Aalto University
School of Science
Degree Programme of Computer Science and Engineering

Subhradeep Kayal

Audiovisual Speaker Clustering for News Broadcast Videos

Master's Thesis
Espoo, May 6, 2015

Supervisor:	Professor Mikko Kurimo, Aalto University
Instructor:	Docent Jorma Laaksonen D.Sc. (Tech.), Aalto University

Aalto University
School of Science
Degree Programme of Computer Science and Engineering

ABSTRACT OF
MASTER'S THESIS

Author:	Subhradeep Kayal		
Title: Audiovisual Speaker Clustering for News Broadcast Videos			
Date:	May 6, 2015	**Pages:**	viii + 47
Professorship:	Speech and Language Processing	**Code:**	S-89
Supervisor:	Professor Mikko Kurimo, Aalto University		
Instructor:	Docent Jorma Laaksonen D.Sc. (Tech.), Aalto University		

With the rapid growth in the amount of multimedia data available on the internet, speaker diarization and person identification have been an important topics of research in the recent years, with many practical applications in the area of automatic labeling and indexing. This can be achieved using solely image based information, speech information, or the fusion of multiple cues such as speech, image, subtitles in the video, etc, which have been known to improve system performance. In this thesis, we perform clustering and diarization experiments on broadcast news video data, provided by the Finnish broadcasting company *YLE* under the *Next Media programme*, financed by *TEKES*, the Finnish Funding Agency for Technology and Innovation.

We first outline some algorithms based on the popular Hierarchical Agglomerative Clustering algorithm and compare there performances using only the image information, in the form of faces extracted, from the videos. Next, a speech based diarization system, based on previous work and the speech recognition software of the Aalto University, is experimented with and evaluated. Finally, some simple combination techniques are documented. As can be seen, the different algorithms have their own strengths and drawbacks, and the efficiency depends largely on the problem to be solved.

Keywords:	Speaker Diarization, Face Clustering, Audiovisual Clustering, Multimodal
Language:	English

Acknowledgements

I started working on what forms the basis for this thesis during early 2013 with a lot of vigor and excitement, hoping to do the best that I can with it and continue towards a PhD, once I finished. Looking back at those times now (as more than 2 years have passed since then), I realize that one cannot possibly comprehend what life has in store. Having been through the highs and the lows of trying to consolidate and complete this piece of work, which I still hold so very close to my heart, I am now filled with a sense of happiness and relief that my thesis will finally see the light of day!

A lot has changed since I migrated to Finland in the autumn of 2011. Through all the changes the faith and support of my parents, and the motivation provided by my friends (Jon, Dan and Markku to say the least) remained the same. To them I'm grateful, as I would have given up a long time ago otherwise. However, I am most indebted to my supervisors Mikko Kurimo and Jorma Laaksonen, whose patience and understanding are things which I value and intend to honor by working hard in whatever I choose to do henceforth. I also owe this to my friend, my wife and my proofreader, Shreya, for pushing me to tie up the loose ends and for staying up through the nights with me. And this section wouldn't be complete without mentioning Antonio, whose example I have thought of in my head when I was unsure of where all this was going, and who kept all of this from being stressful with his lightheartedness!

However it is, good or bad, I have tried to be simple and honest with the thesis and I will always be proud of it.

Thank you all!

Espoo, May 6, 2015

Subhradeep Kayal

Abbreviations and Acronyms

Algorithm Related	
EM	Expectation-Maximization
GMM	Gaussian Mixture Model
HAC	Hierarchical Agglomerative Clustering
VAD	Voice Activity Detection
BIC	Bayesian Information Criterion
Image Related	
HSV	Hue-Saturation-Value
PCA	Principal Components Analysis
LBP	Local Binary Patterns
Speech Related	
MFCC	Mel Frequency Ceptral Coefficients
STFT	Short-time Fourier-Transform
DCT	Discrete Cosine Transform
Results Related	
EP	Estimated Purity
GTP	Ground Truth Purity
DER	Diarization Error Rate
MS	Minimum Speech
MNS	Minimum Non-speech
MWS	Minimum Window Size
WS	Window Step
DWS	Delta Window Size
Misc.	
RT	Rich Transcription
NIST	National Institute of Standards and Technology

Contents

Abbreviations and Acronyms **iv**

1 Introduction **1**

1.1 Setting the Context 1

1.2 Motivation and Scope of the Thesis 2

1.3 Thesis Overview 3

2 Background **5**

2.1 Information Retrieval Research at Aalto University 6

3 Face Clustering **8**

3.1 Face Detection and Filtering 8

3.1.1 Face Detection 8

3.1.2 Filtering of Non-faces 10

3.1.2.1 Skin-colour based Filtering 10

3.1.2.2 Face Eigenspace based Filtering 12

3.2 Feature Extraction and Fusion 13

3.2.1 Gabor Features 14

3.2.2 Local Binary Patterns 15

3.2.3 Feature Fusion 16

3.3 Face Clustering 16

3.3.1 Clustering based on Gaussian Mixture Models and the EM algorithm 17

3.3.2 Hierarchical Agglomerative Clustering (HAC) 19

3.3.3 Modified HAC using Position and Time Information . . 20

3.3.3.1 Temporal Clustering 20

3.3.3.2 Reclustering using Positional Information . . 21

3.3.3.3 Clustering based on HAC 21

4 Speech Clustering **24**
4.1 Feature Extraction . . . 24
4.1.1 Low-level Features . . . 24
4.1.2 Mel Frequency Cepstral Coefficients . . . 25
4.2 Voice Activity Detection . . . 27
4.3 Speaker Segmentation . . . 29
4.3.1 Bayesian Information Criterion . . . 29
4.3.2 Growing Window . . . 30
4.4 Speaker Clustering . . . 31

5 Audiovisual Speaker Diarization **32**
5.1 Initial Segmentation . . . 32
5.1.1 Speaker Segmentation . . . 32
5.1.2 Adding Face Information . . . 32
5.2 Speaker Clustering . . . 33
5.3 Motivation for Suggested Method . . . 35

6 Experiments and Results **36**
6.1 Chapter Overview . . . 36
6.2 Dataset Description . . . 36
6.3 Results of Face Clustering . . . 37
6.3.1 Mechanism of Cluster Analysis . . . 37
6.3.2 Results . . . 38
6.4 Results of Speech Based Diarization . . . 38
6.4.1 Defining the Parameters and the Evaluation Metrics . . . 38
6.4.2 Results . . . 40
6.5 Results of Using Audiovisual Information . . . 40
6.5.1 Results . . . 41

7 Conclusions **42**

List of Tables

6.1 Results of Face Clustering Experiments 38
6.2 Different Parameters of the Diarization System 39
6.3 Best result for the Diarization System 40
6.4 Results of Audiovisual Diarization Experiments 41

List of Figures

3.1 Face detection from a video frame (shown on top). The faces extracted (shown below) are labelled such that they contain information about the frame number and position. 9

3.2 Examples of false positives (non-faces detected as faces). . . . 10

3.3 Distribution of pixels for human skin in the HSV space, averaged over 100 skin-patch images. It can be readily inferred from this figure that hue and saturation properties of the human skin have good clustering properties. 11

3.4 Examples of skin-colour based segmentation. 11

3.5 Mean image. 12

3.6 Typical distances for face and non-face images. 13

3.7 Gabor filter bank for 5 scales (rows) and 8 orientations (columns). 14

3.8 The LBP operator. 16

3.9 HAC algorithm performed on 6 faces 19

3.10 An example of position-information based reclustering. Shown are three consecutive frames, with the faces detected within them in blue and red boxes. The rounded squares show the temporal grouping of the faces on the basis of time and position. 22

5.1 Schematic of the speaker diarization algorithm, with multimodal addition. 33

5.2 Illustration of how the combination works. 34

Chapter 1

Introduction

1.1 Setting the Context

In this section, we go through what Speaker Clustering (or, as is used interchangeably in this thesis, Diarization) is, a brief history of it and how useful it is proving to be with all the applications that it has. A lot of the material in this chapter is influenced by the excellent summarization provided by [3].

Automatic determination of the number of speakers, along with the speaker turns, in a multimedia recording, more formally known as speaker diarization, has evolved to be an increasingly important and dedicated domain of research. While speaker and speech recognition involves recognising a speaker's identity or the transcription of their speech, speaker diarization tries to answer the question : "who spoke when?". This entails the identification of each speaker within an audio stream, without any supervision, and also the intervals of time during which each speaker is active. Being able to solve this problem with high accuracy would benefit a variety of applications related to audio and/or video document processing, such as information retrieval.

In real use cases, audio and/or video recordings often contain more than one person, such as in telephone conversations, broadcast news, debates, shows, movies, meetings, or even lecture or conference recordings. In such cases, it is important to determine the number of speakers automatically, in addition to the periods when each speaker is active. Some examples of the application of speaker diarization include speech and speaker indexing, document content structuring, speaker recognition, speaker attributed speech-to-text, translation and Rich Transcription (RT), a community within which the current state-of-the-art technology has been developed. Research was originally within the telephony domain, and has subsequently spread to the domain of broadcast news and conference meetings. Thus, speaker diarization is an extremely important area of research.

The early work with telephony data, gave way to broadcast news being the main focus of research from the late 1990's onwards, and the use of speaker diarization was aimed at automatic annotation of TV and radio broadcasts. While with the radio transmissions the diarization was done only based on speech features, with the increase in the number of video clips available online, and given the fact that human perception is multimodal in nature, a lot of research has also been focussed on combining visual features (face, lip, gestures are a few) with speech features to improve system performance. These technologies have to meet demands such as indexing, linking and/or summarization of live or archived audio/video transmissions, and this can be done by exploiting several data streams (audio, video and textual information) that are able to capture different kinds of information which complement each other in the analysis of multimedia content. Through speaker diarization, multimedia data can be structured in speaker turns, to which linguistic content and other metadata can be added, for easy browsing or retrieval.

Although this field has gained immense popularity in recent years, it is still relatively nascent when compared to the well established areas of speech and speaker recognition, or face recognition. Thus, there are still very many opportunities for contribution and this thesis is a humble attempt to do the same!

1.2 Motivation and Scope of the Thesis

In a nutshell, this thesis is aimed at providing outlines of two ways of person clustering in audiovisual news broadcast videos, the first using only speech features, and the second utilizing face images, and then experimenting with simple methods of combination of the two. It aims to answer the following broad questions:

- Is Hierarchical Agglomerative Clustering (HAC) [16] a good way to perform clustering with features derived from face images? Can it be improved by adding temporal and positional knowledge derived from the frames of a video?

- How is the performance of HAC for speaker diarization (with MFCC features) with suitable preprocessing (Voice Activity Detection (VAD) [33] and initial segmentation [34])?

- How to combine the above processes into a single HAC framework?

- What are the metrics which can be used to evaluate the aforementioned systems?

In trying to answer the aforementioned questions, we wish to see how each individual modality can be used to perform the clustering task by itself, and whether there is any scope for improvement by simple fusions of the two. Any improvement upon the fusion, at the cost of minimal computational overhead due to it, is a step in the right direction and holds promise for future explorations.

As for the methods of evaluation, they are no less significant than the questions themselves, as whether the results are 'good' or 'bad' largely depends on how they are interpreted. To give an example, a metric which assesses cluster purity (by checking each data point in a cluster) is a correct metric for evaluating the face clusters, whereas for the speaker diarization, a correct metric is one which checks the fraction of speaker time that has not been labelled correctly by the diarization process. Thus, the selection of the evaluation metric completely depends on the final goal of the task at hand.

