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]

D3.4: Description and evaluation of advanced algorithms for uni-modal authentication

Due date: 31/10/2009 **Submission date:** 30/10/2009
Project start date: 01/01/2008 **Duration:** 36 months
WP Manager: Tim Cootes **Revision:**

Author(s): Timo Ahonen (UOULU)

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

D3.4: Description and evaluation of advanced algorithms for uni-modal authentication

Abstract:

This deliverable describes and evaluates the advanced unimodal systems for face detection, face point localisation, face verification and speech verification that were developed within the MOBIO project. In face detection, the presented advanced algorithms are based on using novel features and more efficient detection post-processing techniques. The proposed face point localisation method introduces a new local and global constraint model for the point locations. In face verification, advanced feature extraction and score normalization strategies are employed, whereas the advanced speech verification systems are based on using the joint factor analysis. All the developed algorithms are evaluated against the baseline systems described in project deliverable D3.2, showing clear improvement in verification performance both in the face and the speech modality.

Contents

KEY

1. AMS - Adaptive Mean Shift
2. ASM - Active Shape Model
3. CFD - Context-based Face Detector
4. CLM - Constrained Local Model
5. c-MCT-C - cascaded-Modified Census Transform-Classifier
6. DCT - Discrete Cosine Transform
7. DET - Detection Error Trade-off
8. EFLDM- Enhanced Fisher Linear Discriminant Model
9. EM - Expectation Maximization
10. EPC - Expected Performance Curve
11. FA - Factor Analysis
12. FAR - False Acceptance Rate
13. FLD - Fisher Linear Discriminant
14. FRR - False Rejection Rate
15. GMM - Gaussian Mixture Model
16. HLDA - Hierarchical Linear Discriminant Analysis
17. HMM - Hidden Markov Models
18. HTER - Half Total Error Rate
19. JFA - Joint Factor Analysis
20. LBP - Local Binary Pattern
21. LFB-GMM - Local Frequency Band Gaussian mixture model
22. LDM - Linear Discriminant Model
23. LFA - Latent Factor Analysis
24. LFCC - Linear Frequency Cepstral Coefficients

25. LLR - Log Likelihood Ratio
26. LPQL - Local Phase Quantization Label face detector
27. MCT - Modified Census Transform
28. MFCC - Mel Frequency Cepstral Coefficients
29. MRF - Markov Random Field
30. PCA - Principle Component Analysis
31. PS_MLBPHLDA_tnorm - Multi-scale Local Binary Pattern Histogram Discriminant Analysis with Score Normalization
32. SVM - Support Vector Machines
33. UBM - Universal Background Model
34. VAD - Voice Activity Detection
35. VJFD - Viola Jones Face Detector

1 Introduction

The MOBIO project aims at developing new, effective methods for face and speech based authentication in the context of portable devices. The project studies both uni-modal authentication using either face or speech modality, and bi-modal authentication combining these two modalities.

The components of face authentication system considered in the project are face detection, face point localization and face verification. For the speech authentication system, voice activity detection and speaker verification components are needed.

Existing state-of-the-art methods for these tasks were implemented for the project deliverable *D3.1: Baseline systems for uni-modal authentication.* These methods were described and evaluated in *D3.2: Report on the description and evaluation of baseline algorithms for unimodal authentication* [5].

The project deliverable *D3.3: Advanced systems for uni-modal authentication* builded on D3.1, aiming at developing novel methods for face and speech based authentication that exceed the baseline systems in performance. The purpose of this report is to describe and evaluate the advanced systems and perform comparison to baseline systems to assess the progress achieved in the project.

1.1 Face Authentication

The goal of face authentication is to determine, based on facial images of a user, whether or not a claimed identity is true.

Aiming for a complete face authentication subsystem, methods for three necessary modules in the processing chain have been studied: (1) face detection, (2) facial feature localization and (3) face verification.

The face detection module determines if there are faces in an input image and returns their locations and sizes. Three baseline face detectors were presesented in D3.2:

- **VJFD**: The Viola-Jones face detector
- **LBP-SVM**: Local Binary Pattern features combined with Support Vector Machines
- **c-MCT-C**: A cascade of Modified Census Transform features based classifiers

In this report, two advanced systems are described and evaluated against the baseline systems:

- **LPQL**: Local Phase Quantization Label face detector
- **CFD**: Context-based Face Detector

The second module in the processing chain, face point localization, aims to determine the exact locations of anatomical landmarks, such as the eyes, using the output of face detector as a starting point. D3.2 presented Constrained Local Models (CLM) as the baseline face point localization method. In this report, an advanced system combining local and global shape models [55] is presented.

After the face has been localized and possibly geometrically normalized using the detected landmarks, the actual face verification is performed. Here the goal is to accept or reject the identity claimed by the user based on the input face images and a model of the user. The baseline face verification methods are the following:

- **EFLDM**: An Enhanced Fisher linear discriminant model
- **PB-GMM**: A parts-based Gaussian mixture model
- **ULBPH**: A uniform local binary pattern histogram-based method

This report presents two advanced systems:

- **LFB-GMM**: Local Frequency Band Gaussian mixture model
- **PS_MLBPHLDA_tnorm**: Multi-scale Local Binary Pattern Histogram Discriminant Analysis with Score Normalization for robust face recognition

1.2 Speech Authentication

In speaker authentication, the goal is to accept or reject the claimed identity of a user based on a speech utterance. For this purpose, voice activity detection followed by speaker verification are needed.

The purpose of voice activity detection (VAD) is to detect speech frames from an input speech signal. D3.2 presented an compared two methods for voice activity detection (GMM-VAD and NN-VAD) and as their performance is adequate for the purposes of speech authentication, no further work on voice activity detection was done but the baseline systems were used.

Speaker verification system either accepts of rejects the claimed user identity based on the speech frames from VAD subsystem and a model of the user voice. The two baseline speaker verification systems of D3.2 were

- **GMM-SVM**: Factor analysis with support vector machines and
- **GMM-UBM**: GMM-UBM with channel compensation technique.

