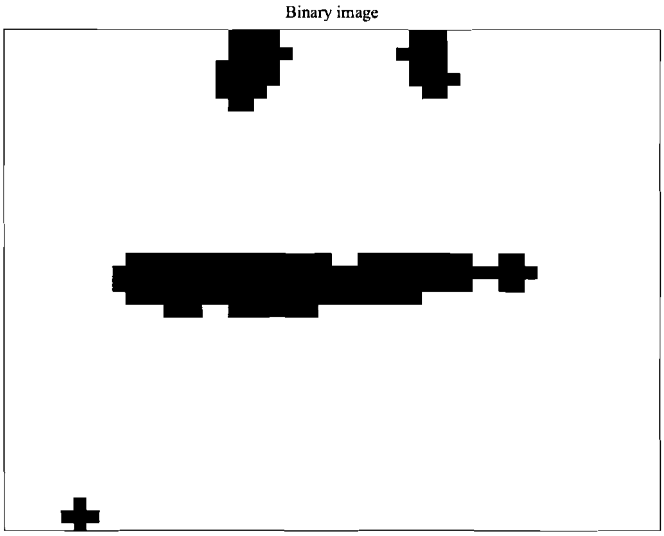


Figure 3.5: The binary image of the mouth region of interest

The dark inner mouth pixels are clearly pronounced in the binary image $B(x, y, n)$ as shown in figure 3.5.

Step 2: *Morphological operations*
Morphological operations are performed on $B(x, y, n)$ to remove noise pixels and to merge possibly separated blobs. Let $E(\bullet; \mathrm{S})$ be the operation of erosion with structuring element S and let $D(\bullet; \mathrm{S})$ be the operation of dilation with structuring element S. Structuring element S was chosen to be a horizontally oriented array because the mouth blob in the binary image was occasionally horizontally split in halves as a result of bright teeth:

$$\mathrm{S} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}.$$

Consecutive morphological operations gives binary image $B'(x, y, n)$:

$$B'(x, y, n) = E\big(D(B(x, y, n), \mathrm{S}), \mathrm{S}\big). \tag{3.8}$$

The next step is to group connected pixels with value 1 to blobs with a so-called grassfire algorithm.

Step 3: *Grassfire algorithm*[3]
The algorithm starts by locating a pixel with value 1 in image B' and copying it to a temporary binary image B'_i, where i is the blob index. The temporary binary image B'_i is then dilated

[3] The description of this algorithm was retrieved from a Philips restricted document.

with structuring element $\mathbf{S_g}$ and compared (logical and) with the original image B'.

$$\mathbf{S_g} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}.$$

The comparison result is put in the temporary image B_i'. If the number of pixels in the temporary image B_i' has increased compared to the previous execution loop, the process restarts with a dilation of the temporary image. If the number of pixels stayed the same, the blob of connected set pixels is completely detected. This blob is then cleared from the original image (logical xor). The process continues as long as set pixels are present in the original image. The next step is to find the most likely mouth blob B_j'.

Step 4: *Classification of blobs*
The most likely mouth blob is selected by several weak classifications of the dimensions of blobs B_i'. Define $\chi_i(n)$ as the set of all x-coordinates at which $B_i'(x, y, n)$ has binary value 1 and define $\psi_i(n)$ as the set of al y-coordinates at which $B_i'(x, y, n)$ has binary value 1. The horizontal and vertical bounds of the blobs are then[4]:

$$x_{l,i}(n) = \mathtt{min}\big(\chi_i(n)\big) \tag{3.9a}$$

$$x_{r,i}(n) = \mathtt{max}\big(\chi_i(n)\big) \tag{3.9b}$$

$$y_{t,i}(n) = \mathtt{min}\big(\psi_i(n)\big) \tag{3.9c}$$

$$y_{b,i}(n) = \mathtt{max}\big(\psi_i(n)\big) \tag{3.9d}$$

The dimensions of the blobs that are used for weak classification are:

$$\text{width of blob } i\text{:} \quad W_i(n) = x_{r,i}(n) - x_{l,i}(n) \tag{3.10a}$$

$$\text{height of blob } i\text{:} \quad H_i(n) = y_{b,i}(n) - y_{t,i}(n) \tag{3.10b}$$

$$\text{ratio of } W_i(n) \text{ and } H_i(n) \text{ of blob } i\text{:} \quad r_{WH,i}(n) = \frac{W_i(n)}{H_i(n)} \tag{3.10c}$$

$$\text{number of pixels in blob } i\text{:} \quad N_i(n) = \text{number of elements in}(\chi_i(n)) \tag{3.10d}$$

$$\text{horizontal center of blob } i\text{:} \quad x_{c,i}(n) = \frac{1}{N_i(n)} \sum \chi_i(n) \tag{3.10e}$$

$$\text{vertical center of blob } i\text{:} \quad y_{c,i}(n) = \frac{1}{N_i(n)} \sum \psi_i(n) \tag{3.10f}$$

The four weak classifier scores shown in figure 3.7 are manually tuned and based on visual inspection of the histograms of the dimensions of blobs shown in figure 3.6.

By the multiplication of the individual classification values we achieve a single classification value $C_{B_i'}$ for all blobs B_i' in the MROI.

$$C_{B_i'} = C_{1i}C_{2i}C_{3i}C_{4i} \tag{3.11}$$

[4] The top left pixel of an image has coordinates $(x,y)=(0,0)$.

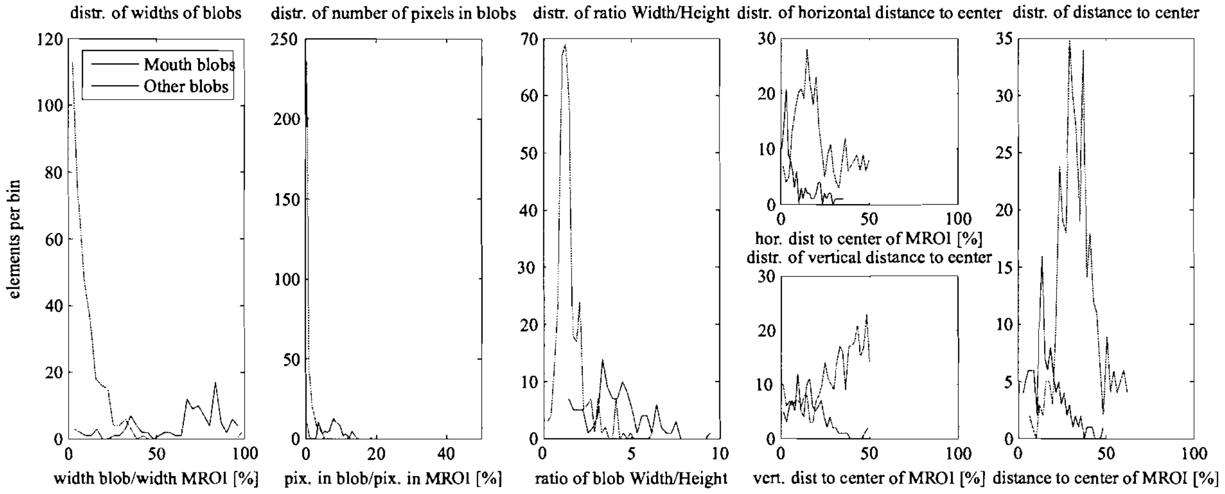


Figure 3.6: Histograms of dimensions of blobs in the binary image

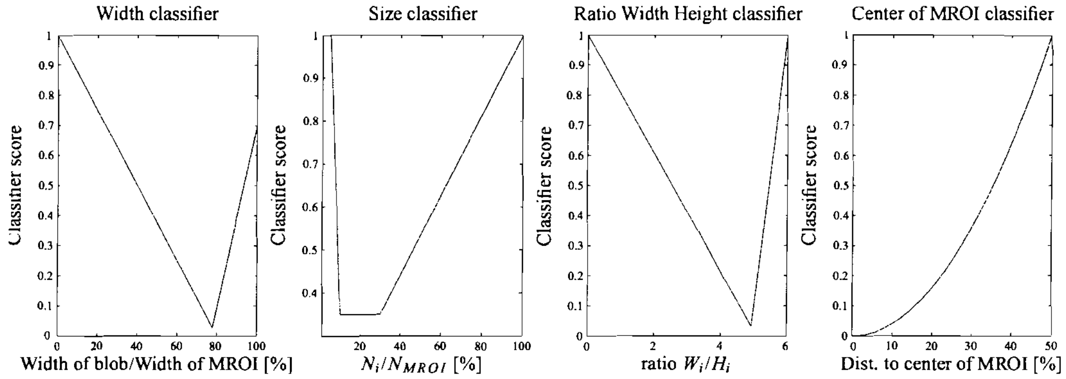


Figure 3.7: Score functions of four weak classifiers to select the most likely mouth blob

From all found blobs, the blob that has the lowest $C_{B'_i}$ score is selected as the mouth blob. The mouth blob has index j.

Step 5: *Find mouth corners*
As the orientation of the mouth is assumed to be horizontal[5] the corners of the mouth are located at the most left and most right pixel with value 1 in binary image $B'_j(x, y, n)$. The x-coordinate of the left-hand corner is therefore $x_{l,j}(n)$ and the right-hand corner is $x_{r,j}(n)$. The y-coordinate for the left $y_{l,j}(n)$ and right hand $y_{r,j}(n)$ corner is the median of the y values of the pixels at the most left and most right edge of the mouth blob. Define $\varphi_{l,j}(n)$ as the set of y-coordinates at which $B'_j(x_{l,j}(n), y, n)$ is set and define $\varphi_{r,j}(n)$ as the set of y-coordinates at which $B'_j(x_{r,j}(n), y, n)$ is set. The left and right y-coordinates of the mouth corners are then:

$$y_{l,j}(n) = \mathtt{med}\big(\varphi_{l,j}(n)\big) \tag{3.12a}$$

$$y_{r,j}(n) = \mathtt{med}\big(\varphi_{r,j}(n)\big) \tag{3.12b}$$

where operation med{A,B,C} is defined by[6] [35]:

$$\mathtt{med}\{A, B, C\} = \begin{cases} A, (B < A < C) \vee (C < A < B) \\ B, (A \leq B \leq C) \vee (C \leq B \leq A) \\ C, \text{otherwise} \end{cases} \tag{3.13}$$

Step 6: *Get lines across mouth*
The mouth model will be matched over the edges of the top and bottom lip and the two corner positions. The edges of the lips will be found on vertical lines perpendicular to the line from the left and right corner. To bound the search area for the edges the lines are chosen as shown in figure 3.8. The yellow line shows the line from left to right mouth corner. Perpendicular to this line are light green lines of different lengths across the mouth. The darker green pixels are the actual pixels that form the lines on which the edges of the lips are searched. The mouth lines will be denoted by index l starting at the left side with line number 0. The pixel number p of the top pixel in each pixel line is 0.

Step 7: *Colour transformation*
For accurate lip edge localisation, the discriminating properties of lip and skin are exploited with a linear discriminant analysis (LDA). In an LDA approach a classification to groups in an m-dimensional space can be done with a linear function of the m variables. Thresholding on the function output will determine to which group an element belongs. The function is formed in such a way that the separation between the groups is maximised, and the distance within the groups is minimised. The general form of an LDA function for the classification in an m-dimensional space is given in equation 3.14.

$$f_{LDA}(u_1, u_2, ..., u_m) = \alpha_0 + \alpha_1 u_1 + \alpha_2 u_2 + ... + \alpha_m u_m \tag{3.14}$$

The values for the coefficients α_0 to α_n can for instance be determined by a principle component analysis (PCA) where an m-dimensional space is converted to a space of eigenvectors. The

[5] The used face detector only detects straight faces.