A good diarization system would be of immense use, in the context of news broadcast videos, as the speaker clusters can be help tag the video, depicting the time intervals when important people are speaking, for ease of the viewers in browsing. While the system needs to have an acceptable performance in both accuracy and speed, it does not need to operate online, meaning that it can be used afterwards on the recorded videos.

1.3 Thesis Overview

Apart from this introduction, the rest of the thesis is arranged as follows:

- Chapter 2 provides a short review of existing literature for speaker diarization and it's applications.

- Chapter 3 contains the description of the methodologies related to the face image based clustering. This means that the process for face image extraction and filtering, and the various clustering algorithms and measures have been defined in this chapter.

- Chapter 4 contains the description of the processes for the speech based diarization, namely the extraction of the MFCC features, the speech/non-speech detector (or the voice activity detector), the initial speaker segmentation step and the final cluster combination process.

- Chapter 5 describes a simple way to integrate the face features with the speech diarization process.
- Chapter 6 provides a brief description of the data used, outlines the metrics needed for evaluating the aforementioned systems and lists the results of the experiments.

Chapter 2

Background

This chapter is aimed at providing a short review of literature relevant to audiovisual speaker recognition techniques and applications.

Early work on audiovisual person recognition was dependent on the fact that the person in the video is speaking and the features extracted from audio and visual domain are associated with each other throughout the video. However, this assumption is invalid for unconstrained videos. Unobtrusive person recognition has been studied extensively using both face [50] and voice-based [41] biometrics. In the video domain, approaches that use both voice and face modalities have been shown to outperform unimodal methods in noisy environments [7], [32]. The principal concept in these approaches is to make use of the modality that is less effected by noise, thereby improving system performance.

In [13], speaker recognition is addressed in the framework of clustering in a generative setting, based on a finite mixture model which explores the idea of non-uniform weighting of observations, and a weighted data mixture model and the associated Expectation-Maximization (EM) procedure is introduced. [18] also proposes clustering in a generative setting, to map the data to a representation of the 3D scene-space, but uses binaural and binocular sensors to gather both auditory and visual observations.

[28] outlines a Bayesian approach to speaker identification, where the temporal sequence of audio and visual observations obtained from speech and shape of the mouth are modeled using a set of coupled Hidden Markov Models (HMM). The likelihood obtained from the CHMM is combined with that obtained from an Embedded HMM which models the face features. The likelihoods obtained are combined using some weights.

[25] outlines different integration schemes as to when the features should be fused, namely, Early integration (EI), in which acoustic and visual speech stimuli are synchronized and merged in some manner for joint learning and

classification, Middle integration (MI), which attempts to learn and classify acoustic and visual speech cues independently, and Late integration (LI), which assumes complete independence between the acoustic and visual modalities.

Besides the above, [32] and [8] provide excellent overviews of audiovisual speech recognition.

While the aforementioned literature is aimed at introducing and outlining various novel methodologies for speaker identification using audiovisual information, there is also significant literature dedicated to suggesting innovative applications of the same.

[35] proposes a system which segments the face and the audio tracks separately, and then combines them to find consistent one-to-one associations between the face tracks and the speaker segments, for the purpose of celebrity recognition in unconstrained web videos.

Speaker identification has also been used for movie content analysis. An interesting method is provided in [24] wherein a likelihood-based approach is first applied for speaker identification using pure speech data, and techniques such as face detection/recognition and mouth tracking are applied for talking face recognition using pure visual data, which are then integrated under a probabilistic framework.

[27] talks about how audio-based speaker identification degrades severely when there is a mismatch between training and test conditions and investigates various techniques to fuse video based speaker identification with audio-based speaker identification; specifically, the techniques to optimally determine the relative weights of the independent decisions based on audio and video, to achieve the best combination.

As can be understood, the field of speaker identification has generated a lot of interest amongst researchers in the past years, and there is still a lot of scope for novel approaches.

2.1 Information Retrieval Research at Aalto University

While speaker clustering research was set in motion with the work done in [30], information retrieval has long been studied at Aalto University, with the Self-Organizing Map (SOM) [19] being heavily used in the beginnings in applications concerning spoken documents [21], images [22] and videos [38].

More recently, a lot of work has been done on the image based retrieval front, such as improving it using gaze and speech [49] and text [39], in video retrieval [20] and in speech based retrieval, such as by using morphs for retrieval [46]. Repeated participation and success in the TRECVID[1] evaluations [15], [2], as well as the conceptualization of the Morpho Challenge [48], for morpheme discovery, are only a few examples of Aalto University's dedication to information retrieval research.

[1] The TREC conference series is sponsored by the National Institute of Standards and Technology (NIST) with additional support from other U.S. government agencies. The goal of the conference series is to encourage research in information retrieval by providing a large test collection, uniform scoring procedures, and a forum for organizations interested in comparing their results. In 2001 and 2002 the TREC series sponsored a video ”track” devoted to research in automatic segmentation, indexing, and content-based retrieval of digital video. Beginning in 2003, this track became an independent evaluation (TRECVID) with a workshop taking place just before TREC.

Chapter 3

Face Clustering

3.1 Face Detection and Filtering

3.1.1 Face Detection

The task of detecting faces from a video essentially translates to detecting the faces present in, ideally, every frame of the video. Taking into account that most videos have a frame rate of 24 frames per second or more, detecting faces from every frame is a time-consuming task. For news broadcasts and similarly 'paced' videos, which almost surely do not contain rapid scene changes within them, detecting faces from every frame is unnecessary. Thus, for such types of videos, faces may be detected from frames which are extracted at a much slower rate, as compared to the frame rate of the video, without losing any significant amount of information. This extraction rate has to be optimal, as a high extraction rate will result in the increase in processing time, whereas a low frame extraction rate will fail to capture all of the information available in the video.

Once a frame has been captured, face-like regions are detected by running the Viola-Jones detector [47] on it and keeping only those regions which are reasonably sized. The argument for discarding small face areas, in this work, is that the dataset consists of news videos, wherein important people, whose faces should be effectively detected and clustered, will have close-up shots and, hence, reasonably large face regions.

The face-like regions are saved with a label which contains two valuable pieces of information: the time at which the face occurred in the video and the positional coordinates of the bounding box which contains the face. This information, in conjunction with the features extracted, will be used to enhance the efficiency of the clustering process. An example of the above is provided and explained in the following figure.

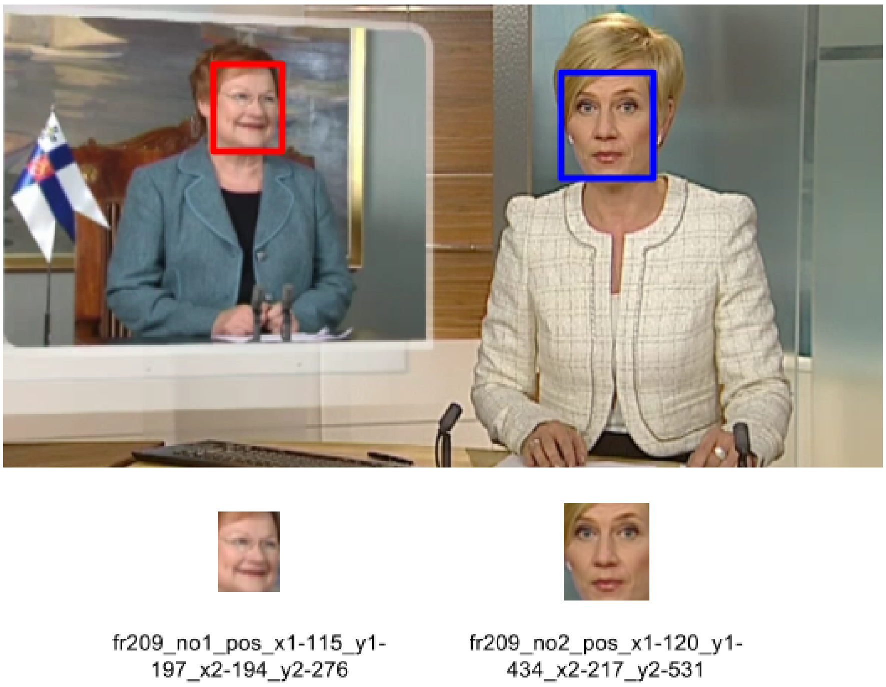

fr209_no1_pos_x1-115_y1-197_x2-194_y2-276
fr209_no2_pos_x1-120_y1-434_x2-217_y2-531

Figure 3.1: Face detection from a video frame (shown on top). The faces extracted (shown below) are labelled such that they contain information about the frame number and position.

Figure 3.1 shows a frame containing two faces (shown in red and blue bounding boxes), which are extracted and suitably labelled. The label is of the form:

fr<*frame_number*>_no<*face_number*>_pos<*bounding-box_coordinates*>

where the bounding-box coordinates are formed by top-left and bottom-right coordinates of the face bounding-box. Taking one label provided in Figure 3.1 and explaining, the label for the newsreader's face reads 'fr209_no2_pos_x1-120_y1-434_x2-217_y2-531', which implies that the face:

- has been extracted from frame number 209,
- is the second face in the scene, as detected by the Viola-Jones detector,
- is bounded by a box with coordinates (120,434) for the top-left corner and (217,531) for the bottom-right.

3.1.2 Filtering of Non-faces

Amongst all the face-like regions extracted from a video, some are erroneously detected non-face regions (Figure 3.2). To reject these regions and perform a clean-up, two filtering steps are performed. These two steps together form an accurate filter to remove non-face regions. Feature extraction and clustering is performed only on those images which pass through both of these steps and, thus, get validated as a true face region.

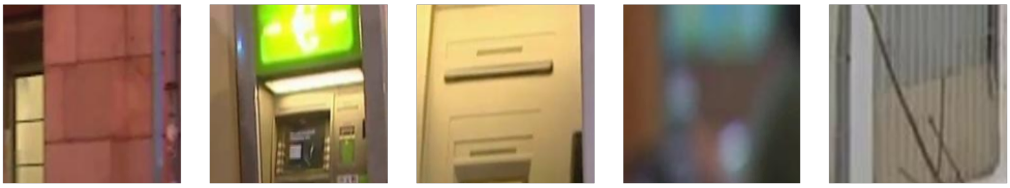


Figure 3.2: Examples of false positives (non-faces detected as faces).

3.1.2.1 Skin-colour based Filtering

The first step is based on the observation that human skin colour can be used to distinguish between face and non-face regions. It has been found that the chrominance component of human skin has good clustering properties [12], especially in the HSV and YCbCr colour-spaces.

To elaborate, the HSV colour space is the representation of the RGB colour space in the cylindrical coordinate system, where the 'chroma' (colour) and the 'luma' (light) components are distinguishable. 'HSV' stands for Hue, Saturation and Value, which are calculated as:

$$H = \arccos \frac{0.5(2R - G - B)}{\sqrt{(R - G)^2 - (R - B)(G - B)}} \tag{3.1}$$

$$S = \frac{\max(R, G, B) - \min(R, G, B)}{\min(R, G, B)} \tag{3.2}$$

$$V = \max(R, G, B). \tag{3.3}$$

The Hue component encodes the colour information or the 'chroma', whereas the Saturation component represents the degree of colour purity. The Value is distinct from both Hue and Saturation, and stores information about the 'luma' or lighting. The Hue component of the skin-colour has excellent clustering properties, as seen in Figure 3.3.