In this report, two advanced systems based on factor analysis are described:

- **GMM-LFA**: GMM-Latent Factor Analysis and
- **GMM-JFA**: GMM-Joint Factor Analysis.

1.3 Evaluation

To allow for direct comparison to performance of baseline systems reported in D3.2, the same evaluation protocols are used. Specifically, the face detection and face point localisation methods are compared using BANCA, XM2VTS, BioSign and BioID face image datasets and comparing the system outputs to manually annotated groud truth. Face verification systems are evaluated using BANCA image data and speech verification systems using the speech data from BANCA database. For details of the datasets, see Appendix A of D3.2.

2 Face Detection

The aim of face detection is to determine if there are any faces in an input image and return their location and size. Face detection is typically the first step in a face authentication system, and its result is cruicial for a successful authentication. Two new face detectors are described in this report, namely the Context-based Face Detector (CFD) developed at IDIAP research institute and Local Phase Quantization Label (LPQL) face detector developed at University of Oulu.

2.1 Related work

Deliverable D3.2 [5] prerented a review of existing methods for face detection, and the reader is referred to that report for an introduction to face detection methods.

Recently, clearly more publications in computer vision venues have been targeting general object recognition than face detection specifically. There have however been some noteworthy works on face detection as well. Destrero et al. [17] considered the problem of feature selection in face detection and face recognition, using a regularization approach aiming at a sparse set of features. Chen et al. [13], on the other hand, presented a method for creating artificial samples for training a face detector. Butko and Movellan [9] aimed at speeding up the face detection process by simulating the visual search in humans and were able to double the speed of the well known Viola-Jones detector. A face detector for uncontrolled outdoor conditions for privacy protection purposes was presented in [23], achieving a detection rate of about 90 %.

2.2 Sliding window detectors

Like most recent face detectors, the systems presented in this report are based on the sliding window approach, which is reviewed in the following.

To detect objects one usually proceeds by *scanning* the image at different positions and scales. At each position and scale a subwindow is formed and tested against a classifier previously trained with geometric normalized samples of size $S_c = (W_c, H_c)$. This is often refered to as a *sliding window approach*.

There are two main sliding window (scanning) methods: the multiscale and the pyramid [46]. The *multiscale* approach varies the size of the scanning subwindow and the classifier has to interpolate its content to S_c in order to decide if the subwindow contains the object or not. The *pyramid* approach computes a set of scaled versions of the original image and for each one varies the position of a S_c fixed size subwindow. No interpolation is needed for this approach (the subwindow and the classifier have the same size), but the image pyramid must be computed first. It can be shown that both methods test the same number of subwindows and the experimental results have shown that they produce similar results.

For any sliding window approach the total number of subwindows to classify is quadratic to the number of pixels. This is because every position in the image must be matched at every possible scale. Therefore, there can be billions of subwindows even for small images and thus it is inefficient to do an exhaustive search. Typically one uses some heuristics that reduce this number to practical values by limiting the number of scales or searching every N pixels (by having an offset of N pixels between subsequent subwindows) [56]. Real-time performance can be achieved by limiting the number of subwindows to process but at the cost of missing some objects.

Usually applying the sliding window approach with any face detector will result in multiple detections and false alarms (see Fig. 1). These detections must be further processed, this is often refered to as *pruning* false alarms and *merging* multiple detections [46].

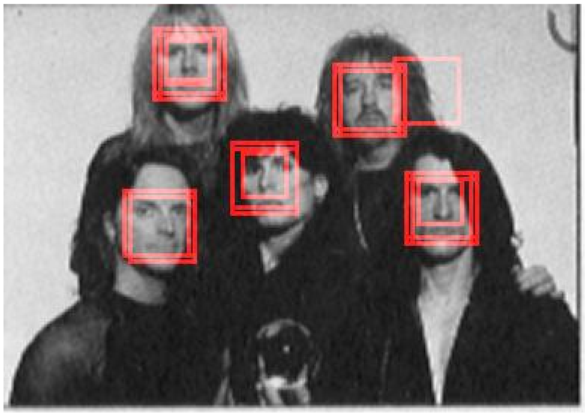

Figure 1: Typical face detections using the multiscale approach and the MCT boosting cascade classifier described in [21] (without clustering multiple detections and removing false alarms).

Since the true location and the number of objects is not known it is prefered to have a finer scanning than to miss any object or misspredict its location too much. However this will also increase the number of false alarms because a larger number of subwindows has to be explored. This is because even a state of the art classifier has a false acceptance rate (FAR) that is not zero, usually of the order of 0.1% to achieve good performance.

2.3 Context-based face detection

2.3.1 Overview

The Idiap advanced face detection system is based on modelling the detections from a face detector. This is a generic technique that could be applied to any object detection problem (provided there is an associated object classifier). The goal is to classify false alarms and merge multiple detections in a principled way without using heuristics. A classifier is built using the contextual information obtained from the object classifier to discriminate between false alarms and true detections. The contextual information is represented by the detection distribution around a target subwindow, this context can then be used to iteratively refine the detections. Finally the detections are clustered using a modified version of the Adaptive Mean Shift algorithm.

2.3.2 Handling multiple detections

The most common approach to solve the multiple detection problem is to heuristically merge them based on the overlapping percentage. Below we present the heuristic methods that have been presented in the literature.

In [56] detections are partitioned into disjoint subsets, by associating two detections to the same one if they overlap. The final step consists of composing for each subset just one subwindow having the average coordinates of all subwindows in that subset. Some restrictions can be further imposed on the partitioning: two detections are considered in the same subset if they overlap more than a threshold (typically 60% of the area of the bigest) and a detection is removed if it is contained by another one. The selection of the best detections (one per subset typically) is done iteratively. At each step the most overlapping detections are merged using a score weighted average and then the subsets are computed again.

Our experiments have shown that this method is sensitive to the scanning parameters: when too coarse, the true detections can be isolated and considered as false alarms and when too fine, false alarms appear in clusters and are usually considered as a final detection. Some variations are implemented in face detection libraries like *OpenCV* or *Torch3vision*, which we refer to as HMergeO, and HMergeT respectively.