[6] The definition of the median is given here for three input values only. It is assumed to be known that this definition can be generalised to any number of input values

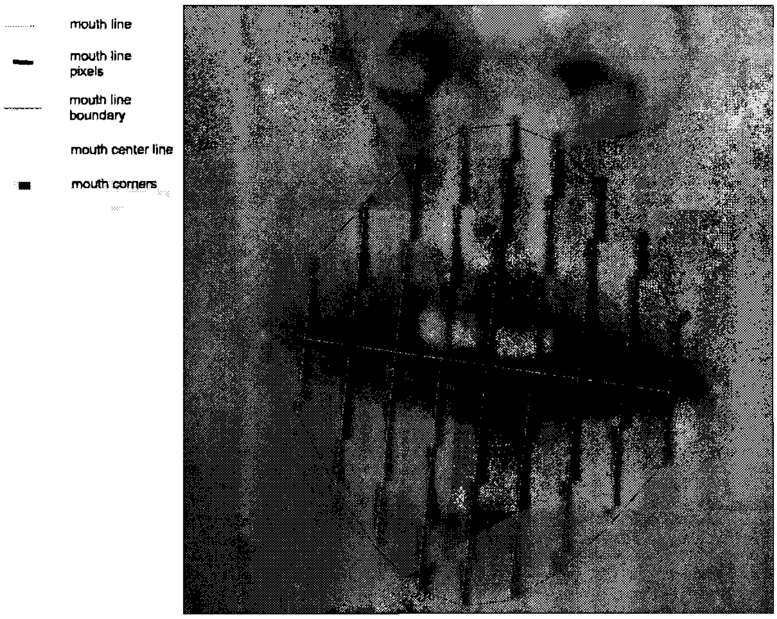


Figure 3.8: Selection of lines across mouth

eigenvector with the highest eigenvalue is the principal component of the space. A projection on to this principal component vector reduces the m-dimensional space to a 1 dimensional space in which a threshold can classify groups. The coefficients α_0 to α_n that form the LDA equation of this operation can be derived directly from the the principal component vector.

In the case of pixel classification (based on pixel colour) to the group of lip pixels and the group of skin and other pixels, the LDA function is a function of the U and V values of the pixels. Histogram information of the U and V values of lips and skin (that is generated from images of the AR faces database [32] and "Philips ToUcam II" webcam images), shows a distribution of lip pixels that have higher U and V values than the group of skin pixels. The LDA function for lip and skin discrimination (equation 3.15) is essentially the projection on the vector in the UV-plane pointing from the gravity point of skin pixels to the gravity point of lip pixels. The LDA function for discrimination in the UV-plane becomes:

$$M_l(p,n) = \alpha_0(n) + \alpha_1(n)V_l(p,n) + \alpha_2(n)U_l(p,n), \tag{3.15}$$

where α_1 and α_2 actually form the linear projection basis vector in the UV-plane. α_o can be left out as it only influences the level of the threshold which is determined dynamically. The index l denotes the line number and p is the pixel number in mouth line l as explained in the previous step of the algorithm. A UV-plane with contour lines of the histogram of skin and lip pixels is shown in figure 3.9.

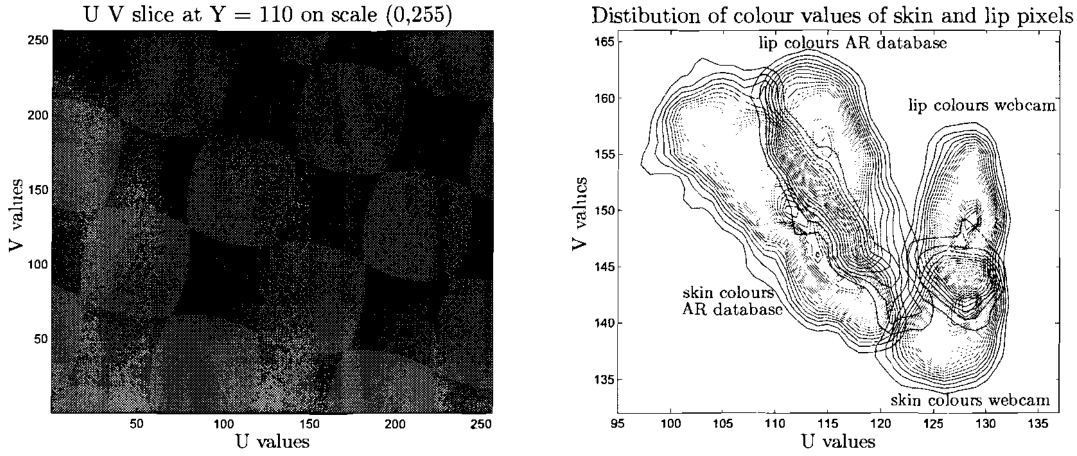


Figure 3.9: Left: A UV-slice in the YUV-space for Y=110. Right: histogram of U and V values of skin and lip pixels for AR database and webcam images (The area is shown as a rectangle in the UV-slice).

For each individual, for different camera's and for different lighting conditions the discriminating function differs, but generally lips contain a stronger red component than skin. U and especially V values of lip pixels are higher than U and V values of skin pixels. The initial discriminating function (equation 3.15) will therefore be based on statistical data retrieved from the U V histograms. The initial values for α_1 and α_2 are 0.88 and 0.48. After each frame the discriminating function adapts to the U and V values of a located person when a correct mouth matching is done. The adaptation is however bound to the knowledge that generally the U and V values of lips are higher than the U and V values of skin. α_1 and α_2 will therefore always remain positive.

Step 8: *Luminance correction*

The distinction between skin and lips is achieved by re-mapping the colour values as described in the previous step. The colour of teeth and the parts inside the mouth in some occasions however is similar to lips and skin. To cope with this situation, the fact that teeth generally have a much higher luminance and the parts inside the mouth have a much lower luminance value than the rest of the MROI is exploited for distinction. These distinctive properties are applied by subtracting very bright and very dark pixel values (respectively the absolute value of the difference to a high and low luminance threshold) from the output of the colour transformation. The luminance corrected mouth line $L_l(p,n)$ is calculated by

$$L_l(p,n) = M_l(p,n) - r\left\{\max\left(\left(F_l(p,n) - th_{high,l}\right),0\right) + \max\left(\left(-F_l(p,n) + th_{low,l}\right),0\right)\right\}, \quad (3.16)$$

where r is a scaling term to normalise the luminance correction term to $M_l(p,n)$. The high and low luminance thresholds th_{low} and th_{high} are calculated with the average luminance value $\mu_{lum,l}$ and the standard deviation $\sigma_{lum,l}$ of the luminance values of the mouth lines.

$$th_{high,l} = \mu_{lum,l} \quad (3.17a)$$

$$th_{low,l} = \mu_{lum,l} - 0.8\sigma_{lum,l} \tag{3.17b}$$

Step 9: *Mark lip*
On each colour transformed and luminance corrected mouth line ($L_l(p,n)$) the lip part is marked. $L_l(p,n)$ has a high value where it is crossing lip area and a low value across skin, teeth and inner mouth parts. A lip threshold $th_{lip,l}$ is calculated based on the average ($\mu_{L,l}$) and standard deviation ($\sigma_{L,l}$) of $L_l(p,n)$

$$th_{lip,l} = \mu_{L,l} + 0.3\sigma_{L,l} \tag{3.18}$$

Where $L_l(p,n)$ exceeds $th_{lip,l}$ lip is detected.

$$L_{lip}(p,n) = \mathtt{sign}(L_l(p,n) - th_{lip,l}) \tag{3.19}$$

where $\mathtt{sign}(a)$ is defined as:

$$\mathtt{sign}\{a\} = \begin{cases} 1, (a \geq 0) \\ 0, otherwise \end{cases} \tag{3.20}$$

Erosion and dilation steps (equation 3.21) on $L_{lip}(p,n)$ remove noise and fill gaps in lips. The erosion and dilation step is skipped when the length of $L_{lip}(p,n)$ is smaller than 18 pixels because the thickness of the lip on these lines often does not exceed one or two pixels. An erosion step would in that case incorrectly remove the detected lip. Three morphological steps are done with structuring elements $\mathbf{S}_{l1}$ and $\mathbf{S}_{l2}$.

$$\mathbf{S}_{l1} = \begin{bmatrix} 1 & 1 & 1 \end{bmatrix}$$

$$\mathbf{S}_{l2} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \end{bmatrix}$$

$$L'_{lip}(p,n) = E\Big(D\big(E(L_{lip}(p,n), S_{l1}), S_{l2}\big), S_{l1}\Big) \tag{3.21}$$

Let $\mathbf{P}_{l,lip}$ be the set of pixel indices on mouth line l where $L'_{lip}(p,n)$ is one. Let $p_{l,c}$ be the pixel number on $L'_{lip}(p,n)$ where the luminance of the mouth line around the center is lowest. The two groups of adjacent pixel indices in set $\mathbf{P}_{l,lip}$ closest to pixel index $p_{l,c}$ are the upper and lower lip pixel groups $\mathbf{P}_{l,up}$ and $\mathbf{P}_{l,low}$. In case a group overlaps pixel $p_{l,c}$, that group will be split at $p_{l,c}$ into the two adjacent groups $\mathbf{P}_{l,up}$ and $\mathbf{P}_{l,low}$. The pixel group with the lowest numbers is defined $\mathbf{P}_{l,up}$ the group with highest indices is defined $\mathbf{P}_{l,low}$. The maximum and minimum values of $\mathbf{P}_{l,up}$ and $\mathbf{P}_{l,low}$ are the points where mouth line l crosses the edges of the upper and lower lip.

$$p_{l,up,top} = \mathtt{min}(\mathbf{P}_{l,up}) \qquad \textbf{top} \text{ edge of } \textbf{up}\text{per lip} \tag{3.22a}$$

$$p_{l,up,bot} = \mathtt{max}(\mathbf{P}_{l,up}) \qquad \textbf{bot}\text{tom edge of } \textbf{up}\text{per lip} \tag{3.22b}$$

$$p_{l,low,top} = \mathtt{min}(\mathbf{P}_{l,low}) \qquad \textbf{top} \text{ edge of } \textbf{low}\text{er lip} \tag{3.22c}$$

$$p_{l,low,bot} = \mathtt{max}(\mathbf{P}_{l,low}) \qquad \textbf{bot}\text{tom edge of } \textbf{low}\text{er lip} \tag{3.22d}$$

Figure 3.10 shows the colour transformation process and lip edge detection in an overview for one mouth line as described by step 6 to step 9.

