MOBIO

Mobile Biometry

http://www.mobioproject.org/

Funded under the 7th FP (Seventh Framework Programme)
Theme ICT-2007.1.4
[Secure, dependable and trusted Infrastructure]

D3.2: Report on the description and evaluation of baseline algorithms for unimodal authentication

Due date: 31/12/2008 **Submission date:** 19/12/2008
Project start date: 01/01/2008 **Duration:** 36 months
WP Manager: Tim Cootes **Revision:**

Author(s): H. Bhaskar (UMAN) and P. Tresadern (UMAN)

Project funded by the European Commission in the 7th Framework Programme (2008-2010)		
Dissemination Level		
PU	Public	Yes
RE	Restricted to a group specified by the consortium (includes Commission Services)	No
CO	Confidential, only for members of the consortium (includes Commission Services)	No

D3.2: Report on the description and evaluation of baseline algorithms for unimodal authentication

Abstract:

This deliverable specifies the technological overview of current biometric baseline systems, both state-of-the-art unimodal face and speaker authentication approaches. The deliverable will elaborate the main scientific contributions, details of implementation, evaluation and results on baseline face authentication and speaker authentication through combining techniques of face localisation, facial feature localisation & face verification and silence detection & speaker verification respectively. These baseline algorithms are evaluated independently and as well tested together as unimodal authentication systems using the well-known BANCA database. The evaluation of these baseline algorithms/systems will: a) provide a basis for developing new methodologies to enhance the performance of current unimodal authentication systems, b) help in innovative development of fusion schemes for integrating unimodal systems to form robust bi-modal authentication systems and c) finally allow comparisons with these joint authentication methods.

Contents

KEY

1. AAM - Active Appearance Model
2. ASM - Active Shape Model
3. c-MCT-C - cascaded-Modified Census Transform-Classifier
4. DCT - Discrete Cosine Transform
5. DET - Detection Error Trade-off
6. EFLDM- Enhanced Fisher Linear Discriminant Model
7. EM - Expectation Maximization
8. EPC - Expected Performance Curve
9. FAR - False Acceptance Rate
10. FLD - Fisher Linear Discriminant
11. FRR - False Rejection Rate
12. GMM - Gaussian Mixture Model
13. HLDA - Hierarchical Linear Discriminant Analysis
14. HMM - Hidden Markov Models
15. HTER - Half Total Error Rate
16. LBP - Local Binary Pattern
17. LDM - Linear Discriminant Model
18. LFCC - Linear Frequency Cepstral Coefficients
19. LLR - Log Likelihood Ratio
20. MCT - Modified Census Transform
21. MFCC - Mel Frequency Cepstral Coefficients
22. PCA - Principle Component Analysis
23. SVM - Support Vector Machines
24. UBM - Universal Background Model
25. VAD - Voice Activity Detection
26. VJFD - Viola Jones Face Detector

1 Introduction

MoBio's objective is to develop new mobile services that are secured by joint bi-modal biometric authentication means. The main building blocks of the joint bi-modal system are fused *face authentication* and *speaker authentication*. In identity recognition systems, a database is stored of *client* models that are generated via a process of *enrollment*. This database is then accessed for one of two applications: *authentication* and *identification*.

In *authentication* applications, the user claims to have a specific identity in order to gain access to some resource. Therefore, the system must only compare the captured data with the stored model of the claimed client (one-to-one matching). If the user matches the claimed client model to within some degree of uncertainty, he[1] is accepted and given access to the resource. If the user does not fit the model then he is deemed to be an *impostor* and denied access. Passport control at an international border is an example of such a system. In these cases, enrollment and testing is typically a co-operative process with the client providing the required data willingly.

In *identification* applications, however, it is not known who the user claims to be. Therefore, the system must search its entire database of client models to find a possible match (one-to-many matching). For example, a witness to a crime may be presented with a photo album of known criminals in order to identify a potential suspect. Enrollment and testing in identification applications may be co-operative or covert (e.g. building a model/probe of a suspect's face from surveillance data).

In MoBio, we are concerned with identity *authentication* for applications such as telephone banking. However, identity authentication using the face or the voice information is a challenging research area due to natural and non-intrusive interaction with the authentication system. In order to develop a reliable and robust joint bi-modal authentication system, it is important to study and devise methods of uni-modal authentication. This essentially requires the development of state-of-the-art face and speaker authentication systems. Such uni-modal authentication systems will provide basis components for bi-modal authentication and model adaptation, also allowing further enhancement of current technology.

This report will focus on the implementation and evaluation of baseline systems, both state-of-the-art uni-modal face and speaker authentication approaches as well as their associated pre-processing steps such as face detection and voice activity detection algorithms.

[1] For the sake of simplicity, we refer to the user in the masculine form

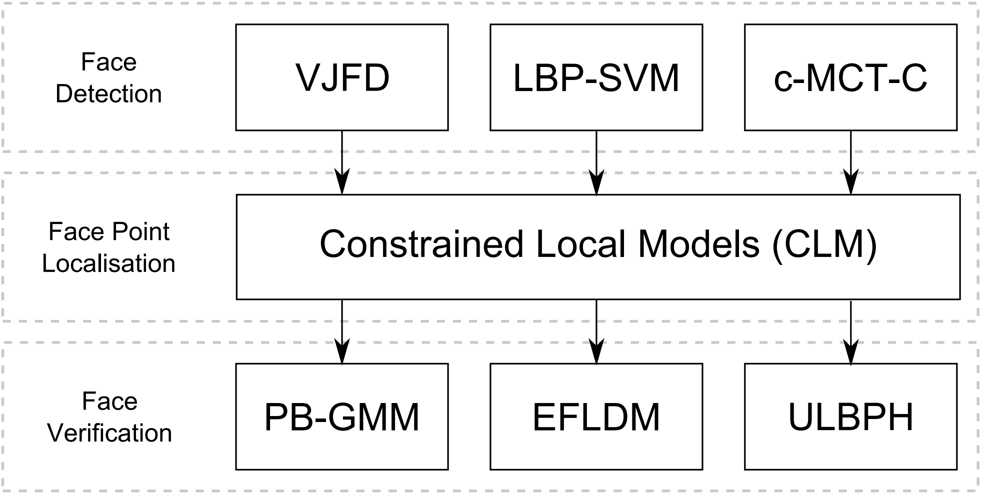


Figure 1: Baseline Algorithms for Face Authentication

1.1 Face Authentication

The goal of any face authentication system is to reliably validate (i.e. accept or reject) a claimed identity based on images of the user's face and a model of the claimed identity. The main steps required to accomplish reliable uni-modal face authentication are: face detection, face point localisation and face verification[2] (see Figure 1 for a schematic representation). In this report, a number of baseline algorithms are implemented and compared for the following three steps.

The first step toward face authentication is *face detection* that attempts to roughly determine the location of any faces in a given image. Face detection is a challenging problem because faces vary greatly in size, shape, orientation, colour and texture, and are influenced by lighting conditions, expression and occlusion. In this report, three baseline systems for face detection are presented:

- **VJFD**: The Viola-Jones face detector [89]
- **LBP-SVM**: Local Binary Pattern features [67, 68] combined with Support Vector Machines [34]
- **c-MCT-C**: A cascade of Modified Census Transform features based classifiers [30]

Once a face has been detected (and approximately localised), *face point localisation* determines the precise location of individual facial features such as eyes, nose and mouth. Most face point localisation systems are based on statistical deformable models of shape (e.g.

[2] We use the term 'verification' to describe the computational process of client/impostor classification, as opposed to 'authentication' that describes the overall application

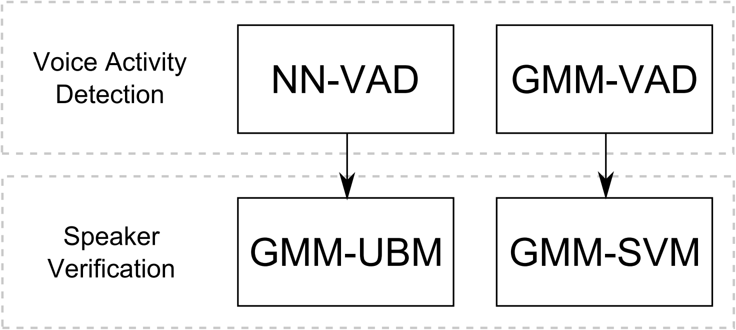


Figure 2: Baseline Algorithms for Speech Authentication

the Active Shape Model [17]) and/or appearance (e.g. the Active Appearance Model [16]) that can be fitted to an image. Using a statistical model helps to address the problem of variation in shape, size and appearance of the human face in addition to differences in illumination and expression. An advanced facial feature localisation technique based on Constrained Local Models (CLM) [21], is implemented as a baseline algorithm and evaluated. This method uses an appearance model similar to that of the AAM but for the generation of feature templates as against approximating image pixels directly. The method has been demonstrated to improve the localisation accuracy of facial features.

Finally, *face verification* refers to the computational process of validating a claimed identity based on one or more images of a face, and either accepting or rejecting the identity claim. Here, three verification strategies are compared:

- **EFLDM**: An Enhanced Fisher linear discriminant model [52, 53, 97]
- **PB-GMM**: A parts-based Gaussian mixture model [13, 57]
- **ULBPH**: A uniform local binary pattern histogram-based method [68, 2, 1]

1.2 Speech Authentication

The purpose of a speaker authentication system is to decide whether a given speech utterance was pronounced by a claimed client or by an impostor. The main components of such speech authentication systems are voice activity detection followed by speaker verification (see Figure 2 for a block diagram).

Voice activity detection (VAD) consists of isolating speech frames (relevant information for speaker authentication) from the rest of the audio signal. The silence segments obviously do not contain much speaker information, but may contain detrimental information such as ambient noise, music and background speech. This report, presents two baseline systems for voice activity detection:

- **GMM-VAD**: GMM-based voice activity detection [60]
- **NN-VAD**: VAD via neural network-based phoneme recognition, where all phonemes are merged together to form speech class [62]

Following the detection of voice activity, *speaker verification* determines whether the detected voice activity corresponds to the claimed client or an impostor. Speaker verification systems are generally classified into text-dependent and text-independent types. Various technologies such as Hidden Markov Models (HMMs), Gaussian mixture models (GMMs), Support Vector Machines (SVMs), pattern matching algorithms, Neural networks (NNs), decision trees, etc. have been used to process voice activity for speaker verification. The report describes two state-of-the-art verification systems:

- **GMM-SVM**: Factor analysis with support vector machines [60]
- **GMM-UBM**: GMM-UBM with channel compensation technique [11]

1.3 Evaluation protocols

Identity verification technology is still developing and many novel methods have been proposed in recent years. However direct comparison of the reported methods is often difficult if tests are performed on different data with large variations. Evaluation protocols alleviate this problem by defining a set of data, and how it should be used in a system in order to conduct systematic unbiased experiments and record performances. Over the last several years, different datasets for face verification and speaker verification have become available. In this report, the BANCA dataset and its corresponding protocols has been used for testing the uni-modal face authentication system (See Appendix A for further details.). Individual algorithms of face detection and face point localisation were also tested using other datasets including XM2VTS, BioID and BioSign.

1.4 Report structure

The report has been organised as follows. In Section 2, face detection methods are introduced, independently evaluated and their performances compared against one another using different datasets. A similar evaluation of face point localisation is presented in Section 3. The implementation of the state-of-the-art uni-modal face verification schemes is presented along with evaluation results and comparison in Section 4. Similarly, voice activity detection algorithms are introduced in Section 5 and speaker verification techniques are evaluated in Section 6. Finally, the uni-modal systems for face and speaker authentication are summarised in Section 7.

2 Face Detection

The first step of any face verification system is detecting the locations of faces present within images. Face detection is particularly challenging because of variability in scale, location, orientation, pose (frontal, profile, angular, etc.), appearance and lighting conditions during capture.

There are several terms related to face detection and they are used in different meanings by different authors. For the purposes of this report, the following definition of face detection is considered:

> "Given an arbitrary image, the goal of face detection is to determine whether or not there are any faces in the image and, if present, return the image location and extent of each face." [96].

Among the many face detection methods, the ones based on learning algorithms and classification have attracted much attention recently and have demonstrated significantly better results than earlier, hand-crafted knowledge based methods. These methods are essentially data-driven and rely heavily on training sets which in turn trigger discussions on the suitability of particular datasets for such a task. Another important aspect of face detection techniques is the evaluation of its performance. Most methods use metrics such as detection rates and false alarm rates in addition to learning time, execution time and the number of samples required in training for comparing performances.

In the remainder of this section, a short review of several face detection methods (Section 2.1) in addition to detailed implementations of competetive baseline face detectors are presented in Section 2.2. Performance evaluation metrics are described in Section 2.3 and results comparing these techniques are illustrated in Section 2.4

2.1 Related work

A number of different techniques have been proposed for face detection. In a recent survey [96], these methods have been classified into four different types: a) Knowledge-based methods, b) Feature invariant approaches, c) Template matching methods and d) Appearance-based methods.

Knowledge-based methods include studies that are rule-based, typically utilising human knowledge of the constitution of a face. These top-down methods typically use manually tailored rules for what a face looks like [96]. *Feature invariant* techniques include algorithms that aim at determining structural components that exist in the face and use them in turn to find faces. The main aim of these approaches is to extract facial components such as eyes, nose, etc. and use them in conjunction with coded rules for identifying face candidates. Though these techniques are quite straightforward, translating human knowledge into well-defined rules is often hard.

In contrast, number of bottom-up strategies have also been proposed. These techniques work with the underlying assumption that identifying invariant features in faces that allow reliably detecting faces under changing lighting condition and poses.

The *template-based methods* aim at matching the input image to a stored set of patterns containing the description of the face and its features, usually using correlation-type techniques.

Finally, in contrast to the template-based technique, the *appearance-based models* learn to capture the variability in appearance from a set of training images and further use these learnt models for face detection.

A lot of progress has been made especially in real-time face detection since the work of [96], but in terms of detection accuracy, some of the methods presented in this survey are still top-class. One of the best performing detectors mentioned in this survey is the one developed by Schneiderman and Kanade [79] which is based on wavelet transform and AdaBoost learning. The problem of this detector is its high computational cost.

The work by Viola and Jones [89] can be considered a breakthrough in face detection: the combination of an efficient computation of Haar-like features and a simple-to-complex cascade of AdaBoost based classifier result in a face detector capable of real time face detection from videos.

Stan Z. Li *et al.* [50] developed an extended system of Viola and Jones by replacing AdaBoost with FloatBoost and improvising the cascade of classifiers into a classifier pyramid which is able to detect faces at different poses in real time.

Support vector machines, having been developed specifically for solving two-class classification problems, are well suited for the face detection task. This was first proposed by Osuna *et al.* [70]. A problem with support vector machines in face detection is that a large number of support vectors may be needed resulting in rather slow detection systems. However, in [51] the authors reported to have developed a support vector machine based face detector working at 4 frames / second.

2.2 Baseline systems

2.2.1 Viola-Jones Face Detector (VJFD)

The first baseline method for face detection is based on the Viola-Jones face detector [89]. This detector is well known for its high detection accuracy under limited computational overload. An OpenCV library [69] based implementation of the face detector is also available. The detection system is widely used and is highly robust and efficient, therefore, making it a suitable baseline method for comparison against different other detection methods. Viola and Jones use features that resemble Haar wavelet responses as an input for their detectors. These features, albeit very simple, seem to provide enough information for reliable face detection. The most prominent advantage of these features is their speed: using so called *integral images*, these features can be computed in constant time from any subwindow of an image.

AdaBoost [29] is used to select the most prominent features among a large number of extracted features and construct a strong classifier from boosting a set of weak classifiers. The use of a cascade of classifiers made their system one of the first real-time frontal-view face detector. The system has resulted in a large amount of research and publications concerning face detectors of similar nature. In summary, the three factors behind the success of this type of a detector are:

- Haar-like features which are very fast to compute – can be computed in constant time.
- A fast and reliable classifier resulting from boosting
- Cascade of classifiers where most windows can be discarded in a very early stage of the cascade resulting in fast processing.

2.2.2 Local Binary Pattern-Support Vector Machine (LBP-SVM) Model

An alternative baseline model for face localisation is using LBP features [67, 68] with SVM [18]. Below, a brief description of the system is provided. More details on the face detection method can be found in [34].

LBP Based Description for Face Detection

The basic methodology for LBP based face description is as follows: The facial image is divided into local regions and LBP texture descriptors are extracted from each region independently. The descriptors are then concatenated to form a global description of the face.

For face detection, an LBP based representation that is suitable for low-resolution images and has a short feature vector needed for fast processing is used. A specific of this representation is the use of overlapping regions and a 4-neighbourhood LBP operator ($LBP_{4,1}$) to avoid statistical unreliability due to long histograms computed over small regions. Additionally, the holistic description of a face is enhanced by including the global LBP histogram computed over the whole face image. A standard resolution of 19×19 is considered and the LBP facial representation is derived as follows (see Figure 3): Divide a 19×19 face image into 9 overlapping regions of 10×10 pixels (overlapping size=4 pixels). From each region, compute a 16-bin histogram using the $LBP_{4,1}$ operator and concatenate the results into

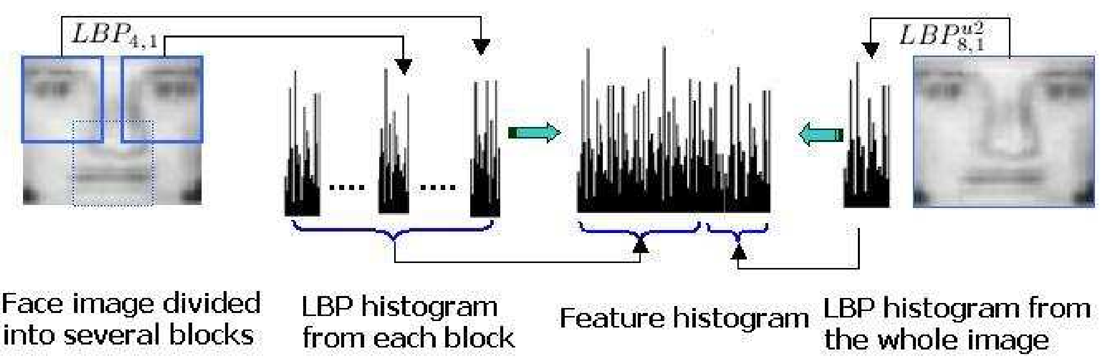

$LBP_{4,1}$
$LBP_{8,1}^{u2}$
Face image divided into several blocks
LBP histogram from each block
Feature histogram
LBP histogram from the whole image

Figure 3: Facial representation for low-resolution images: a face image is represented by a concatenation of a global and a set of local LBP histograms.

a single 144-bin histogram. Additionally, apply $LBP_{8,1}^{u2}$ to the whole 19 × 19 face image and derive a 59-bin histogram which is added to the 144 bins previously computed. Thus, obtain a (59+144=203)-bin histogram as a face representation.

Training Data

Generally, building appearance based face detectors requires large training sets of images in order to capture the variability of facial appearances. However, only a set of 2900 face images is used in this model as positives samples. To collect nonface patterns, the ”boostrap” strategy is used in five iterations. First, 4500 patterns from a set of natural images which do not contain faces are randomly extracted. Then, at each iteration the system is trained, the face detector is run, and all the nonface patterns that were wrongly classified as faces are collected and used for training. Overall, 6155 nonface patterns are obtained as negative training examples.