A similar heuristic approach has been proposed by Rowley et al. [47]. They preserve the detections with higher number of overlapping detections within a small neighborhood and eliminate the other ones. The final output is given by the centroids of the preserved detections.

A more principled approach was recently proposed in [52] and [53] where the authors study the score distribution in both location and scale space. Their experimental results have shown that the score distribution is significatively different around a true object location than around a false alarm location, thus making possible to build a model to better distinguish the false alarms and enhance detection. This approach is motivated by the fact that the object classifier is usually trained with geometric normalized positive samples and it does not process the *context* (area around given samples). Also, some false alarm subwindows may have a higher score than a true detection nearby and may be selected as the final detections using an heuristic merging technique.

2.3.3 Our approach

The advanced Idiap system is a new generic method for face (object) detection that makes use of the contextual information for improving detection accuracy. We address some problems of the previous methods such as: multiple detections, false alarms and sensitivity to scanning parameters (it should work equally well for fine or coarse face scanning).

We build a context-based model used for pruning false detections based on the work of [52] and [53]. Similarly we define the *context* as the detection distribution around a target subwindow, by varying its scale and position and checking it against a classifier. The context is described using multiple features such as its density, the geometric distribution for scale and position axis and some score statistics. Our model automatically selects the best features from the contextual information and optimizes its internal parameters.

We also use this contextual information to improve the object detections. We argue that we can estimate from the context the direction where an object is more likely to reside and using this idea in a greedy way it is possible to devise an algorithm for refining object detections.

Finally, a modified version of the Adaptive Mean Shift (AMS) is used to solve the problem of clustering multiple detections in a more principled way. The advantage over previous methods is that it uses practically no parameters, it has no heuristics, the number of clusters does not need to be known apriori and its properties are theoretically established.

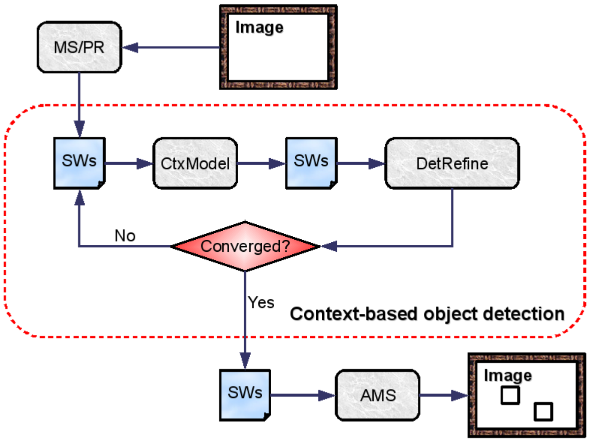

Figure 2: Context-based face (object) detection method consisting of the following blocks: i) CtxModel - the context-based model to discriminate and remove false alarms, initialized with the multiscale or pyramid scanning (MS/PR) and ii) DetRefine - one step of the method to refine detections. The final block is the AMS clustering algorithm to process the converged collection of subwindows (SWs).

The proposed context-based face detection method is presented in Fig. 2. The first step is to run the multiscale or the pyramid (MS/PR) scanning over the input image using a face (object) classifier. The detections (SWs) are checked against the context-based model (CtxModel) to remove false alarms. These are further refined and a new collection of subwindows is generated. If the refinement has not converged, the subwindows are again checked against the context-based model. Otherwise they are clustered using the AMS algorithm to merge the detections that were converged closely.

2.4 Local Phase Quantization Labels (LPQL) face detector

The Local Phase Quantization Label (LPQL) face detector is based on the discrete labels produced by the Local Phase Quantization (LPQ) operator [42]. The LPQ operator, originally developed for blur tolerant texture recognition, has shown good performance in recognition of textures even when there is no blur [42] as well as in face recognition [2, 12]. The prior works utilizing LPQ operator have, however, considered only histograms of LPQ labels. In face detection, the LPQ labels are used directly in a boosting framework.

The LPQ operator is based on examining the local phase in $(2R+1)$-by-$(2R+1)$ neighborhoods $\mathcal{N}_{(x,y)}$ at each pixel position (x, y) of the image $I(x, y)$. These local spectra are computed using a short-term Fourier transform defined by

$$F(u, v, x, y) = \sum_{s=-R}^{R} \sum_{t=-R}^{R} I(x - s, y - t) e^{-j2\pi(us+vt)}. \tag{1}$$

This transform is separable, so the computation of STFT coefficients can be done efficiently using simply 1-D convolutions for the rows and columns successively.

Following the procedure by [42], local Fourier coefficients are computed at four frequency points $(\beta, 0)$; $(0, \beta)$; (β, β); $(\beta, -\beta)$, where β is a small scalar. The phase information in the Fourier coefficients is recorded by observing the signs of the real and imaginary parts of each component in $F(u, v, x, y)$, i.e. quantizing the phase into 4 discrete levels. For this, the coefficients are first reordered into a vector

$$\mathbf{I}(x, y) = \begin{bmatrix} \mathrm{Re}\{F(\beta, 0, x, y)\} \\ \mathrm{Im}\{F(\beta, 0, x, y)\} \\ \mathrm{Re}\{F(0, \beta, x, y)\} \\ \mathrm{Im}\{F(0, \beta, x, y)\} \\ \mathrm{Re}\{F(\beta, \beta, x, y)\} \\ \mathrm{Im}\{F(\beta, \beta, x, y)\} \\ \mathrm{Re}\{F(\beta, -\beta, x, y)\} \\ \mathrm{Im}\{F(\beta, -\beta, x, y)\} \end{bmatrix},$$

which is then encoded into a discrete label by observing the sign of each of the components

$$I_{LPQ}(x, y) = \sum_{j=1}^{8} s(I_j(x, y)) 2^{j-1} \tag{2}$$

where $I_j(x, y)$ is the j-th component of the vector $\mathbf{I}(x, y)$ and $s(z)$ is the thresholding function

$$s(z) = \begin{cases} 1, & z \geq 0 \\ 0, & z < 0 \end{cases} \tag{3}$$

To detect faces an image, the sliding window approach using cascade of GentleBoost based classifier is applied. A subwindow scans the image at different scales and locations, and each subwindow is fed into a cascade of classifiers.