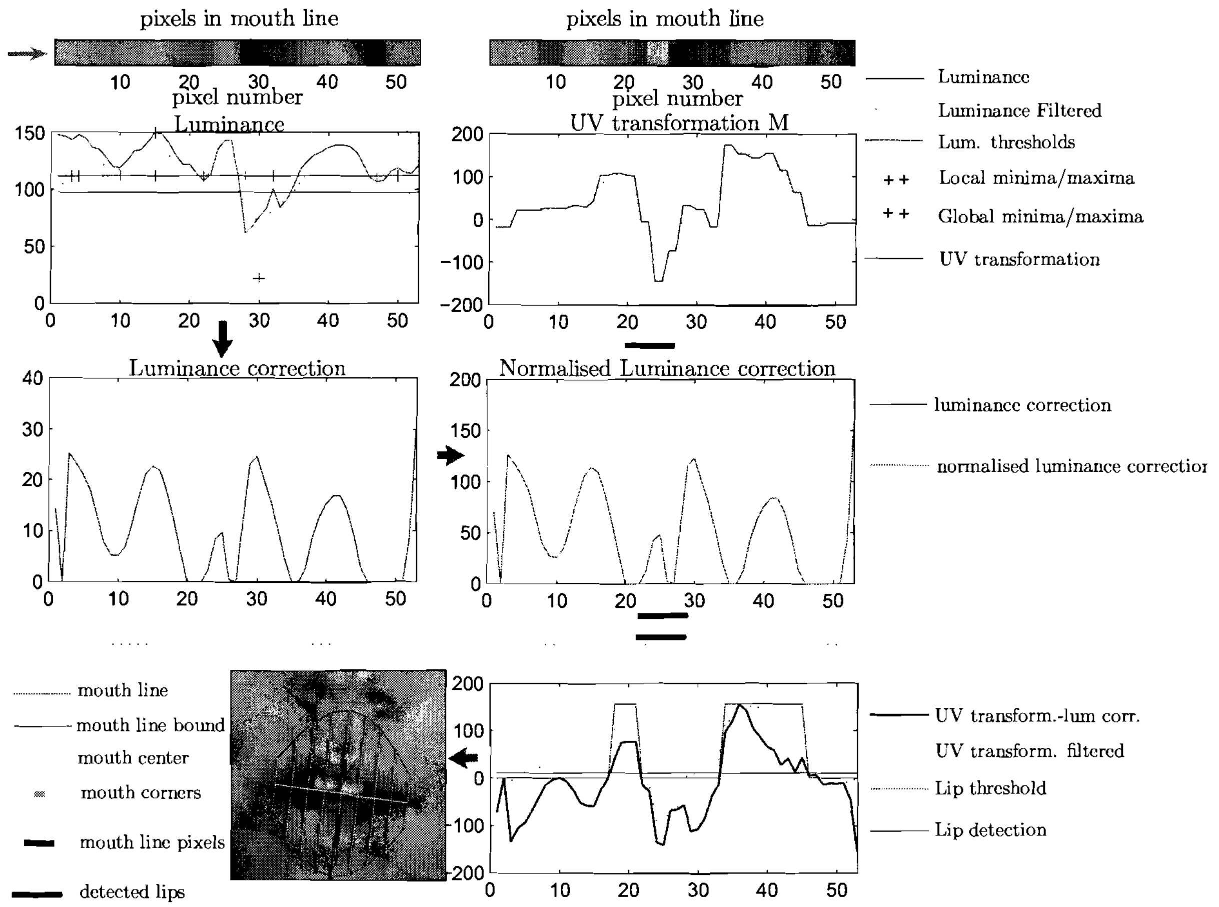


Figure 3.10: Colour transformation, luminance correction and lip edge detection for one mouth line

Step 10: *The mouth model*

The mouth shape can be found by fitting a least squares curve through all found edge points. We selected an exact parabolic fitting through each edge point of a lip contour and the two corner points. The median of all parabolic fittings of each lip contour is then taken to prevent the influence of an outlier edge point.

A parabola can be determined when two zero crossings are known and one more point on the parabola is defined. The edges defined in equation 3.22 (in combination with the mouth corners) each define a parabola $PAR_{l,up/low,top/bot}$ when assumed that the peak of those parabola is equally far from both corner points of the mouth. Let $o_{l,up/low,top/bot}$ be the distance of the peak of parabola $PAR_{l,up/low,top/bot}$ to the line between the mouth corners. And define:

$$\mathbf{O}_{up,top}(n) \text{ is the set of peak values } o_{(1,2,\ldots,\#mouthlines,l_m),up,top} \tag{3.23a}$$

$$\mathbf{O}_{up,bot}(n) \text{ is the set of peak values } o_{(1,2,\ldots,\#mouthlines,l_m),up,bot} \tag{3.23b}$$

$$\mathbf{O}_{low,top}(n) \text{ is the set of peak values } o_{(1,2,\ldots,\#mouthlines,l_m),low,top} \tag{3.23c}$$

$$\mathbf{O}_{low,bot}(n) \text{ is the set of peak values } o_{(1,2,\ldots,\#mouthlines,l_m),low,bot} \tag{3.23d}$$

The peak values calculated from the mouth line edges in the middle of the mouth are more reliable than the peak values calculated from the mouth lines near the edges of the mouth. In general, when an exact fitting is done (which is the case when fitting a parabola through 3 points) through 3 points, an error in one point influences the fitting more when that point is close to another point. A weighting is applied to the set of peak values by adding the 'middle values' (index numbers l_m) multiple times to the set of peak values. The median of each set $\mathbf{O}_{up/low,top/bot}(n)$ gives the peak value $O(n)$ of the matching parabola for each edge of the bottom and top lip.

$$O_{u,t}(n) = \mathrm{med}\big(\mathbf{O}_{up,top}(n)\big) \tag{3.24a}$$

$$O_{u,b}(n) = \mathrm{med}\big(\mathbf{O}_{up,bot}(n)\big) \tag{3.24b}$$

$$O_{l,t}(n) = \mathrm{med}\big(\mathbf{O}_{low,top}(n)\big) \tag{3.24c}$$

$$O_{l,b}(n) = \mathrm{med}\big(\mathbf{O}_{low,bot}(n)\big) \tag{3.24d}$$

Figure 3.11 shows the matching of the parabolic lip models to the detected edge points on the mouth lines. The mouth feature values horizontal and vertical mouth opening are now

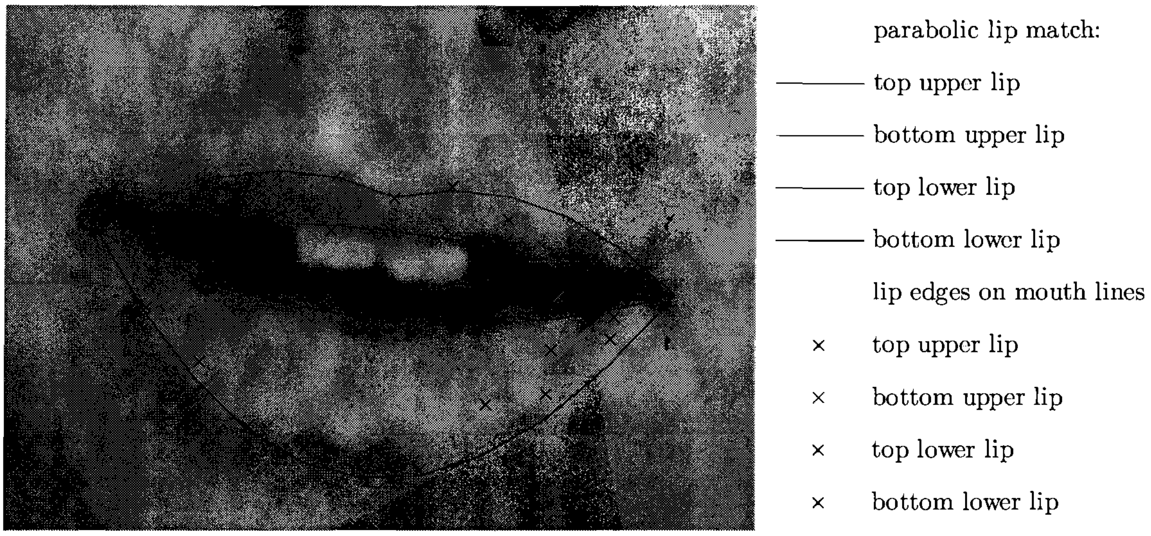


Figure 3.11: Matching of parabolic mouth model through lip edges on mouth lines and mouth corners

available in each frame n for visual speech detection.

$$O_v(n) = O_{u,b}(n) - O_{l,t}(n) \tag{3.25a}$$

$$O_h(n) = W_j(n)\text{as given in equation 3.10a.} \tag{3.25b}$$

The recognition of visemes with these features will be discussed in section 3.4.

3.3.2 Mouth finding fall-back procedure

The mouth finding algorithm as described by step 1 through step 5 of the lip finding algorithm in section 3.3.1, can fail due to the following reasons:

- There is no mouth blob, because the mouth area is not dark enough or the mouth is occluded for instance.
- Mouth blob is separated (because of for instance bright teeth) into two smaller blobs that fail for the mouth classification step.
- Mouth blob is too large, because it is connected to for instance a moustache blob or a shade blob.
- Only half of the mouth is found due to asymmetrical lighting conditions.

To address this problem, a fall-back procedure is done, in case no valid mouth blob was found with mouth finding algorithm. This fall-back procedure will attempt to find a correct mouth blob in an alternative way that is based on the colour information in the MROI. If this fall-back procedure also fails in finding the correct position of the mouth, the mouth is declared not valid and discarded by the rest of the algorithm.

The colour based mouth finding algorithm

From the U and V values of the pixels in the MROI an image is created in which red is more pronounced. An example can be seen in image b of figure 3.12. This transformation is the same as transformation of the mouth lines in step 7 of the lip finding algorithm. In this case it is applied to an area in stead of lines. In equation 3.26 the transformation is given for image n.

$$M_{MROI}(x, y, n) = \alpha_1(n)V(x, y, n) + \alpha_2(n)U(x, y, n) \tag{3.26}$$

An example of result of this operation is shown in figure 3.12b. The image which is the result of equation 3.26 is transformed into a binary image as described in step 1 of the lip finding algorithm. This reddish binary image will be named $B_r(x, y, n)$. This is shown in image c of figure 3.12. The pixel values in this binary image are based on colour. The pixels with value one, will therefore be named reddish pixels.

If this fall-back procedure was called because the blobs based on luminance were to big or there were no blobs at all, the rest of the fall-back procedure is the same as step 2 through 5 of the lip finding algorithm as described in section 3.3.1. The mouth corner localisation is in