SVM Classification

A face detection system using a Support Vector Machine (SVM) classifier [18] is built since it is well founded in statistical learning theory and has been successfully applied to various object detection tasks in computer vision.

In short, given training samples (face and nonface images) represented by their extracted LBP features, an SVM classifier finds the separating hyperplane that has maximum distance to the closest points of the training set. These closest points are called *support vectors*. To perform a nonlinear separation, the input space is mapped onto a higher dimensional space using Kernel functions. In this approach, to detect faces in a given target image, a 19 × 19 subwindow scans the image at different scales and locations. A downsampling rate of 1.2 and a moving scan of 2 pixels are considered. At each iteration, the representation $LBP(w)$ is computed from the subwindow and fed to the SVM classifier to determine whether it is a face or not ($LBP(w)$ denotes the LBP feature vector representing the region scanned by the subwindow). The classifier decides on the "faceness" of the subwindow according to the sign of the following function:

$$F(LBP(w)) = Sign(\sum_{i=1}^{l} \alpha_i y_i K(LBP(w), LBP(t_i)) + b) \tag{1}$$

where $LBP(t_i)$ is the LBP representation of the training sample t_i, y_i is 1 or -1 depending on whether t_i is a positive or negative sample (face or nonface), l is the number of samples, b is a scaler (bias), and $K(.,.)$ the kernel function.

Post-Processing

Additionally, given the results of the SVM classifier, a set of heuristics is performed to merge multiple detections and remove the false ones. A merging procedure similar to that of the Viola and Jones's system [89] is adopted.

Optimizations

So far, the optimization of the system has not been considered as the original goal behind LBP-SVM face detector was only to show the discriminative power of the LBP based facial representation. Therefore, the performance of the system could be easily enhanced by using larger sets of training samples. Also, one can largely speed-up the system by using the notion of integral histograms for fast LBP face representation extraction. Finally, a cascade of classifiers would also improve the speed of the system.

2.2.3 Cascade of Boosted LBP-based Classifiers (c-MCT-C)

The third state-of-the-art frontal face detection system considered in this report is based on the combined use of the cascade paradigm of [89] and Modified Census Transform (MCT) features [30]. MCT belongs to the family of LBP features.

Like most face detection systems, the face detector scans the input image at many scales. The conventional approach is to compute a pyramid of sub-sampled images. A fixed scale sub-window $\mathbf{x}$ is then scanned across each of these images (Figure 4) and sent to the cascade.

Let $F(\mathbf{x})$ denote a n-level cascade of strong classifiers with increasing complexity. The strong classifier $f^j(\mathbf{x}) = \sum_i w_i^j \cdot h_i^j(\mathbf{x})$ is a linear combination of weak classifiers h_i (w_i is the associated weight) that are trained using a boosting algorithm [29]. For a given stage j, if $f^j(\mathbf{x}) < \theta_j$ then the window $\mathbf{x}$ is rejected (considered as a non-face) otherwise the window is sent to the next level for a more detailed evaluation. Finally at the last stage, the sub-window is considered as a face if $f^n(\mathbf{x}) \geq \theta_j$. The threshold θ of each stage is determined to detect close to 100% of the faces.

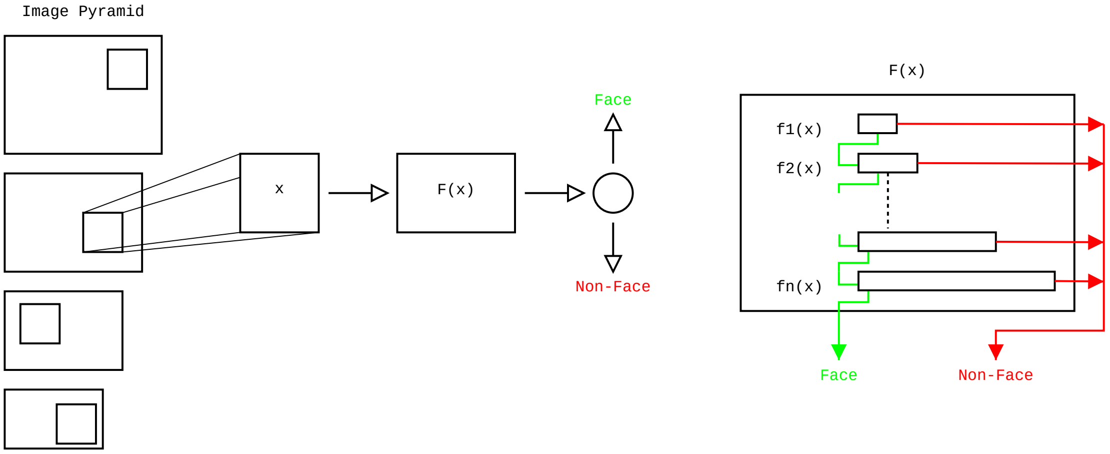


Figure 4: Face Detection using a Pyramid scanning and a cascade.

The objective of a cascade is to eliminate as many negative examples as possible at the earliest stage possible: simple classifiers are used to reject the majority of the sub-windows while detecting almost all of the positive instances. More complex classifiers can then focus on a reduced number of sub-windows. Thus, the construction of a cascade of classifiers reduces the computation time.

The MCT operator is a non-parametric 3 × 3 kernel which summarizes the local spatial structure of an image. At a given pixel position (x_c, y_c), MCT is defined as an ordered set of binary comparisons of pixel intensities between every pixel and the mean of the pixel values within the mask. The decimal form of the resulting 9-bit word (MCT code) can be expressed as follows:

$$MCT(x_c, y_c) = \sum_{n=0}^{8} s(i_n - \mu)2^n \tag{2}$$

where μ corresponds to the mean, i_n to the grey values of the 8 surrounding pixels and the centre pixel, and function $s(x)$ is defined as:

$$s(x) = \begin{cases} 1 & \text{if} \quad x \geq 0 \\ 0 & \text{if} \quad x < 0 \,. \end{cases} \tag{3}$$

2.3 Evaluation

Labelling a ground truth box around a face is a highly subjective operation. As a result, quantifying the degree to which a detected box is correctly positioned and scaled with respect to the true face is a somewhat difficult task. Furthermore, many datasets do not provide ground truth (presumably for this very reason).

However, the locations of specific facial features (e.g. the centres of the eyes) are much less ambiguous. Therefore, in order to evaluate the accuracy of the face detectors, we define a fixed mapping that determines predicted locations of the eye centres, p_l and p_r, from the detected bounding box. We then compute the maximum distance, d_{max}, from these points to their ground truth counterparts, q_l and q_r, normalised with respect to the inter-ocular distance, $|q_l - q_r|$, as our error metric [41]:

$$d_{max} = \frac{\max(|p_l - q_l|, |p_r - q_r|)}{|q_l - q_r|}. \tag{4}$$

This has the added benefit that exactly the same metric can be applied in order to compare eye localisation before and after face point localisation (see Section 3.3). To summarise the distribution of this metric over all images in each dataset, we present the median (i.e. 50th percentile) and 90th percentile statistics. In addition, the number of images where no face was detected at all are also recorded.

2.4 Results

The three baseline face detectors were applied to the evaluation datasets described in Appendix A and the results are shown in Table 1. The results show that the Viola-Jones detector typically out-performs the other baseline methods. However, it should be noted that the performance of a particular detector is also highly dependent on the training data. In this case, the relatively poor performance of the LBP-SVM detector can be attributed to a small training set. In contrast, the VJFD implemented in OpenCV was trained with a large and varied dataset and therefore exhibits superior performance.

However, it can be observed that the VJFD has more missed detections than c-MCT-C on the BANCA database. Note that those results on BANCA could be refined as this database offers various conditions. Finally, it should be noted that the VJFD crops the face larger than the two others baseline systems. It thus suggests that the performance of the system is also linked to the size of the observation window. Therefore, a higher performance of the other systems could be probably obtained with a similar observation window. The high rates of missed detections for the LBP-SVM adn c-MCT-C methods on the BioID dataset suggest that the training data used for these implementations was not representative of the test data in BioID. Therefore, it is likely that performance can be improved also by including more varied data in the training set.

Finally, examples of face detection on images from four datasets (Figures 5-8) and one example of a failed detection are illustrated (Figure 9).

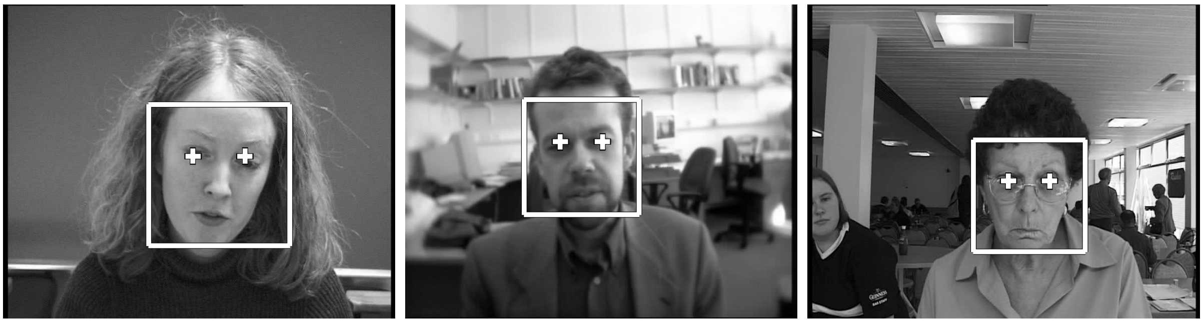


Figure 5: Detected examples from the BANCA dataset, using the Viola-Jones detector.

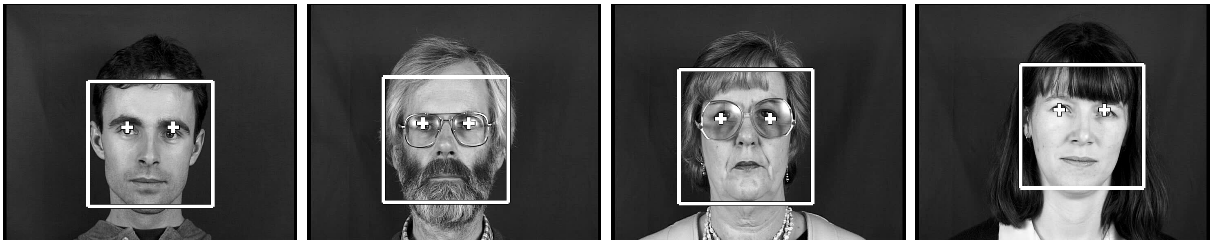


Figure 6: Detected examples from the XM2VTS dataset, using the Viola-Jones detector.

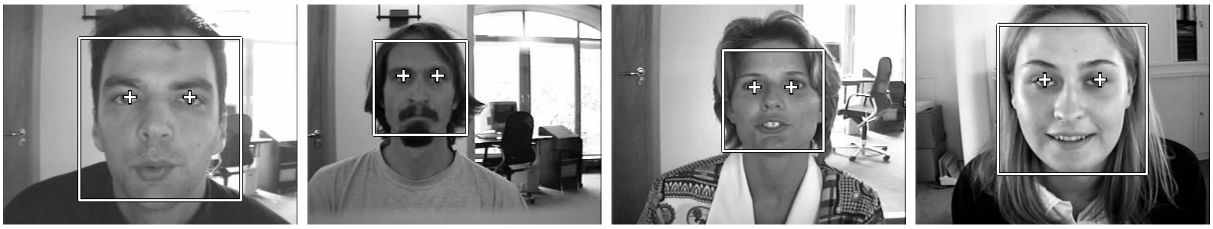


Figure 7: Detected examples from the BioID dataset, using the Viola-Jones detector.

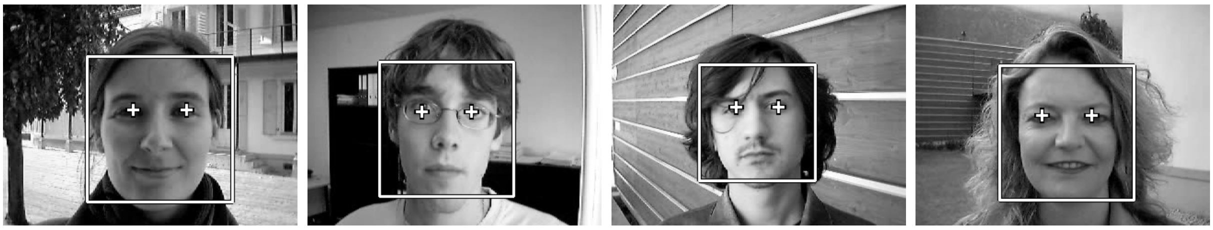


Figure 8: Detected examples from the BioSign dataset, using the Viola-Jones detector.

		Missed detections	d_{max}	
			Med.	90%
BANCA	VJFD	189 (2.9%)	0.09	0.15
	LBP-SVM	424 (6.5%)	0.24	2.2
	c-MCT-C	154 (2.4%)	0.11	0.18
XM2VTS	VJFD	12 (0.51%)	0.11	0.18
	LBP-SVM	13 (0.55%)	0.17	0.39
	c-MCT-C	71 (3%)	0.14	0.21
BioID	VJFD	56 (3.7%)	0.092	0.15
	LBP-SVM	345 (23%)	0.19	0.46
	c-MCT-C	372 (24%)	0.14	0.25
BioSign	VJFD	5 (0.91%)	0.094	0.14
	LBP-SVM	57 (10%)	0.3	0.68
	c-MCT-C	21 (3.8%)	0.11	0.17

Table 1: Missed detections (i.e. images where no face was detected at all) and statistics (specifically the median value and 90th percentile over the entire dataset) of d_{max} for the eye points.

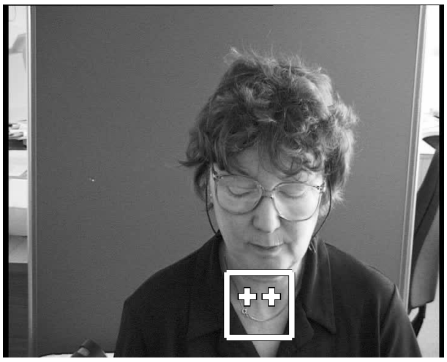


Figure 9: Viola-Jones detection failure on the BANCA dataset. It is likely that the neckline of the subject's jumper was mistaken for a jawline, while the chain may have provided mouth-like features.

3 Face Point Localisation

Following face detection, face point localisation can improve face verification performance. The goal of face point localisation techniques is to detect the presence and location of facial landmarks (e.g. the centre of the eye, a corner of the mouth) using the already located faces obtained from the face detector, and is a challenging problem owing to: (a) high inter-personal variability (e.g. gender and race); (b) the intra-personal changes (e.g. pose, expression, presence/absence of glasses, beard, mustaches); and (c) the acquisition conditions (e.g. illumination and image resolution).

3.1 Related work

Face point localisation techniques attempt to automatically find landmark points on a given face within an image. Several different methods of face point localisation have been proposed and a majority of these techniques can be conveniently categorised into: a) model based methods [17, 20], b) classification type approaches [54, 85] and c) probabilistic techniques [12, 38].

Model based methods aim at building generic models of the face so that the model can be fit to any new instance of the face automatically. A standard approach to this type of problem is collecting a manually labelled training set of images that enable the model to learn the shape and texture variation typically present in the faces. In contrast, the *classification type* methods treat the problem of face point localisation as a score/cost maximisation problem of a trained two class classifier. However, the *probabilistic formulations* for face point localisation aim at finding the best estimation of human face configuration in a given image. Probabilistic formulation of such problems involve deducing unknown parameters given known observations based on specific estimators such as Maximum A Posteriori or Maximum Likelihood and using an inference strategy such as the Condensation algorithm [39], Belief Propagation, etc.

Model based methods combine both shape and texture to build models and match to unseen images [17, 20, 23, 64, 86]. The input to such model is an approximate localisation of the face (either segmented manually or found automatically using a global detector such as the ones described in the previous section) and the goal would be to automatically locate prominent internal features on the face.

There are two different approaches to this problem. The first approach fits a *generative model* to the face region. The best match of the model simultaneously calculates feature point locations. Examples of this approach are the Active Appearance Model (AAM) [16] and 3D-Morphable Model [88] methods. The second approach is *feature-based* and splits the face into separate subregions that can be found using feature detection methods, with constraints on the relative configuration of feature points. The Active Shape Model (ASM) [17] and Pictorial Structure Model (PSM) [26] are examples of this type.

Generative model-based

A popular example of the generative approach is the Active Appearance Model (AAM) algorithm [16], which uses a combined model of shape and texture. A Principal Component Analysis (PCA) model of shape is learnt from a set of manually labelled images and also a model of the triangulated texture variation across the training set. A joint model of the shape and texture parameters allows the generation of new object instances, which resemble the training set. The AAM then searches new images by using the texture residual between the model and the target image to iteratively update model parameters to provide an estimate of the current image.

A related method is the 3D-Morphable Model due to Vetter *et al.* [88]. This method constructs a 3D model of the whole head from textures and 3D vertices obtained using a laser range finder. PCA is applied to the 3D coordinates and surface texture to build the generative model. The Morphable model is then fitted to new 2D images using a coarse to fine correspondence method based on optical flow. The method requires a few key points to be hand labelled to produce the dense correspondence from 3D to 2D. However recent approaches aim to mitigate the requirement for manual intervention. For example Romdhani *et al.* [77] use SIFT features [56], an appearance based rejection and a projection constraint to initialise the 3D morphable model automatically.

Due to the success of generative models, they have been developed significantly in many ways. Matthews *et al.* [61] have developed a more efficient update scheme which utilises the inverse compositional update algorithm [5]. Xiao *et al.* have developed hybrid 2D-3D versions of the AAM and applied them to human faces [94]. In the medical domain, AAM's have been extended to 3D by Mitchell *et al.* [64] and applied to cardiac image volumes. Van Ginneken *et al.* [86] compare the AAM and ASM with pixel based segmentation and also investigate hybrid approaches that improve on the normal pixel based AAM. Scott *et al.* [82] extend the texture sampling part of the algorithm to edge and cornerness values, instead of the normalised pixel values used in the original AAM, and demonstrate significantly improved results on faces and spinal images.

A recent alternative to the AAM algorithm that has been applied to both human face photographs and cardiac images is the RankBoost approach to shape prediction described by Zheng *et al.* [99]. This classification type method uses Rankboost [28] to rank the possible image warpings from the mean shape to the an unseen image and thus compute feature points.

Feature-based

A classic example of the feature based approach to model matching is the Active Shape Model (ASM) algorithm [17]. This method learns a statistical model of shape from manually labelled images and also PCA models of intensity profiles around individual feature points. When applied to an unseen image the best local match of each feature is found and the shape model fitted to the updated points. Therefore individual false detections which invalidate the learnt shape configuration are avoided. The search proceeds iteratively until the feature points converge.