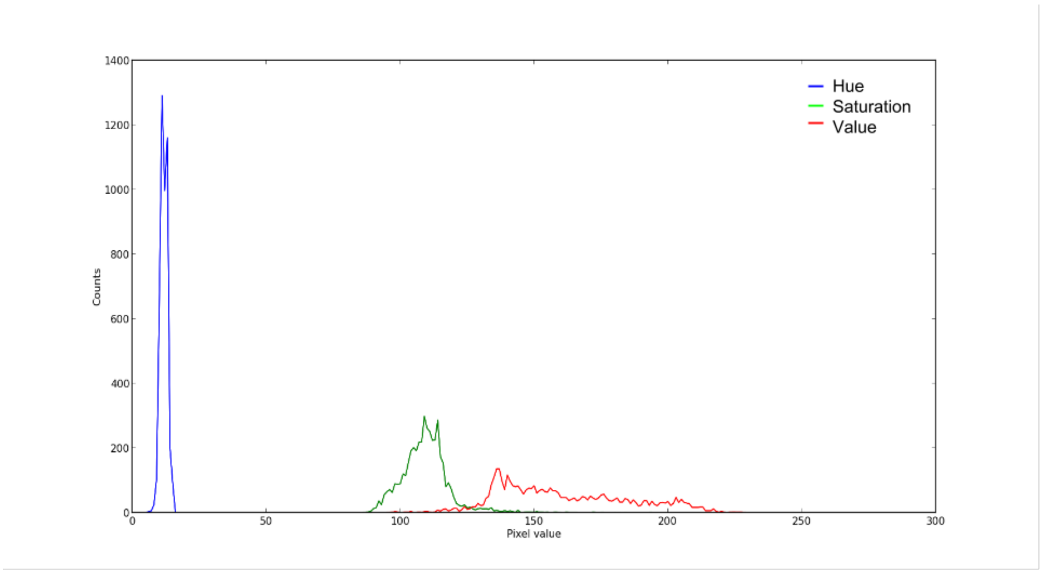


Figure 3.3: Distribution of pixels for human skin in the HSV space, averaged over 100 skin-patch images. It can be readily inferred from this figure that hue and saturation properties of the human skin have good clustering properties.

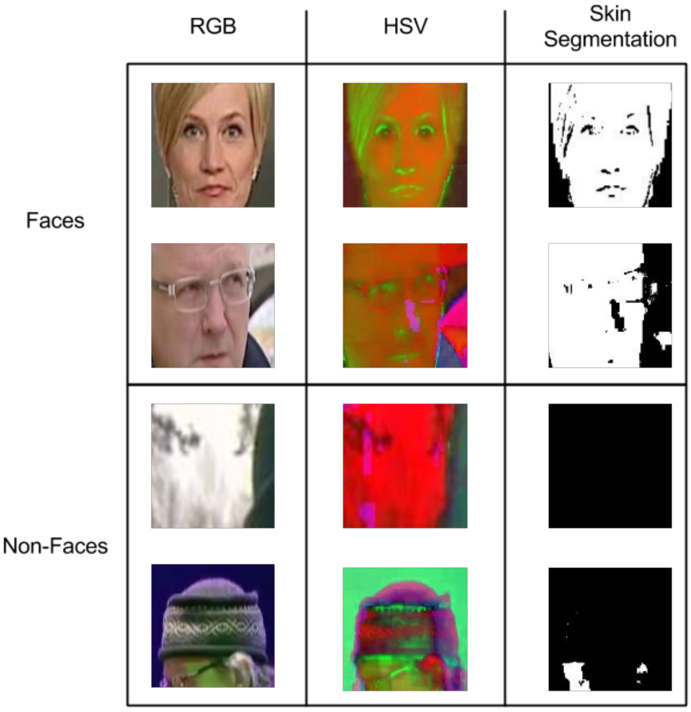


Figure 3.4: Examples of skin-colour based segmentation.

Using this property, each pixel in the image can be classified as a skin pixel or a non-skin pixel, depending upon whether the HSV values of the pixel are within the range depicted in Figure 3.3, where the hue value is between 5 and 25, and the saturation is between 90 and 150. After the pixels have been labelled such, the fraction of the total pixels which qualify as skin pixels is counted. If this fraction is not substantial, and less than a predetermined threshold, the region is discarded as a non-face region. A good threshold has been found to be 0.8, i.e., at least 80% of the region must comprise skin pixels for it to be a face region. Some examples of skin colour based thresholding are given in Figure 3.4.

3.1.2.2 Face Eigenspace based Filtering

The second step uses a simple PCA-based filtering algorithm [45] to get rid of the remaining non-face regions which might pass through step 1. Here, a set of example face images is chosen and projected to the 'face-space' using PCA. Then, a region is accepted as a face if the distance between its projection on the 'face-space' and itself is less than a threshold.

A subset is made from the faces extracted from the news broadcast videos, for training, and PCA is performed on them, after mean-adjustment (Figure 3.5 shows the mean image). Then the largest eigenvectors, which explain most of the variance, are kept. Let the mean face be denoted as F_{mean} and the i_{th} eigenvector as E_i.

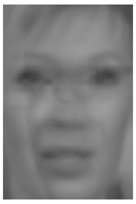


Figure 3.5: Mean image.

A new face-like image, F_i is projected onto the face-space, so that:

$$F_{i_p} = \sum_{k=1}^{N} w_k E_k + F_{mean} \tag{3.4}$$

$$w_k = E_k^T (F_i - F_{mean}), \tag{3.5}$$

where, N is the number of eigenvectors. In other words, the new face-like image is mean adjusted, and reconstructed using the eigenvectors of the training set. The distance between the reconstructed image and the original gives a measure of the 'faceness' of the original:

$$D = \|F_i - F_{i_p}\|. \tag{3.6}$$

If this distance is less than a threshold, then F_i is classified as a face, otherwise discarded as a non-face. Figure 3.6 shows some typical distances.

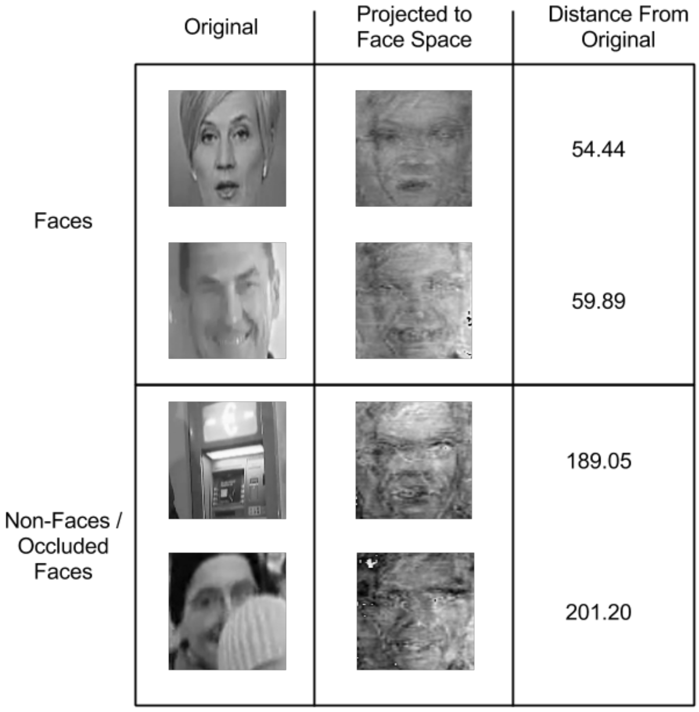


Figure 3.6: Typical distances for face and non-face images.

This method also gets rid of faces with occlusions. In the context of news broadcast videos, wherein important people can be assumed to have clear frontal or, at least, partially frontal views of their faces, faces with occlusions can be disregarded as unimportant. These occluded face images, if present during the clustering process, will act as noise and result in a lower clustering efficiency.

3.2 Feature Extraction and Fusion

Gabor wavelets and Local Binary Patterns (LBP) have been two of the most successful local appearance descriptors for face recognition. While LBP captures small scale details, the Gabor features encode facial shape and appearance information over a range of coarse scales. Both of these descriptors are rich in information and hence chosen for experimentation. Their complementary nature also makes them suitable for feature fusion.

3.2.1 Gabor Features

The Gabor filter is a linear filter which is similar to the human visual system in its ability to capture information about spatial orientation and signal frequency. It is an excellent way to isolate and assess texture features. It behaves like a band pass filter for the local spatial frequency distribution in the texture. Since the pioneering work in [23], it has been widely used in face recognition. The Gabor features are the result of convolving each image with a bank of Gabor filters at various scales and orientations, and then taking the absolute values of the results (also known as the 'energy image').

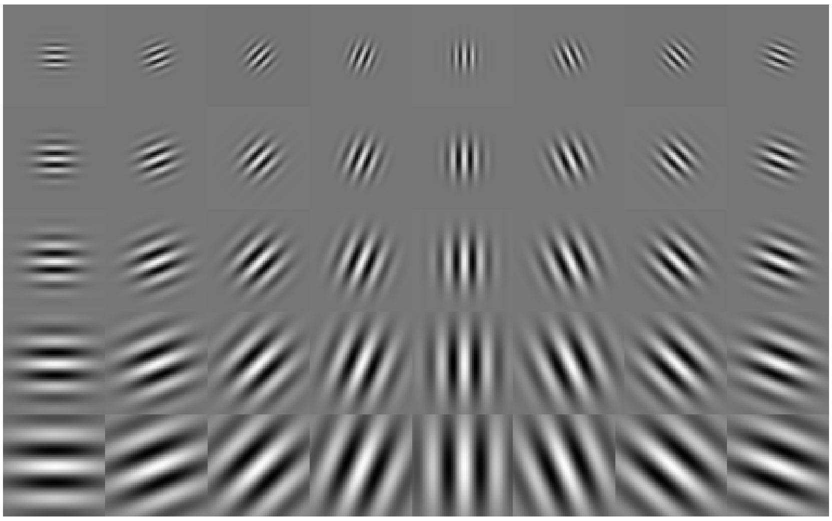


Figure 3.7: Gabor filter bank for 5 scales (rows) and 8 orientations (columns).

Mathematically, the 2D Gabor filter is a Gaussian kernel modulated by a sinusoidal plane wave [9]. There are many representations of the filters, the most common, in context of face recognition, being:

$$\phi_{f,\theta,\gamma,\eta}(x,y) = \frac{f^2}{\pi\gamma\eta} e^{-(\alpha^2 x'^2 + \beta^2 y'^2)} e^{j2\pi f x'}, \tag{3.7}$$

where

$$x' = x\cos\theta + y\sin\theta \text{ and } y' = -x\sin\theta + y\cos\theta.$$

Here, f is the central frequency of the sinusoidal plane wave, θ is the anti-clockwise rotation of the Gaussian and the plane wave, and α, β control the sharpness of the Gaussian along its axes. γ and η keep the ratio between the frequency and sharpness as constant, where

$$\gamma = \frac{f}{\alpha} \text{ and } \eta = \frac{f}{\beta}.$$

Usually, a bank of filters, at several scales and frequencies, is used, instead of a single filter, to most effectively capture all the available information contained in a facial image. The most suitable filter bank, as past studies suggest [42], uses 5 scales and 8 orientations (Figure 3.7).

Then, the Gabor feature for each image is the result of convolution of 40 Gabor filters with the image. Each such 2D output of convolution is concatenated row-wise to yield 40 different row vectors, which are then concatenated to form one single high-dimensional feature vector for each image. While taking the result of convolution, which is a complex quantity, the absolute values are considered.