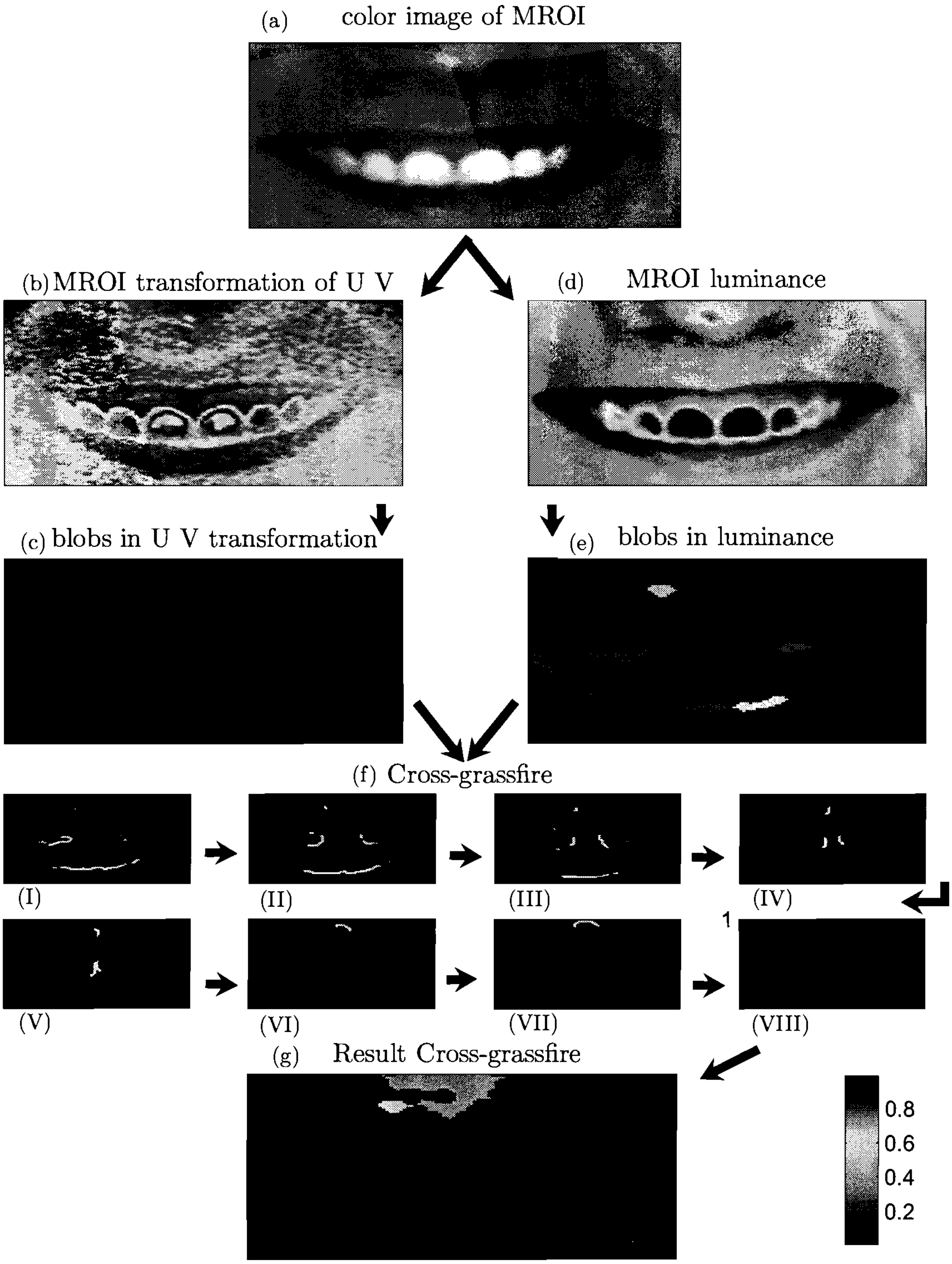


Figure 3.12: mouth blob localisation fall-back procedure

that case based on the reddish pixels blobs.
If the fall-back procedure was called because the widths of the blobs (that were found based on luminance only) were too small or there was no valid blob at all, the morphological step is replaced by the following steps which we shall call *cross-grassfire algorithm*. It was inspired by the grassfire algorithm[7]:
Cross-grassfire
The luminance blobs (image c in figure 3.12) are allowed to grow further only where the pixel values of the reddish binary image (image e in figure 3.12) are one. The growing process is described by the following operations:
The luminance blobs in $B'(x, y, n)$ are dilated with structuring element $\mathbf{S_g}$ as defined in step 3 of the lip finding algorithm in section 3.3.1. The resulting temporary binary image is compared (logical and) with binary image $B_r(x, y, n)$. The resulting binary image is added (logical or) with binary image $B'(x, y, n)$. The number of pixels with value one are counted. These operations are continued until the number pixels with value one does not increase anymore. Eight cycles of this operation are shown in image f-I to f-VIII of figure 3.12. The number of blobs is reduced to the resulting blobs as shown in image g of figure 3.12.
Morphological operations as described in step 2 of the lip finding algorithm are performed on the resulting binary image. The rest of the fall-back procedure is the same as step 4 and 5 of the lip finding algorithm as described in section 3.3.1.

3.4 Visual speech detection

The visual features, that are extracted as described in the previous chapter, are used to classify the mouth when one is detected in the sequence. The classification was first done for sequences with isolated vowel utterances and thereafter with sequences of natural speech. The first subsection of this chapter will describe how non-speech is detected. Subsection 3.4.2 gives a description of the classification to the vowel groups based on the visual features.

3.4.1 Visual non-speech detection

The mouth features that are used for the visual non-speech detector are vertical O_v and horizontal O_h mouth opening. Visual non-speech is declared when:

- The mouth is closed for 0.5 seconds.
- Horizontal and vertical mouth opening is approximately constant for 0.8 seconds.

The detection is implemented by storing a history of 0.8 seconds of mouth openings. If the maximum vertical mouth opening in the frames of the last 0.5 seconds is not larger than $O_{V_{thCIM}}(n)$, visual non-speech is declared. Also, when the difference between the largest vertical mouth opening minus the smallest vertical mouth opening and the difference between

[7] The author has no knowledge whether a similar method already exists under a different name

the largest horizontal mouth opening minus the smallest horizontal mouth opening the during the last 0.8 seconds are smaller than the vertical and horizontal movement thresholds $O_{V_{th}}(n)$ and $O_{H_{th}}(n)$, visual non-speech is declared. The vertical and horizontal movement thresholds are both derived from the width of the MROI. This straight forward algorithm is also be described by equations 3.27,3.28,3.29 and 3.30.

$$fl_{NoVM}(n) = \begin{cases} 1 & \text{,if } \max\left(O_v(n - \lfloor 0.8F_v \rfloor), ...O_v(n)\right) - \min\left(O_v(n - \lfloor 0.8F_v \rfloor), ...O_v(n)\right) < O_{V_{th}}(n) \\ 0 & \text{,otherwise} \end{cases} \tag{3.27}$$

$$fl_{NoHM}(n) = \begin{cases} 1 & \text{,if } \max\left(O_h(n - \lfloor 0.8F_v \rfloor), ...O_h(n)\right) - \min\left(O_h(n - \lfloor 0.8F_v \rfloor), ...O_h(n)\right) < O_{H_{th}}(n) \\ 0 & \text{,otherwise} \end{cases} \tag{3.28}$$

$$fl_{CIM}(n) = \begin{cases} 1 & \text{if } \max\left(O_v(n - \lfloor 0.5F_v \rfloor), ...O_v(n)\right) < O_{V_{th}CIM}(n) \\ 0 & \text{,otherwise} \end{cases} \tag{3.29}$$

$$fl_{Non-Speech}(n) = fl_{CIM}(n) \vee (fl_{NoVM}(n) \wedge fl_{NoHM}(n)) \tag{3.30}$$

$O_v(n)$ is the vertical mouth opening in frame n, $O_h(n)$ is the horizontal mouth opening in frame n and F_v is the frame rate. The tests and results of this non-speech detection algorithm are given in section 3.5.3.

3.4.2 Visual vowel detection

It has been reported in literature (e.g. [28]) that it is not possible to visually distinguish all pronounced speech from (still) images of mouths. We conducted a preliminary test with 6 individuals to find out how well people were able to recognise the spoken vowels from still images of well-articulated vowels. Based on these findings and the author's own intuition, the idea rose that only three vowel groups are visually very distinct. These three groups are illustrated on an F1-F2-plane in figure 3.13. Visual recognition of these vowel groups and matching with the audio features is one of the goals of the research in this report. In this case, the visual recognition will be based on distinctive properties of the mouth shape for each vowel group. The three chosen visual vowel groups that are shown in figure 3.13 will be named A-, O- and E-group. The visual A-group is located on the right of the acoustical F1-F2 plane. The visual O-group is located at the bottom of the acoustical F1-F2 plane and the E-group at the top of the F1-F2 plane. This division was based on preliminary tests. It is an assumption that may profit from refinement with further research.
First, the reference shape of the mouth needs to be determined for each speaker. The sizes of the reference mouth is needed to determine threshold of mouth shape classification. For now, this problem is solved by calculating the average mouth shape with the history of found mouth shapes. The algorithm that was implemented to visually recognise the vowel groups consists of the the following steps that will be explained in this section:
Step 1 :Extract the mouth features.
Step 2 :Determine the reference mouth shape of the current person.
Step 3 :Classify the mouth by comparing the features to the reference mouth shape.

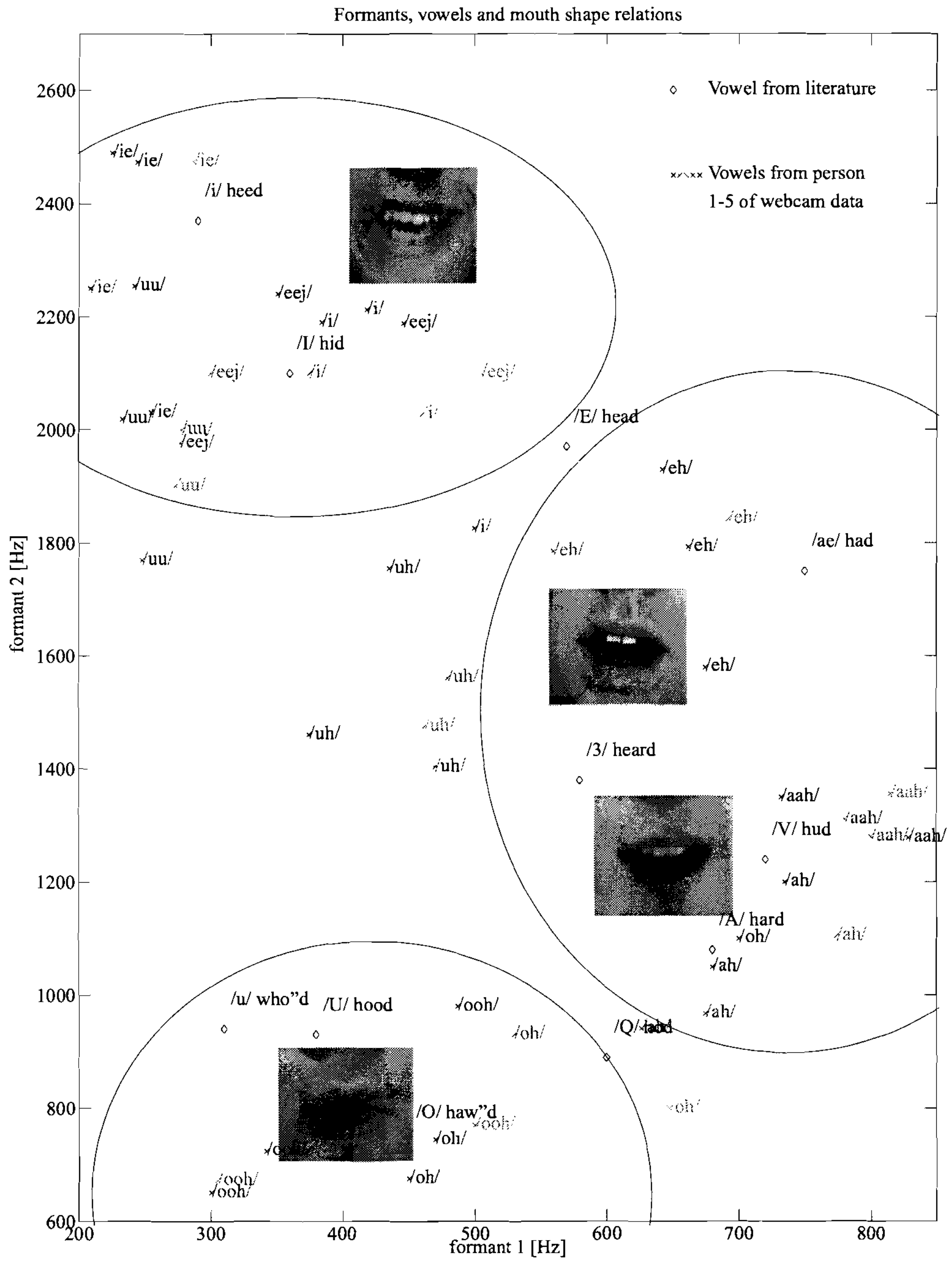


Figure 3.13: Distinct mouth shapes and vowel groups in formant1-formant2 plane (F1-F2 plane)

Step 1:*Extract the mouth features.*
The mouth features are extracted as explained in the section 3.3.1.
Step 2: *Determine the reference mouth shape*
The reference mouth depends on:

- The size of the mouth of the person.
- The distance from the person to the camera.