Another elegant method of combining feature responses and shape constraints is is that of the Pictorial Structure Model (PSM) [26]. This approach is very efficient due to the use of pairwise constraints and a tree structure. The method uses the whole response surface of each detector and a dynamic programming search to find the global optimum set of final feature locations. However the PSM is mainly a global search method and does not use a fully-connected shape model. Therefore locally it is less accurate compared to methods that use the full shape model (e.g. ASM). Recently a denser graph matching approach using k-fans has been suggested to provide a stronger shape constraint [27]. The single tree based PSM is however very useful to provide initialisation points for local search methods that do use a fully-connected shape model (such as the AAM), given a rough localisation of the object.

Another method that combines shape and feature detection is the Simultaneous Modeling And Tracking (SMAT) algorithm described by Dowson and Bowden [23]. The SMAT method tracks an object given an initialisation and generates new templates from a clustered set of templates sampled from previous frames. It uses a shape model to constrain feature configurations, but does not form a combined model of shape and texture. Similar template selection methods were also published around the same time [19].

An extension of the ASM approach is the Shape Optimised Search (SOS) [19]. The SOS computes the response surface around each individual feature point, instead of merely computing the best response (as in the ASM). The model is then fitted to this set of response surfaces by allowing the shape parameters to vary whilst optimising the sum of feature responses (using the Nelder-Mead simplex optimiser [66]). Limits on the shape parameters enforce the relative shape constraint and utilising the whole response surface provides more robustness compared to the ASM approach [19].

3.2 Baseline systems

3.2.1 Constrained Local Models (CLM)

In contrast, the baseline system presented in this section builds on the template selection methods [23, 19] by using an appearance model, learnt from a fixed training set, to generate templates (rather than using selection methods such as nearest neighbour). This method, termed the Constrained Local Model (CLM) is therefore unable to generate false templates (from false matches) and can be applied to search static images as well as video, but at the expense of requiring a manually labelled training set.

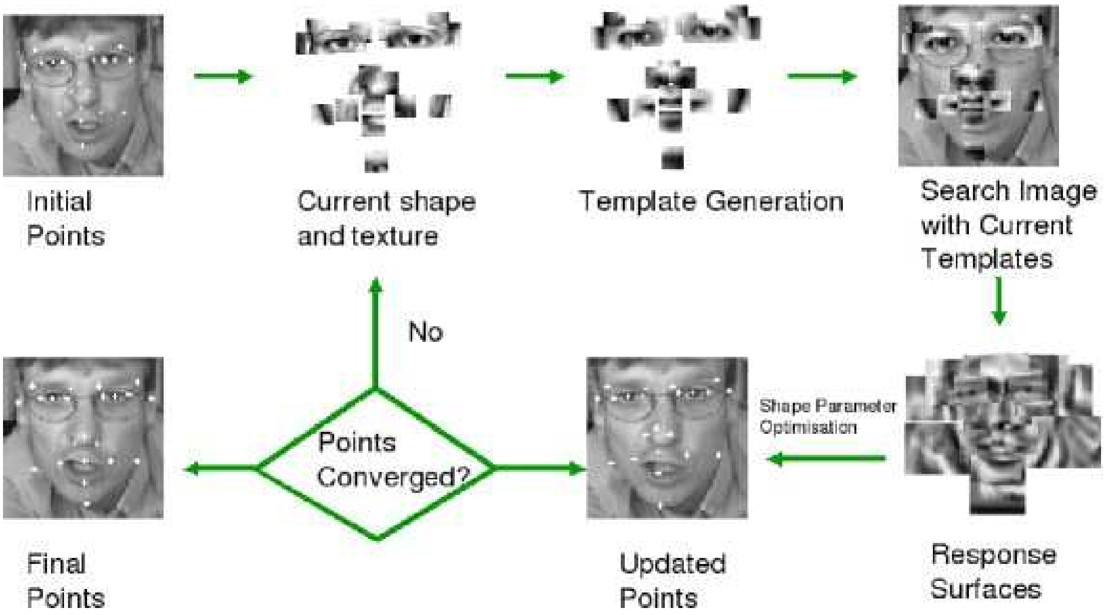


Figure 10: CLM Algorithm

The CLM algorithm [21] is an efficient and robust model matching method that uses a joint shape and texture appearance model. The main distinction of CLM as against AAM is that the appearance model is used in CLM to generate a set of feature templates as against trying to approximate the image pixels directly. The CLM approach attempts to learn the variation in appearance of a set of region templates surrounding individual features and matches the model to unseen images using a search mechanism. The process work iteratively by generating templates using the joint model to regions sampled around each feature point, correlating the templates to the target image to generate a response and finally optimising the shape parameters so as to maximise the sum of responses (see Figure 10 for a schematic illustration of the CLM process).

The CLM algorithm combines the constraints on appearance and shape as follows:-

1. Initialise the set of feature points by applying individual feature detectors, constrained using an algorithm such as the Pictorial Structure Model (PSM) [26], within the detected face region. This PSM model is very efficient due to the use of pairwise constraints and a tree structure, but is mainly a global search method that does not use a full shape model.

2. Repeat:-

   (a) Fit the joint model to the current set of feature points to generate a set of templates.

(b) Use the shape constrained search method to predict a new set of feature points.

Until Converged.

When tracking the initial points are propagated from the previous frame. On a new sequence (or if tracking failed) a global search can be used.

Constrained Local Appearance Models

A joint shape and texture model is built from a training set of 1052 manually labelled faces using the method of Cootes *et al.* [17]. However in the CLM approach the texture sampling method is different. A training patch is sampled around each feature and normalised such that the pixel values have zero mean and unit variance.[3] The texture patches from a given training image are then concatenated to form a single grey value vector. The set of grey scale training vectors and normalised shape co-ordinates are used to construct linear models, as follows.

$$\mathbf{x} = \overline{\mathbf{x}} + \mathbf{P}_s\mathbf{b}_s \qquad \mathbf{g} = \overline{\mathbf{g}} + \mathbf{P}_g\mathbf{b}_g \tag{5}$$

Where $\overline{\mathbf{x}}$ is the mean shape, $\mathbf{P}_s$ is a set of orthogonal modes of variation and $\mathbf{b}_s$ is a set of shape parameters. Similarly $\overline{\mathbf{g}}$ is the mean normalised grey-level vector, $\mathbf{P}_g$ is a set of orthogonal modes of variation and $\mathbf{b}_g$ is a set of grey-level parameters. The shape and template texture models are combined using a further PCA to produce one joint model. The joint model has the following form.

$$\mathbf{b} = \mathbf{P}_c\mathbf{c} \qquad \text{where} \qquad \mathbf{P}_c = \begin{pmatrix} \mathbf{P}_{cs} \\ \mathbf{P}_{cg} \end{pmatrix} \qquad \& \qquad \mathbf{b} = \begin{pmatrix} \mathbf{W}_s\mathbf{b}_s \\ \mathbf{b}_g \end{pmatrix} \tag{6}$$

Here $\mathbf{b}$ is the concatenated shape and texture parameter vector, with a suitable weighting $\mathbf{W}_s$ to account for the difference between shape and texture units (see []). $\mathbf{c}$ is a set of joint appearance parameters. $\mathbf{P}_c$ is the orthogonal matrix computed using PCA, which partitions into two separate matrices $\mathbf{P}_{cs}$ and $\mathbf{P}_{cg}$ which together compute the shape and texture parameters given a joint parameter vector $\mathbf{c}$.

Shape Constrained Search

Given a set of initial feature points, the joint shape and texture model is used to generate a set of grey value texture patches. The templates are applied to the search image and response images computed. Let (X_i, Y_i) be the position of feature point i and $I_i(X_i, Y_i)$ be the response of the i^{th} feature template at that point. The positions can be concatenated into a vector $\mathbf{X}$,

[3] The face regions from the training images are re-sampled to a fixed sized rectangle to allow for scale changes

$$\mathbf{X} = (X_1, \ldots, X_n, Y_1, \ldots, Y_n)^T \tag{7}$$

where $\mathbf{X}$ is computed from the shape parameters $\mathbf{b}_s$ and a similarity transformation T_t from the shape model frame to the response image frame. $\mathbf{X}$ is calculated as follows.

$$\mathbf{X} \approx T_{\mathbf{t}} \left(\overline{x} + \mathbf{P}_s \mathbf{b}_s \right) \tag{8}$$

The parameters of the similarity transform, $T_{\mathbf{t}}$ and, shape parameters $\mathbf{b}_s$ are concatenated into $\mathbf{p} = \left(\mathbf{t}^T | \mathbf{b}_s^T \right)^T$. Therefore $\mathbf{X}$ can be represented as a function of $\mathbf{p}$. Given a starting value for $\mathbf{p}$ the search proceeds by optimising a function $f(\mathbf{p})$ based on the image response surfaces I_i and the statistical shape model learnt from the training set. The objective function is

$$f(\mathbf{p}) \quad = \quad \sum_{i=1}^{n} I_i(X_i, Y_i) + K \sum_{j=1}^{s} \frac{-b_j^2}{\lambda_j} \tag{9}$$

The second term is an estimate of the log-likelihood of the shape given shape parameters b_j and eigenvalues λ_j. This log-likelihood follows the approach of Dryden [24] in assuming the b_j are independent and Gaussian distributed. The parameter K is a weight determining the relative importance of good shape and high feature responses. The value of K can be determined by computing the ratio of $\sum_{i=1}^{n} I_i(X_i, Y_i)$ and $\sum_{j=1}^{s} \frac{b_j^2}{\lambda_j}$ when applied to a verification set with human labelled ground truth. The optimisation of $f(\mathbf{p})$ is performed using the Nelder-Mead simplex algorithm [66].

3.3 Evaluation

In order to quantify improvement in eye location with points predicted from the detected bounding box of the face, the evaluation using d_{max} for the eye points (see Section 2.3) is repeated following face point localisation.

In addition, three measures over the set of *all* points for every processed image are computed: (i) maximum Euclidean distance from ground truth, d_{max}; (ii) the 90th percentile value of Euclidean distance from ground truth, d_{90}; and (iii) the mean Euclidean distance from ground truth,

$$d_{mean} = \frac{1}{n} \sum_{i=1}^{n} \frac{|p_i - q_i|}{|q_l - q_r|}, \tag{10}$$

where n is the number of labelled points. As in Section 2.3, all quantities are normalised with respect to the inter-ocular distance, $|q_l - q_r|$, to provide scale invariance. Computing d_{max} permits a direct comparison with the corresponding values for the eye points. The 90th percentile value gives a similar measure but is slightly more robust in the presence of one or two outliers. The mean Euclidean distance is a standard metric used in other studies [21]. Having computed each measure for every image in each dataset the cumulative frequency curve is plotted (Figure 11), and the median value and 90th percentile value over all *images* for each dataset is computed.

As in Section 2.4, the method is evaluated on four datasets: BANCA, XM2VTS, BioID and BioSign (see Appendix A). Note that the images used to train the CLM are completely independent of these datasets, since they consisted of different subjects imaged under different conditions. Face points localisation is initialised using the output of each presented face detector to evaluate any effect this has on point localisation accuracy.

3.4 Results

The results of this evaluation are summarised in Table 2 and a specific example of a cumulative frequency curve is shown in Figure 11. Due to the small number of ground truth points provided in BANCA and BioSign (two and four, respectively), values for d_{90} are not presented on these datasets. Similarly, the only labelled points in the BANCA dataset are the eye points so d_{max} is not reproduced for the eyes in this dataset.

From Table 2, it appears that the final error is largely unaffected by the initialisation i.e. that the CLM typically converges on the same stable solution. Furthermore, for datasets with a dense labelling of points (XM2VTS and BioID, in this case), d_{max} increases quite significantly. However, comparing this measure with d_{mean} suggests that this increase is largely as a result of a small number of outliers. Note also that the eye centres are typically among the most salient of the face points and can usually be localised with high accuracy. Therefore, including less salient points naturally increases the average error, as evidenced by d_{mean} for the XM2VTS and BioID datasets (with 22 landmarks each).

From Figure 11, the error between localised eye points and ground truth was reduced in over 90% of examples for the BANCA dataset. As a result, the maximum distance between eye points was less than 10% of the inter-ocular distance in around 70% of cases, and less than 16% of inter-ocular distance in 90% of cases following face point localisation.

Finally, randomly selected examples from each dataset are shown (Figures 12-15) in addition to one example of a failure mode (Figure 16).

		Eye points		All points					
		d_{max}		d_{max}		d_{90}		d_{mean}	
		Med.	90%	Med.	90%	Med.	90%	Med.	90%
BANCA	VJFD	0.069	0.14	-	-	-	-	0.054	0.11
	LBP-SVM	0.078	2.4	-	-	-	-	0.063	2.2
	c-MCT-C	0.075	0.16	-	-	-	-	0.06	0.14
XM2VTS	VJFD	0.049	0.095	0.19	0.34	0.13	0.24	0.067	0.11
	LBP-SVM	0.049	0.12	0.19	0.37	0.13	0.27	0.067	0.13
	c-MCT-C	0.052	0.16	0.21	0.57	0.14	0.42	0.071	0.22
BioID	VJFD	0.044	0.13	0.17	0.34	0.12	0.24	0.064	0.12
	LBP-SVM	0.055	0.23	0.18	0.49	0.13	0.4	0.071	0.21
	c-MCT-C	0.046	0.13	0.17	0.34	0.12	0.24	0.065	0.12
BioSign	VJFD	0.041	0.12	0.067	0.23	-	-	0.042	0.14
	LBP-SVM	0.043	0.14	0.067	0.2	-	-	0.041	0.12
	c-MCT-C	0.041	0.11	0.065	0.18	-	-	0.041	0.099

Table 2: Statistics (specifically the median value and 90th percentile over the entire dataset) of d_{max} (both for the eyes and all points), d_{90} and d_{mean}.

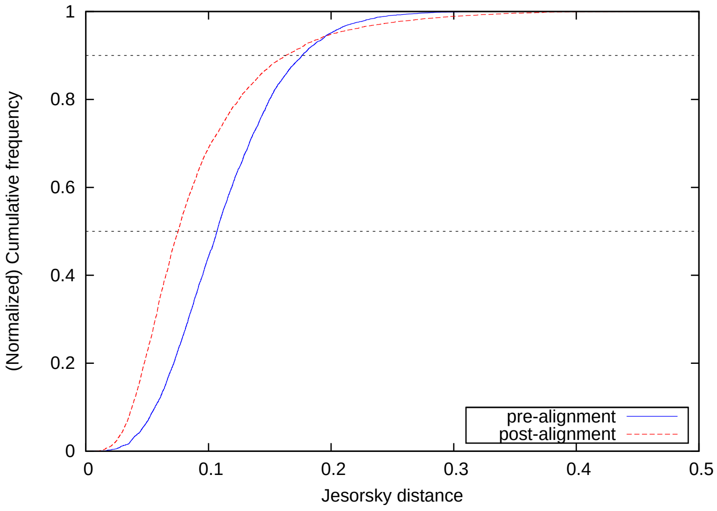


Figure 11: Cumulative distribution of d_{max}, used by Jesorsky [41], for the eyes over the BANCA dataset, initialized using the c-MCT-C face detector.

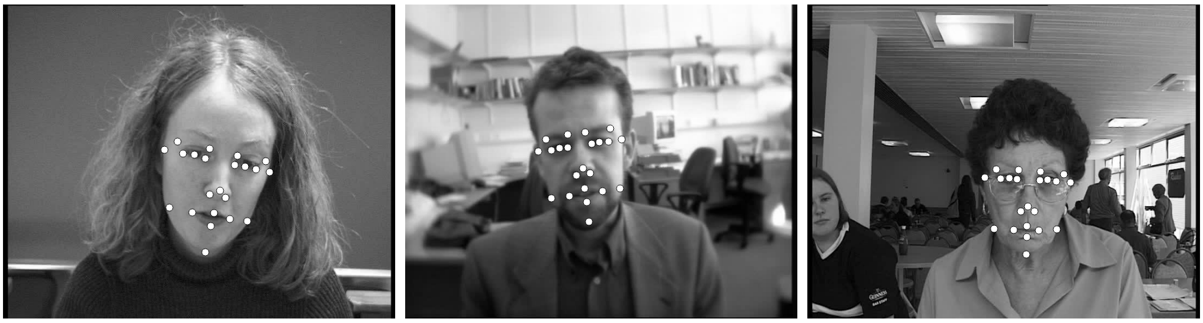


Figure 12: Localized examples from the BANCA dataset, initialized using the Viola-Jones detections (Figure 5).

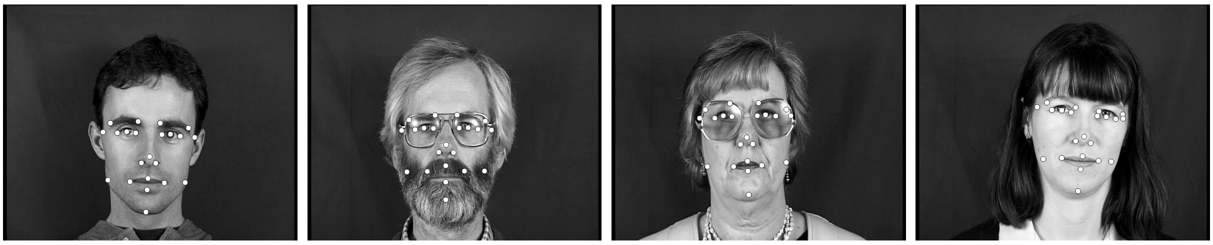


Figure 13: Localized examples from the XM2VTS dataset, initialized using the Viola-Jones detections (Figure 6).

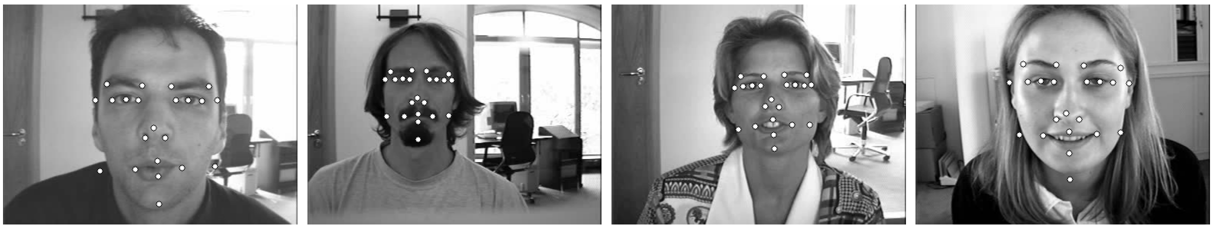


Figure 14: Localized examples from the BioID dataset, initialized using the Viola-Jones detections (Figure 7).

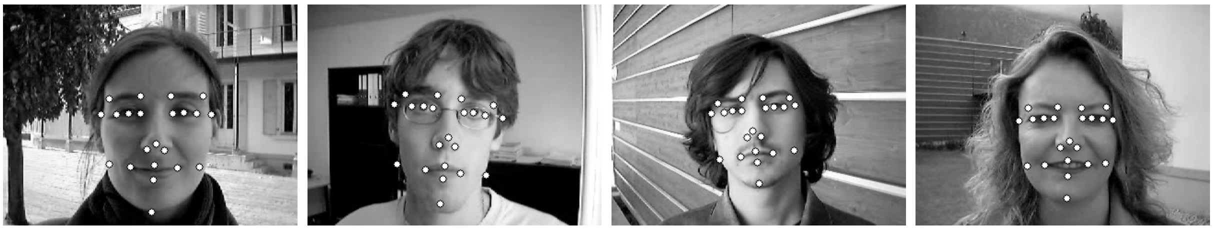


Figure 15: Localized examples from the BioSign dataset, initialized using the Viola-Jones detections (Figure 8).