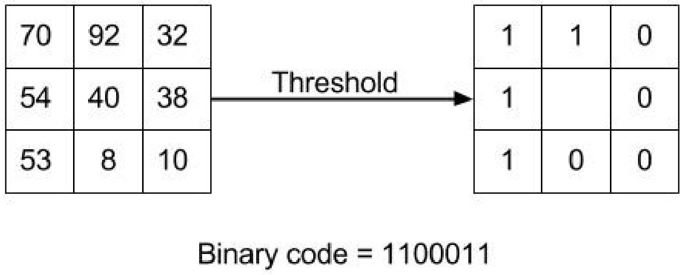
3.2.2 Local Binary Patterns

Local Binary Patterns (LBP) [1][29] are simple texture features which label each pixel of an image by thresholding its neighbourhood. The size and shape of the neighbourhood to be operated upon can be varied. Apart from the computational simplicity with which it can be calculated, LBP features are quite ideal for our experiments as they are invariant to rotation and robust against illumination and contrast changes. Mathematically, the LBP operator has been described as:

$$LBP(x_c, y_c) = \sum_{n=0}^{7} 2^n s(i_n - i_c). \tag{3.8}$$

In this case, n runs over the 8-neighbourhood of a pixel c. i_c and i_n are the pixel values at c and n, and $s(v)$ is 1 if $v \geq 0$ and 0 otherwise. This is illustrated in Figure 3.8.

These LBP operators are the basis of the face recognition method in [1] where the face image is divided into regular grid, and histograms are made from the LBP within each cell. Then, all these histograms are concatenated to form a global high-dimensional descriptor vector for each image. This more accurately captures the local properties of the image and makes the LBP suitable for face recognition.



70	92	32
54	40	38
53	8	10

1	1	0
1		0
1	0	0

Figure 3.8: The LBP operator.

3.2.3 Feature Fusion

Following the work in [43], which elicits the promise of combining two complementary and heterogeneous feature sets, such as the LBP and the Gabor features, in boosting the recognition rates, this paper also experiments with fused features of the same type. As in [43], PCA is performed on the feature sets (LBP and Gabor) and 'significantly non-zero' eigenvalues are kept when reconstructing the data in a lower dimension. Then the each of the PCA-reduced feature set is variance normalized and concatenated to form the fused feature vector.

If x_1, x_2 are the different feature sets, U_1, U_2 are the matrices with the significant eigenvectors of x_1, x_2 as columns, and μ_1, μ_2 are the means of x_1, x_2, then the PCA projected data is given by:

$$y_1 = U_1^T(x_1 - \mu_1) \text{ and } y_2 = U_2^T(x_2 - \mu_2). \tag{3.9}$$

The combination is given by the z-score normalized concatenation:

$$z = (\frac{y_1}{\sigma_1}, \frac{y_2}{\sigma_2}), \tag{3.10}$$

where σ_1, σ_2 are the standard deviations of y_1, y_2.

3.3 Face Clustering

Two commonly used methods of clustering are inspected and compared to a variant, which incorporates positional and temporal information to enhance the clustering efficiency.

3.3.1 Clustering based on Gaussian Mixture Models and the EM algorithm

Gaussian Mixture Models (GMM) are mixtures of two or more Gaussians, which are linearly superposed. If suitably tuned, GMMs can model any data distribution to an arbitrary degree of accuracy. A K-component GMM is given by:

$$p(x) = \sum_{k=1}^{K} \pi_k N(x|\mu_k, \Sigma_k), \tag{3.11}$$

where the constituent Gaussians have their own mean and covariance, mixed with a degree π_k known as the mixing coefficient, such that,

$$\sum_{k=1}^{K} \pi_k = 1, \;\; 0 \leq \pi_k \leq 1. \tag{3.12}$$

Before performing the Expectation-Maximization (EM) algorithm to fit the GMM to the training data, it is important to define a few more variables. A K-dimensional binary random variable z is defined, having a 1-of-K representation, in which a particular element is 1 and all others are 0. Then,

$$p(z_k = 1) = \pi_k. \tag{3.13}$$

Since z uses a 1-of-K representation, it can be written as:

$$p(z) = \prod_{k=1}^{K} \pi_k^{z_k}. \tag{3.14}$$

Similarly:

$$p(x|z) = \prod_{k=1}^{K} N(x|\mu_k, \Sigma_k)^{z_k}. \tag{3.15}$$

Having defined equations (3.14) and (3.15), another quantity, known as 'responsibility', is now defined, which is conditional probability of z given x. Using the Bayes' theorem:

$$p(z_k = 1|x) = \gamma(z_k) = \frac{p(z_k = 1)p(x|z_k = 1)}{\sum\limits_{j=1}^{K} p(z_j = 1)p(x|z_j = 1)}. \tag{3.16}$$

Putting the value of equations (3.14) and (3.15) in equation (3.16), and writing the equation to represent the value of 'responsibility' at each data point x_n:

$$\gamma(z_{nk}) = \frac{\pi_k N(x_n|\mu_k, \Sigma_k)}{\sum\limits_{j=1}^{K} \pi_j N(x_n|\mu_j, \Sigma_j)}. \tag{3.17}$$

Now, it is desired to find the values of π, μ and Σ to 'fit' the GMM to the training data, thereby making it a problem of parametric density estimation. An easy way to do it is to maximize the log-likelihood of these parameters, given the data, and this is what the EM algorithm does. The log-likelihood is given by:

$$L(\pi, \mu, \Sigma|X) = \sum_{n=1}^{N} \log\{\sum_{k=1}^{K} \pi_k N(x_n|\mu_k, \Sigma_k)\}, \tag{3.18}$$

where K is the number of mixture components and N is the number of data points.

The derivation of the EM algorithm is straightforward and proceeds by differentiating equation (3.18) with respect to π, μ, Σ, setting the results to zero, and solving the resultant equations (see [5] for more details). The outline of the algorithm is as follows:

1. Initialize values of mixing coefficients, means and covariances.

2. E-step: Evaluate responsibilities using equation (3.17) and the current parameter values.

3. M-step: Re-estimate the parameter values using equations (3.19), (3.20) and (3.21):

$$\mu_k^{new} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk}) x_n \tag{3.19}$$

$$\Sigma_k^{new} = \frac{1}{N_k} \sum_{n=1}^{N} \gamma(z_{nk})(x_n - \mu_k^{new})(x_n - \mu_k^{new})^T \tag{3.20}$$

$$\pi_k^{new} = \frac{N_k}{N}, \tag{3.21}$$

   where N_k is the number of points belonging to the k^{th} component.

4. Evaluate log-likelihood using equation (3.18) and check for convergence. If not converged, return to step 2.

After convergence of the EM algorithm, each data point is assigned to a cluster from 1 to K, depending on the probability of the point belonging to that particular component Gaussian.

3.3.2 Hierarchical Agglomerative Clustering (HAC)

Hierarchical Clustering [16], simply put, tries to create a hierarchy of clusters. There are two approaches which can be used, a top-down approach (Divisive Clustering) and a bottom-up approach (Agglomerative Clustering). Agglomerative clustering works by placing each point in its own cluster and combining clusters, according to some similarity metric, until there is one cluster left. Thus, it creates a cluster-tree, with different levels representing different clustering outcomes. The divisive approach is the opposite, meaning that it begins with all the points in one cluster, and proceeds by gradually dividing the cluster into smaller clusters. The agglomerative approach is more widely used because it is easier to understand and implement.

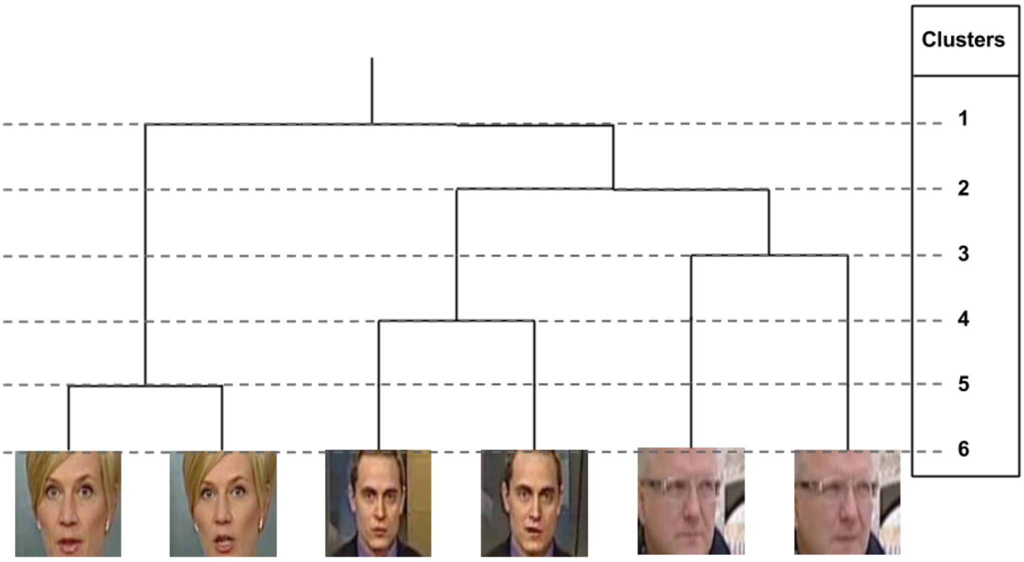


Figure 3.9: HAC algorithm performed on 6 faces

In Hierarchical Agglomerative Clustering (HAC), clusters are combined based on some similarity measure. Three commonly used similarities are single-link (SL), complete-link (CL) and average-link (AL) similarities, described in the following equations:

$$SL(i,j) = \min\{d(a,b) \forall a \in C_i, b \in C_j\} \tag{3.22}$$

$$CL(i,j) = \max\{d(a,b) \forall a \in C_i, b \in C_j\} \tag{3.23}$$

$$AL(i,j) = \frac{1}{|C_i||C_j|} \sum_{a \in C_i} \sum_{b \in C_j} d(a,b), \tag{3.24}$$

where $d(a, b)$ is the distance metric between points a and b, such that point a belongs to cluster C_i, and point b belongs to cluster C_j. $|C_i|$ refers to the size of cluster C_i.

The process of HAC can be outlined as:

1. Calculate the distance matrix D_s for the data points in the dataset, such that $D_s(i, j)$ is the distance between the i^{th} and j^{th} points.

2. Initialize the algorithm such that all the datapoints represent a singleton cluster.

3. Calculate similarity between clusters using any suitable similarity metric (see equations (3.22), (3.23) and (3.24)).

4. Combine the nearest pair of clusters.

5. Continue till one cluster remains.

As can be seen from Figure 3.9, HAC forms a cluster tree, whose different labels represent different clustering outcomes. The required number of clusters can be selected by choosing the appropriate level on the clustering tree.

3.3.3 Modified HAC using Position and Time Information

The following section presents modifications to the work done in [17], such that the efficiency of clustering of the HAC algorithm is improved.

3.3.3.1 Temporal Clustering

The temporal clustering in [17] proceeds via selection of a window parameter (w), which is the maximum allowable temporal difference between two faces for them to be in the same cluster. This step is modified entirely to avoid the heuristic selection of the parameter w.

The algorithm for organizing the face regions into clusters, by their temporal similarity, automatically detects 'jumps' in the sequence of the time-stamps by simple peak detection. It is stated as follows:

1. Sort the unique time-stamps in ascending order of the time-stamps.

2. Detect the 'jumps' in this sequence by detecting the peaks of the first difference. This can be done by detecting the non-zero second differences of the sequence of temporal labels. Let the corresponding temporal labels for the detected peaks be $T = \{t_1, t_2, \dots, t_n\}$.

3. Partition the face-samples according to their time-stamps and according to T. This means that all the faces with time-stamps between 0 and t_1 are in cluster 1, from t_1 to t_2 are in cluster 2, and so on.