The reference mouth shape is based on only the widths of the history of mouth shapes. A reference value for the vertical mouth opening cannot simply be based on the history of vertical mouth openings as it can be possible that a person is keeping his mouth closed for a very long period. The reference vertical mouth opening is therefore based on the reference mouth width. The thresholds that are used in the next step of the mouth shape classification are therefore actually only based on the approximation of the average width of the mouth. In the test sequences the distance of the speaker to the camera is kept constant.
Step 3: *Mouth classification*
For the classification of the mouth shapes to a vowel group, the distinctive visual features are retrieved from well-articulated vowel pronunciations. An example of a very distinct feature is the strong decrease in width of the mouth when the vowel O is uttered. To retrieve features, sequences were recorded with people saying isolated vowels. The thresholds that were used for classification are:

- swthr (relative small width threshold) ; The portion of the reference width that determines the threshold for mouth width.
- vopthrrat (relative vertical opening threshold) ; The portion of the reference vertical opening that determines the threshold for vertical mouth opening.
- vthr (vertical opening threshold) ; A small absolute threshold for vertical mouth opening in pixels.
- hwratthr (height width ratio threshold) ; Threshold for ratio between height and width of the mouth.
- brightrat (brightness ratio) ; High threshold for the ratio of the average luminance of the mouth opening and the average luminance of the lips.
- darkrat (dark ratio) ; Low threshold for the ratio of the average luminance of the mouth opening and the average luminance of the lips.

Table 3.4.2 shows the conditions of the features of the mouth for classification to the vowel groups.
These values were heuristically chosen based on observations from the data. The available training set is not large enough to produce a more solid base for these choices.
The method for visual vowel detection is used and tested on the sequences with isolated vowel utterances and on sequences with natural speech. The experiments are described in section 3.5.4.

Table 3.1: Visual feature conditions for the vowel groups

	swthr	vopthrrat	vthr	hwratthr	brightrat	darkrat
	= 0.9	= 0.2	= 3	= 0.16	= 0.85	=0.75
I	x	>	>	>	>	x
A	x	>	>	>	<	x
O	<	>	>	x	x	<

3.5 Experiments

3.5.1 Mouth Movement Dynamics

Introduction This brief paragraph addresses the movement of the mouth and the minimal image acquisition speed. The determination of the minimum frame rate at which visual speech detection should be possible, is necessary for a decision on the visual acquisition hardware and/or the available processing power.

Method To determine the minimum frame rate, a natural speaking person facing the camera is recorded at maximum frame rate of the available webcam. The lip finding algorithm, as described in section 3.3.1, is used to record the movement of the lips. Also the images are manually examined to make sure no lip movement is missed by the limited frame rate of the webcam.

Analysis In the recorded sequences at 30 frames per second, the major mouth movements were all present in the footage. The inspection of the signals revealed typical opening and closing times of the mouth in the order of 0.1 seconds. The spectrum of the vertical opening movements are plot in figure 3.14. The spectrum shows a drop at 7.5Hz for the mouth movement.

Conclusion From this brief analysis of natural visual speech, we conclude that the absolute minimum frame rate to record major mouth movement is 15Hz.

3.5.2 Lip and Skin Colour adaptation

Introduction

For the lip finding algorithm, the parameters for the colour transformation of the U and V values (equation 3.15) are initially based on statistical data retrieved from the AR database and webcam images. When a satisfactory matching of lips is done, the colours of the lips and skin of the found person can be used to adapt the parameters for the colour transformation for better matching in the subsequent frames. This paragraph describes the illustrative experiment for this addition to the lip finding algorithm. The aim of these experiments is to find out if altering the transformation parameters based on the previously matched mouth, is a useful addition to the lip finding algorithm.

The parameters of the transformations are changed when a correct match to the mouth was done by the mouth matching algorithm. The criterion to determine if a mouth was matched correctly in this implementation is based on the consistency of the colours within each area of

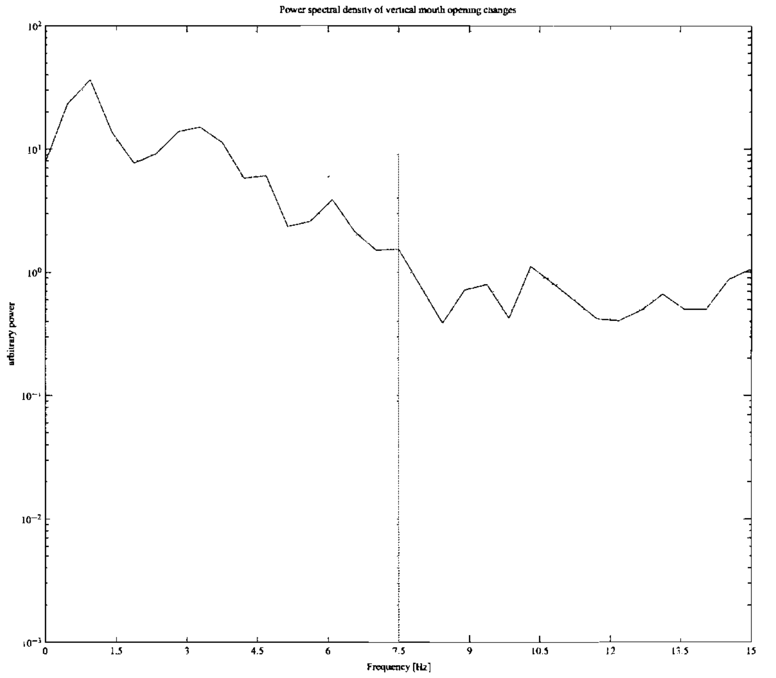


Figure 3.14: Spectrum of vertical mouth opening movement in a natural speech sequence of 40 seconds

the mouth template and whether the mouth blob was localised correctly. In essence when the value of the weak classifier score is below a threshold the parameters of the transformation are not adapted. The consistency of the colours within each area of the mouth template is determined by the maximum of the standard deviations of the U and V values of the pixels in each mouth template area divided by the number of elements in each template area. The threshold for the maximum of this scaled standard deviation in colour within each area of the matched mouth template is heuristically determined to 0.06 .

If the decision was made to adapt the transformation parameters, they are changed in a recursive way. In this experiment, the new parameters are 0.9 times the old parameters plus 0.1 times the optimal new parameters which are based on the average colours of the lips and skin in the matched mouth template. After each correctly matched mouth, the transformation parameters should iterate one step towards the optimal parameters to distinguish lip colours from the skin colours of the current person.

Method

The adaptation algorithm was tested by initiating the transformation parameters with sub optimal values. The iteration process of the transformation parameters is visualised by feeding the same colour image of a face several times to the algorithm.

Analysis

The analysis is done by manually inspecting the transformation image of the MROI.

An example of the influence of 5 iterations of parameters of the colour transformation is shown in figure 3.15.

Conclusion

The tests showed a very promising accuracy improvement in the matching of the lips. In a few occasions however, the matching iterated to an incorrect matching. The parameter diverted

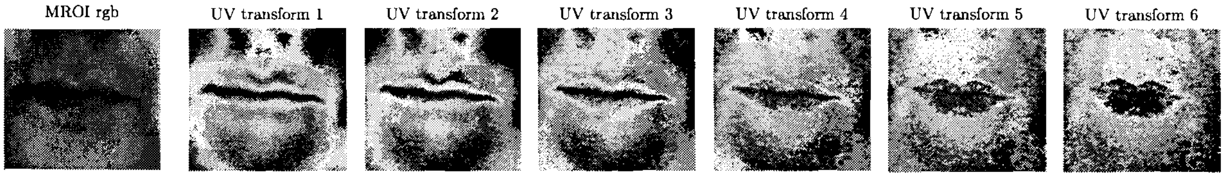


Figure 3.15: Influence of updating the parameters of the colour transformation with U and V values of lip and skin with correctly found mouth. The initial parameters were deliberately chosen suboptimal to clearly show the iteration steps.

from the optimal parameters. The iteration process can therefore not be regarded as stable. The reason for that was an incorrect initial match of the mouth.
For now, this colour adaptation is added to the lip finding algorithm, but with the following restrictions and features:

- The criterion on which to determine if a mouth was matched correctly is set very high. Only when colour within the areas of the mouth model are very consistent (very low standard deviation of the colours), the parameters are updated.
- Boundaries for the transformation parameters are applied to prevent divergence.
- When the boundaries for the parameters are exceeded, they will be reset to initial values.

At least the following problems need to be addressed for improving the adaptation algorithm:

- The standard deviation of the colours in mouth match only is not a very suitable criterion to decide whether the mouth was matched properly. A better criterion needs to be found.
- It is unclear what the optimal iteration speed parameter should be to prevent divergence of the transformation parameters and simultaneously a fast enough response.
- Divergence of the parameters is not detected.

Unfortunately they could not be addressed in this assignment because of time limitations.

3.5.3 Visual non-speech detection tests

Introduction
The non-speech detection tests were performed on 2 sequences of a person alternating talking and being silent. A sequence of a person who was not a part of the training set is used to produce performance figures. The author is aware that a test set with only one person can only give an weak indication of the actual performance. The time to perform large scale performance tests was not available. The algorithm has also been tested on footage of newsreaders recorded from television broadcasts. Because of the large distance of the newsreaders

Table 3.2: Visual non-speech detector performance on two test sequences

	test sequence 1	test sequence 2
Visual non-speech detections	11 of 11	12 of 12
False non-speech detection during speech,	0	0

from the camera the resolution of the video data was too low for accurate lip finding which was one of the conditions of this assignment. It is concluded that as assumed the algorithm is not suitable to locate lips accurately in low resolution video. The tests on the television sequences are not further discussed.

Method

The performance of the visual non-speech detector was measured by manual inspection of each detection. Also segments in which there is no speech for 1 second were analysed. When it was not detected by the visual non-speech detector it counts as a missed detection. For illustrative purpose figure 3.16 shows mouth feature values and the non-speech detection flag on a training sequence.

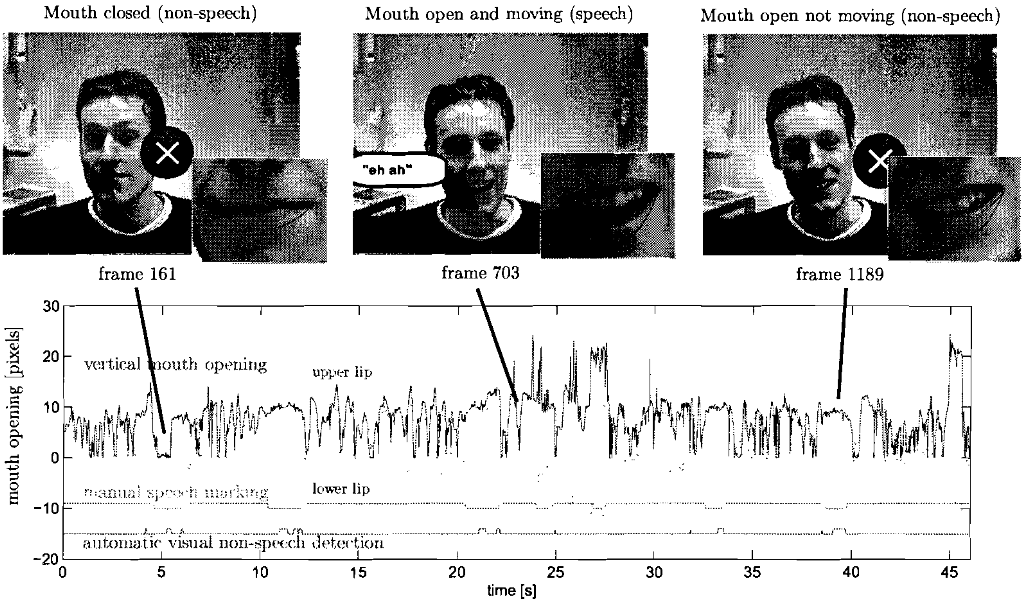


Figure 3.16: Example of non-speech detection in a sequence. Three frames are highlighted

Analysis

In the first test sequence the person in the camera's field of view was asked to utter isolated Dutch vowels and to be silent between every vowel for one second. The second test sequence consisted of short sentences separated by silences of about one second. The performance of the non-speech detector on both these sequences is shown in table 3.5.3 In figure 3.17 the visual non-speech detection flag is plotted in red under the true speech in green. The non-speech detection went up in all 12 silent segments of the sequence. Note that the detection flag in many case goes down again before speech was present. The reason for this is the opening of the mouth for instance for inhaling prior to speech.