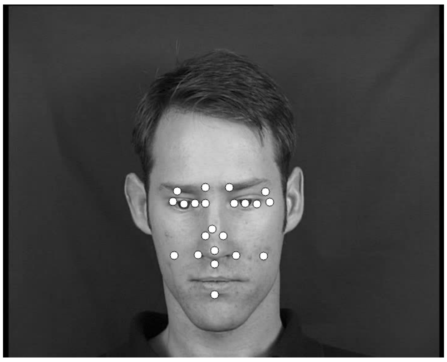


Figure 16: Localization failure on the XM2VTS dataset, initialized using the Viola-Jones detector. Although the eyes are located accurately lower face points such as the mouth and chin are not located correctly.

4 Face Verification

Face verification, an application of face recognition, authenticates a person's identity by comparing the captured image with the user's own template(s) stored in the system. Such systems perform one to one comparison to determine whether the person presenting himself to the system is the person he claims to be. Zhao *et al.* [98] and others [40] have discussed extensively the challenges of face recognition which raise issues in mathematics, computing, engineering, psycho-physics and neuroscience. These challenges can be summarised in two points: (1) A large variability in facial appearance of the same person and (2) High dimensionality of data and small sample size.

A large variability in facial appearance of the same person is caused by variations of facial pose, illumination, and facial expression. These variations are further increased by changes in the camera parameters, such as aperture, exposure time, lens aberrations and sensor spectral response. As mentioned in [45, 40], the intra-personal variations are usually larger than the image variation due to change in the face identity, called the inter-personal variation. This variability makes it difficult to build a simple model to describe an individual from a small number of sample images or perform linear discriminant analysis to separate different persons. Mathematically, the face manifold is highly complicated and non-linear. Furthermore, it is difficult to perfectly align the face to a canonical co-ordinate frame under these conditions. Therefore, misalignments are caused by translation, rotation, occlusion and scale errors in the normalised face image, thus making the recognition problem even more difficult.

In addition there exist issues surrounding the high dimensionality and small sample size of data. In general, the number of image samples per person available is much smaller than the dimensionality of the image space. Therefore, the system cannot build reliable models of each individual to recognise the face identity from a probe image. This is called the generalisation problem. Also, a small sample size may lead to numerical problems in matrix operations because of the singularity of within-class covariance matrices [6].

In general, two directions, feature representation and pattern classification based on the extracted features, must be pursued to deal with the first challenge. The first problem is concerned with the representation of a face image in a "good" feature space where the face manifolds become simpler. Both image normalisation and face representation can help in this respect. The second direction relates to the design of a classifier to solve the difficult non-linear classification and regression problems in the new face space and obtain good generalisation. In other words, the face image is segmented and then normalised by geometric and photometric normalisations which eliminate the effect of face rotation in plane, and scaling, and improve the face image quality. Then, a face representation, such as Gabor wavelets which reduce the non-linear behaviour of face data due to intra-personal variation, is extracted from the normalised image. Although good normalisation and face representation methods help in reducing the degree of non-linearity, commonly the dimensionality of the face representation is increased. Therefore, an effective dimensionality reduction method and a classifier are needed to deal with the above problem. The development of a successful algorithm requires the exploration of both directions.

One of the greatest challenges of the still face recognition is to solve the generalisation problem in the context of limited sample size. However, the video face recognition can overcome this problem as video can provide abundant image data for establishing the face model.

4.1 Related work

Video-based recognition has been overviewed in [92, 100] and can be divided into the following three categories:

- 3D/super-resolution model: Video data can contain face images taken from significantly different viewing angles and/or resolutions. For example, face can be captured in the surveillance scenarios. As a result, it may be possible to use this data jointly to reconstruct a 3D model or a super resolution face image.

- Facial dynamics: Video data can provide temporal continuity (dynamic information) which can potentially improve the recognition performance. Such continuity can be facial expression, geometric continuity related to head and/or camera movement, or photometric continuity. Lee *et al.* [49] performed video-based face recognition using probabilistic appearance manifolds, where K-means clustering with PCA is applied to model different head poses. The transition probability between images in each of the pose manifold is used to encode the connectivity between the pose manifolds in order to model appearance under different poses. Zhou and Chellappa [101] proposed a generic probabilistic framework to track and recognise face simultaneously by estimating the joint posterior probability distribution of motion vector and identity variable. Liu and Chen [55] proposed to use adaptive hidden Markov Model to model the dynamics of face rotation sequences. An empirical comparative studies [35, 49] showed that facial dynamics information is useful for recognition.

- Unordered multiple images: Video data can be treated as an unordered image set, ignoring the temporal information, which not only helps solving the generalisation problem in the training stage, but also improving the classification accuracy by choosing the good image frame, (for example, frontal face image), to match in the test stage. In general, there are two common approaches to recognition. The first approach is to compute the similarity score for each query image frame in the video with the gallery and then combine the scores over the query video sequence. For example, Gong *et al.* [33] applied the voting based fusion method to summarise the similarity. The baseline methods proposed in this report in Section 4.2 takes the average of the similarities between the query and gallery image frame for face verification. The second approach is directly to compute a similarity between the probe and gallery image sets. Yamaguchi *et al.* [95] applied canonical correlation to measure the similarity between the image sets based on the angle between subspaces. Wolf and Shashua [93] introduced the kernel principle angle to measure the similarity of image sets. Kim *et al.* [46] proposed Discriminative Canonical Correlation based on maximising the canonical correlations of within-class sets while minimising the canonical correlations of between-class sets in order to improve recognition accuracy.

4.2 Baseline systems

4.2.1 Parts-Based Gaussian Mixture Model (PB-GMM)

The first face verification baseline model implementation presented in this report combines part-based approaches and GMM modeling. Parts-based approaches divide the face into blocks, or parts, and treats each block as a separate observation of the same underlying signal (the face). According to this technique, a feature vector is obtained from each block by applying the Discrete Cosine Transform (DCT) and the distribution of these feature vectors is then modelled using GMMs. Several advances have been made upon this technique, for instance, Cardinaux *et al.* [13] proposed the use of background model adaptation while Lucey and Chen [57] examined a method to retain part of the structure of the face utilising the parts-based framework as well as proposing a relevance based adaptation.

Feature Extraction

The feature extraction algorithm is described by the following steps. The face is normalised, registered and cropped. This cropped and normalised face is divided into blocks (parts) and from each block (part) a feature vector is obtained. Each feature vector is treated as a separate observation of the same underlying signal (in this case the face) and the distribution of the feature vectors is modelled using GMMs. This process is illustrated in Figure 17.

The feature vectors from each block are obtained by applying the DCT. Even advanced feature extraction methods such as the DCTmod2 method [78] use the DCT as their basis feature vector; the DCTmod2 feature vectors incorporate spatial information within the feature vector by using the deltas from neighbouring blocks. The advantage of using only DCT feature vectors is that each DCT coefficient can be considered to be a frequency response from the image (or block). This property is exploited by the JPEG standard [72] where the coefficients are ranked in ascending order of their frequency.

Feature Distribution Modelling

Feature distribution modelling is achieved by performing background model adaptation of GMMs [13, 57]. The use of background model adaptation is not new to the field of biometric authentication; in fact, it is commonly used in the field of speaker verification [22].

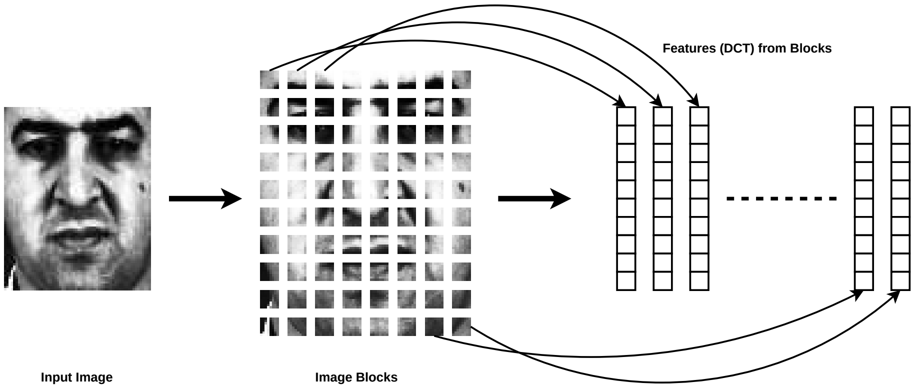


Figure 17: A flow chart of describing the extraction of feature vectors from the face image for Parts-Based approaches.

Background model adaptation first trains a world (background) model Ω_{world} from a set of faces and then derives the client model for the i^{th} client Ω^{i}_{client} by adapting the world model to match the observations of the client.

Two common methods of performing adaptation are mean only adaptation [74] and full adaptation [48]. Mean only adaptation is often used when there are few observations available because adapting the means of each mixture component requires fewer observations to derive a useful approximation. Full adaptation is used where there are sufficient observations to adapt all the parameters of each mode. Mean only adaptation is the method chosen for this work as it requires fewer observations to perform adaptation, this is the same adaptation method employed by Cardinaux *et al.* [13].

Verification

To verify an observation, $\boldsymbol{x}$, it is scored against both the client (Ω^{i}_{client}) and world (Ω_{model}) model, this is true even for methods that do not perform background models adaptation [78]. The two models, Ω^{i}_{client} and Ω_{world}, produce a log-likelihood score which is then combined using the log-likelihood ratio (LLR),

$$h(\boldsymbol{x}) = \ln(p(\boldsymbol{x} \mid \Omega^{i}_{client})) - \ln(p(\boldsymbol{x} \mid \Omega_{world})), \tag{11}$$

to produce a single score. This score is used to assign the observation to the world class of faces (not the client) or the client class of faces (it is the client) and consequently a threshold τ has to be applied to the score $h(\boldsymbol{x})$ to declare (verify) that $\boldsymbol{x}$ matches to the i^{th} client model Ω^{i}_{client}, i.e if $h(\boldsymbol{x}) \geq \tau$.

4.2.2 Enhanced Fisher Linear Discriminant Model (EFLDM)

The second baseline system considered in this report is to project each of the image frames to the Fisherface space, and then use normalised correlation to measure the similarity between the images in template and query in the projected space. The final similarity score is the average of those similarities. A brief description of this system is given.

Fisher Linear Discriminant Model (FLD)

Fisher Linear Discriminant aims to find a linear projection from an original fixed size feature vector such that in the projected space, objects are best separated. The objective criterion, in a two class case, can be formally described as maximizing the ratio of (squared) between-class centroids and the sum of within-class variances. In the case of multiple classes, this criterion becomes maximizing the trace of the product of two matrices: inverse averaged within-class covariance matrix and the sum of between-class covariance matrix (of the projected space).

Belhumeur *et al.* [6] first proposed to use FLD, also called Fisherface, to reduce the dimensionality of face image. However, due to large pixel dimensions and few image samples per user, the within-class covariance matrix is often non-singular. In order to overcome this problem, Belhumeur *et al.*proposed first to apply principal component analysis (PCA) to the original face image (stacked in a column vector), such that 95% of the variance is retained.

Having projected face images into the PCA space, it is straightforward to apply the standard FLD, thereby overcoming the problem of non-singular within-class covariance matrix. A special property of FLD is that, if there are C persons in a development database, one can have at most $C - 1$ dimensions in the FLD space.

In this implementation, the number of eigenvectors in the PCA transformation matrix is selected by keeping 98% of the eigenvalue energy. In order to improve the generalisation capability of FLD, an Enhanced Fisher Linear Discriminant Model (EFLDM) has been implemented [52, 53, 97]. This method decomposes the FLD procedure into a simultaneous diagonalisation [31] of the two within- and between-class scatter matrices. It first diagonalises the within-class scatter matrix and then the between-class scatter matrix.

Similarity measurement

In the testing stage, the face image is extracted with the eye positions provided from the c-MCT-C face detector and scaled to a size of 60×71 pixels. Photometric normalisation is not performed on the cropped image. EFLDM then projects the high-dimensional raw image vector, $\mathbf{x}$, formed by concatenating pixel values in the image space, to a low dimensional discriminative feature $\mathbf{y}$.

$$\mathbf{y} = \mathbf{W_{EFLDM}}^T \mathbf{x} \tag{12}$$

After projecting into the EFLDM space, the similarity measurement between query image $\mathbf{I_n}$ and the m template images, $Sim(\mathbf{I_n})$, defined in (13), is obtained by taking the average of normalised correlations between the discriminative features of the query image frame $\mathbf{y}_n$ and those of templates $\mathbf{y}'_{[1...m]}$.

$$Sim(\mathbf{I_n}) = \frac{1}{m} \sum_{i=1}^{m} \frac{\mathbf{y}_n^T \mathbf{y}'_i}{\|\mathbf{y}_n\| \|\mathbf{y}'_i\|} \tag{13}$$

The similarity measurement of query video and template is the average of the similarity scores between the query images in the video and the template images.

4.2.3 Uniform Local Binary Pattern Histograms (ULBPH)

The third baseline system for face verification is using uniform local binary pattern histograms (ULBPH) [58, 68] with histogram intersection for face recognition [2, 1]. The system is briefly introduced in this section.

Local Binary Pattern Histogram

The basic methodology for LBPH based face description is first to apply uniform LBP operator to a face image. The resulting LBP image is then divided into non-overlapping sub-regions, $\mathbf{M}_1$, $\mathbf{M}_1$,..$\mathbf{M}_J$. In each region j, a histogram is computed as a regional feature $\mathbf{f}_j$. This regional feature can be used to measure the face similarity by averaging the scores of local similarity of the corresponding regional histograms of the pair of images $\mathbf{I}$ and $\mathbf{I}'$ being compared.

In this implementation, the face image is extracted with the eye positions provided from the c-MCT-C face detector and scaled to a size of 120 × 142 pixels. Photometric normalisation is not performed on the cropped face. The radius and neighbourhood of the LBP operator are set to 2 and 8 respectively, and the resulting LBP image is then divided into 9 × 9 non-overlapping regions.

Similarity measurement

When comparing different distance measures for face recognition, the histogram intersection was found to perform better than chi-squared [14]. Therefore, the similarity measurement of the image pairs $Sim(\mathbf{I}, \mathbf{I}')$ using histogram intersection is defined in (14).

$$Sim(\mathbf{I}, \mathbf{I}') = \frac{1}{J} \sum_{j=1}^{J} \sum_{i} \min(\mathbf{f}_j(i), \mathbf{f}'_j(i)) \tag{14}$$

where i is an index of the histogram bin, $j = 1, \ldots, J$ is an index of each image in the enrollment video, $\mathbf{f}_j(i)$ denotes the i-th histogram bin of image j in video $\mathbf{I}$, and $\mathbf{f}'_j(i)$ is defined similarly but for video $\mathbf{I}'$.

For face verification from an image, the similarity measurement between the query image and template images is defined as the average of the similarity measurements between the query image and each image in the template. Similarly, for face verification from a video, the similarity measurement between the query video and the template images is defined as the average of the similarity measurements between each query image in the video and template images.

4.3 Evaluation

Two types of curve are used to compare the performance: the Detection Error Trade-off (DET) curve [59] and the Expected Performance Curve (EPC) [7]. A DET curve is actually a Receiver Operating Curve (ROC) plotted on a scale defined by the inverse of a cumulative Gaussian density function.

It has been pointed out [7] that two DET curves resulted from two systems are not comparable because such comparison does not take into account how the thresholds are selected.

The annual NIST speaker evaluation has adopted a specific operating threshold, defined as a function of the cost of false acceptance and false rejection error rates, and the class prior probabilities (subject to being either a genuine user or an impostor). EPC is a generalisation of the NIST evaluation method by using multiple threshold, with a normalised cost defined by a weighted error rate (WER):

$$\mathrm{WER}(\beta, \Delta) = \beta \mathrm{FAR}(\Delta) + (1 - \beta)\ \mathrm{FRR}(\Delta), \tag{15}$$

where $\beta \in [0, 1]$ balances between FAR and FRR.

In order to use EPC, two data sets have to be defined: the development score set, obtained from g1 and its evaluation counterpart, which is based on g2. For each chosen β, the development score set is used to minimise (15) in order to obtain an operational threshold. This threshold is then applied to the evaluation set in order to obtain the final pair of false acceptance rate (FAR) and false rejection rate (FRR).

The EPC curve simply plots half total error rate (HTER) versus β, where HTER is the average of FAR and FRR. The lower an EPC curve is the better the performance. An EPC makes possible the comparison of different systems.

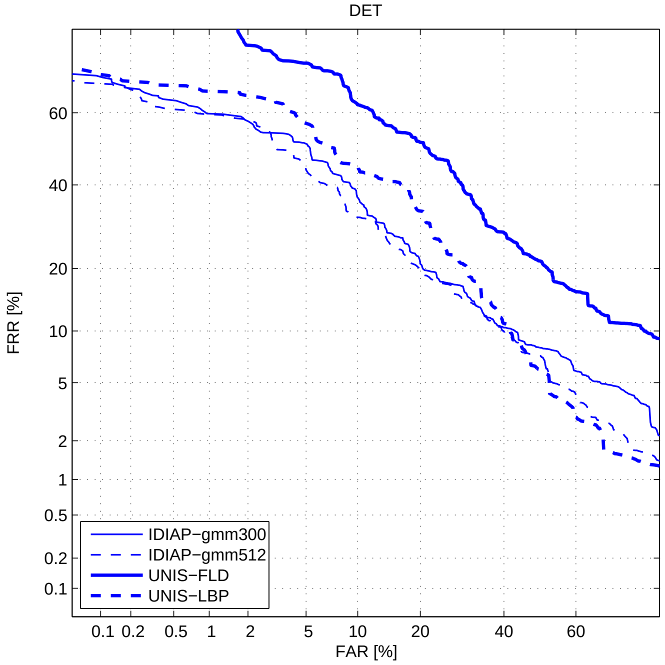


Figure 18: DET curves of the baseline systems computed on the g2 (evaluation set) of the BANCA video according to the P protocol.

4.4 Results

In this report, the baseline systems were evaluated using the BANCA database under the P protocol in an open set verification scenario (see Appendix A). A comparison of all four systems using DET and Expected Performance Curves are shown in Figures 18 and 19, respectively.

From the results, it can be observed that both figures are consistent in showing that the GMM systems are slightly better than ULBPH, and that both the GMM and the ULBPH systems perform significantly better than the EFLDM system based on the Fisher Linear Discriminant.

As a baseline system, EFLDM is suboptimal on this database for at least two reasons: Firstly, the Fisher discriminant is estimated on an independent set of identities different from the actual client identities. Secondly, it operates directly on image pixels, as opposed to GMM (which utilizes DCT coefficients) and ULBPH (which utilizes LBP). In essence, the latter two approaches rely on feature reprentations that are more robust to inaccurate face localization and degraded image conditions. Such an observation is consistent with previous studies in the literature.