3.3.3.2 Reclustering using Positional Information

This step accounts for the fact that every frame in the video may have the presence of two or more faces, at different spatial coordinates in the frame. Hence, every temporal cluster will also contain face-samples of two or more individuals, whose spatial coordinates, or positional label, are different. This information is used in order to further enhance the initialization mechanism to the HAC algorithm.

The algorithm for reclustering based on positional information uses Gaussian Mixture Models (GMMs), to model the clusters, whose parameters are inferred by the Expectation-Maximization algorithm. The problem of preselecting the number of clusters is solved by using the model with the lowest BIC (Bayesian Information Criterion) score [37].

The algorithm is stated as:

1. For a particular temporal cluster, read the positional labels (X-Y coordinates of the center of the face bounding box) for the face-samples in it.

2. Fit GMMs, with components varying from 1 to max, to the position data. The value of max is the upper limit for the number of faces that may be present in the frame. A good guess for max is:

$$max = \frac{\text{Area of frame}}{\text{Area of smallest face sample}}.$$

3. Select the optimal components for the GMM to be that for which the BIC score is minimum, and recluster according to it.

4. Repeat for all temporal clusters.

A pictorial example of the algorithm is shown in Figure 3.10.

3.3.3.3 Clustering based on HAC

The clusters which are formed thus are the seed clusters for the HAC algorithm. Apart from traditional HAC, these seed clusters are also used with the modified HAC algorithm (THAC) described in [17]. A brief description of the THAC algorithm is as follows:

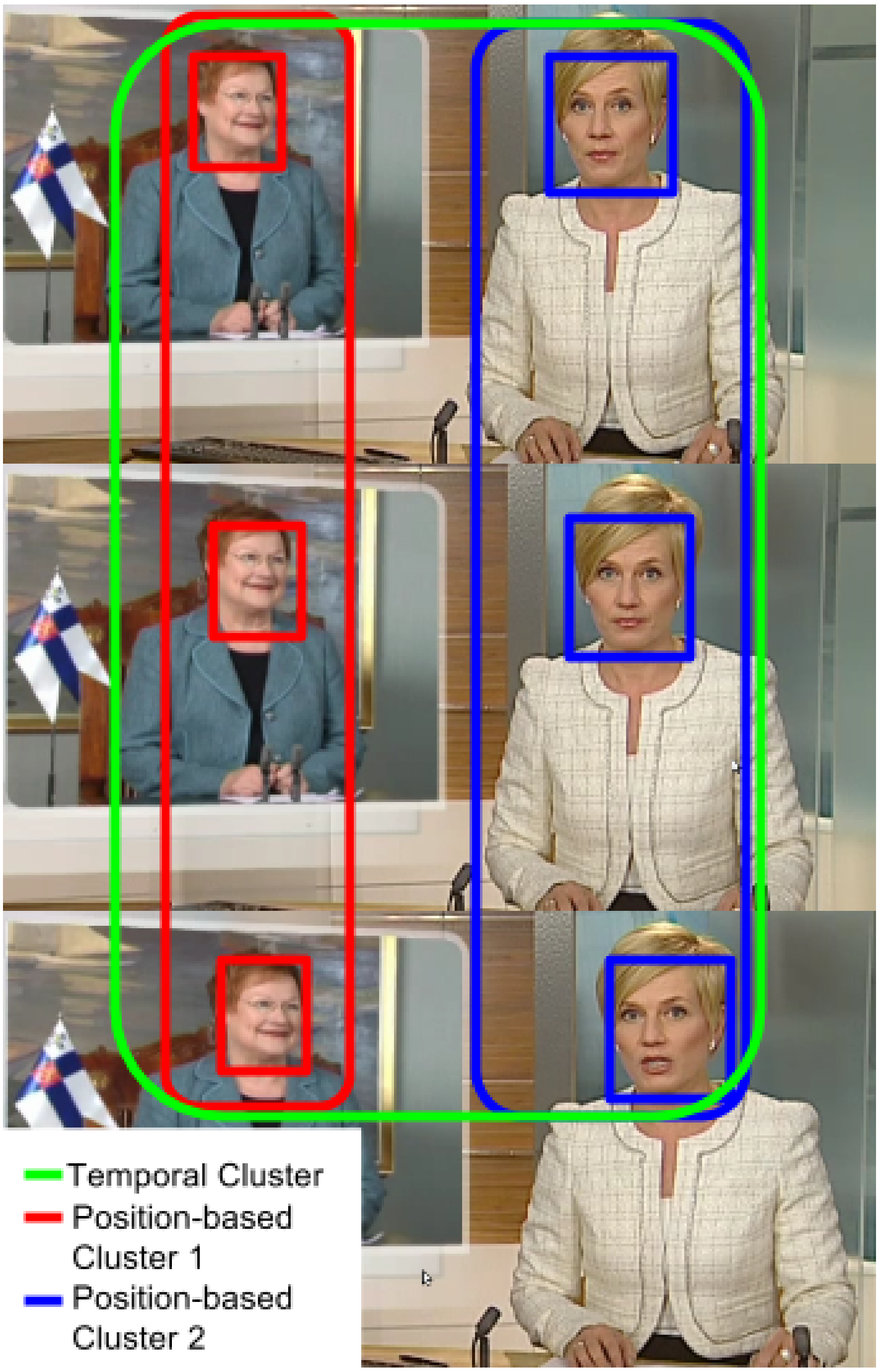


Figure 3.10: An example of position-information based reclustering. Shown are three consecutive frames, with the faces detected within them in blue and red boxes. The rounded squares show the temporal grouping of the faces on the basis of time and position.

1. Calculate the feature distance matrix D_s using the combined feature vectors of the face-samples and, additionally, the temporal distance matrix, D_t, given by:
$$D_t(i, j) = |t_i - t_j|, \tag{3.25}$$
where t_i and t_j are time-stamps.

2. Weight D_t from (3.25) with an exponential decay term and construct the matrix D such that:
$$D(i, j, it) = D_s(i, j) + D_t(i, j) * \exp(-it/N) \tag{3.26}$$
where, it is the present iteration number and N is the total number of samples.

3. Use D as the distance matrix for the HAC algorithm, updating it with every iteration.

Having listed the necessary algorithms, results of the experiments of face clustering, using the aforementioned features and algorithms, can be found in Chapter 6.

Chapter 4

Speech Clustering

4.1 Feature Extraction

4.1.1 Low-level Features

The feature extraction step converts the raw audio data into a numerical dataset with instances and their respective features. These features are extracted from various properties of the audio signal, using several signal processing techniques. They are called 'instantaneous features', since they are extracted from short frames of the audio signal and produce values for each such frame.

[31] lists the following categories of low-level features:

1. Temporal shape – Features that are computed from the signal waveform, e.g., effective duration, attack time.

2. Temporal features – These are computed from the statistical properties of the signal, e.g., auto-correlation.

3. Energy features – Features which reflect the energy content of the signal, e.g., global energy, harmonic energy.

4. Spectral shape features – Features computed from the Short-time Fourier-Transform of the signal, e.g., spectral rolloff, spectrogram, Mel frequency ceptral coefficients (MFCC).

5. Harmonic features – Features which are computed from the sinusoidal harmonic modelling of the signal, e.g., harmonic noise ratio.

6. Perceptual features – Features computed using a model of human hearing or perception, e.g., loudness, sharpness, spread.

Speaker diarization falls into the category of speaker-based processing techniques, and a desirable representation of the input signal would be one which retains enough information to optimally separate the speech, of different speakers in the conversation, into homogeneous segments. Similar to speaker and speech recognition systems, speaker diarization may also use features such as Mel frequency cepstral coefficients (MFCC), linear frequency cepstral coefficients (LFCC), perceptual linear predictors (PLP), linear predictive coding (LPC) amongst others. The MFCCs have been found to be particularly useful in speech recognition [10], and have been known to yield good performance for speaker diarization [4].

4.1.2 Mel Frequency Cepstral Coefficients

The first step in any speech recognition system is to extract features which are good for identifying the linguistic content and discarding all the other information, such as background noise, emotion etc.

An important point to understand is that the sounds generated by a human are filtered by the shape of the vocal tract and it is this shape that determines the nature of the emanating sound. If this shape can determined accurately, it would then correspond to an accurate representation of the phoneme being produced. The shape of the vocal tract manifests itself in the envelope of the short time power spectrum, and the Mel frequency cepstral coefficients (MFCCs) accurately represent this envelope. Since the MFCCs are central to the process of speech clustering, a complete description of the steps to be performed to calculate these features is provided next.

The first step in calculating the MFCCs is to split the input signal into overlapping blocks, or 'frames', of equal length, by choosing a fixed window time duration. If the sampling rate is f_s and the window duration is T_w, the total number of samples in each window is calculated by $N_w = f_s T_w$. The number of overlapping samples is based on the choice of the 'hop-size' h. The input signal, thus, gets split into N_o overlapping frames:

$$N_o = \left\lfloor \frac{N - N_w}{h} \right\rfloor + 1, \tag{4.1}$$

where N is the total number of samples in the input signal. Denoting the discrete-time input signal by $x(t)$, the n^{th} frame, x_n can be written as an

N_w-dimensional vector:

$$x_n = \begin{bmatrix} x[(n-1)h+1] \\ x[(n-1)h+2] \\ . \\ . \\ x[(n-1)h+N_w] \end{bmatrix}. \tag{4.2}$$

In order to prevent spectral leakage, each such frame x_n is multiplied by a Hamming window function [14]:

$$w(t) = 0.54 - 0.46\cos\left(\frac{2\pi t}{N_w - 1}\right), \tag{4.3}$$

followed by the application of the Short-time Fourier-Transform (STFT) on the product, giving:

$$S_n(k) = \sum_{t=0}^{N_w-1} x_n(t)w(t)e^{-j2\pi kt/N_w} \quad 1 \leq k \leq K, \tag{4.4}$$

where K denotes the length of the STFT. The STFT produces complex numbers, the magnitudes of which (power spectrum) are kept for further processing.

The next important step is to map the power spectrum thus obtained onto the 'mel' scale. The mel scale is based on human hearing, motivated by the fact that humans do not hear loudness on a linear scale. The mel scale value for a frequency f is given by:

$$mel(f) = 1127\log\left(1 + \frac{f}{700}\right). \tag{4.5}$$

An overlapping set of n_t triangular filters are created, so that the maximum weight for each filter reduces with increase in frequency. The mel transformation matrix is made, such that:

$$M(i,j) = \begin{cases} \frac{2}{f(i_h)-f(i_l)}\frac{k-f(i_l)}{f(i_m)-f(i_l)}, & \text{if } f(i_l)<k<f(i_m) \\ \frac{2}{f(i_h)-f(i_l)}\frac{f(i_h)-k}{f(i_h)-f(i_m)}, & \text{if } f(i_m)<k<f(i_h) \\ 0, & \text{otherwise} \end{cases} \tag{4.6}$$

where $f(i_l)$, $f(i_m)$ and $f(i_h)$ denote, respectively, the lower, middle and highest frequency of each mel filter bank. The dimensionality of M is then n_t x K.

After the filter bank has been constructed, the spectrogram is then mapped to the log mel-scale as follows:

$$X = \log(M|S|^T). \tag{4.7}$$

The final step is to perform a Discrete Cosine Transform (DCT) on the mel spectrogram. Doing so decorrelates the energies of the overlapping filterbanks, which are otherwise correlated. Only the first 10 - 16 coefficients are kept, for they describe the spectral envelope, which in turn characterises vocal tract shape [10]. In addition to the MFCC, logarithmic speech signal power is also used as a feature.