Conclusion

It should be noted that the performance of the lip matching algorithm was remarkably good on the person in the test sequences. There were only a few mouth mismatches on the total of 1367 frames. In the training set, on the average about once every 50 frames the mouth matching was incorrect.

From the result on the test sequences, the used method for non-speech detection appears very robust. Furthermore, from the data that the non-speech detection flag goes down before actual speech is present, the detection flag could be regarded as an "intend-to-speak" detector. Though on the training sets, the flag is not as robust as appears in the test set.

An elaborate test is necessary using more sequences with different people to confirm the actual performance of this visual-only non-speech detector.

3.5.4 Visual vowel detection tests

Introduction

As explained in chapter 3.4.2 the vowel detection tests were done on sequences with isolated vowels and on sequences with natural speech. The same sequences that were used in the visual non-speech detection test in section 3.5.3 are used for this visual vowel detection test.

Method

All frames of the test sequences were manually marked. Each acoustic frame was assigned: I, A, O, 'indistinct speech' or 'non-speech'. These vowels are encoded with the numbers 1, 2 ,3, 0.5 and 0. The detection was then compared to the automatic visual vowel detection. The detection error cannot be determined by simply subtracting the detection from the true vowel values. In many cases the mouth is already shaped to utter sound which is audible later. Each connected (maximum of 4 frames) visual vowel detection is therefore compared to the true assigned vowel in a few frames around the current frame. When a visually detected vowel was present within the true vowel data of the previous till the next 4 frames it counts as a correct detection. Also each visually detected vowel is counted as correct when an 'indistinct' vowel was found in the true data. When a visual vowel detection cannot be confirmed with the true vowel data is it counted as an error.

The visual vowel detection was also done on the training sequences with isolated vowels. That analysis is added also.

Analysis

Visual vowel detection in isolated utterances

Table 3.5.4 gives an overview of how the true vowels were visually detected in the training set sequences with isolated vowels. It is important to note that the test subjects were not asked to exaggerate their articulation.

Visual vowel detection in natural speech The true vowel data and the visual vowel detection of the test sequence with natural speech are shown in green and red in figure 3.17. It can be seen in the close up of the visual and true vowel data in figure 3.18 that the visual detection is in general early on the true vowel data.

Table 3.4 gives an overview of how the true vowels were visually detected in the test sequence with natural speech segments.

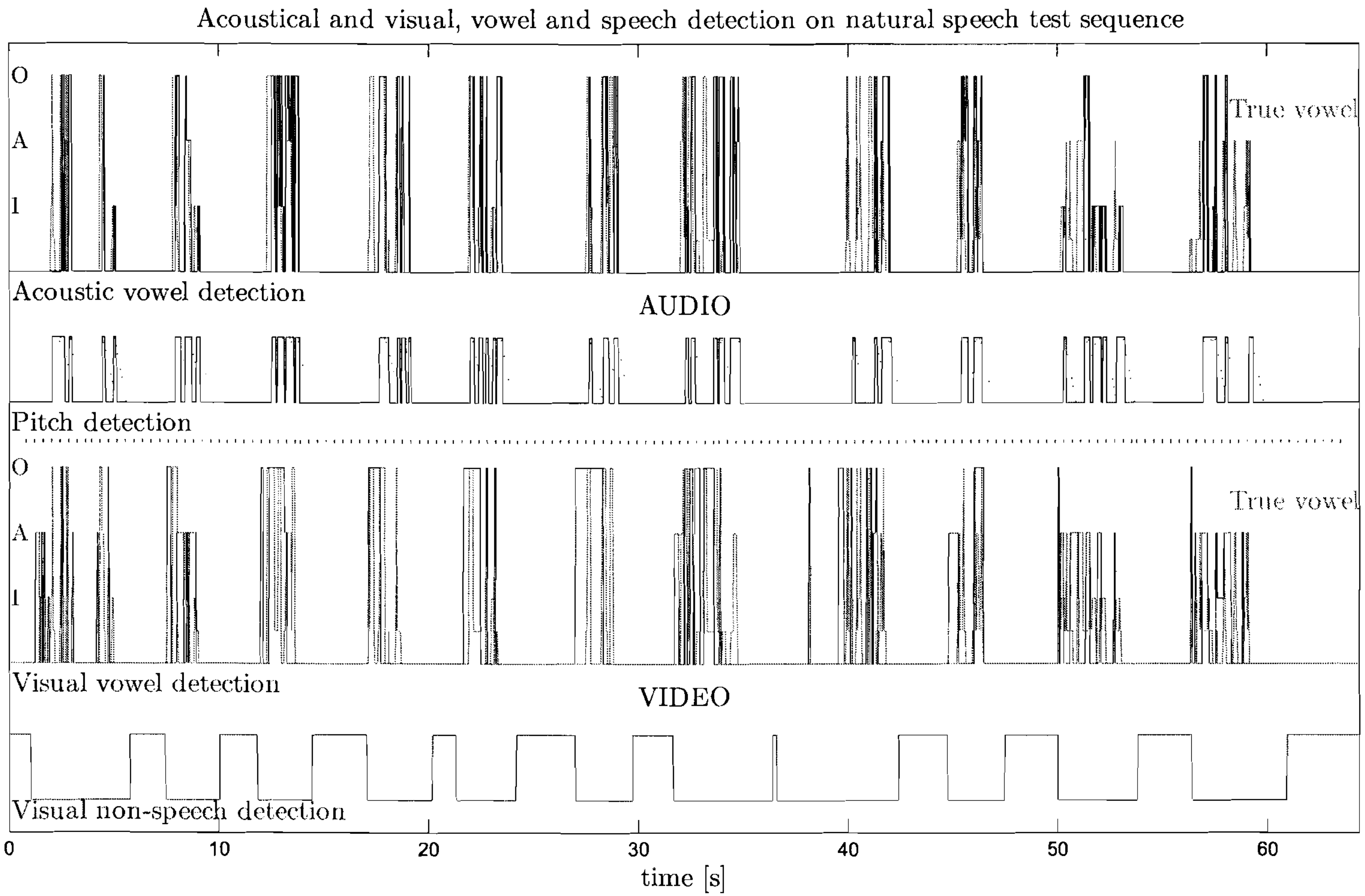


Figure 3.17: Acoustical and visual speech and vowel detection on the test sequence with natural speech intermitted by silences. (No acoustical interference was added.)

Table 3.3: This table gives the detection rate of the video-only vowel detector. The data set consisted of 50 isolated Dutch vowels. This test was done on the training set. It is therefore not an adequate indication of the true performance

Isolated	visually recognised as [%]			
acoustic vowels	I	A	O	none
eej, i, iee (15)	**73**	20	0	7
eh, ah, aah (15)	27	**53**	0	20
oh, ooh, uh, uu (20)	5	0	90	5
	number of false detections			
In long segments of non-speech	**6**	0	2	rest

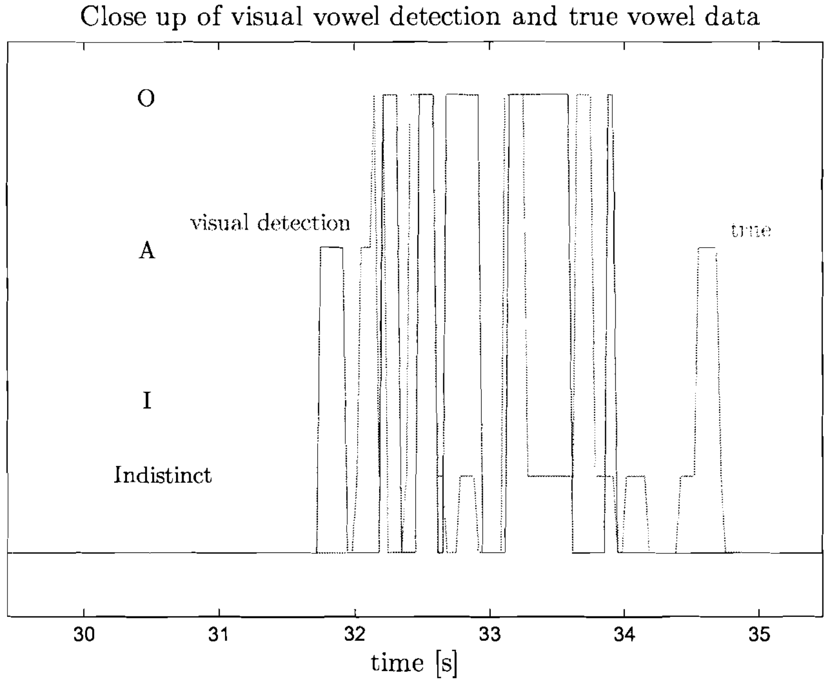


Figure 3.18: A close-up of visually detected vowel data comparison to true vowel data

Table 3.4: Visual detection of acoustic vowels in the test sequence with natural speech intermitted by silences

acoustic vowels	visually detected within 0.2 seconds [%]			
	I	A	O	missed
eej, i, iee (7)	**14**	43	43	0
eh, ah, aah (15)	0	**87**	0	13
oh, ooh, uh, uu (21)	0	5	**81**	14
Indistinct speech (23)	48	30	4	17
	number of false detections			
In long segment of non-speech (12)	0	0	**1**	

Conclusion

The performance criterion was chosen very lenient. From the detection rates it can be concluded that for articulated speech the A en O sounds can be visually detected. The visual distinction of the I group is not very large because with the used visual vowel detector it was often mistaken for an O or an A. Note that a row with indistinct speech is added to table 3.4. In the recording of natural speech there were many frames which could not be manually identified as an element of one of the vowel groups, but were definitely identified as human speech when manually marking the frames. Detection by the visual vowel detector in these frames often occurred. The number of false viseme detections for the test sequence was remarkably low. It has to be noted that the test person was asked not to try and fool the visual detection but to maintain a neutral face when being silent.

Chapter 4

Audiovisual Speech Detection

The previous two chapters described the acoustical speech and visual speech. This chapter will describe the approach to make an audio-only speech detector more robust by integrating the audio and video features. The primary idea of this approach is that voiced speech can only originate from an opened mouth. This extra visual criterion for speech allows the audio speech detector to be less strict because the absence of an opened moving mouth can exclude false positives. This makes it more robust in acoustically noisy environments or speech interference. The extended idea of the approach in this document is that speech is more than pitch alone. Normal speech consists of vowels and consonants.

Speaker independent vowel recognition has been widely reported in the literature on audio-only speech recognition. In the research described in this document there was no attempt to create a speech recognition system that distinguishes all sounds, but to match a few well-distinguishable mouth shapes to a few acoustically well-distinguishable vowels. This provides more exclusion opportunities than pitch detection alone and allows the acoustic speech detector to be even less strict and thus even more robust to noise and speech interference. This approach could also be useful to detect the active speaker in a scene with multiple persons.