The two GMM systems reported here are similar except that the number of Gaussian

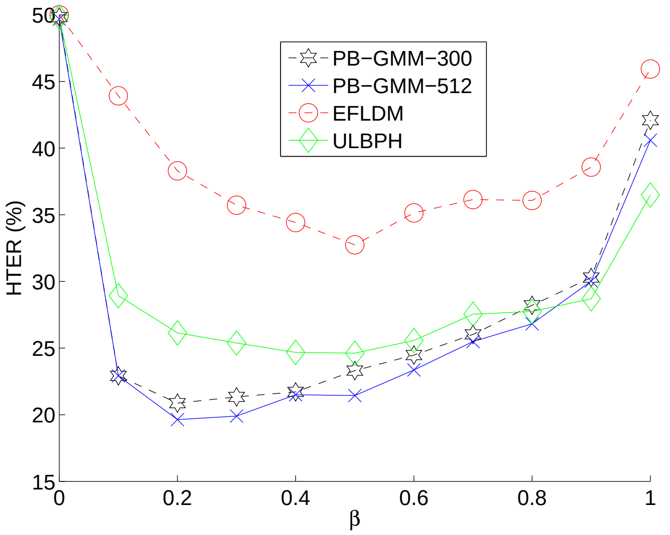


Figure 19: Expected performance curves of the baseline systems evaluated on the g2 (evaluation set) of the BANCA video according to the P protocol.

components are different (300 versus 512). However, the performance of both systems are not significantly different. This implies that the choice of number of Gaussian components is not very important. Although no efforts were made to optimize the systems, one recommended way to tune the number of Gaussian components of a GMM system is by using a series of 2^n for a linear function of n, for instance, $32, 64, 128, 256, 512, 1024, 2048$ etc.

An important difference between ULBPH and GMM is that ULBPH represents a holistic approach to face recognition whereas GMM represents a part-based approach. The advantage of a holistic approach is its ability to gauge the overall appearance of a face image. However, since a face image is not strictly rigid, the part-based approach can better gauge the local facial features such as eyes, nose, etc. In so doing, however, the part-based approach foregoes its ability to have a holistic view. In order to demonstrate that the GMM model does not use the spatial configuration, one can, for instance, artificially interchange image blocks (from which the DCT coefficients were derived) such that the resultant image no longer resembles a face.

Note that one should not conclude just based on the current experiment setting that the part-based approach is better than the holistic approach. This is because both systems reported here essentially rely on different feature representations. However, it is certainly sensible to combine both the systems in order to leverage the complementary nature of part-based and holistic information.

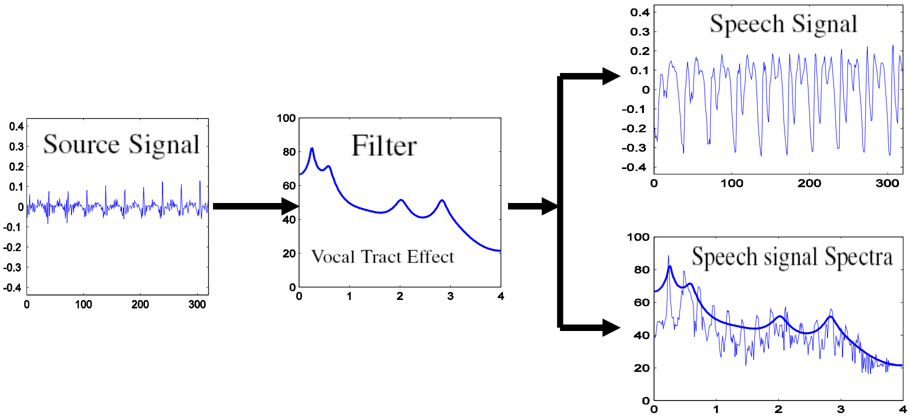


Figure 20: The source-filter model : the source signal, the vocal tract effect (the filter) and the resulting speech signal

5 Voice Activity Detection

Voice activity detection (VAD) (also known as speech activity detection) is a technique used in speech processing wherein the presence or absence of human speech is detected in regions of audio (which may also contain music, noise, or other sound). VAD is a fundamental step towards speech authentication. The main uses of VAD are in speech coding, speech recognition and other speech based applications. It can facilitate speech processing, and can also be used to deactivate some processes during non-speech segments: it can avoid unnecessary coding/transmission of silence packets in VOIP, saving on computation and on network bandwidth. VAD is usually language independent.

However, speech is a complex signal subject to several effects throughout its production occurring at several levels: semantic, linguistic, articulatory, and acoustic. The differences of these effects causes differences in the acoustic properties of the resulting speech signal. The speaker-dependent differences are used to discriminate between speakers: vocal tract characteristics, learned speaking habits characteristics. All these characteristics are hidden in the speech signal produced by the speaker, and the fundamental question is ”how to locate and extract these speaker-dependent effects from the speech signal?”.

The speech signal can be seen as the result of the passage of a signal source through the vocal tract, which gives it its spectral form characteristics: its frequency content (spectrum) is altered by the resonances (formants) of the vocal. This point of view is called the source-filter model (see Figure 20 for illustration). The most powerful systems are based on coefficients characterizing the evolution of the vocal tract shape throughout the speech signal production.

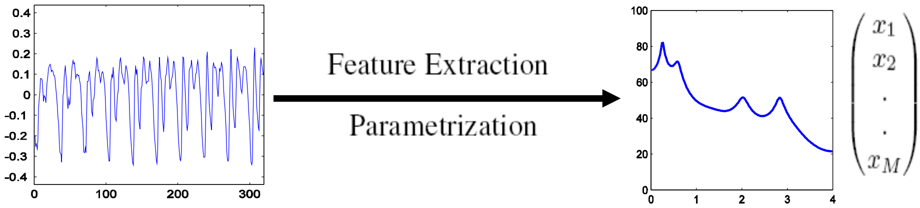


Figure 21: Feature Extraction :vector characterizing the vocal effect (the filter)

The vocal tract characteristics vary slowly during the production of the speech signal: they can be seen as constants during some tens of milliseconds (usually T=10, 20 or 30ms). From the point of view of the speaker identity (using only the vocal tract characteristics), the useful information is a vectors sequence (one vector every T ms). The nth vector comprises information about the vocal tract shape at time nT (see Figure 21 for illustration scheme).

VAD is an important enabling technology for a variety of speech-based applications. Therefore various VAD algorithms have been proposed that provide different compromises between latency, sensitivity, accuracy and computational cost. There are algorithms based on energy envelope, GMM modeling of speech and silence working with various kinds of features or alternative approaches like using phone recognizer to analyze where is speech.

5.1 Baseline systems

5.1.1 GMM-based (GMM-VAD)

The various sentences pronounced by a speaker can be seen as a sequence of vectors belonging to R^M, M being the number of coefficients necessary to characterize the vocal tract shape at a given time. The state-of-the-art systems use the contained information in the various acoustic vectors independently of their generation times. The most popular approaches in this category are based on the use of GMM. In this case it is possible to discard frames which can disturb the speaker verification process. The non-speech frames contain only information about the session (noise, microphone, telephone, ...) and not about the speaker. Thus their use could lead to probable performance decrease. In using GMM modeling, the frames are considered as independent and identical distributed so the non speech frame elimination can be performed without any change in theoretical framework.

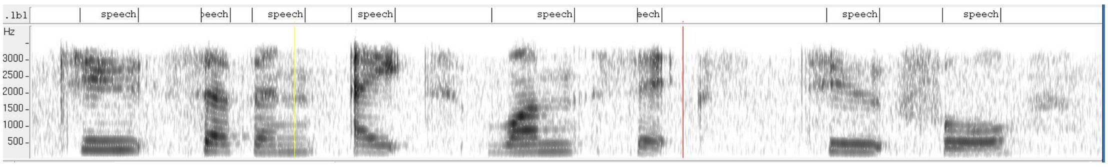


Figure 22: Speech labels given by the Speech Selection System

This frame selection method is based on an energy criterion: frames with low energy are considered as non-speech and frames with high energy are considered as speech (see Figure 22). The short-term log-energy is estimated every 10ms, the resulting value are first normalized using mean removal and variance normalization in order to fit a 0-mean and 1-variance distribution. The resulting scalar values are then used to train a GMM with 3 components, using the Expectation Maximization (EM) algorithm. Indeed $X\%$ of the most energized frames are selected through the GMM, with:

$$X = w_1 + (merged * \alpha * w_2) \tag{16}$$

where w_1 is the weight of the highest (energy) Gaussian component, w_2 is the weight of the middle component, *merged* is an integer which can equal 0 or 1, and α is a parameter fixed to 0.3. The value of *merged* is decided according to the likelihood loss when merging the gaussian components 1 and 2 and the components 2 and 3. If the loss is higher for components 1 and 2, *merged* is set to 0 else to 1.

The 3-Gaussians VAD based system retains frames with the highest energies. This implies that in the case of speech signals with low SNR, the problem of separating high and low energy frames becomes more difficult. In this case only voiced speech frames are retained. However, this is not a disadvantage, since the voiced speech frames contains the most information about the speaker identity (with respect to unvoiced frames).

The segmentation into speech (and silence) ranges has been smoothed to get rid of very small segments. The model uses a moving window with a length set to 10 frames and if there is a total of at least 3 frames in all the speech segments starting in this window, it builds a contiguous segment from the first to the last segment in the window. Otherwise, the segments are not kept for being part of the smoothed segmentation. The next window position starts at the beginning of the last segment starting in the old window.

5.1.2 Neural network-based phoneme recognition (NN-VAD)

Non-speech frames are discarded and only speech frames are considered in the following processing of training models and verification. Speech/non-speech segmentation is performed using a phoneme recognizer [81], where all phoneme classes are linked to speech

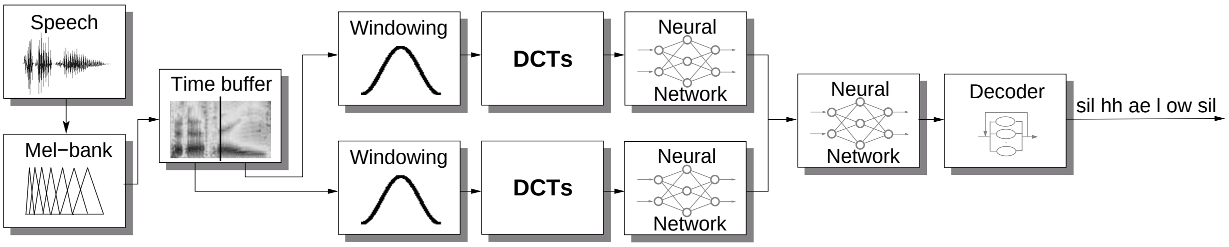


Figure 23: LCRC phone recognizer

class; silence and speaker noises (breathing, coughing, etc.) are omitted. Further post-processing with two rules based on short time energy of the signal is applied:

- if the average energy in speech segment is 30dB less than the maximum energy in the conversation side, then segment is labeled as silence.
- in case of stereo signals, where two speakers are in separate channels of one telephone call, we use the energy from the opposite channel to eliminate cross-talks. If the energy in the opposite conversation is bigger than energy minus 3dB in the processed side, the segment is also labeled as silence.

These two steps eliminate very quiet speech segments and cross-talk regions.

If it could be assumed that the phonemes represent speech, then it is possible to use a phoneme recognizer to find speech segments in audio recordings. In this report, a hybrid system based on Neural Networks (NN) is presented. Feature extraction makes use of long temporal context based on temporal patterns (TRAPs) [37]. First, Mel filter bank energies are obtained in the conventional way [25]. After sentence mean normalization in each band, temporal evolution of critical band spectral densities are taken around each frame. Based on previous work in phoneme recognition [80], the context of 31 frames (310 ms) around the current frame is selected. This context is split into 2 halves: Left and Right Contexts (hence the name "LCRC"). This allows for more precise modeling of the whole trajectory while limiting the size of the model (number of weights in the NN) and reducing the amount of necessary training data [80]. Both parts are processed using DCT to de-correlate and reduce dimensionality. Two NNs are trained to produce phoneme posterior probabilities for both context parts. Third NN functions as a merger and produces final set of phoneme-state posterior probabilities (Figure 23).

A simple Viterbi decoder[4] without any language model constraints processes the output of the merger and produces string of phonemes. In [80], it has been shown that this system outperforms phoneme recognizers with GMM/HMM modeling.

[4] SVite, which is part of STK-toolkit developed at Brno University of Technology: http://www.fit.vutbr.cz/speech/sw/stk.html

5.2 Evaluation

The VAD is not evaluated stand-alone (there is rarely data available with labeling of speech and silence regions). Instead, we report results for speaker verification that uses VAD as a pre-processing step in Section 6.4.

6 Speaker Verification

Speaker recognition systems fall into two categories: *text-dependent* and *text-independent*.

If the text must be the same for enrollment and verification this is called *text-dependent* recognition. In a text-dependent system, prompts can either be common across all speakers (e.g.: a common pass phrase) or unique. In addition, the use of shared-secrets (e.g. passwords and Personal Identification Numbers) or knowledge-based information can be employed in order to create a multi-factor authentication scenario.

Text-independent systems are most often used for speaker identification as they require very little if any cooperation by the speaker. In this case the text during enrollment and test is different. In fact, the enrollment may happen without the user's knowledge, as in the case for many forensic applications. As text-independent technologies do not compare what was said at enrollment and verification, verification applications tend to also employ speech recognition to determine what the user is saying at the point of authentication.

Each speaker recognition system has two phases: Enrollment and verification. During enrollment, the speaker's voice is recorded and typically a number of features are extracted to form a voice print, template, or model. In the verification phase, a speech sample or "utterance" is compared against a previously created voice print. For identification systems, the utterance is compared against multiple voice prints in order to determine the best match(es) while verification systems compare an utterance against a single voice print. Because of the process involved, verification is faster than identification.

6.1 Related work

A number of different systems for speaker authentication have been proposed in the literature and a majority of them are based on a statistical framework. According to this framework, a probabilistic model of a typical human voice is constructed by training on a large collection of voice recordings from different people to produce a universal background model (UBM). From this generic model, a more specific, client-dependent model is then derived using adaptation techniques and data from a particular client (GMM-UBM paradigm). It is then possible to estimate the likelihood ratio of the claimed client model to that of the generic model. The system them accepts or rejects the claim if the likelihood ratio is higher than a given threshold, selected in such a way as to achieve either a low rejection rate, a low acceptance rate, or some combination of both.

There is a number of techniques that have previously proven to improve feature extraction and modeling capability and help fight against the main problem in speaker verification - diversity in channel and acoustic conditions. These techniques are: Cepstral Mean Subtraction, Feature Warping [71], RelAtive SpecTrAl (RASTA) filtering [87], Heteroscedastic Linear Discriminant Analysis (HLDA) [47], Feature Mapping [75] and Eigenchannel Adaptation [9]. These technique are used in systems based on Gaussian Mixture Modeling (GMM) [76], or factor analysis (FA) [44, 91, 60] and systems based on sequence kernel Support Vector Machines (SVM) classifying either GMM mean supervectors [83] or vectors constructed from Maximum Likelihodd Linear Regression (MLLR) transformations [84], which are transformations commonly used in speech recognition for speaker adaptation, system based on prosodic features [42] or system based on phone strings or lattices [36, 65]. For combination of several systems see [10, 42, 73].

6.2 Baseline systems

6.2.1 GMM/SVM-based (GMM-SVM)

The use of GMM in a GMM-UBM framework has been a standard in the speaker verification [8]. In addition to this framework, the Latent Factor Analysis (LFA) is systematically applied for all systems in training and testing [43, 90, 60]. From the resulting session compensated model it is possible to extract supervectors by concatenating Gaussian means. These supervectors can be used directly in a SVM classifier. This association between the factor analysis and SVM allows to benefit from the FA decomposition power and SVM classification power. The implemented baseline system uses Z-T-norm for score normalization.

Feature extraction

The signal is characterized by 50 coefficients including 19 linear frequency cepstral coefficients (LFCC), their first derivative, their first 11 coefficients of second derivatives and the delta-energy. They are obtained as follows: 24 filter bank coefficients are first computed over 20ms Hamming windowed frames at a 10ms frame rate. Bandwidth is limited to the 300-3400Hz range.

Here, the energy coefficients are first normalized using a mean removal and variance normalization in order to fit a 0-mean and 1-variance distribution. The energy component is then used to train a three component GMM, which aims at selecting informative frames. The most energized frames are selected through the GMM. Once the speech segments of a signal are selected, a final process is applied in order to refine the speech segmentation:
1- overlapped speech segments between both the sides of a conversation are removed,
2- morphological rules are applied on speech segments to avoid too short ones, adding or removing some speech frames.

Finally, the parameter vectors are normalized to fit a 0-mean and 1-variance distribution. The mean and variance estimators used for the normalization are computed file by file on all the frames kept after applying the frame removal processing.

World models

Two GMM world models are used, one for males and one for females. The two GMM are trained using Fisher English Training Speech Part 1 (LDC:LDC2004S13), and consists of about 10 million speech frames each for males and females.

Resulting world models are 512 gender dependent GMM's with diagonal covariance matrices. For a better separation of initial classes, frames are randomly selected among the entire learning signal via a probability followed by an iteration of the EM algorithm, to estimate the GMM parameters. During the estimation of the world model parameters, instead of using all the learning signals in their temporal order, 10% of frames is selected randomly at each new iteration. For the two last iterations, the entire signal is classically used in its temporal order. During all the process, a variance flooring is applied so that no variance value is less than 0.5.

Client, test and impostor models with Factor Analysis

A speaker model can be decomposed into three different components: world, a speaker dependent and session dependent components [43, 90, 60]. A GMM mean super-vector is defined as the concatenation of the GMM component means. In the following, (h, s) will indicate the session h of the speaker s. The latent factor analysis model, can be written as:

$$\mathbf{m}_{(h,s)} = \mathbf{m} + \mathbf{D}\mathbf{y}_s + \mathbf{U}\mathbf{x}_{(h,s)}, \tag{17}$$

where $\mathbf{m}_{(h,s)}$ is the session-speaker dependent super-vector mean, $\mathbf{D}$ is $S \times S$ diagonal matrix (S is the dimension of the supervector), $\mathbf{y}_s$ the speaker vector (its size equal S), $\mathbf{U}$ is the session variability matrix of low rank R (a $S \times R$ matrix) and $\mathbf{x}_{(h,s)}$ are the session factors, a R vector. Both $\mathbf{y}_s$ and $\mathbf{x}_{(h,s)}$ are normally distributed among $\mathcal{N}(0, I)$. $\mathbf{D}$ satisfies the following equation $\mathbf{I} = \tau \mathbf{D}^t \mathbf{\Sigma}^{-1} \mathbf{D}$ where τ is the *relevance factor* required in the standard MAP adaptation.

The client model is obtained by performing the decomposition of equation 17 and by retaining only the speaker dependent components:

$$\mathbf{m}_s = \mathbf{m} + \mathbf{D}\mathbf{y}_s, \tag{18}$$

The success of the factor analysis model relies on a good estimation of the $\mathbf{U}$ matrix, thanks to a sufficiently high amount of data, where a high number of different recordings per speaker is available. In these experiments the U matrix is trained by using about 240 speakers (120 males and 120 females) coming from NIST'04. For each speaker about 20 sessions are considered.