The features outlined above do not have any temporal information. In order to incorporate the changes over multiple frames, the first and second time derivatives are added to the basic feature vector. The so-called 'Delta Coefficients' are calculated by:

$$\Delta_n = \frac{\sum\limits_{i=1}^{L} i(c_{n+i} - c_{n-i})}{2\sum\limits_{i=1}^{L} i^2}, \tag{4.8}$$

where c_n is the MFCC for the n^{th} frame. The second differentials or the delta-deltas are calculated by replacing c_n by Δ_n in equation (4.8).

Calculating these Delta coefficients and appending them to the original MFCC feature vector makes these features even more efficient in modelling speech.

4.2 Voice Activity Detection

A costly drawback affecting a lot of speech processing systems is the presence of environmental noise and it's harmful effect on the system performance. Voice Activity Detection (or VAD) is the process of detecting those parts of the audio stream which contain voice, and discarding those parts which are non-speech. Removing the non-speech parts from the audio signal considerably speeds up the latter tasks and also increases their accuracy. However, mistakes in the VAD could remove audio segments with speech in them, and cause errors later.

In the context of broadcast news, non-speech can come from several sources, starting to from music, to background noise of all kinds when doing outside interviews. Thus, a voice activity detection system based on energy levels with a threshold would not work properly, since these levels are variable.

Following the work done in [30], in this thesis, a two-class model based system was used, with one model trained on speech, and another trained on crowd noise. An effective decision rule [40], which can be used, is based on a statistical likelihood ratio test. Given an observation to be classified, the problem is reduced to selecting the class with the largest posterior probability:

$$H = \begin{cases} H_1, & \text{if } P(H_1|x) > P(H_0|x) \\ H_0, & \text{otherwise} \end{cases} \tag{4.9}$$

Using the Bayes rule leads to a statistical likelihood ratio test:

$$H = \begin{cases} H_1, & \text{if } \frac{P(x|H_1)}{P(x|H_0)} > \frac{P(H_0)}{P(H_1)} \\ H_0, & \text{otherwise} \end{cases} \tag{4.10}$$

In order to evaluate the test, an assumption must be made that the speech (S_j) and noise (N_j) features are asymptotically independent Gaussian random variables:

$$p(x|H_0) = \prod_{j=0}^{C-1} \frac{1}{\pi \lambda_N(j)} \exp\left(-\frac{|X_j|^2}{\lambda_N(j)}\right) \tag{4.11}$$

$$p(x|H_1) = \prod_{j=0}^{C-1} \frac{1}{\pi \lambda_S(j)} \exp\left(-\frac{|X_j|^2}{\lambda_S(j)}\right), \tag{4.12}$$

where X_j represents the MFCC coefficients, and $\lambda_N(j)$ and $\lambda_S(j)$ denote the variances of N_j and S_j for component j. Then, the decision rule becomes:

$$\prod_{j=0}^{C-1} \frac{\lambda_N(j)}{\lambda_S(j)} \exp\left(\frac{|X_j|^2}{\lambda_N(j)} - \frac{|X_j|^2}{\lambda_S(j)}\right) > \eta, \tag{4.13}$$

where η is an appropriate threshold.

This is a simple method of VAD using the likelihood ratio test. For other robust VAD algorithms, [33] is a good reference.

Once the segments are detected, they are enlarged before and after by a small amount of time, to avoid missing any speech because of tight bounds. Although, ideally, the best results would be obtained if the non-speech model was trained with the kind of non-speech to be expected in broadcast news, due to unavailability of such data, the non-speech model is trained with crowd noise. Even so, it gives an adequate performance.

4.3 Speaker Segmentation

After the segments of speech in the audio stream have been detected and labelled, it is still important to know if they are composed of only one speaker or more. Speaker based segmentation aims at dividing the input audio to personalized speaker turns. Given no prior information about speakers, the task is to detect where the speaker changes take place and then label the speaker turns. There are several methods proposed in literature which achieve this task of speaker segmentation, a nice overview of which may be found in [34], which can be broadly divided into:

1. Metric-based speaker change detection - where the speaker change boundaries are found using a distance measure that illustrates the dissimilarity between two speech segments.

2. Model-based speaker change detection - would use, for example, Gaussian mixture models trained for different acoustic classes and assign speech segments to the classes according to maximum likelihood principle.

Features extracted from the speech signal characterise both the spoken message, and the speaker and acoustic conditions. However, features collected from more than few seconds of speech are expected to fill the feature space in a way that depends primarily on the speaker and acoustic conditions, and not the particular text spoken [34]. Thus, if there are no changes in acoustic conditions, speaker change at time t can be detected comparing feature sets X and Y that correspond to speech uttered before and after time t.

Various distance measures have been used to compare the feature sets and detect speaker changes. The most common ones use the generalised likelihood ratio (GLR), the Bayesian information criterion (BIC), or the Kullback-Leibler (KL) divergence [34]. The approach used in this thesis involves a growing window and the use of the Bayesian Information Criterion (BIC) as a distance measure, as introduced in [6].

4.3.1 Bayesian Information Criterion

Bayesian Information Criterion (BIC) is used to select which of a set of models best fits a set of data samples. BIC introduces a penalty term which penalizes more complex models, so as to avoid overfitting, and aims to reach a balance between fitness to the data and complexity of the model.

The likelihood that the feature sets X and Y belong to the same speaker is effectively a measure of how well a Gaussian model can explain both of these observations, whereas the likelihood of having two different speakers is the same as measuring how well a model with two Gaussian components can explain these. Mathematically, given a segment of consecutive data samples $x = \{x_0, ..., x_n\}$, the task is to choose between two models, the first where all the samples are drawn from one multivariate Gaussian distribution, $x \sim N(\mu, \Sigma)$, and the second, where $\{x_0, ..., x_i\} \sim N(\mu_1, \Sigma_1)$ and $\{x_{i+1}, ..., x_n\} \sim N(\mu_2, \Sigma_2)$. The BIC metric is:

$$\Delta BIC = \frac{-N}{2}\log(|\Sigma|) + \frac{N_1}{2}\log(|\Sigma_1|) + \frac{N_2}{2}\log(|\Sigma_2|) + \lambda(\frac{1}{2}(d + \frac{1}{2}d(d+1))\log(N)) \tag{4.14}$$

where $N = N_1 + N_2$, N_1, N_2 are the number of features in X, Y respectively, d is the feature dimensionality, and λ is the penalty weight.

When $\Delta BIC < 0$, then the model with two Gaussians is preferred over the single Gaussian model.

4.3.2 Growing Window

Having discussed about model selection using BIC given a set of trial points, it is important to discuss the selection of these trial points, which is a non-trivial task.

An accurate way of achieving this is by using a growing window, as described in [6]. The process starts with a minimum window size, wherein ΔBIC values are calculated at each position using equation (4.14). If the most negative ΔBIC is less than 0, then the change point has been found, and the process begins again with a minimum-length window right after the point found. Otherwise, the window size is increased by a fixed value and the process is repeated again. Although quite accurate, this algorithm is quite slow, as the same hypothesis is tested repeatedly, with more data being added at each step.

This basic algorithm is improved in [44] and [30]. In [44], instead of enlarging the window by a fixed value, the step-size is small at first and progressively increased. In [30], the BIC computations are not performed for every point, but points spaced by a fixed step distance, Δ_{step}, in a first pass, to get the best candidate, say t_{can}, followed by a second pass of BIC computations from $t_{can} - \Delta_{step}$ to $t_{can} + \Delta_{step}$, in order to fine-tune the result. Using these improvements, the algorithm is sped up considerably.

4.4 Speaker Clustering

The final step for speaker diarization is to effectively perform speaker clustering. There are several approaches to this, with the most popular ones reviewed in [3].

The problem here is of measuring similarity and grouping together the similar samples. Hence, the BIC can be used as a metric here again, similarly as in segmentation. Standard agglomerative hierarchical clustering (HAC) (see Section 3.3.2 of Chapter 3) can be used here, with BIC as the similarity metric, on the segments, iteratively merging similar segments, and stopping when the most similar clusters have a BIC distance of more than 0.

Together with Chapter 3, this chapter describes the processes of clustering using face-images and speech segments independently. The next chapter talks about the fusion of the two.

Chapter 5

Audiovisual Speaker Diarization

Having defined the individual processes for clustering using face image information and diarization using speech MFCCs, in Chapters 3 and 4 respectively, a simple way to combine the two modes of information is outlined the next sections. The methods which will be reused are not explicitly defined again in this chapter, and may be referred to from the aforementioned chapters if needed.

5.1 Initial Segmentation

5.1.1 Speaker Segmentation

The algorithm for the multimodal diarization branches out from the speaker diarization process after the initial speaker segmentation step, as shown in Figure 5.1. As described in detail in Section 4.3, the speaker segmentation step follows the voice activity detection process. This initial segmentation step uses the excerpts of speech, as detected and labelled by the VAD, and divides them into personalized speaker turns, such that within each such small division, the speech is solely from a single speaker. Once this has been done, the face image information corresponding to each such segment is added next.

5.1.2 Adding Face Information

Given a speech segment (belonging to one speaker), we identify the continuous time interval to which it belongs. Then, from that time interval, all the faces are extracted, after the preprocessing steps mentioned in Section 3.1. Finally, features describing the face images are extracted as per Section 3.2.

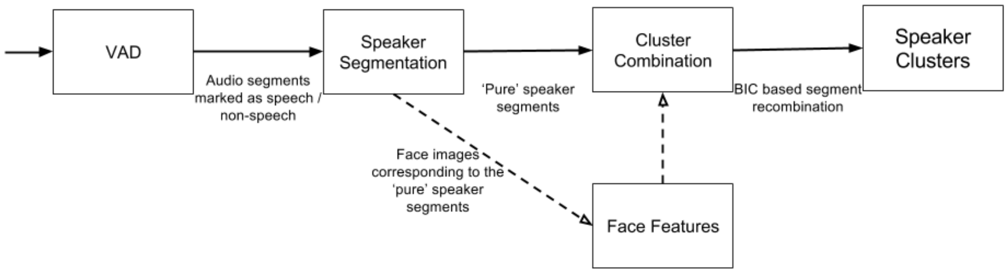

VAD
Audio segments marked as speech / non-speech
Speaker Segmentation
'Pure' speaker segments
Cluster Combination
BIC based segment recombination
Speaker Clusters
Face images corresponding to the 'pure' speaker segments
Face Features

Figure 5.1: Schematic of the speaker diarization algorithm, with multimodal addition.

One important note here is that, during the preprocessing steps, the threshold for the skin color based filtering is set to be quite high and that for the PCA based filtering is set as very low to ensure that little to no noise can creep into the initialization step. This is done because the errors that are accumulated here will only be propagated to the next steps of the algorithm, and that is undesirable.

5.2 Speaker Clustering

The final step in the process is to perform speaker clustering, so that the similar speakers and faces are grouped together. We experiment with a few ways to combine the information, all of them using the BIC similarity metric.

- Scheme 1:

$$\Delta BIC = \begin{cases} \Delta BIC_{speech}, & \text{either segment has only speech} \\ w \ \cdot \Delta BIC_{speech} + \\ (1-w) \cdot \Delta BIC_{face}, & \text{otherwise} \end{cases} \tag{5.1}$$

  Here, the BIC metric is calculated using a linear combination of the speech BIC metric and that from the face features, provided that both modes of information exist for a certain segment. Otherwise, only the speech BIC is used.