The choice of the fusion of the acoustical and visual data retrieved from different modalities is given in section 4.1. The integration of and experiments with the audiovisual speech detection algorithm are given in section 4.2.

4.1 Modality fusion

In the introduction in chapter 1 the two often-used methods of data fusion are already described. Based on the results of the visual non-speech and vowel detector it was decided to first attempt a data fusion on decision level. The data fusion on feature level requires a much higher understanding of the true acoustical features and was unfeasible in this assignment.

The visual non-speech detector was designed to be conservative. This means that the decisions based on the video data have a low false positive rate. In essence this means that visual non-speech detection has a high confidence value which makes the detections in the visual domain suitable to overrule the detections in the audio domain. The audiovisual speech detector

will therefore be based on the pitch based speech detector and aided by the detections in the visual domain. The visual non-speech detection flag will overrule false speech detections by the acoustical speech detector.
The visual vowel detector was also designed to be conservative. Missed speech by the acoustical speech detector can also be overruled by the visual vowel detector. We decided not to include this in the tested decision based audiovisual speech detector because the acoustically missed speech detections are not numerous and the visual vowel detector can give false positive on pre-speech mouth movements.
The current reliability and accuracy of the vowel detector is not yet high enough for accurate matching to acoustical vowels on decision level. Furthermore, the performance of the acoustical vowel detector quickly drops when acoustical noise is added. An integration at feature level is probably more promising.

4.2 Integration of audiovisual speech detection

The flow-chart of the framework of the audiovisual speech detector that was written in a Matlab environment is shown in figure 4.1. A screen shot of the output is shown in figure 4.2. This screen shot shows the underlying calculations in both audio and video domain. The calculations in the audio domain are shown on the left-hand side and the calculations in the video domain are shown on the right-hand side. The two graphs on the top left show the spectra of audio block a and b in yellow. The red contour is the formant structure in which the blue circles mark formant 1 and formant 2. The two plots with the title 'autocorr. ..' show the autocorrelation of audio block a and b from which the pitch detection is determined. The green circles mark the the peak at lag zero and the peak for the lag corresponding to pitch of the speech.
The result of the grouping of the Viola and Jones face detector algorithm is shown in the top right image. In the bottom right, the binary image is shown from which the mouth blob and mouth corners are derived. The matching to the mouth curves is shown in the image on the right-hand side in the middle. The main output image is shown in the center. In this example frame it shows a speech balloon based on the audio on the left and a speech balloon based on visual features on the right-hand side of the face. In this example frame both the audio based and video based vowel detector agree that an A sound was produced. When visual non-speech is determined by the visual non-speech detector it features a non-speech sign at the right-hand side of the face. Numerical audio and video data is shown under the main image.

4.3 Audiovisual speech detection test

The performance of the audio-only speech detector was already shown in section 2.4. In this paragraph the influence of addition of the decisions in the video domain is demonstrated.
Method
The test sequence with natural speech is used to demonstrate the performance. The method is the same as described in section 2.4.1 where the true speech signal is determined by the

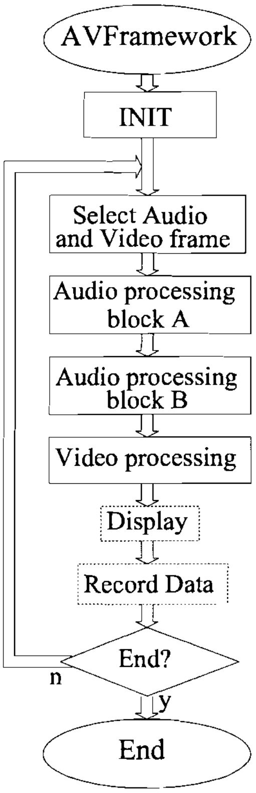
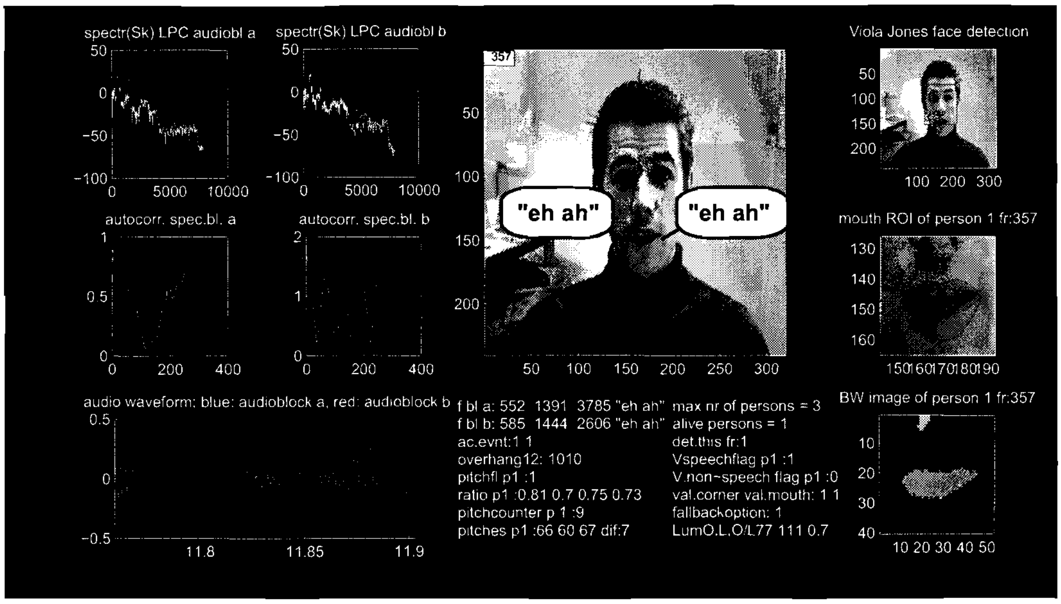


Figure 4.1: Flowchart audiovisual speech detector framework


spectr(Sk) LPC audiobl a
spectr(Sk) LPC audiobl b
autocorr. spec.bl. a
autocorr. spec.bl. b
audio waveform: blue: audioblock a, red: audioblock b
357
"eh ah"
"eh ah"
Viola Jones face detection
mouth ROI of person 1 fr:357
BW image of person 1 fr:357

```
f bl a: 552  1391  3785 "eh ah"   max nr of persons = 3
f bl b: 585  1444  2606 "eh ah"   alive persons = 1
ac.evnt:1 1                       det.this fr:1
overhang12: 1010                  Vspeechflag p1 :1
pitchfl p1 :1                     V.non-speech flag p1 :0
ratio p1 :0.81 0.7 0.75 0.73      val.corner val.mouth: 1 1
pitchcounter p 1 :9               fallbackoption: 1
pitches p1 :66 60 67 dif:7        LumO.L.O/L77 111 0.7
```

Figure 4.2: Screen shot of the output of the audiovisual speech detector featuring underlying calculations

pitch-based speech detection in a low noise environment.

Analysis

Figure 4.3 shows an increasing number of false audio-only speech detections in blue for increasing acoustical interference. The output of the (decision level) audiovisual speech detector is shown in red. The audiovisual speech detector shows less false detections. Figure 4.4 shows the increasing number of false detection in a graph for increasing acoustical interference of both the audio-only and the audiovisual speech detector.

Conclusion

The decision based audiovisual speech detector as demonstrated in this chapter is believed to only be an initial example of the possibilities of the visual features. The fusion of the audio and video features on decision level did not exploit any correlation between the two modalities. For future work it is recommended to exploit the fusion on feature level.

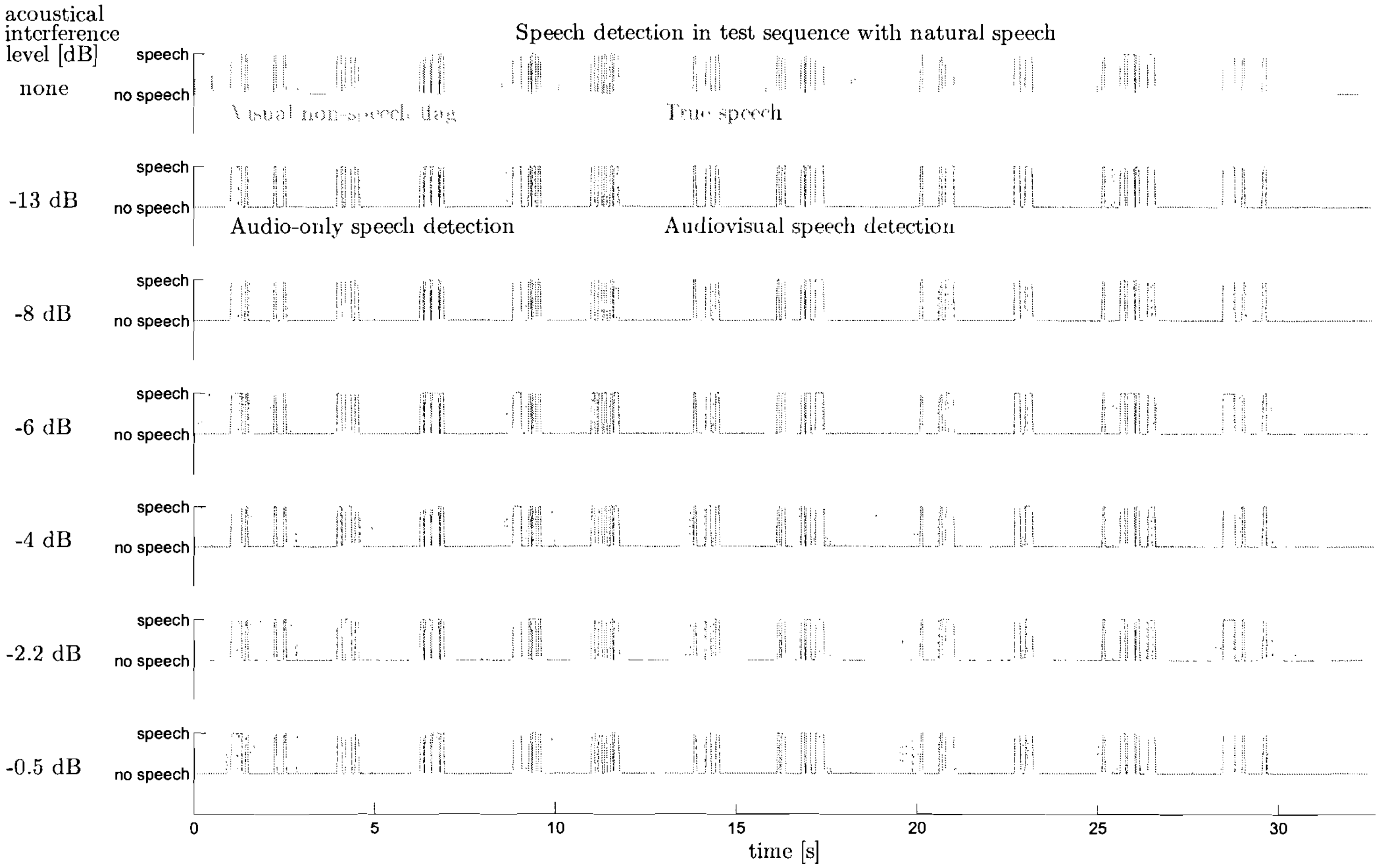


Figure 4.3: The detection errors for increasing acoustical interference.