Kernel based scoring and SVM modeling

By using (18), the factor analysis model estimates supervectors containing only speaker information, normalized with respect to the session variability. A probabilistic distance kernel that computes a distance between GMM's, well suited for a SVM classifier. Let $\mathcal{X}_{\mathbf{s}}$ and $\mathcal{X}_{\mathbf{s}'}$ be two sequences of speech data corresponding to speakers $\mathbf{s}$ and $\mathbf{s}'$, the kernel formulation is given below.

$$K(\mathcal{X}_{\mathbf{s}}, \mathcal{X}_{\mathbf{s}'}) = \sum_{g=1}^{M} \left(\sqrt{\alpha_g}\mathbf{\Sigma}_g^{-\frac{1}{2}}\mathbf{m}_s^g\right)^t \left(\sqrt{\alpha_g}\mathbf{\Sigma}_g^{-\frac{1}{2}}\mathbf{m}_{s'}^g\right). \tag{19}$$

This kernel is valid when only means of GMM models are varying (weights and covariance are taken from the world model). $\mathbf{m}_{\mathbf{s}}$ is taken here from the model in eq. 18, *i.e.* $\mathbf{m}_s = \mathbf{m} + \mathbf{D}\mathbf{y}_s$.

The LIA_SpkDet toolkit benefits from the LIBSVM [15] library to induce SVM and to classify instances. SVM models are trained with an infinite (very large in practice) C parameter thus avoiding classification error on the training data (hard margin behavior). The negative labeled examples are speakers from the normalization cohort.

6.2.2 Gaussian Mixture Model-based (GMM-UBM)

The second baseline system for speaker verfication is a GMM system that is based on standard GMM-UBM paradigm [74]. It employs number of techniques that has previously proven to improve GMM modeling capability and help fighting against the eternal problem in speaker verification - diversity in channel and acoustic condition. These techniques are: Cepstral Mean Subtraction, Feature Warping, HLDA and eigenchannel adaptation.

Features

The features used in the system are 13 Mel frequency cepstral coefficients (MFCC) coefficients (including zero'th cepstral coefficients, 20ms window, 10ms shift, 23 bands in Mel filter bank). To compensate for channel mismatch in different conversations, three simple feature processing techniques are successively applied: cepstral mean over whole conversation is subtracted from the features, Feature Warping (3sec window, warping into normal distribution) is applied and finally, temporal trajectories of individual feature vector coefficients are filtered using standard RASTA filter. After this processing, each feature vector is augmented with its first second and third order derivatives. This results in 52 dimensional feature vectors containing information about context of 13 frames.

HLDA

As the next step, Heteroscedastic Linear Discriminant Analysis (HLDA) [47, 32], which is also in common use in speech recognition systems, is employed. HLDA provides a linear transformation that can de-correlate the features and reduce the dimensionality while preserving the discriminative power of features. HLDA needs classes to estimate its class-covariance statistics (which are then used to estimate the transform matrix). For this purpose, GMM with 2048 Gaussian components is trained on test data from SRE2004 and the feature frames aligned with individual GMM mixture components are considered as classes. HLDA transformation reducing the dimensionality from 52 to 39 is estimated. GMM is then updated in the new HLDA space (by projecting collected class-covariance and mean statistics through HLDA transformation). Features are also projected into HLDA space and GMM is re-estimated (still only on SRE2004 test data) by few additional EM iterations to obtain the UBM model.

Training speaker model and verification

Each speaker model is obtained by traditional *relevance MAP* adaptation [74] of UBM using enrollment conversation. Only the means are adapted with relevance factor $\tau = 19$.

In verification phase, standard Top-N Expected Log Likelihood Ratio (ELLR) scoring [74] is used to obtain verification score, where $N = 10$ in the baseline system. However, for each trial, both speaker model and UBM are adapted to channel of test conversation using simple eigenchannel adaptation prior to computing the log likelihood ratio score. Note, that when T-norm [3] is used to normalize the score, each T-norm model is also adapted to channel of relevant tested conversation.

Eigenchannel subspace estimation

Let *supervector* be a MD dimensional vector constructed by concatenating all GMM mean vectors and *normalized by corresponding standard deviations*. M is the number of Gaussian mixture components in GMM and D is the dimensionality of features. Before eigenchannel adaptation can be applied, the directions in which *supervector* is mostly affected by changing channel are identified. These directions, which are refered to as eigenchannels, are defined by columns of $MD \times R$ matrix $\mathbf{V}$, where R is the chosen number of eigenchannels ($R = 30$ in our system). The matrix $\mathbf{V}$ is given by R eigenvectors of average within class covariance matrix, where each class is represented by supervectors estimated on different segments spoken by the same speaker.

More precisely, all speakers from NIST SRE2004 data for which at least two conversations are available are selected. For each speaker, i, and all his conversations, $j = 1, \ldots, J_i$, UBM is adapted to obtain a supervector, $\mathbf{s}_{ij}$. The corresponding average speaker supervector given by $\bar{\mathbf{s}}_i = \sum_{j=1}^{J_i} \mathbf{s}_{ij}/J_i$ is subtracted from each supervector, $\mathbf{s}_{ij}$, and resulting vectors form columns of $MD \times J$ matrix $\mathbf{S}$, where J is the number of all conversations from all selected speakers ($J = 2961$ in our case). Eigenchannels (columns of matrix $\mathbf{V}$)

are given by R eigenvectors of $MD \times MD$ matrix $\mathbf{SS}^T$ corresponding to R largest eigenvalues. Unfortunately, for this system, where $MD = 2048 \cdot 39 = 79872$, direct computation of these eigenvectors is unfeasible. A possible solution is to compute eigenvectors, $\mathbf{V}'$, of $J \times J$ matrix $\mathbf{S}^T\mathbf{S}$, eigenchannels are then given by $\mathbf{V} = \mathbf{SV}'$. In case the MAP criterion is used for eigenchannel adaptation (see below), the length of each eigenchannel must be also normalized to the standard deviation of $\mathbf{S}$ columns along the direction of the eigenchannel. This normalization is irrelevant in the case of ML criterion.

Eigenchannel adaptation

Once the eigenchannels are identified, speaker model (or UBM) can be adapted to a test conversation by shifting its supervector in the directions given by eigenchannels to better fit the test conversation data. Mathematically, this can be expressed as finding the *channel factors*, $\mathbf{x}$, that maximize the following MAP criterion:

$$p(\mathbf{O}|\mathbf{s} + \mathbf{Vx})\mathcal{N}(\mathbf{x}; \mathbf{0}, \mathbf{I}), \tag{20}$$

where $\mathbf{s}$ is supervector representing the model to be adapted[5], $p(\mathbf{O}|\mathbf{s}+\mathbf{Vx})$ is likelihood of the test conversation given the adapted supervector (model) and $\mathcal{N}(\cdot; \mathbf{0}, \mathbf{I})$ denote normally distributed vector. Assuming fixed occupation of Gaussian mixture components by test conversation frames, $\mathbf{o}_t, t = 1, \ldots, T$, it can be shown that $\mathbf{x}$ maximizing criterion (20) is given by:

$$\mathbf{x} = \mathbf{A}^{-1} \sum_{m=1}^{M} \mathbf{V}_m^T \sum_{t=1}^{T} \gamma_m(t) \frac{\mathbf{o}_t - \boldsymbol{\mu}_m}{\boldsymbol{\sigma}_m}, \tag{21}$$

where $\mathbf{V}_m$ is $M \times R$ part of matrix $\mathbf{V}$ corresponding to m^{th} mixture component, $\gamma_m(t)$ is the probability of occupation mixture component m at time t, $\boldsymbol{\mu}_m$ and $\boldsymbol{\sigma}_m$ are the mixture component's mean and standard deviation vectors and

$$\mathbf{A} = \mathbf{I} + \sum_{m=1}^{M} \mathbf{V}_m^T \mathbf{V}_m \sum_{t=1}^{T} \gamma_i(t). \tag{22}$$

In this implementation, occupation probabilities, $\gamma_m(t)$, are computed using UBM and assumed to be fixed for given test conversation. This allows to pre-compute matrix $\mathbf{A}^{-1}$ only once for each test conversation. For each frame, only Top-N occupation probabilities are assumed not to be zero. In the following ELLR scoring, also only the same top-N mixture components are considered. All these facts ensure that adapting and scoring different speaker or T-norm models on a test conversation can be performed very efficiently.

Eigenchannel adaptation can also be performed by maximizing ML criterion instead of MAP criterion. This correspond to dropping the prior term, $\mathcal{N}(\mathbf{x}; \mathbf{0}, \mathbf{I})$, in criterion (20) and term $\mathbf{I}$ in (22). In the experiments conducted, there is always enough adaptation data

[5] Note again that by our definition, supervector is mean supervector normalized by the corresponding standard deviations.

(test conversations contains approximately 2.5 minutes of speech) making the prior term in MAP criterion negligible. Therefore, no difference in performance when using the two criteria was found.

6.3 Evaluation

GMM-UBM

In the following experiments, results will be presented for 1-side training, 1-side test, all trials condition from SRE2005 NIST evaluation, which were used for system development, and for primary condition (1-side training, 1-side test, English only trials) from SRE2006 NIST evaluation.

GMM-SVM

The NIST SRE 2006 evaluation is used as development to determine the decision thresholds. For the GMM-UBM based systems, 200 male speakers and 119 female speakers from NIST 2004 are used as background data for TZ-norm[6]. For the SVM systems, NIST'04 and fisher (380 male sessions and 299 female sessions) are used to perform score normalization (TZ-norm) and as negative examples to train the SVM classifiers.

6.4 Results

GMM-UBM

Table 3 documents the process of building our system. It shows line-by-line the improvements in performance obtained by successively adding different techniques. The starting point was GMM system with 2048 Gaussian mixture components, features were 13 MFCC coefficients augmented with their derivatives and processed by cepstral mean normalization. The error rate of this system is very high and is almost halved by adding simple RASTA filtering. Replacing RASTA with Feature Warping improved the performance, however, further small gain was obtained from combination of both techniques. Application of RASTA filtering on top of Feature Warping appeared to be slightly more advantageous then doing it in opposite order. In next two steps, features were also augmented with second and third derivative coefficients. While adding second derivatives is clearly beneficial for both SRE2005 and SRE2006 evaluation sets, advantage of adding third derivatives, which were observed during the development on SRE2005 data, was not confirmed on SRE2006.

The following two steps, each significantly improving the system performance, were: projection of 52 dimensional features into 39 dimensional HLDA space and application of

[6] Application of T-norm prior to Z-norm

Eigenchannel adaptation.

So far, all the presented results were obtained without normalizing the verification scores by any standard technique, such as T-norm or Z-norm [3]. As it can be seen in Table 3, T-norm was not effective in improving performance of our full system. Experimentation with the Z-norm and ZT-norm provided results that were mixed and unconvincing. This contradicted the conclusions drawn in [44, 91], where Z-norm or ZT-norm were found necessary for making channel variability modeling techniques really effective. Later it was found that some systems are invariant against normalization and for others it is crucial.

The results are presented for our baseline system which is a "stable" BUT GMM system designed for NIST SRE 2006 evaluation, for more details see [11]. However, lots of work was done at BUT for NIST SRE 2008 (especially in the area of joint factor analysis), the experimental results show that these techniques can perform almost two times better on the same data.

GMM-SVM

In this section results of the best system submitted to NIST SRE08 are presented: We have participated to 2 kind of tasks, the short2-short3 task and the 3conv-short3 task. In the short2-short3 task one session of about 2.5 minutes of telephone speech is used in training and testing. In the 3conv-short3 task 3 sessions of about 2.5 minutes of telephone speech is used in training and 1 session is used for test.

GMM-SVM with TZ-norm. Figures 24, 25 and 26 show that the system is much more powerful for males than for females, even though the system is the same for males and females. The only difference lies in the used world models and eigen-channel matrix $\mathbf{U}$. At this stage we are seeking the causes of this performance shift.

Figures 24 and 25 show that the performance worse in det6 case (all languages) than in det7 case (English only). This performance degradation was expected because the world models are trained on English Only sessions.

Very good performances can be observed in the case of 3conv-short3 condition with respect to the short2-short3 condition (by comparing our results with the results of other sites submissions). This is probably caused by our selection frames which make lot of false reject frames.

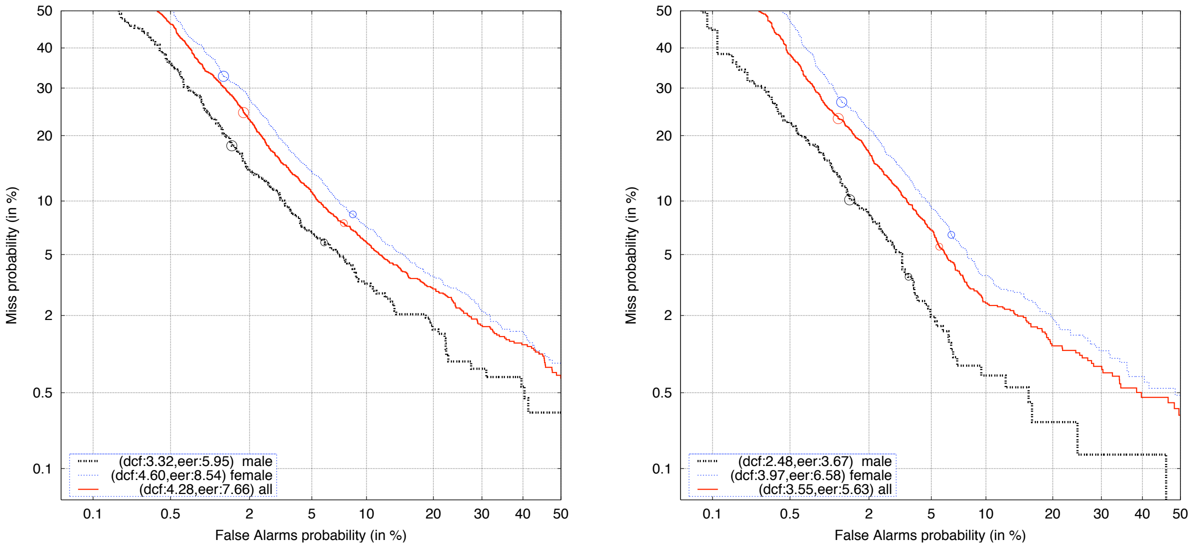
For the 3conv-short3 condition, impostor models, used for score normalization, are trained using only 1 session (instead of 3 sessions). This is also the case of negative examples used for training SVM classifiers. In condition 3conv-short3, the LIA submitted system (SVM-GMM with TZ-norm) was the best one among all participants systems in the case of native English trails (det8).

System	SRE2005		SRE2006	
	EER	DCF	EER	DCF
MFCC+Δ, CMN, 2048 G.	26.6%	.089	23.8%	.088
+ RASTA	14.3%	.055	11.8%	.059
+ Feature Warping	12.4%	.052	10.0%	.051
+ ΔΔ	11.2%	.047	9.1%	.049
+ ΔΔΔ	10.6%	.047	9.3%	.048
+ HLDA (52→39)	9.7%	.042	8.2%	.041
+ Eigenchannel adapt.	4.6%	.020	4.0%	.020
+ T-norm	4.6%	.020	4.0%	.018
Full system, 512 Gauss.	4.9%	.026	4.7%	.024

Table 3: The improvements in performance obtained by successively adding different techniques.

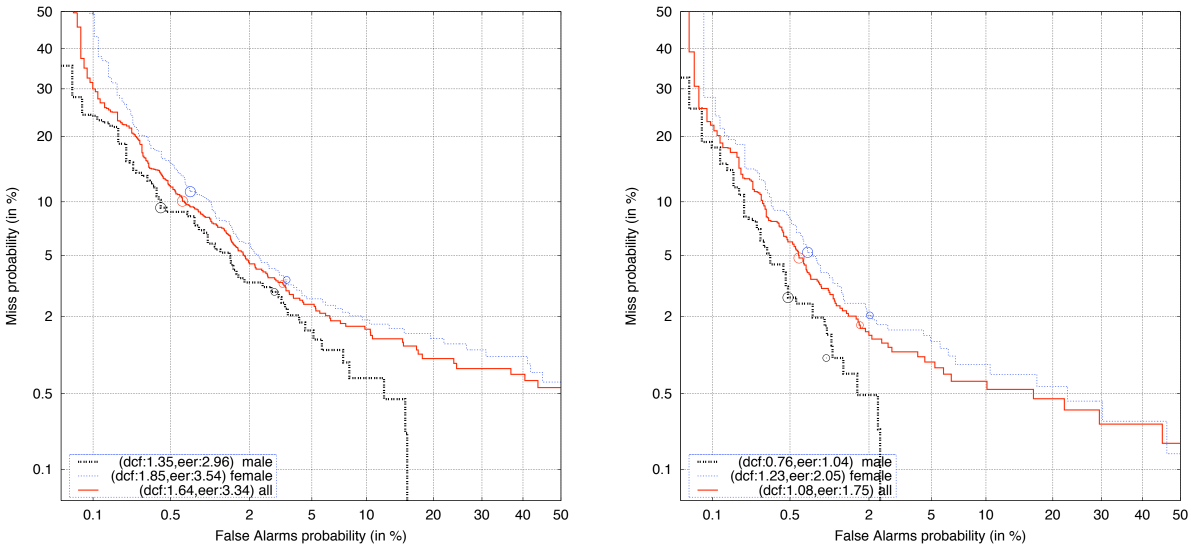
System	BANCA - EER[%]	
	subset G1	subset G2
GMM-UBM	11.06	8.21
GMM-SVN	9.0	9.0

Table 4: Comparison of technique on BANCA database with P protocol.



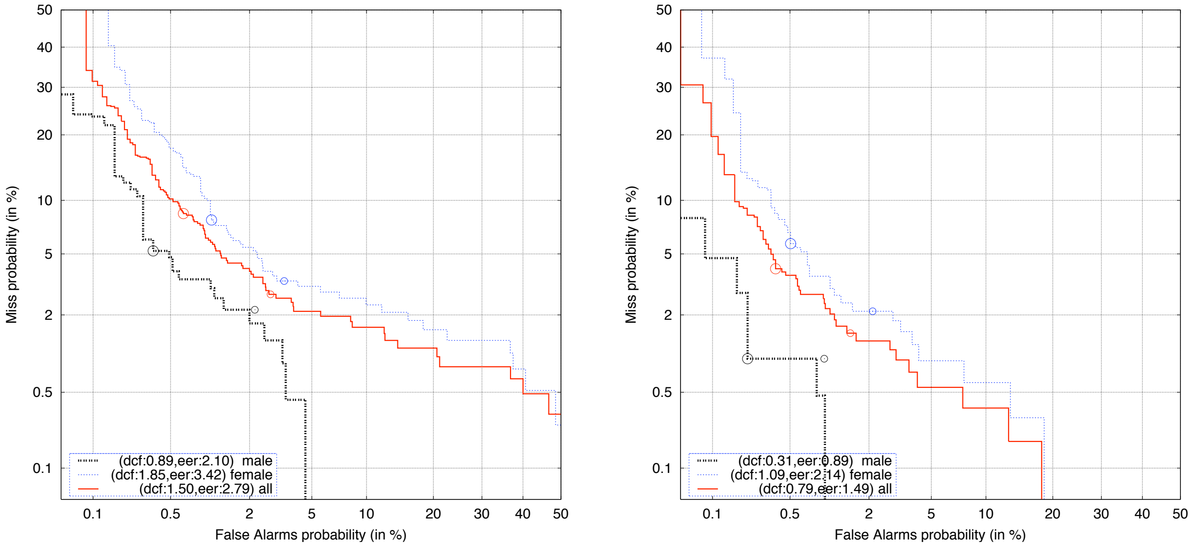
(a) (b)

Figure 24: DET Curves for male, female and gender-independent, all languages trials (det6): (a) short2-short3 condition; (b) 3conv-short3 condition.