- Scheme 2:

$$\Delta BIC = \begin{cases} \Delta BIC_{speech}, & \text{either segment has only speech} \\ \Delta BIC_{face}, & \text{either segment has only faces} \\ w \cdot \Delta BIC_{speech} + \\ (1-w) \cdot \Delta BIC_{face}, & \text{otherwise} \end{cases} \tag{5.2}$$

The process iteratively merges similar clusters, and stops when the most similar clusters have a BIC distance of more than 0. A simple illustration is provided in Figure 5.2.

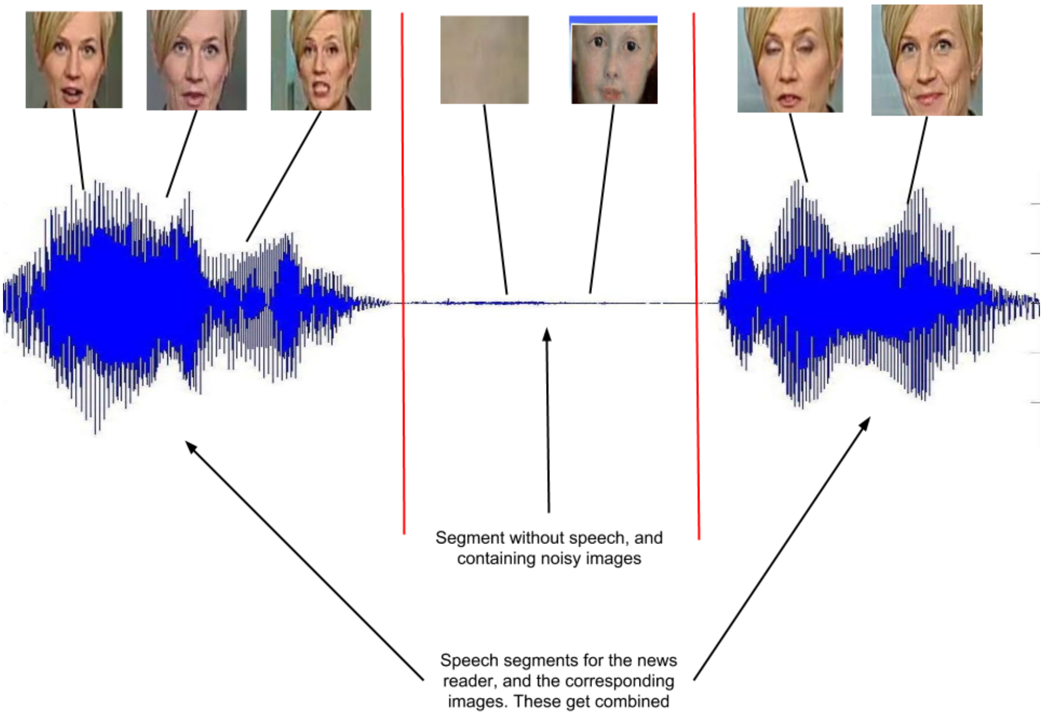

Segment without speech, and containing noisy images
Speech segments for the news reader, and the corresponding images. These get combined

Figure 5.2: Illustration of how the combination works.

5.3 Motivation for Suggested Method

The above schemes of combining the modalities through a linear combination of the BIC metric is admittedly naïve and the reasons for trying them out are four-fold:

1. We want to check whether combining the modalities is a step in the right direction, and if a simple combination shows even slight improvements, it could be an interesting avenue to explore.

2. The calculation overheads due to the addition of the facial information should be minimum.

3. The number of parameters to be tuned should not drastically increase because of the addition of the image information.

4. As HAC has been used as the clustering algorithm, it was desirable to keep it so for the multimodal clustering.

If the above schemes work, many modifications could be made to improve the system in the form of efficient preprocessing steps (one of them is suggested in Chapter 7) or a more efficient algorithm altogether.

Chapter 6

Experiments and Results

6.1 Chapter Overview

This chapter contains the results of the experiments performed on the YLE news broadcast videos dataset (see Section 6.2 for a brief description). The rest of the chapter is divided into sections which contain the description of the metrics which help quantify the results and the results of the experiments themselves for the three individual cases, namely, clustering using only face images, speaker clustering using only speech features, and diarization using both.

6.2 Dataset Description

The dataset for the experiments was generously provided by YLE [1], a Finnish broadcasting company, and consists of news broadcast videos for the year 2012. One week of these videos were manually annotated with speaker labels to compare the diarization system performance to. Each of the videos is approximately 20 minutes long, making the total amount of annotated content worth about 140 minutes. Apart from manual annotation of speaker labels, ground truth clusters were manually formed for the faces extracted from one week of videos for evaluating the face clustering algorithm. The number of speakers in per video varies from 10 to 17 for the 7 days of videos, while the number of different persons whose faces appear in the video varies from 10 to 23.

[1] http://yle.fi/

6.3 Results of Face Clustering

6.3.1 Mechanism of Cluster Analysis

Many ways exist in literature to evaluate clustering quality: intrinsic methods, which measure the within and between cluster proximities, and extrinsic methods, which use manual ground-truth classification. Here, we base our results on the metrics proposed in [36] to measure the quality of clustering for the algorithms described in Chapter 3.

[36] proposes an evaluation measure based on the purity and inverse-purity of clusters. The clustering created by the various algorithms in question is the 'estimated clustering', whereas that made manually is the 'ground-truth clustering'. Then, 'estimated purity' (EP) and 'ground-truth purity' (GTP) are defined as:

$$EP = \frac{1}{D}\sum_{k}\max_{j}|E_k \cap GT_j| \tag{6.1}$$

$$GTP = \frac{1}{D}\sum_{k}\max_{j}|GT_k \cap E_j|, \tag{6.2}$$

where D is the number of samples, and GT_i and E_i are the i^{th} ground-truth and estimated clusters respectively.

So, the higher the EP is, the lesser the chance of a cluster getting 'mixed', or including faces of different people. Similarly, the higher the GTP, the lesser the likelihood of a person's face appearing in multiple clusters. However, neither alone is a good criterion for judgement. Thus, an F-measure is defined. But before that, the EP and GTP measures should be scaled so that they are between 0 and 1. In order to scale them, consider a single cluster with all the faces, such that the $GTP = 1$ and

$$EP_{min} = \frac{\max_j |GT_j|}{D}. \tag{6.3}$$

Similarly for the case where each image is a singleton cluster, $EP = 1$ and,

$$GTP_{min} = \frac{|GT|}{D}. \tag{6.4}$$

Using the above equations, to scale the values of the estimated purity and the ground-truth purity, we finally arrive at:

$$EP_n = \frac{EP - EP_{min}}{1 - EP_{min}} \tag{6.5}$$

$$GTP_n = \frac{GTP - GTP_{min}}{1 - GTP_{min}}. \tag{6.6}$$

Then, the F-measure is constructed as:

$$F = 2\frac{EP_n \cdot GTP_n}{EP_n + GTP_n}. \tag{6.7}$$

It is equation (6.7) which is used to assess the capabilities of the face clustering algorithms earlier described.

6.3.2 Results

Table 6.1: Results of Face Clustering Experiments

Method	**EP**	**GTP**	**F-measure**
EM-GMM	0.728	0.695	0.711
HAC	0.754	0.714	0.732
TP_HAC	0.809	**0.824**	0.816
TP_T-HAC	**0.876**	0.815	**0.844**

The results of this thesis are based on the following algorithms (described in Chapter 3):

1. Gaussian Mixture Model based clustering or **EM-GMM**.
2. Hierarchical Agglomerative Clustering with a cosine distance metric and average-link similarity measure, or **HAC**.
3. HAC initialized with novel method, or **TP_HAC**.
4. Spatio-temporal HAC [17] with novel initialization, or **TP_T-HAC**.

The results in Table 6.1 list the average of the values for EP, GTP and F-measure for the 7 transcribed and labelled videos in the dateset (as described in Section 6.2). It can be seen that the novel initialization step improves the HAC algorithm, with the spatio-temporal HAC performing even better.

6.4 Results of Speech Based Diarization

6.4.1 Defining the Parameters and the Evaluation Metrics

While there is a significant amount of literature focussing on correctly evaluating the performance of a speech recognition system, there is considerably less literature on how to evaluate the performance of the speech diarization task. In this case, we will consider the metrics defined in [6] and [19], which came as a result of the Rich Transcription (RT) evaluation series performed by the National Institute of Standards and Technology (NIST).

The main metric used is the Diarization Error Rate or DER, which is basically the fraction of speaker time that is not labelled correctly. In the present case, assuming no overlapping speakers (which is usually a fair assumption to make for news broadcasts), DER is given by:

$$DER = \frac{\sum\limits_{Segs} \Big(ST \cdot \big(\max(IS?, SD?) - CD? \big) \Big)}{\sum\limits_{Segs} (ST * IS)}, \tag{6.8}$$

where $Segs$ is Segments, ST is the segment time in seconds, $IS?$ is a boolean denoting whether the real speaker is talking, $SD?$ is a boolean denoting whether the speaker has been detected, and $CD?$ is a boolean which is 1 if we have the correct label and 0 otherwise. The lower the value of the DER metric, the better the system is.

Apart from this, it is important to specify that NIST ignores speech pauses of 0.3 seconds or less, as they are considered to be normal in continuous speech and are an approximation of an utterance boundary pause. Also, mismatches between the detected times and human labelled ground truth of less than 0.25 seconds are considered to be correct, to account the difficulties for a human to correctly label the ground truth transcription with exact precision. These assumptions will be kept in mind while calculating the DER.

Having defined the evaluation metric, the set of parameters to be tuned (which the speaker diarization system is dependent upon) is defined as follows.

Table 6.2: Different Parameters of the Diarization System

Parameter	**Description**
Min. Speech	Minimum duration of speech that can be a turn
Min. Non-Speech	Minimum silence to split speech segment
Min. Window Size	Minimum size of the growing window
Window Step	The maximum amount the window can grow by
Delta Window Size	Window minimum growing
Lambda	Penalty weight for the BIC

As mentioned in Chapter 4, the speaker diarization system proceeds through a pipeline containing voice activity detection, speaker segmentation and speaker clustering processes. In this context, the minimum speech (MS) and minimum non-speech (MNS) are parameters for the voice activity detection, the minimum window size (MWS), window step (WS) and delta window size (DWS) are for the speaker segmentation, while the lambda (λ) is the regularisation weight for the BIC. A grid search is performed, with the DER as the optimisation function, to determine the best values for these parameters.

While the DER is a metric to evaluate the final results of the diarization process, other metrics, such as the VER (VAD error rate) or the MTER (missing turns error rate) may be computed to test the individual components of the diarization system (VER for the voice activity detection subsystem, MTER for the speaker segmentation subsystem in this particular example). However, this thesis will simply outline the final results of the diarization system, i.e., the DER, as the main aim of the thesis is to report the performance of the audiovisual fusion. For an in-depth overview of all the metrics and the importance of individual parameters, [30] should be referred to.

6.4.2 Results

The values of the aforementioned parameters was found by a grid search on the first 3 of the 7 annotated videos. The table below shows the values of the parameters obtained as a result of the grid search, as well as the value of DER for those parameters when diarization was performed on all 7 videos.

Table 6.3: Best result for the Diarization System

MS	MNS	MWS	WS	DWS	λ	**DER**
0.6s	1.5s	1s	3s	0.1s	1.35	**0.145**

For tasks involving audio data in English, the DER typically varies between 0.075 and 0.11 for state-of-the-art systems. In this light, the current system performance for the Finnish audio data seems acceptable.

6.5 Results of Using Audiovisual Information

For the clustering and diarization based on audiovisual cues, the above metrics (F-score for clustering, DER for diarization) are calculated separately for each combination scheme outlined in Section 5.2. For the DER, the parameters calculated in the previous section are reused, although this way might be suboptimal.