The classifiers in the cascade are based on a sum of functions (weak classifiers). One such funcion f_m maps the possible LPQ codes at one specific neighborhood size R_m and location (x_m, y_m) within the window into real values:

$$f_m(w_{LPQ}) = w_{LPQ}(x_m, y_m, R_m) \mapsto \mathcal{R}, \tag{4}$$

and the decision of the classifier is

$$\text{sign}\left(\sum_m f_m(w_{LPQ}) - t\right). \tag{5}$$

The functions f_m are learned using the GentleBoost approach in a similar manner to Multi-block LBP face detector described in [61].

2.5 Evaluation

To allow for direct comparison between advanced systems presented above and the baseline systems, the same performance measures as in D3.2 are used. The performance measure is based on predicted locations of eye centers, p_l and p_r, and the corresponding ground truth locations $q_l$and q_r. The normalised maximum distance is then used as the performance measure:

$$d_{max} = \frac{\max(|p_l - q_l|, |p_r - q_r|)}{|q_l - q_r|}. \tag{6}$$

The median and 90th percentile statistics are reported for each test image dataset as a performance measure in addition to the number of missed detections.

2.6 Results

The results of evaluating the performance of the advanced face detectors as well as the results for the baseline face detectors are shown in Table 1.

The results show that the performance of advanced face detectors is higher than that of the baseline detectors. It should be noted that, as already reported in D3.2, the Viola-Jones detector shows very good performance and in some of the datasets, it still gives the best performance.

On the BANCA dataset, the Context-Based Face Detector results in the smallest number of missed faces, followed by the Local Phase Quantization Labels detector. Measured by the detection accuracy, the Viola-Jones still performs best, however the difference to LPQL detector is very small.

LPQL and VJ detector show very similar performance on the BioID and BioSign datasets, and the number of missed faces by the CFD is somewhat higher. On the XM2VTS dataset, LPQL detector shows better accuracy and fewer missed detections than the VJ 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
	CFD	105 (1.7%)	0.19	0.28
	LPQL	129 (2.1%)	0.092	0.16
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
	CFD	39 (1.7%)	0.17	0.23
	LPQL	4 (0.17%)	0.097	0.16
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
	CFD	535 (35%)	0.2	0.28
	LPQL	26 (1.7%)	0.094	0.16
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
	CFD	60 (11%)	0.17	0.24
	LPQL	5 (0.91%)	0.092	0.14

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.

3 Face Point Localisation

The aim of this module is to determine the exact locations of anatomical landmarks on the surface of the face (*e.g.* eyes, nose, corners of the mouth). Having recovered these locations, we can stretch the image of the face, as if it were printed on a sheet of rubber, so that the feature locations are shifted to 'normalized' locations. This has shown to be effective in adjusting for small rotations of the head and facial expression, thus potentially making verification more robust.

This report presents the advanced system developed at UMAN that combines a *local* model of shape, encapsulated by a small number of constraints between neighbouring features, with a *global* shape model that enforces constraints between every pair of features on the face [55]. The motivation behind this is to ensure that sensible candidates are selected for the feature locations *before* regularization and to permit solutions that are close to but not exacty within the shape space, allowing a better fit to the data.

3.1 Related work

Since a detailed review of existing methods was presented in D3.2 (Section 3.1), here we refer again only the most relevant works. In particular, this method builds on the Active Shape Model (ASM) [14] and Pictoral Structure Matching (PSM) [20] methodologies to implement guided candidate selection before regularization.

Under the ASM paradigm, feature locations are first predicted in the image (*e.g.* based on the output of a face detector). For each feature, nearby matches are searched within a region around the predicted position, using a measure such as normalized correlation to determine goodness of fit. Since it is not computationally feasible to evaluate the probability of every set of candidates, only the best match for each feature is retained and this set of matches regularized using a global shape model derived from PCA. This process is iterated to convergence within a multi-scale framework.

In contrast, under the PSM paradigm *every* image location is considered a candidate as opposed to just the best match for each feature. To find the most probable set of candidates, the PSM uses an approximation to the global shape model that applies constraints only between pairs of neighbouring points in a tree structure. As a result, dynamic programming can be used to solve this Markov Random Field (MRF) and find the most probable set of candidates efficiently.

The shortcoming of the ASM is that it does not account for spatial constraints when selecting candidates such that one false match can corrupt all feature locations following regularization (where error is distributed approximately evenly among all points). The shortcoming of the PSM is that it uses only an approximation to the true probability distribution over shape and therefore can permit more variability than desired.

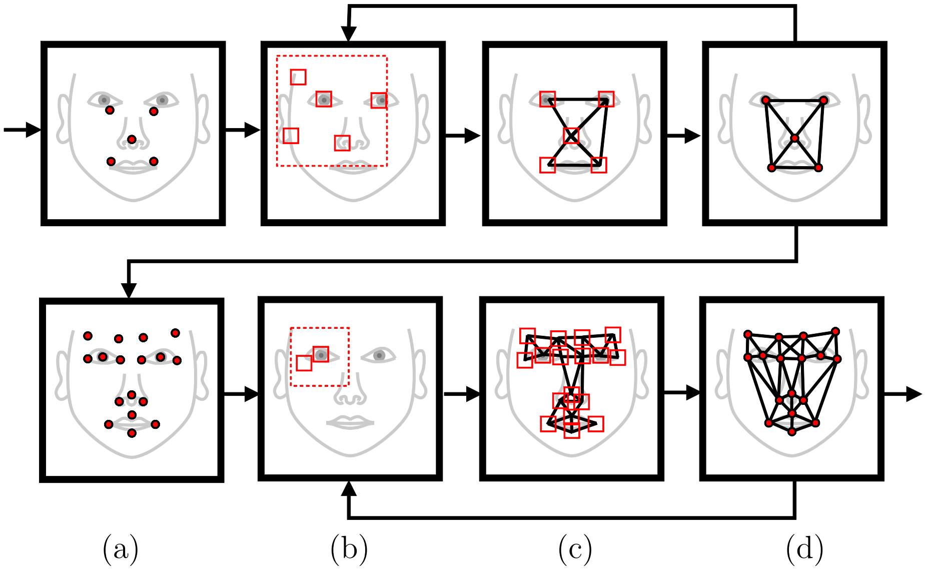

Figure 3: Schematic of a two-level cascade search: (a) initialize; (b) find candidate points; (c) select best set of candidates using MRF; (d) regularize using global model and either iterate, go to next level or finish.