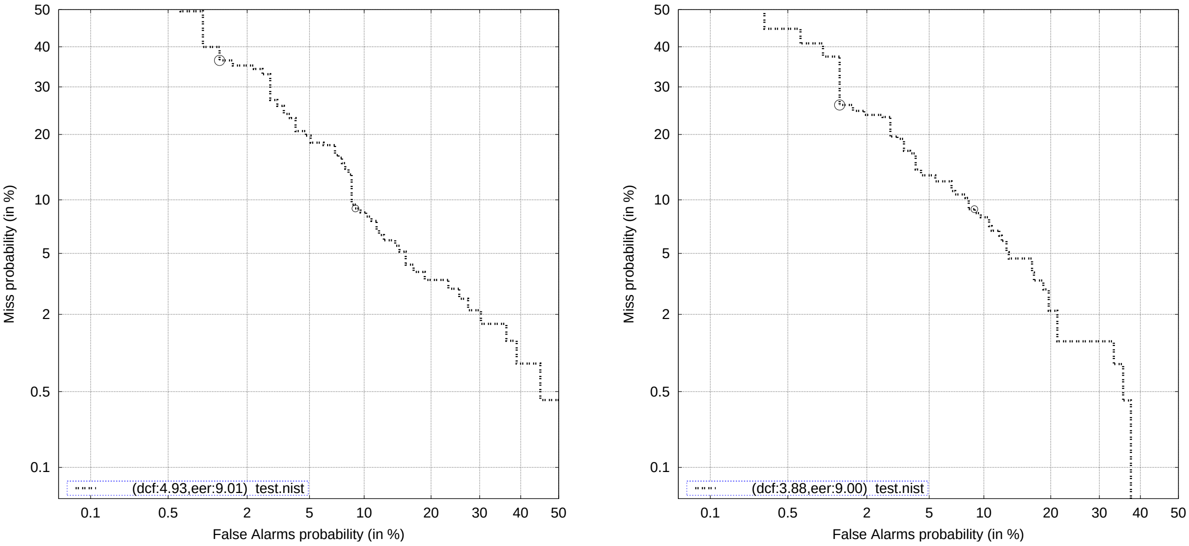
(a) (b)

Figure 25: DET Curves for male, female and gender-independent, only English trials (det7): (a) short2-short3 condition; (b) 3conv-short3 condition.



(a) (b)

Figure 26: DET Curves for male, female and gender-independent, native English trials (det8): (a) short2-short3 condition; (b) 3conv-short3 condition.



(a) (b)

Figure 27: DET curves for the BANCA dataset: (a) group *g1*; (b) group *g2*

7 Summary

This document described some of the state-of-the-art unimodal biometric baseline systems for face and speaker authentication. The report presents detailed implementation, evaluation and results on consituent baseline algorithms on face detection, facial feature localisation & verification techniques for unimodal face authentication and speech/silence detection & verification methods for unimodel speech authentication. The methods were chosen based on: a) a detailed survey of the literature in the repective areas and b) their proven performance capacity to cope with MOBIO's concept. These representative methodologies were selected to perform experimental comparison on the face and speaker authentication tasks using the BANCA database and its corresponding experimental protocol. The mechanisms have also been tested individually using other datasets including the XM2VTS, BioID and BioSign.

Following detailed experimentation, the following conclusions can be made:

- *Face Detection:* Three baseline methods, VJFD, LBP-SVM and c-MCT-C of face detection have been compared in this report. The results on different datasets comparing these techniques clearly reveals that the VJFD method is the best followed by the c-MCT-C and the LBP-SVM schemes. The performance was rated based on low error rates and the minimum missed detections on most datasets.

- *Facial Feature Localisation:* Facial feature localisation is performed through the detection of face points in the image. The CLM technique for face points localisation has been reported in this paper. The evaluation of the CLM model indicated best performances when combined with the VJFD face detector as against the c-MCT-C and LBP-SVM schemes. In general the overall accuracy of the system was also found to be low directly indicating the capacity of the face points localisation algorithm to deliver reliable feature detections.

- *Face Verification:* Finally, unimodal face authentication is accomplished through performing verification experiments integrating the face detection and facial feature localisation algorithms. The face verification systems, DCT-GMM, EFLDM and ULBPH, were tested on the BANCA dataset using the BANCA protocol and the results illustrate that the techniques recorded comparably good performances. The DCT-GMM method presented slightly higher precision in comparison to the other methods.

- *Speaker Verification from speech*: Two baseline methods, GMM based with channel compensation and SVM based with factor analysis have been compared in this report. The results (see Table 4) on BANCA database show that Factor analysis perform slightly better in terms of Equal Error Rate.

It is foreseen that the study of unimodal authentication systems in this report will help building enhanced models of unimodal authentication system for face and speech verification also simultaneously helping the integration of successful schemes to form robust bi-modal systems.

Acknowledgements

The authors would like to thank Tim Cootes (UMAN), Sebastien Marcel and Chris McCool (IDIAP), Matti Pietikäinen (UOULU), Timo Ahonen (UOULU), Abdenour Hadid (UOULU), Josef Kittler (UNIS), Norman Poh (UNIS), Honza Cernocky (BUT), Pavel Matejka (BUT), Driss Matrouf (LIA) and Jean-Francois Bonastre (LIA) for their contributions to and review of this report.

References

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

[2] Timo Ahonen, Abdenour Hadid, and Matti Pietikäinen. Face recognition with local binary patterns. In *Proc. European Conference on Computer Vision (ECCV)*, pages 469–481, 2004.

[3] R. Auckenthaler, M. Carey, and H. Lloyd-Tomas. Score normalization for text-independent speaker verification systems. *Digital Signal Processing*, 10:42–54, 2000.

[4] Enrique Bailly-Baillire, Samy Bengio, Frdric Bimbot, Miroslav Hamouz, Josef Kittler, Johnny Marithoz, Jiri Matas, Kieron Messer, Vlad Popovici, Fabienne Pore, Belen Ruiz, and Jean-Philippe Thiran. *The BANCA Database and Evaluation Protocol*, volume 2688 of *Proc. International Conference on Audio- and Video-based Biometric Person Authentication.* Springer Berlin / Heidelberg, 2003.

[5] S. Baker and I. Matthews. Lucas-Kanade 20 years on: A unifying framework. *International Journal of Computer Vision*, 56(3):221–255, 2004.

[6] P.N. Belhumeur, J.P. Hespanha, and D.J. Kriegman. Eigenfaces vs. fisherfaces: recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):711–720, Jul 1997.

[7] S. Bengio and J. Marithoz. The Expected Performance Curve: a New Assessment Measure for Person Authentication. In *The Speaker and Language Recognition Workshop (Odyssey)*, pages 279–284, Toledo, 2004.

[8] F. Bimbot, J.-F. Bonastre, C. Fredouille, G. Gravier, I. Magrin-Chagnolleau, S. Meignier, T. Merlin, J. Ortega-Garcia, D. Petrovska, and D. A. Reynolds. A tutorial on text-independent speaker verification. *EURASIP Journal on Applied Signal Processing, Special issue on biometric signal processing*, 2004.

[9] N. Brümmer. Spescom DataVoice NIST 2004 system description. In *Proc. NIST Speaker Recognition Evaluation 2004*, Toledo, Spain, June 2004.

[10] N. Brümmer, L. Burget, J. Černocký, O. Glembek, F. Grézl, M. Karafiát, D. A. van Leeuwen, P. Matějka, P. Schwarz, and A. Strasheim. Fusion of heterogeneous speaker recognition systems in the STBU submission for the NIST speaker recognition evaluation 2006. *IEEE Transactions on Audio, Speech and Language Processing*, 15(7):2072–2084, September 2007.

[11] L. Burget, P. Matejka, P. Schwarz, O. Glembek, and J. Cernocky. Analysis of feature extraction and channel compensation in GMM speaker recognition system. *IEEE*

Transactions on Audio, Speech and Language Processing, 15(7):1979–1986, September 2007.

[12] Michael C. Burl, Markus Weber, and Pietro Perona. A probabilistic approach to object recognition using local photometry and global geometry. In *Proc. European Conference on Computer Vision (ECCV)*, pages 628–641. Springer, 1998.

[13] F. Cardinaux, C. Sanderson, and S. Marcel. Comparison of MLP and GMM classifiers for face verification on XM2VTS. In *Proc. International Conference on Audio- and Video-based Biometric Person Authentication*, pages 1058–1059, 2003.

[14] Chi-Ho Chan, Josef Kittler, and Kieron Messer. Multi-scale local binary pattern histograms for face recognition. In Seong-Whan Lee and Stan Z. Li, editors, *ICB*, volume 4642 of *Lecture Notes in Computer Science*, pages 809–818. Springer, 2007.

[15] Chih-Chung Chang and Chih-Jen Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[16] T. F. Cootes, G. J. Edwards, and C. J. Taylor. Active appearance models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6:681–685, June 2001.

[17] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham. Active shape models – Their training and application. *Computer Vision and Image Understanding*, 61(1):266–275, January 1995.

[18] N. Cristianini and J. Shawe-Taylor. *An introduction to Support Vector Machines and other kernel-based learning methods.* Cambridge University Press, 2000.

[19] D. Cristinacce and T. F. Cootes. A comparison of shape constrained facial feature detectors. In *Proc. International Conference on Automatic Face and Gesture Recognition (AFGR)*, 2004.

[20] D. Cristinacce and T. F. Cootes. Facial feature detection and tracking with automatic template selection. In *Proc. International Conference on Automatic Face and Gesture Recognition (AFGR)*, 2006.

[21] D. Cristinacce and T. F. Cootes. Automatic feature localisation with constrained local models. *Pattern Recognition*, 41:3054–3067, 2008.

[22] G. Doddington, M. Przybocki, A. Martin, and D. Reynolds. The NIST speaker recognition evaluation — overview, methodology, systems, results, perspective. *Speech Communication*, 31(2-3):225–254, 2000.

[23] N. Dowson and R. Bowden. Simultaneous modeling and tracking (smat) of feature sets. In *Proc. International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2005.

[24] I. Dryden and K. V. Mardia. *Statistical Shape Analysis*. Wiley, London, 1998.

[25] S. Young et al. *The HTK Book*. University of Cambridge, 2005.

[26] P. Felzenszwalb and D. Huttenlocher. Pictorial structures for object recognition. *International Journal of Computer Vision*, 61(1):55–79, January 2005.

[27] P. Felzenszwalb and D. C. D. Huttenlocher. Spatial priors for part-based recognition using statistical models. In *Proc. International Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, 2005.

[28] Y. Freund, R. Iyer, R. E. Schapire, and Y. Singer. An efficient boosting algorithm for combining preferences. *Journal of Machine Learning Research*, 4:933–969, November 2003.

[29] Y. Freund and R. E. Schapire. A short introduction to boosting. *Journal of the Japanese Society for Artificial Intelligence*, 14(5):771–780, September 1999.

[30] B. Froba and A. Ernst. Face detection with the modified census transform. In *Proc. International Conference on Automatic Face and Gesture Recognition (AFGR)*, pages 91–96, Seoul, Korea, May 2004.

[31] Keinosuke Fukunaga. *Introduction to statistical pattern recognition (2nd ed.)*. Academic Press Professional, Inc., San Diego, CA, USA, 1990.

[32] M. J. F. Gales. Semi-tied covariance matrices for hidden markov models. *IEEE Transactions on Speech and Audio Processing*, 7:272–281, 1999.

[33] Shaogang Gong, Stephen J. McKenna, and Alexandra Psarrou. *Dynamic Vision: From Images to Face Recognition*. Imperial College Press, London, UK, UK, 2000.

[34] A. Hadid, M. Pietikäinen, and T. Ahonen. A discriminative feature space for detecting and recognizing faces. In *IEEE Conference on Computer Vision and Pattern Recognition*, volume II, pages 797–804, 2004.

[35] Abdenour Hadid and Matti Pietikäinen. From still image to video-based face recognition: An experimental analysis. In *Proc. International Conference on Automatic Face and Gesture Recognition (AFGR)*, pages 813–818, 2004.

[36] A. Hatch, B. Peskin, and A. Stolcke. Improved phonetic speaker recognition using lattice decoding. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Philadelphia, PA, USA, March 2005.

[37] H. Hermansky and S. Sharma. Temporal patterns (TRAPS) in ASR of noisy speech. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2427–2431, Phoenix, Arizona, March 1999.

[38] Rui Huang, Dimitris N. Metaxas, and Vladimir Pavlovic. A hybrid face recognition method using markov random fields. In *Proc. International Conference on Pattern Recognition (ICPR)*, volume 3, pages 157–160. IEEE Computer Society, 2004.

[39] M. Isard and A. Blake. ConDensAtion – conditional density propagation for visual tracking. *International Journal of Computer Vision*, 29(1):5–28, August 1998.

[40] Anil K. Jain and Stan Z. Li. *Handbook of Face Recognition.* Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2004.

[41] O. Jesorsky, K. J. Kirchberg, and R. W. Frischholz. Robust face detection using the Hausdorff distance. In *Proc. International Conference on Audio- and Video-based Biometric Person Authentication*, pages 90–95, June 2001.

[42] S. Kajarekar, L. Ferrer, E. Shriberg, K. Sonmez, A. Stolcke, A. Venkataraman, and J. Zheng. SRI's 2004 NIST speaker recognition evaluation system. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Philadelphia, PA, USA, March 2005.

[43] P. Kenny, G. Boulianne, and P. Dumouchel. Eigenvoice Modeling With Sparse Training Data. *IEEE Transactions on Speech and Audio Processing*, 13(3):345, 2005.

[44] P. Kenny and P. Dumouchel. Disentangling speaker and channel effects in speaker verification. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, volume 1, pages 47–40, Montreal, Canada, May 2004.

[45] Tae-Kyun Kim and Josef Kittler. Locally linear discriminant analysis for multimodally distributed classes for face recognition with a single model image. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(3):318–327, 2005.

[46] Tae-Kyun Kim, Josef Kittler, and Roberto Cipolla. Discriminative learning and recognition of image set classes using canonical correlations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(6):1005–1018, 2007.

[47] N. Kumar. *Investigation of Silicon-Auditory Models and Generalization of Linear Discriminant Analysis for Improved Speech Recognition.* PhD thesis, John Hopkins University, Baltimore, 1997.

[48] C. Lee and J. Gauvain. Bayesian adaptive learning and map estimation of hmm. In C.-H. Lee, F. Soong, and K. Paliwal, editors, *Automatic speech and speaker recognition : Advanced topics*, pages 83–107. Kluwer Academic Publishers, Boston, Massachusetts, USA, 1996.

[49] Kuang-Chih Lee, Jeffrey Ho, Ming-Hsuan Yang, and David J. Kriegman. Video-based face recognition using probabilistic appearance manifolds. In *Proc. International Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 313–320, 2003.

[50] Stan Z. Li and ZhenQiu Zhang. Floatboost learning and statistical face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(9):1112–1123, 2004.

[51] Yongmin Li, Shaogang Gong, and Heather Liddell. Support vector regression and classification based multi-view face detection and recognition. In *Proc. International Conference on Automatic Face and Gesture Recognition (AFGR)*, page 300, Washington, DC, USA, 2000. IEEE Computer Society.

[52] Yongping Li. *Linear Discriminant Analysis and its application to Face Identification.* PhD thesis, University of Surrey, 2000.

[53] Chengjun Liu and H. Wechsler. Gabor feature based classification using the enhanced fisher linear discriminant model for face recognition. *IEEE Transactions on Image Processing*, 11(4):467–476, Apr 2002.

[54] Xiaoming Liu. Generic face alignment using boosted appearance model. In *Proc. International Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1–8. IEEE Computer Society, 2007.

[55] Xiaoming Liu and Tsuhan Chen. Video-based face recognition using adaptive hidden markov models. In *Proc. International Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 340–345, 2003.

[56] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[57] S. Lucey and T. Chen. A GMM parts based face representation for improved verification through relevance adaptation. In *Proc. International Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 855–861, 2004.

[58] Topi Mäenpää, Timo Ojala, Matti Pietikäinen, and Soriano Maricor. Robust texture classification by subsets of local binary patterns. In *Proc. International Conference on Pattern Recognition (ICPR)*, volume 3, page 3947, Los Alamitos, CA, USA, 2000. IEEE Computer Society.

[59] A. Martin, G. Doddington, T. Kamm, M. Ordowsk, and M. Przybocki. The DET Curve in Assessment of Detection Task Performance. In *Proc. European Conference on Speech Communication and Technology (EuroSpeech)*, pages 1895–1898, Rhodes, 1997.

[60] D. Matrouf, N. Scheffer, B. Fauve, and J-F. Bonastre. A straightforward and efficient implementation of the factor analysis model for speaker verification. In *INTERSPEECH Conference, Antewerp, Belgium*, 2007.

[61] I. Matthews and S. Baker. Active appearance models revisited. *International Journal of Computer Vision*, 60(2):135–164, 2004.

[62] P. Matějka, L. Burget, P. Schwartz, O. Glembek, M. Karafiát, F. Grézl, J. Černocký, D. A. van Leeuwen, N. Brummer, and A. Strasheim. STBU system for the NIST 2006 speaker recognition evaluation. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2007.

[63] K. Messer, J. Matas, J. Kittler, J. Luettin, and G. Maitre. XM2VTSDB: The extended M2VTS database. In *Proc. International Conference on Audio- and Video-based Biometric Person Authentication*, 1999.

[64] S. Mitchell, B. Lelieveldt, J. Bosch, R. van der Geest, J. Reiber, and M. Sonka. Segmentation of cardiac MR volume data using 3D active appearance models. In *Proc. International SPIE Conference in Medical Imaging*, volume 4684, pages 433–443, 2002.

[65] J. Navratil, Qin Jin, W.D. Andrews, and J.P. Campbell. Phonetic speaker recognition using maximum-likelihood binary-decision tree models. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, volume 4, pages 796–799, Hong Kong, China, April 2003.

[66] J. A. Nelder and R. Mead. A simplex method for function minimization. *Computer Journal*, 7:308–313, 1965.

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

[68] T Ojala, M. Pietikäinen, and T. Mäenpää. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987, July 2002.

[69] OpenCV library. http://opencvlibrary.sourceforge.net/.

[70] Edgar Osuna, Robert Freund, and Federico Girosi. Training support vector machines: an application to face detection. In *Proc. International Conference on Computer Vision and Pattern Recognition (CVPR)*, page 130, Washington, DC, USA, 1997. IEEE Computer Society.

[71] J. Pelecanos and S. Sridharan. Feature warping for robust speaker verification. In *Proc. A Speaker Odyssey*, pages 213–218, Crete, Grece, 2001.

[72] W. B. Pennebaker and J. L. Mitchell. *JPEG still image data compression standard.* New York: Van Nostrand Reinhold, 1993.