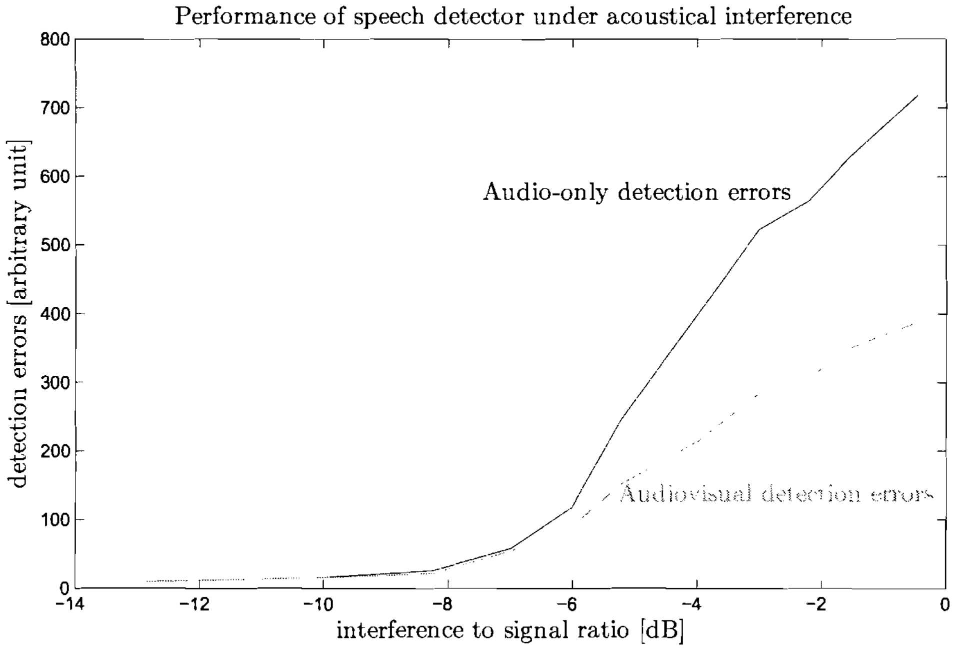


Figure 4.4: The detection errors for increasing acoustical interference.

Chapter 5

Conclusions and Recommendations

For the assignment to design and realise an audiovisual speech detector two domains were explored. In this research assignment the following tasks were achieved:

- A literature study on mouth feature extraction was done for the video domain and literature study on speech production, detection and recognition was done for the audio domain.
- An available C implementation of a pitch-based speech detector was reprogrammed in Matlab environment.
- An acoustical detector for distinct vowels was designed and programmed in a Matlab environment based on knowledge found in literature.
- An existing C implementation of a face finding algorithm was altered to detect faces in useful scales for the conditions of the assignment.
- An algorithm was programmed to group the detections of the face detection algorithm to unique non overlapping face locations.
- A luminance based mouth localisation algorithm was designed and implemented in a Matlab environment which is based on data retrieved from a faces database and webcam images.
- A secondary colour-based mouth localisation algorithm was designed and implemented in a Matlab environment for the cases when the luminance-based solution fails.
- A robust lip finding algorithm was designed and implemented that is based on colour and luminance information. A colour faces database and webcam images were used for the design.
- A conservative visual non-speech detection algorithm based on the features from the lip finding algorithm was designed and implemented.

- A conservative visual distinct vowel detector based on the features from the lip finding algorithm was designed and implemented.
- The audio and video algorithms were combined in one Matlab framework to form an offline audiovisual speech detector.
- A demonstration tool was programmed to illustrate the performance of the audiovisual speech detector.
- The pitch-based method for speech detection is not robust for a noisy environment.
- The method for acoustical vowel detection is not robust for a noisy environment.
- The A and O group are visually very distinct and detectable with the implemented visual vowel detector.
- The I group is visually confusable with the A group with the used video features.

These tasks resulted in an audiovisual speech detector which outperforms an audio-only speech detector. It can be concluded that the lip finding algorithm is robust and that the visual non-speech detector has a high confidence. It can also be noted that the lip finding algorithm is not a costly algorithm, which makes it suitable for implementation in for instance mobile devices.

The research on visual speech is certainly not finished. Based on the results, the chosen method to robustly detect a few distinct speech properties with basic features appears useful. In this research there is still room for much improvement. The following recommendations are made for future work:

- The audiovisual speech detector should be implemented for real-time operation to get a better view of its performance.
- Confidence values of the audio and video decisions should be used in to the decision-level audiovisual speech detector to increase its accuracy.
- Instead of decision-level modality fusion, fusion of the audio and video features at feature level should be further investigated.
- It is suggested to find a more direct relation between the area of the opening of the mouth and acoustical frequencies. A model in which the mouth is represented as the opening of the last cavity of vocal tract is suggested.
- Further exploitation of the motion of the detected mouth will increase the robustness of the visual (non-)speech detector.
- The visual I, A and O detectors can be grouped to one video-only conservative visual speech detector that detects speaking mouth shapes. Other distinct speaking mouth shapes like distinct consonants can be added to the detector.

- The parameter setting of the visual (non-)speech detection algorithm can be trained for one person which should increase robustness and accuracy. The training can for instance be continuously done when it is detected that the user is in a low-noise environment, which can be detected by an audio noise estimator.
- The number of visual features can be expanded by for instance a teeth and tongue detection. Also the area of the opening of the mouth and protrusion of the mouth has influence on the speech signal.
- Instead of the parabolic mouth matching through the detected edge points of the lips, a 'Hough transform based' mouth shape matching could give more detailed features of the mouth shape (at the cost of robustness).
- Temporal filtering can be applied to reduce outliers in the mouth matching.
- The position of the eyes and nose can be extracted to more accurately determine the locations of the mouth (corners).
- The colour adaptation algorithm can further improve the lip finding algorithm when the problems described at the end of the experiment in section 3.5.2 are addressed.
- To increase the robustness for changing lighting conditions it is suggested that a white-point correction at the camera might be useful in stead of the transformation adaptation described in section 3.5.2.

References

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

[2] "AT&T labs-research," http://www.research.att.com/history/70picture.html.

[3] A. Latour, "Future of video calling starts coming into focus." *The Wall Street Journal Europe*, July 24, 2004, http://pww.research.philips.com/natlab/hvetech/vidtel.htm#Vifo.

[4] H. McGurk and J. MacDonald, "Hearing lips and seeing voices," *Nature*, vol. 264, no. 5588, pp. 746-748, December 1976.

[5] C. Fisher, "Confusions among visually perceived consonants," *Journal on Speech and Hearing Research*, vol. 11, pp. 796–804, 1968.

[6] A. Summerfield, *Some preliminaries to a comprehensive account of audio-visual speech perception.* in: Hearing by eye: The Psychology of Lipreading (eds.) B. Dodd and R. Campbell, Lawrence Erlbaum Press, 1987, pp. 5–31.

[7] M. N. Kaynak, Q. Zhi, A. D. Cheok, K. Sengupta, Z. Jian, and K. C. Chung, "Analysis of lip geometric features for audio-visual speech recognition," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 34, no. 4, pp. 564–570, July 2004.

[8] R. Cutler and L. Davis, "Look who's talking: Speaker detection using video and audio correlation," *IEEE International Conference on Multimedia and Expo*, pp. 1589–1592, 30 July - 2 August 2000.

[9] P. L. Chu, "Voice-activated AGC for teleconferencing," *IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP-96*, pp. 929–932, May 1996.

[10] R. Martin, "Spectral subtraction based on minimum statistics," *EUSIPCO-94*, pp. 1182–1185, September 1994.

[11] G. Peterson and H. Barney, "Control methods used in a study of vowels," *Journal of Acoustical Society of America*, vol. 24, pp. 175–184, March 1952.

[12] A. M. Noll, "Short-time spectrum and cepstrum techniques for vocal-pitch detection," *Journal of the Acoustical Society of America,*, vol. 36, no. 2, pp. 296–302, February 1964.

[13] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *International Journal of Computer Vision*, vol. 1, no. 4, pp. 321–331, 1987.

[14] A. R. Mirhosseini, H. Yan, K.-M. Lam, and C. Chen, "A hierarchical and adaptive deformable model for mouth boundary detection," *Proceedings, International Conference on Image Analysis and Processing (ICIAP97)*, 1997.

[15] M.-H. Yang, D. J. Kriegman, and N. Ahuja, "Detecting faces in images: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 1, pp. 34–58, January 2002.

[16] M. Bartula, "Audio-visually steered video enhancements in videotelephony," Master's thesis, Wroclaw University of Technology, Faculty of Electronics, July 2005.

[17] P. A. Viola and M. J. Jones, "Robust real-time face detection," *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137–154, September 2004.

[18] S. Basu, C. Neti, N. Rajput, A. Senior, L.Subramaniam, and A. Verma, "Audio-visual large vocabulary continuous speech recognition in the broadcast domain," *IEEE 3rd Workshop on Multimedia Signal Processing*, pp. 475–481, September 1999.

[19] G. Fant, *Acoustic Theory of Speech Production.* Mouton, The Hague, 1960, no ISBN.

[20] R. A. Roberts and C. T. Mullis, *Digital Signal Processing.* Addison-Wesley Publishing Company, May 1987, ISBN 0-201-16350-0.

[21] S. Kay, *Modern Spectral Estimation: Theory and Application.* Prentice-Hall, 1988, ISBN 0-13-015159-9.

[22] J. Durbin, "The fitting of time-series models," *Rev. Int. Inst. Statist./International Statistical Review*, vol. 28, no. 3, pp. 233–243, 1960.

[23] RME, Hammerfall DSP Multiface and RME QuadMic microphone amplifier, http://www.rme-audio.com.

[24] L. Lachs, "Research on spoken language processing," *Speech Research Laboratory, Department of Psychology, Indiana University*, no. 23, pp. 81–88, 1999.

[25] M. Reuvers, "Face detection on the INCA+," Master's thesis, University of Amsterdam, Faculty of Science, 2004.

[26] M. Heckmann, K. Kroschel, C. Savariaux, and F. Berthommier, "DCT-based video features for audio-visual speech recognition," *Proc. Int. Conf. Spoken Lang. Process.*, pp. 1925–1928, September 2002.

[27] M. Li, D. Li, N. Dimitrova, and I. Sethi, "Audio-visual talking face detection," *Proceedings. 2003 International Conference on Multimedia and Expo, ICME '03,*, vol. 2, pp. II - 473–476, July 2003.

[28] L. G. D. Silveira and F. L. B. Jacques Facon, "Visual speech recognition: a solution from feature extraction to words classification," *Computer Graphics and Image Processing, 2003. SIBGRAPI 2003. XVI Brazilian Symposium on*, pp. 399–405, 2003.

[29] A. Montgomery and P. Jackson, "Physical characteristics of the lips underlying vowel lipreading performance," *Journal of the Acoustical Society of America*, vol. 73(6), pp. 2134–2144, June 1983.

[30] G. Rabi and S. Lu, "Energy minimization for extracting mouth curves in a facial image," *Proc. of IASTED International Conference on Intelligent Information Systems*, pp. 381–385, December 1997.

[31] R.-L. Hsu, M. Abdel-Mottaleb, and A. Jain, "Face detection in color images," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 24, no. 5, pp. 696–706, May 2002.

[32] A. Martinez and R. Benavente, "The AR face database," *Computer Vision Center, Purdue University, Indiana USA, Technical Report*, no. 24, June 1998.

[33] A.M.Tekalp, *Digital Video Processing*, ser. Signal Processing Series, A. Sullivan, Ed. Prentice Hall, 1995, ISBN 0-13-190075-7.

[34] F. Czerni, "Face tracking for mobile devices." Philips Semiconductors - CoC VCS, Tech. Rep. VCS001, December 2 2002.

[35] G. de Haan, *Video Processing*, 2nd ed. University Press, Eindhoven, 2000, ISBN 90-9014015-8.