3.2 Advanced system

The method we describe in the following sections addresses these shortcomings by imposing local constraints when selecting candidates before regularizing using the true shape model (see Figure 3). Furthermore, this is implemented in a hierarchical (multi-scale) framework [33] that is shown to improve performance. Currently, the approximate shape distribution employed in the candidate selection process is defined by hand; however, recent research has shown that this can be automated to improve performance further [26].

To be more specific, in our proposed technique we formulate the deformable object matching as a global shape alignment problem combined with MRF-based local modeling. We represent the model as a set of N points, $\mathbf{X} = \{\mathbf{x}_i = (u_i, v_i)\}$. Given a query image, $\mathbf{I}$, our aim is then to find the optimal set, $\mathbf{X}^*$, that maximizes the posterior,

$$p(\mathbf{X}|\mathbf{I}) \propto p(\mathbf{I}|\mathbf{X})p(\mathbf{X}). \tag{7}$$

Since the number of possible positions for each $\mathbf{x}_i$ is very large, however, considering the combinatorial number of all possible $\mathbf{X}$ is computationally intractable, even if we restrict the set of candidates for each $\mathbf{x}_i$ to some smaller number (*e.g.* by considering only locally optimal candidates within a region of interest). By making certain assumptions of conditional independence between features, however, we can approximate the joint prior with an MRF that reduces the complexity of the problem such that an approximate solution, $\mathbf{Y}$, can be found efficiently. We can then *regularize* this approximation to give a solution that is closer to the optimum, $\mathbf{X}^*$.

We summarize this approach as follows:

1. Initialize point locations, $\mathbf{X}_0^*$
2. For $t = 1 \ldots T$
   (a) Select most promising candidates, $\mathbf{Y}_t = \arg\max_{\mathbf{Y}} p(\mathbf{I}|\mathbf{Y})p(\mathbf{Y}|\mathbf{X}_{t-1}^*)$
   (b) Regularize candidates in order to update points, $\mathbf{X}_t^* = \arg\max_{\mathbf{X}} p(\mathbf{X}|\mathbf{Y}_t)$

In the following sections, we describe these two steps in detail before outlining how they are integrated within a hierarchical 'cascade' framework.

3.2.1 MRF-guided Candidate Selection

The first step involves finding the set of candidates, $\mathbf{Y}_t$, that maximizes the posterior,

$$p(\mathbf{Y}|\mathbf{I}) = p(\mathbf{I}|\mathbf{Y})p(\mathbf{Y}|\mathbf{X}_{t-1}^*). \tag{8}$$

The first term in (8) is the likelihood that indicates how well the image data supports the hypothesized candidates. It is common to assume that the patch, Q_i, associated with each candidate, $\mathbf{y}_i$, does not overlap with any other patch such that

$$p(\mathbf{I}|\mathbf{Y}) = \prod p(Q_i|\mathbf{y}_i) \tag{9}$$

$$\Rightarrow -\log p(\mathbf{I}|\mathbf{Y}) = -\sum \log p(Q_i|\mathbf{y}_i) = \sum \phi(\mathbf{y}_i) \tag{10}$$

where $\phi(\cdot)$ is an error function that indicates goodness of fit with the image data. In our case, we use (negated) normalized correlation over the x- and y-gradient images for accurate feature localization. To make inference practical, we restrict the set of candidates considered for each $\mathbf{y}_i$ to the k_i lowest-scoring local minima of $\phi(\mathbf{y}_i)$ within a search region of size r_i.

The second term in (8) is the MRF prior, where the conditional dependence on $\mathbf{X}_{t-1}^*$ is introduced as a result of taking all measurements in a normalized (with respect to scale and orientation) coordinate frame defined by the current estimate of $\mathbf{X}^*$. By making certain assumptions of conditional independence between points, we can approximate the joint prior with a more sparsely connected graph. In the case where we consider only dependencies between pairs of points,

$$p(\mathbf{Y}|\mathbf{X}_{t-1}^*) \approx \prod p(\mathbf{y}_i|\mathbf{y}_j, \mathbf{X}_{t-1}^*) \tag{11}$$

$$\Rightarrow -\log p(\mathbf{Y}|\mathbf{X}_{t-1}^*) \approx -\sum \log p(\mathbf{y}_i|\mathbf{y}_j, \mathbf{X}_{t-1}^*) = \sum \psi(\mathbf{y}_i, \mathbf{y}_j) \tag{12}$$

where $\psi(\cdot)$ is an error function that indicates goodness of fit between a pair of candidates corresponding to connected nodes in the graph (we have omitted the dependency on $\mathbf{X}_{t-1}^*$ for clarity). Typically, it is assumed that

$$p(\mathbf{y}_i|\mathbf{y}_j) \sim N(\mathbf{y}_i - \mathbf{y}_j; \mu_{ij}, \Sigma_{ij}) \tag{13}$$

where μ and Σ are the mean and covariance of the displacement distribution, learned from training data. As a result, $\psi(\cdot)$ is the Mahalanobis distance from the mean displacement. When combining global and local models, however, we instead model relationships between residuals:

$$\begin{aligned} p(\mathbf{y}_i|\mathbf{y}_j, \mathbf{X}^*) &\sim N((\mathbf{y}_i - \mathbf{x}_i^*) - (\mathbf{y}_j - \mathbf{x}_j^*); \mu_{ij}, \Sigma_{ij}) &(14)\\ &\sim N((\mathbf{y}_i - \mathbf{y}_j) - (\mathbf{x}_i^* - \mathbf{x}_j^*); \mu_{ij}, \Sigma_{ij}) &(15)\end{aligned}$$

In the case of a rigid shape model, this is equivalent to modelling the raw displacements. When the shape is allowed to vary with respect to its coordinate frame, however, the distance $\mathbf{x}_i^* - \mathbf{x}_j^*$ varies and modifies the pairwise potential. In practice, we maximize (8) by minimizing an energy function,

$$E = \sum \phi(\mathbf{y}_i) + \lambda \sum \psi(\mathbf{y}_i, \mathbf{y}_j), \tag{16}$$

where λ is a parameter that weights the influence of the prior and likelihood terms. This energy function is minimized using inference algorithms such as dynamic programming (as used in this work), belief propagation [15] or tree re-weighted message passing [31].