6.5.1 Results

Table 6.4: Results of Audiovisual Diarization Experiments

Method	**F-measure**	**DER**
Scheme 1	0.751	**0.141**
Scheme 2	0.812	0.16

The results for the different schemes are listed in table 6.4.

As can be observed, the addition of the face features marginally improves the diarization result. However, since the method used here to combine the segments is primarily speech focussed, the F-measure for the face clustering is not as good as the numbers obtained by the face clustering focussed algorithms.

Chapter 7

Conclusions

The master's thesis outlines and experiments with various algorithms which process single and multiple cues for speaker identification and diarization. Various evaluation metrics are defined and calculated, as well, to measure system performance.

Using only face images, two novel variants of the traditional HAC algorithm are evaluated, which use spatial and temporal information of 'where' (spatial coordinates) the faces are present in a frame and 'when' (timestamps) they are present. Using the evaluations metrics described, it was found that the novel variants improve the face clustering performance significantly.

Coming to the speech diarization, an existing implementation [30] based on the Aalto University speech recognition software [11], was tried and the optimal parameters calculated. The DER evaluation metric suggests good performance on the dataset provided.

Finally, some simple fusion techniques are evaluated, which give marginally better results, but show promise for future work.

It is important to note that although the DER is slightly improved for the combination, the F-score for the face clustering is found to decrease. This is probably because the Gaussian distributions used to model the image features do so inadequately. Moreover, there may be two faces in the frame (one speaking and one not), which are then put in the same cluster, thereby resulting in poor performance, when the face clustering metric is concerned. This may be improved by first filtering out non-speaking faces using some mouth-openness metric via keypoint detection of the lips in the faces [26], which has not been tried here. In all, the combination (even a simple one) seems to be a step in the right direction!

Bibliography

[1] Ahonen, T., Hadid, A., and Pietikainen, M. Face description with local binary patterns: Application to face recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence 28*, 12 (2006), 2037–2041.

[2] Amid, E., Mesaros, A., Palomaki, K., Laaksonen, J., and Kurimo, M. Unsupervised feature extraction for multimedia event detection and ranking using audio content. In *IEEE International Conference on Acoustics, Speech and Signal Processing, 2014* (May 2014), pp. 5939–5943.

[3] Anguera Miro, X., Bozonnet, S., Evans, N., Fredouille, C., Friedland, G., and Vinyals, O. Speaker diarization: A review of recent research. *IEEE Transactions on Audio, Speech, and Language Processing 20*, 2 (2012), 356–370.

[4] Anguera Miro, X., Wooters, C., and Pardo, J. M. Robust speaker diarization for meetings: Icsi rt06s meetings evaluation system. In *Proceedings of the Fourth International Conference on Spoken Language Processing, 2006* (2006), vol. 4299, Springer, pp. 346–358.

[5] Bishop, C. M. *Pattern Recognition and Machine Learning.* Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[6] Chen, S. S., and Gopalakrishnan, P. S. Speaker, environment and channel change detection and clustering via the bayesian information criterion. pp. 127–132.

[7] Chen, T. Audiovisual speech processing. *Signal Processing Magazine, IEEE 18*, 1 (2001), 9–21.

[8] Chibelushi, C., Deravi, F., and Mason, J. A review of speech-based bimodal recognition. *IEEE Transactions on Multimedia 4*, 1 (2002), 23–37.

[9] Daugman, J. Complete discrete 2-d gabor transforms by neural networks for image analysis and compression. *IEEE Transactions on Acoustics, Speech and Signal Processing 36*, 7 (1988), 1169–1179.

[10] Davis, S., and Mermelstein, P. Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Transactions on Acoustics, Speech and Signal Processing 28*, 4 (1980), 357–366.

[11] Department of Signal Processing and Acoustics, Aalto School of Electrical Engineering. Aalto ASR Public Repository. https://github.com/aalto-speech/AaltoASR, 2014.

[12] Garcia, C., and Tziritas, G. Face detection using quantized skin color regions merging and wavelet packet analysis. *IEEE Transactions on Multimedia 1*, 3 (1999), 264–277.

[13] Gebru, I., Alameda-Pineda, X., Horaud, R., and Forbes, F. Audio-visual speaker localization via weighted clustering. In *2014 IEEE International Workshop on Machine Learning for Signal Processing (MLSP)* (2014), pp. 1–6.

[14] Harris, F. On the use of windows for harmonic analysis with the discrete fourier transform. *Proceedings of the IEEE 66*, 1 (1978), 51–83.

[15] Ishikawa, S., Koskela, M., Sjöberg, M., Laaksonen, J., Oja, E., Amid, E., Palomäki, K., Mesaros, A., and Kurimo, M. Picsom experiments in trecvid 2013. In *Proceedings of the TRECVID 2013 Workshop* (2013).

[16] Jain, A. K., and Dubes, R. C. *Algorithms for Clustering Data.* Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1988.

[17] Kayal, S. Face clustering in videos: GMM-based hierarchical clustering using spatio-temporal data. In *13th UK Workshop on Computational Intelligence (UKCI)* (2013), pp. 272–278.

[18] Khalidov, V., Forbes, F., Hansard, M., Arnaud, E., and Horaud, R. Audio-visual clustering for multiple speaker localization.

[19] Kohonen, T., Ed. *Self-organizing Maps.* Springer-Verlag New York, Inc., 1997.

[20] Koskela, M., Sjöberg, M., and Laaksonen, J. Improving automatic video retrieval with semantic concept detection.

[21] Kurimo, M. Fast latent semantic indexing of spoken documents by using self-organizing maps. In *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing, 2000* (2000), vol. 6, pp. 2425–2428 vol.4.

[22] Laaksonen, J., Koskela, M., and Oja, E. Picsom: self-organizing maps for content-based image retrieval. In *International Joint Conference on Neural Networks, 1999* (1999), vol. 4, pp. 2470–2473 vol.4.

[23] Lades, M., Vorbruggen, J., Buhmann, J., Lange, J., Von Der Malsburg, C., Wurtz, R., and Konen, W. Distortion invariant object recognition in the dynamic link architecture. *IEEE Transactions on Computers 42*, 3 (1993), 300–311.

[24] Li, Y., Narayanan, S., and c. Jay Kuo A, C. Adaptive speaker identification with audiovisual cues for movie content analysis. *Pattern Recognition Letters 25* (2004), 777–791.

[25] Lucey, S., Chen, T., Sridharan, S., and Chandran, V. Integration strategies for audio-visual speech processing: applied to text-dependent speaker recognition. *IEEE Transactions on Multimedia 7*, 3 (2005), 495–506.

[26] Luzardo, M., Viitaniemi, V., Karppa, M., Laaksonen, J., and Jantunen, T. Estimating head pose and state of facial elements for sign language video. In *Proceedings of 9th Language Resources and Evaluation Conference (LREC 2014)* (2014), European Language Resources Association.

[27] Maison, B., Neti, C., and Senior, A. Audio-visual speaker recognition for video broadcast news: Some fusion techniques. In *IEEE Multimedia Signal Processing (MMSP99)* (1999).

[28] Nefian, A., Liang, L., Fu, T., and Liu, X. A bayesian approach to audio-visual speaker identification. In *Audio- and Video-Based Biometric Person Authentication*, vol. 2688. 2003, pp. 761–769.

[29] Ojala, T., Pietikäinen, M., and Harwood, D. A comparative study of texture measures with classification based on featured distributions. *Pattern Recognition 29*, 1 (1996), 51–59.

[30] Ojeda, A. M. M. Speaker diarization. Master's thesis, Aalto University, School of Science, 2014.

[31] Peeters, G. A large set of audio features for sound description (similarity and classification) in the CUIDADO project. Tech. rep., Icram, 2004.

[32] Potamianos, G., Neti, C., Gravier, G., Garg, A., and Senior, A. Recent advances in the automatic recognition of audiovisual speech. *Proceedings of the IEEE 91*, 9 (2003), 1306–1326.

[33] Ramirez, J., Gorriz, J. M., and Segura, J. C. *Voice Activity Detection. Fundamentals and Speech Recognition System Robustness.* In-Tech, 2007.

[34] Remes, U. Speaker-based segmentation and adaptation in automatic speech recognition. Master's thesis, Helsinki University of Technology, 2007.

[35] Sargin, M., Aradhye, H., Moreno, P., and Zhao, M. Audio-visual celebrity recognition in unconstrained web videos. In *IEEE International Conference onAcoustics, Speech and Signal Processing, 2009* (2009), pp. 1977–1980.

[36] Schwab, S., Chateau, T., Blanc, C., and Trassoudaine, L. A multi-cue spatio-temporal framework for automatic frontal face clustering in video sequences. *EURASIP Journal on Image and Video Processing 2013*, 1 (2013).

[37] Schwarz, G. Estimating the Dimension of a Model. *The Annals of Statistics 6*, 2 (1978), 461–464.

[38] Sjöberg, M., Ishikawa, S., Koskela, M., Laaksonen, J., and Oja, E. Picsom experiments in trecvid 2011. In *Proceedings of the TRECVID 2011 Workshop* (2011).

[39] Sjöberg, M., Laaksonen, J., Honkela, T., and Pöllä, M. Inferring semantics from textual information in multimedia retrieval. *Neurocomputing 71*, 13?15 (2008), 2576 – 2586.

[40] Sohn, J., Kim, N. S., and Sung, W. A statistical model-based voice activity detection. *IEEE Signal Processing Letters 6*, 1 (1999), 1–3.

[41] Stolcke, A., Kajarekar, S. S., Ferrer, L., and Shrinberg, E. Speaker recognition with session variability normalization based on mllr adaptation transforms. *IEEE Transactions on Audio, Speech, and Language Processing 15*, 7 (2007), 1987–1998.

[42] Struc, V., and Pavesic, N. The complete gabor-fisher classifier for robust face recognition. *EURASIP Journal on Advances in Signal Processing 2010*, 1 (2010), 847680.

[43] Tan, X., and Triggs, B. Fusing gabor and lbp feature sets for kernel-based face recognition. In *Proceedings of the 3rd International Conference on Analysis and Modeling of Faces and Gestures* (2007), pp. 235–249.

[44] Tritschler, A., and Gopinath, R. A. Improved speaker segmentation and segments clustering using the bayesian information criterion. In *EUROSPEECH* (1999).

[45] Turk, M., and Pentland, A. Face recognition using eigenfaces. In *Proceedings of the 1991 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)* (1991), pp. 586–591.

[46] Turunen, V. T., and Kurimo, M. Speech retrieval from unsegmented finnish audio using statistical morpheme-like units for segmentation, recognition, and retrieval. *ACM Trans. Speech Lang. Process. 8*, 1 (2008), 1–25.

[47] Viola, P., and Jones, M. Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)* (2001), vol. 1, pp. 511–518.

[48] Virpioja, S., Turunen, V. T., Spiegler, S., Kohonen, O., and Kurimo, M. Empirical comparison of evaluation methods for unsupervised learning of morphology. *Traitement Automatique des Langues 52*, 2 (2011), 45–90.

[49] Zhang, H., Ruokolainen, T., Laaksonen, J., Hochleitner, C., and Traunmüller, R. Gaze and speech-enhanced content-based image retrieval in image tagging. In *Artificial Neural Networks and Machine Learning ? ICANN 2011*, vol. 6792. Springer Berlin Heidelberg, 2011, pp. 373–380.

[50] Zhao, W., Chellappa, R., Phillips, P. J., and Rosenfeld, A. Face recognition: A literature survey. *ACM Comput. Surv. 35*, 4 (2003), 399–458.