[73] D. Reynolds, W. Campbell, T Gleason, C. Quillen, D. Sturim, P. Torres-Carrasquillo, and A. Adami. The 2004 MIT lincoln laboratory speaker recognition system. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Philadelphia, PA, USA, March 2005.

[74] D. A. Reynolds. Comparison of background normalization methods for text-independent speaker verification. In *Proc. European Conference on Speech Communication and Technology (EuroSpeech)*, pages 963–966, Rhodes, Greece, September 1997.

[75] D. A. Reynolds. Channel robust speaker verification via feature mapping. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, volume II, pages 53–56, Hong Kong, China, April 2003.

[76] D. A. Reynolds, T. F. Quatieri, and R. B. Dunn. Speaker verification using adapted gaussian mixture models. *Digital Signal Processing*, 10(1–3):19–41, 2000.

[77] S. Romdhani and T. Vetter. 3D probabilistic feature point model for object detection and recognition. In *Proc. International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007.

[78] C. Sanderson and K. K. Paliwal. Fast feature extraction method for robust face verification. *Electronic Letters*, 38(25):1648–1650, 2002.

[79] H. Schneiderman and T. Kanade. A statistical method for 3D object detection applied to faces and cars. In *Proc. International Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 746–751, Washington, DC, USA, 2000. IEEE Computer Society.

[80] P. Schwarz, P. Matějka, and J. Černocký. Towards lower error rates in phoneme recognition. In *Proc. International Conference on Text, Speech and Dialogue*, pages 465–472, Brno, Czech Republic, September 2004.

[81] P. Schwarz, P. Matějka, and J. Černocký. Hierarchical structures of neural networks for phoneme recognition. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 325–328, Toulouse, France, May 2006.

[82] I. M. Scott, T. F. Cootes, and C. J. Taylor. Improving appearance model matching using local image structure. In *Proc. International Conference on Information Processing in Medical Imaging*, 2003.

[83] A. Solomonoff, W. Campbell, and I. BoardmanCampbell. Advances in channel compensation for SVM speaker recognition. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, volume I, pages 629–632, Philadelphia, PA, USA, March 2005.

[84] A. Stolcke, L. Ferrer, S. Kajarekar, E. Shriberg, and A. Venkataraman. MLLR transforms as features in speaker recognition. In *Proc. European Conference on Speech Communication and Technology (EuroSpeech)*, pages 2425–2428, Lisbon, Portugal, September 2005.

[85] Jilin Tu, Zhenqiu Zhang, Zhihong Zeng, and Thomas Huang. Face localization via hierarchical condensation with fisher boosting feature selection. In *Proc. International Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 2, pages 719–724. IEEE Computer Society, 2004.

[86] B. van Ginneken, M. Stegmann, and M. Loog. Segmentation of anatomical structures in chest radiographs using supervised methods: a comparative study on a public database. *Medical Image Analysis*, 10(1):19–40, February 2006.

[87] S. van Vuuren and H. Hermansky. On the importance of components of the modulation spectrum for speaker verification. In *Proc. International Conference on Spoken Language Processing*, volume 7, pages 3205–3208, Sydney, Australia, May 1998.

[88] T. Vetter and V. Blanz. Estimating coloured 3D face models from single images: an example based approach. In *Proc. European Conference on Computer Vision (ECCV)*, volume 2, 1998.

[89] Paul Viola and Michael J. Jones. Robust real-time face detection. *International Journal of Computer Vision*, 57(2):137–154, 2004.

[90] R. Vogt, B. Baker, and S. Sridharan. Modelling Session Variability in Text-Independent Speaker Verification. In *Proc. European Conference on Speech Communication and Technology (EuroSpeech)*, 2005.

[91] R. Vogt and S. Sridharan. Experiments in session variability modelling for speaker verification. In *Proc. International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, volume 1, pages 897–900, Toulouse, France, May 2006.

[92] Harry Wechsler. *Reliable Face Recognition Methods: System Design, Implementation and Evaluation (International Series on Biometrics).* Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[93] Lior Wolf and Amnon Shashua. Kernel principal angles for classification machines with applications to image sequence interpretation. In *Proc. International Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, pages 635–642, 2003.

[94] J. Xiao, S. Baker, I. Matthews, and T. Kanade. Real-time combined 2D+3D active appearance models. In *Proc. International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2004.

[95] Osamu Yamaguchi, Kazuhiro Fukui, and Ken-ichi Maeda. Face recognition using temporal image sequence. In *Proc. International Conference on Automatic Face and Gesture Recognition (AFGR)*, pages 318–323, 1998.

[96] M.-H. Yang, D. J. Kriegman, and N. Ahuja. Detecting face in images: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(1):34–58, January 2002.

[97] Baochang Zhang, Peng Yang, Shiguang Shan, and Wen Gao. Discriminant gaborfaces and support vector machines classifier for face recognition. In *Proc. Asian Conference on Computer Vision (ACCV)*, pages 37–41, 2004.

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

[99] Y. Zheng, X. S. Zhou, B. Georgescu, S. Zhou, and D. Comaniciu. Example based non-rigid shape detection. In *Proc. European Conference on Computer Vision (ECCV)*, 2006.

[100] Shaohua Kevin Zhou. Face recognition using more than one still image: What is more? In *Proc. 5th Chinese Conference on Biometric Recognition, (SINOBIOMETRICS)*, pages 212–223, 2004.

[101] Shaohua Kevin Zhou, Volker Krüger, and Rama Chellappa. Probabilistic recognition of human faces from video. *Computer Vision and Image Understanding*, 91(1-2):214–245, 2003.

A Evaluation datasets

A.1 BANCA

The BANCA dataset [4] is the principal evaluation dataset for the purposes of this report. It is a collection of multi-modal (video and audio) data in four different European languages (although only English is used in this report) and is intended for evaluating personal identity verification systems.

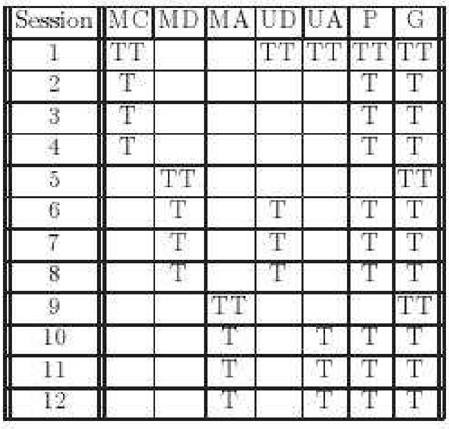
For each language, the dataset principally contains data from 52 subjects that are split into two gender-balanced groups, *g1* and *g2*, with 13 males and 13 females in each group. There is an additional group, *wm*, of 30 subjects (15 male and 15 female) that is used only for building a world model. Each subject participated in 12 recording sessions, grouped into three different scenarios:

- **Controlled (Sessions 1-4)**: Captured in a controlled environment (quiet office) using a high quality camera and two microphones (high and low quality).
- **Degraded (Sessions 5-8)**: Captured in a less-controlled environment (an occupied office) using a low quality camera (a web-cam, in this case) and two microphones (high and low quality).
- **Adverse (Sessions 9-12)**: Captured in an uncontrolled environment (staff cafeteria) using a high quality camera and two microphones (high and low quality).

In each recording session, two recordings were captured: (a) a client access where the user matched their claimed identity; (b) an impostor attack where the user did not match their claimed identity. The impostor attacks were balanced so that each client every other client once only (conversely, every client is attacked once only). Furthermore, the attacks were balanced so that there are an equal number of attacks for a given client/impostor within each condition (controlled, degraded, adverse).

The high quality video data consists of PAL (i.e. 720 × 576 pixels) video at a colour sampling resolution of 4:2:0, compressed at a ratio of 5:1. The low quality video was upsampled in resolution to match that of the high quality video. The audio data was captured at a frequency of 32 kHz in both 12-bit (low quality) and 16-bit (high quality), and was not compressed.

An evaluation protocol defines a set of data, how it should be used by a system to perform a set of experiments and how the performance should be computed such that no bias is introduced. In this case, the BANCA protocol was intended to the development of *open-set* verification systems such that new clients can be added to the test list wihout having to redesign the verification system (including parameters etc.). This is achieved by splitting the data into two sets: the *development* set used to calibrate the system is calibrated with



Session	MC	MD	MA	UD	UA	P	G
1	TT			TT	TT	TT	TT
2	T					T	T
3	T					T	T
4	T					T	T
5		TT					TT
6		T		T		T	T
7		T		T		T	T
8		T		T		T	T
9			TT				TT
10			T		T	T	T
11			T		T	T	T
12			T		T	T	T

Figure 28: Usage of different sessions in BANCA configurations

the knowledge of the subjects identity in the test phase; and the *evaluation* set with the fixed calibrations obtained from the development, with which training (enrollment) and testing (access) is to be conducted. For fusion verification systems, a third set called the fusion tuning set is introduced.

In BANCA, the following distinct experimental configurations have also been specified: Matched Controlled (MC), Matched Degraded (MD), Matched Adverse (MA), Unmatched Controlled (UC), Unmatched Degraded (UD), Unmatched Adverse (UA), Pooled test (P) and Grand test (G). The table in Figure 28 summarises the usage of the different sessions in each configuration. "TT" refers to the client training while "T" depicts the client and imposter test sessions. For example, in configuration MC the true client data from session 1 is used for training and the true client data from sessions 2, 3 and 4 are used for testing. All the imposter attack data from all sessions are used for imposter testing.

It is anticipated that from the measure performances on all configurations, it is possible to deduce information regarding the intrinsic performance given a particular condition, performance in varied condition and a potential gain from more representative training condition. The protocol also specifies the use of the false acceptance and the false rejection rates as measures of performance.

A.2 XM2VTS

The XM2VTS dataset [63], containing 295 subjects, is an extension of the M2VTS dataset and was designed for the development of personal identity verification systems. The data were recorded in four sessions, spread out over five months, in order to capture variation in appearance (e.g. hairstyle, wearing glasses etc.). Each session captured two PAL video

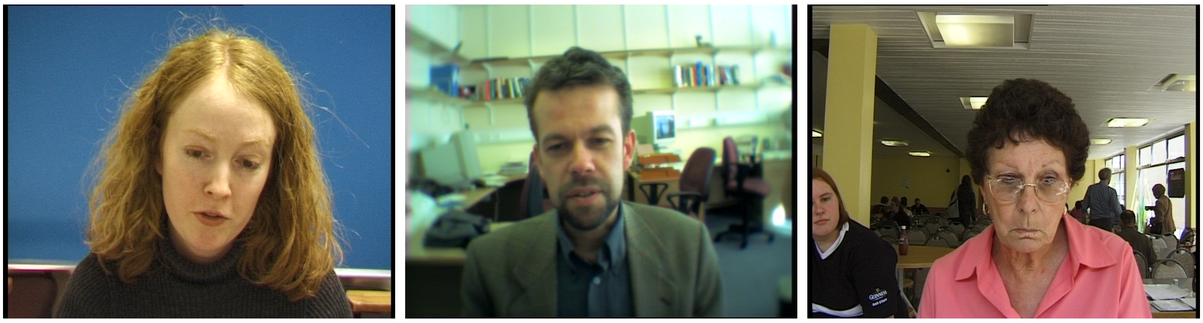


Figure 29: Examples from the BANCA dataset, representing (from left) controlled, degraded and adverse capture conditions.

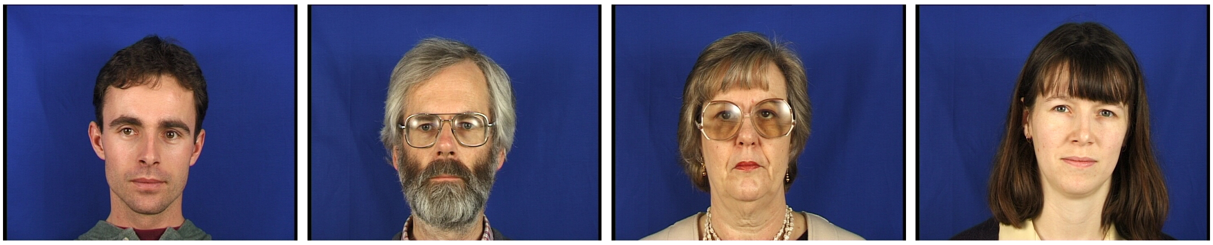


Figure 30: Examples from the XM2VTS dataset.

sequences of each subject: (i) a frontal view of the subject reading pre-defined text and (ii) a sequence where the subject rotates the head. For the purposes of this report, however, we use only two still images (720 by 576 pixels) of each subject per session (see Figure 30) in order to evaluate the face detection and point localisation modules (a total of 2360 images). In addition to the image data, a dense labelling of 68 facial features was available with which to evaluate the face point localisation module.

A.3 BioID

The BioID face dataset, containing 23 subjects, consists of 1521 greyscale images at a resolution of 384 by 286 pixels (see Figure 31). Each image shows a frontal view of one of the test subjects in an indoor environment and is manually labelled with the locations of the eye centres. As a result, this dataset is used principally to evaluate the face detector.

A.4 BioSign

The BioSign dataset, containing 50 subjects, consists of 550 colour images at a resolution of 320 by 240 pixels (see Figure 32). Each image, captured using a mobile phone, shows a frontal view of each subject at a range of locations (both indoor and outdoor) and is labelled

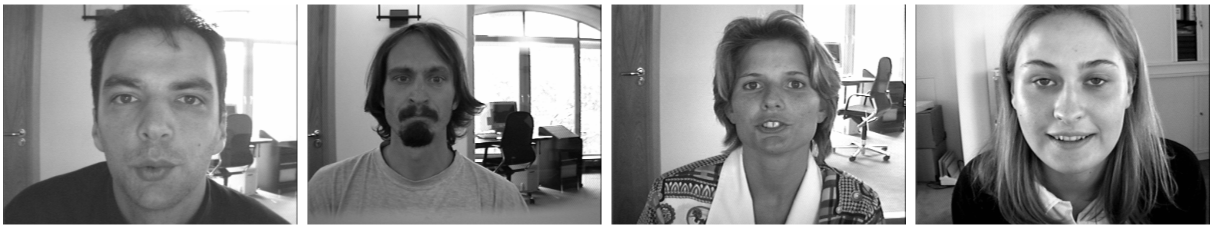


Figure 31: Examples from the BioID dataset.

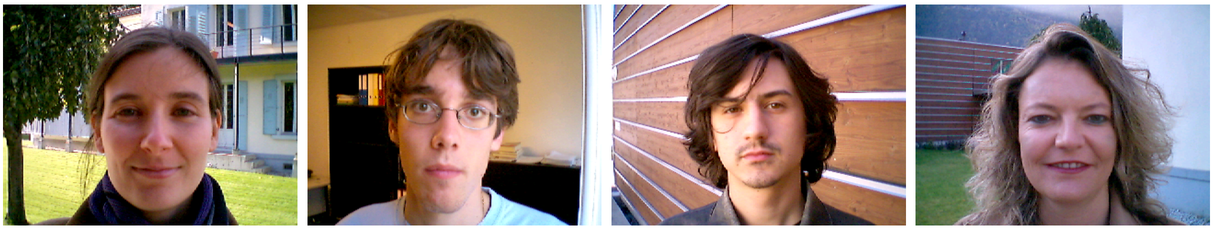


Figure 32: Examples from the BioSign dataset.

with the locations of the eye centres and corners of the mouth. As a result, this dataset is used to evaluate both face detection and (to a limited extent) face point localisation.

B Parsing tools

In order for partners to develop system components independently, we use a system whereby each component is developed as a stand-alone command-line executable. The output of one component is saved as a text file that serves as the input to the next component in the processing chain. Text file input/output was provided via a software library written by UMAN.

B.1 Global input file

All tasks use a common `mobio_inputfile` object that defines:

- input/output directories
- file extension of the input files (so that both video and audio can be tested)
- IDs of the participants to be enrolled (with corresponding input filenames)
- authentication challenges (defined by a true ID with corresponding files, plus the claimed IDS to be tested against)

An example is given below:

```
input_dir: ./videos/
model_dir: ./models/
output_dir: ./output/
file_ext: avi

enrol:
{
  ID: { name: person0 videos: { face0_1 face0_2 face0_3 } }
  ID: { name: person1 videos: { face1_1 face1_2 face1_3 } }
  ID: { name: person2 videos: { face2_1 face2_2 face2_3 } }
}

tests:
{
  test: { ID: { name: person0 videos: { face0_4 face0_5 } }
          challenge: { person0 person1 } }
  test: { ID: { name: person1 videos: { face1_4 face1_5 } }
          challenge: { ALL } }
}
```

B.2 Face detection output

The face detection component reads a mobio_inputfile object as input in order to specify global parameters. The output of face detection is one mobio_video_faceboxes object (stored as <output_dir>/<videoname>_faceboxes.txt) per video processed. This file contains a list of frames, where each frame contains a list of faceboxes. Each facebox is specified by its centre (in x and y), size (in x and y) and score (indicating the strength of the detector output):

```
filename: face0_1
frame_length: 40
start_frame: 0
frames: {
  frame:  { facebox: { cx: 11 cy: 12 wx: 20 wy: 30 score: 0.95 } }
  frame:  { facebox: { cx: 21 cy: 22 wx: 20 wy: 30 score: 0.97 }
            facebox: { cx: 51 cy: 52 wx: 20 wy: 30 score: 0.92 } }
  frame:  { }
  frame:  { facebox: { cx: 11 cy: 12 wx: 20 wy: 30 score: 0.95 } }
  ...
}
```

B.3 Feature localization output

The feature localization module reads a mobio_inputfile object (that specifies global parameters) and one mobio_video_faceboxes object (created by the face detection module) per video. The output of feature localization is a mobio_video_face_points object (stored as <output_dir>/<videoname>_facepts.txt) per video processed. Like the mobio_video_faceboxes object, this contains a list of frames where each frame contains a list of faces (each corresponding to a facebox in <videoname>_faceboxes.txt). Each face contains a score (based on the feature locations, as opposed to the detection score) and a list of feature locations (x-position, y-position and feature score):

```
filename: face0_1
frame_length: 40
start_frame: 0
frames: {
  frame:
  {
    face:
    {
      score: 0.89
      points:
      {
        pt: { x: 0 y: 1 score: 2 }
        pt: { x: 1 y: 2 score: 3 }
        pt: { x: 2 y: 3 score: 4 }
      }
    }
  }
  frame:
  {
    face: { ... }
    face: { ... }
  }
  frame:  { }
  ...
}
```

B.4 Face authentication output

The face authentication module reads a mobio_inputfile object (that specifies global parameters) and one mobio_video_face_points object (created by the feature localization module) per video. The output of face authentication is a mobio_outputfile object (stored as <output_dir>/<videoname>_results.txt) per video that stores the video name, the true identity corresponding to the video and a score structure for each claimed ID.

Videos are divided into segments for video/audio processing. Each segment is assigned a score. These scores are then combined for the total score of the claimed ID against the video:

```
filename: face0_1
actual_id: person0
results:
{
  result:
  {
    claimed_id: person0
    total_score: 0.87
    segment_length: 40
    start_time: 0
    segments:
    {
      segment: { score: 0.42 }
      segment: { score: 0.42 }
      segment: { score: 0.42 }
      segment: { score: 0.42 }
      segment: { score: 0.42 }
    }
  }
  result:
  {
    claimed_id: person1
    ...
  }
  ...
}
```