3.2.2 PCA-based Regularization

Having selected a set of candidate feature locations, we regularize them by projecting onto a learned subspace of allowable solutions. Specifically, we align the set of $2N$-dimensional training vectors, $\mathbf{X} = (u_1, \ldots, u_N, v_1, \ldots, v_N)^{\mathrm{T}}$, using Procrustes analysis and then learn their underlying linear subspace via PCA [14]. As a result, any $\mathbf{X}$ can be described by a (similarity, in this case) transformation, S, and a vector of shape coefficients, $\mathbf{b}$:

$$\mathbf{X} = S(\overline{\mathbf{X}} + \mathbf{P}\mathbf{b} + \epsilon) \tag{17}$$

where $\overline{\mathbf{X}}$ is the mean shape over the training data (in a normalized reference co-ordinate frame), $\mathbf{P}$ is a set of orthogonal modes of variation, and ϵ accounts for residual displacements associated with every feature point in the global shape. We then compute an optimal value of $\mathbf{X}$ by projecting the selected candidates onto this subspace:

$$\mathbf{X}_t^* = S_t(\overline{\mathbf{X}} + \mathbf{P}\mathbf{P}^{\mathrm{T}}(S_{t-1}^{-1}\mathbf{Y}_t - \overline{\mathbf{X}})) \tag{18}$$

where S_t is the estimated pose, S, at time t.

3.2.3 Cascade Implementation

We extend the proposed model by applying the search algorithm in a hierarchical fashion, first localizing a small subset of highly salient points which are then used as an initialization for a more complex model with a greater number of points and shape modes. Importantly, search parameters and the properties of the MRF are re-learned at each level such that the correct distribution parameters (*i.e.* all μ_{ij} and Σ_{ij}) are employed rather than (incorrectly) assuming constant values for all levels. Our motivation is to use the most salient points to efficiently and accurately estimate the global pose (translation, scale and orientation) of the object before estimating the locations of the remaining features with a more flexible global shape model. Given the cascade of combined global and MRF-based local shape models and a target image, the following algorithm (see also Figure 3) is used to localize the object of interest using C levels:

1. Initialize point locations, $\mathbf{X}^*_{0,0}$, using locations that are fixed with respect to face detector output.
2. For $c = 1 \dots C$
   - (a) Initialize $\mathbf{X}^*_{0,c}$ by fitting the shape model at level c to $\mathbf{X}^*_{T^{c-1},c-1}$ via weighted least-squares with a zero-mean Gaussian prior over shape parameters, $\mathbf{b}$.
   - (b) Compute points $\mathbf{X}^*_{T^c,c}$ using the search algorithm with level-specific values for number of iterations (T^c), MRF parameters (λ^c and all μ^c_{ij} and Σ^c_{ij}) and search parameters (k^c_i and r^c_i). This involves searching in a radius r^c_i around the current estimate of each point position and finding the best k^c_i candidates for each. An MRF solver picks the best combination of candidates by minimising (16). These are then used to update the global pose and shape parameters.

As an example, in experiments with models of the face we use a two-level cascade with 7 points in the first level and 22 points in the second (Figure 4). The feature finder is iterated at each level until E reaches a minimum. Typically this involves only a few iterations, though this can be modified by imposing hard limits on the number of iterations and by modifying termination criteria.

3.2.4 Parameter Estimation

When learning each level of the cascade, we re-estimate the MRF parameters and search parameters from training data to reflect the increasing accuracy of our estimated solution. For example, due to high uncertainty in scale and orientation of the detected face (the MRF potentials are invariant to global translations), the spread of pairwise potentials at the first level is likely to be large. After applying one iteration of the search, however, we would expect the error distribution between the updated points and their true values to have a much smaller spread. Therefore, we re-estimate all μ_{ij} and Σ_{ij} at each level.

We also estimate appropriate values for all r_i (*i.e.* the search radius for point i) at each level because once we have accurately located a small subset of highly salient points at

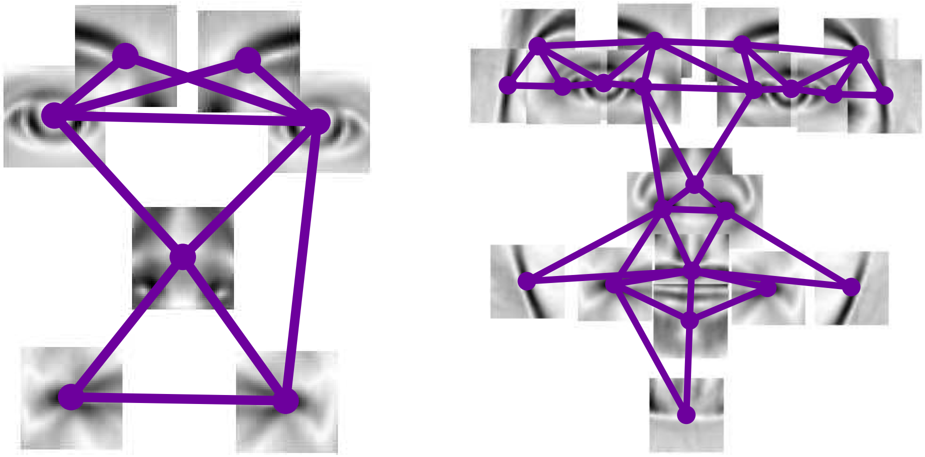

Figure 4: Models from two levels of hierarchy, containing 7 and 22 points, respectively.

one level, we should not need to search as far for these points at the next. Therefore, we set r_i to the maximum distance between the estimated and true location for point i over the training set. In effect, this is a principled method of constraining points that we think have already been accurately localized.

Similarly, we estimate k_i (*i.e.* the number of local minima to consider for point i) by noting the maximum rank among the considered candidates of the true candidate (*i.e.* the closest candidate to the true location). In effect, we consider fewer candidates for highly discriminant landmarks that result in few spurious local minima, maintaining efficiency in a principled manner.

Finally, λ (*i.e.* the weighting between prior and likelihood) is estimated during the training phase via an exhaustive search over a fixed (typically $\sim$50) number of values, based on the mean error after applying the model to the training set of images.

3.2.5 Comparison with Baseline System

There are a number of differences between this system and the baseline system (a Constrained Local Model [16]). This system uses an alternative appearance model, normalized *gradient* correlation; this has been shown to be more accurate than using grey values directly. Where the CLM uses a non-linear optimization in shape space directly, the advanced system permits solutions that lie outside of shape space; this allows the model to fit more closely to the data, improving accuracy further.

3.3 Evaluation

For a direct comparison with D3.2, we repeat evaluations that quantify the improvement in eye location estimation using the d_{max} metric and present statistics for: (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,

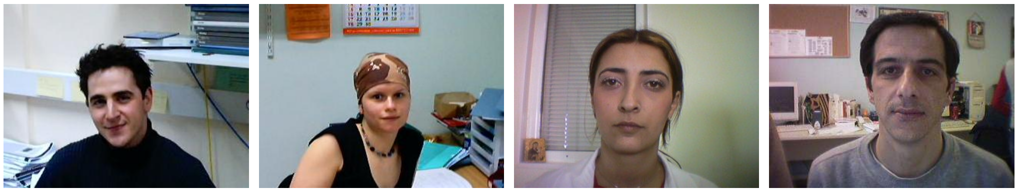

Figure 5: Examples taken from the 1052 images used to train the advanced system.

$$d_{mean} = \frac{1}{n}\sum_{i=1}^{n}\frac{|p_i - q_i|}{|q_l - q_r|}, \tag{19}$$

where n is the number of labelled points. As usual, 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 [16]. Having computed each measure for every image in each dataset the cumulative frequency curve is plotted (Figure 6), and the median value and 90th percentile value over all *images* for each dataset is computed.

As in D3.2, the method is evaluated on four datasets: BANCA, XM2VTS, BioID and BioSign. Again, the images used to train the advanced system are completely independent of these datasets, consisting of different subjects imaged under different conditions (Figure 5). Feature localisation is initialised using the output of the baseline face detector from UOULU (based on the Viola-Jones algorithm). Only this face detector was used to initialize the feature localizer in the following analysis since it exhibited the best performance of the baseline systems.

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 6. 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, we see that the median values are reduced for both d_{mean} over all points and d_{max} over the eye points in almost all cases (median d_{mean} remains unchaged for BioID). In all but one case (BANCA) the 90th percentile of d_{mean} is unchanged or reduced.

However, it can be seen that in several cases (*e.g.* BANCA and BioSign) the 90th percentile increases for d_{max} over the eye points. This suggests a slightly higher rate of failure in the advanced feature localizer that we believe to be a result of the fact that the advanced system does not regularize the final output (whereas the baseline system does). This has the effect that the model is allowed to fit to the data more closely, resulting in

		Eye points		All points					
		d_{max}		d_{max}		d_{90}		d_{mean}	
		Med.	90%	Med.	90%	Med.	90%	Med.	90%
BANCA	Baseline	0.069	0.14	-	-	-	-	0.054	0.11
	Advanced	0.059	0.21	-	-	-	-	0.045	0.14
XM2VTS	Baseline	0.049	0.095	0.19	0.34	0.13	0.24	0.067	0.11
	Advanced	0.025	0.11	0.19	0.38	0.12	0.26	0.059	0.11
BioID	Baseline	0.044	0.13	0.17	0.34	0.12	0.24	0.064	0.12
	Advanced	0.026	0.13	0.21	0.39	0.13	0.26	0.064	0.12
BioSign	Baseline	0.041	0.12	0.067	0.23	-	-	0.042	0.14
	Advanced	0.024	0.18	0.06	0.26	-	-	0.032	0.11

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}. The feature localizer was initialized using the baseline Viola-Jones detector in all cases.

high accuracy when all features can be found. However, in cases where features can not be localized using only image data the estimate is permitted to stray from the true solution. Work is ongoing to detect these cases and account for them accordingly (*e.g.* by including a 'dummy' point to substitute for missing data).

Figure 6 provides a more complete picture of the situation. We see that the 90th percentile for d_{max} over the eye points is indeed higher for the advanced system. However, we also see that the advanced system out-performs the baseline system in over 80% of cases.

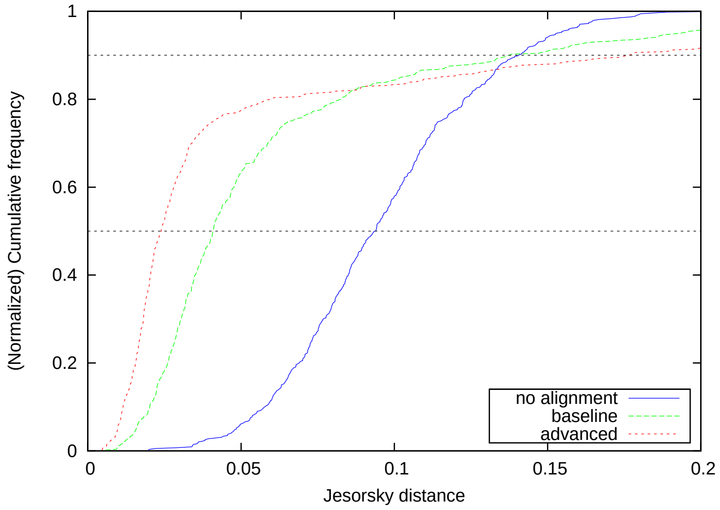

Figure 6: Cumulative distribution of d_{max} over the eye points for the BioSign dataset, using the baseline Viola-Jones detector as initialization. Although the 90th percentile value is greater for the advanced system compared to the baseline, the advanced system is superior in 80% of cases.

4 Face Verification

4.1 Related work

In an unconstrained environment, reliable face recognition is still difficult to achieve. In particular, illumination is known to be the one of the most significant problems. For example, ambient lighting varies greatly everyday, as well as between indoor and outdoor environments. Moreover, directed light source may over-saturate a part of face image and make another part being invisible because of cast and attached shadows. Therefore, photometric normalization and illumination robust features are important for face recognition.

Photometric normalization

Generally, conventional image processing transformations can be applied to reduce the image to a more canonical form where the illumination variations can be suppressed. For example, Gross & Brajovic [25] use an anisotropic smoothing iteration method to estimate the luminance of face image, so that the reflectance of the face image can be extracted by dividing the face intensity image with the luminance. Self Quotient Image model proposed by Wang *et al.*[58] normalizes the illumination by dividing the image by a smoothed version of itself. The advantage of these methods is that it does not involve any training images to model the illumination variations. An overview of the photometric normalization can be found in [62].

Illumination robust feature

In face recognition, researchers commonly use Gabor features which simulate the multi-scale and multi-orientation nature of the receptive field. However, one of the shortcomings of such feature is that the computation cost is high. Therefore, Local Binary Pattern (LBP)[41], derived using ordinal contrast encoding, has been proposed. This coding reflects the intrinsic nature of the face by capturing the mutual ordinal relationships between neighbours at pixel level or region level, and thus provides a degree of response stability in the presence of illumination changes. In order to be robust to face misalignment, a local histogram of the LBP image [1] is generally used as a face descriptor. Alternatively, Heusch *et al.* [27] and Ekenel *et al.* [19] implement DCT on LBP image for face recognition.

Recently, researchers suggested the use of phase information for face recognition because the phase information is invariant to blur. For example, Zhang *et al.* [60] proposed global and local Gabor phase pattern histogram for face recognition. Ahonen *et al.* [2] proposed Local Phase Quantization Histogram (LPQH). More recently, Chan *et al.* [12] improved the LPQH representation by extending it to the multiresolution framework [11] to enhance its robustness to face misalignment.

4.2 Local Frequency Band Gaussian Mixture Model

4.2.1 Overview

The Idiap advanced face verification is an extension of the GMM parts-based approach. The GMM parts-based approach performs a spatial decomposition of the face, and so Idiap's advanced face verification extends this by performing a spatial and frequency decomposition of the face. The frequency decomposition results in a separate image for each frequency response, a GMM parts-based classifier is then derived for each separate image. These classifiers are then combined using weighted summation with the weights being derived using linear logistic regression.

4.2.2 General Text and Related Work

A recent advance in face verification has been the effective use of feature distribution modelling techniques. The first effective method of performing face verification using feature distribution modelling was in 2002 by Sanderson and Paliwal [50]; despite the earlier work of Samaria et al. [48, 49] and Nefian and Hayes [40] who used HMMs.

The GMM Parts-Based approach, introduced by Sanderson and Paliwal, has been employed by several researchers [10, 34]. This method consists of dividing the face into blocks, or parts, and to then consider each block separately. The distribution of these parts is then modelled using Gaussian Mixture Modelling. This technique divides the face into blocks, or parts, and treats each block as a separate observation of the same underlying signal (the face). Feature vectors are obtained from each block by applying the Discrete Cosine Transform 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. [10] proposed the use of background model adaptation while Lucey and Chen [34] 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 7.

The feature vectors from each block are obtained by applying the DCT. Even advanced feature extraction methods such as the DCTmod2 method [50] 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

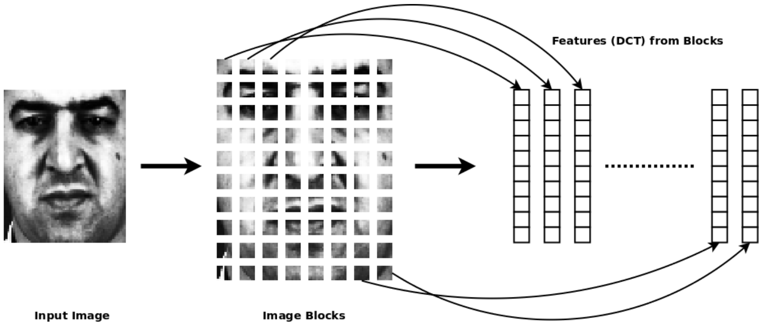

Figure 7: A flow chart of describing the extraction of feature vectors from the face image for Parts-Based approaches.

response from the image (or block). This property is exploited by the JPEG standard [43] 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 [10, 34]. 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 [18]. 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 [44] and full adaptation [32]. 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 when 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. [10].

Verification

A description of the Parts-Based approach is not complete without defining how an observation is verified. 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 model adaptation [50]. The two models, Ω^{i}_{client} and Ω_{world}, produce a log-likelihood score which is then combined using the log-likelihood ratio (LLR),

$$h(\boldsymbol{x}) = \ln(p(\boldsymbol{x} \mid \Omega^{i}_{client})) - \ln(p(\boldsymbol{x} \mid \Omega_{world})), \tag{20}$$

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} when $h(\boldsymbol{x}) \geq \tau$.

4.2.3 Local Frequency Band Approach

The proposed method is to divide the face into separate blocks and to then decompose these blocks in the frequency domain. This can be achieved by treating the frequency response from each block separately to form frequency sub-images. This method is applied to the DCT feature vectors obtained by applying the Parts-Based approach. Each coefficient can be considered independently because each coefficient of the DCT is orthogonal.

The technique is summarised as follows: (1) the face is cropped and normalised to a 68×68 image, (2) this image is divided into 8×8 blocks with an overlap of 4 pixels in the horizontal and vertical axes, (3) the DCT coefficients from each block are separated and used to form their own frequency sub-image, and (4) a feature vector is formed by taking a block from the frequency sub-image and vectorising the block. The way in which the frequency sub-images are formed is demonstrated in Figure 8.

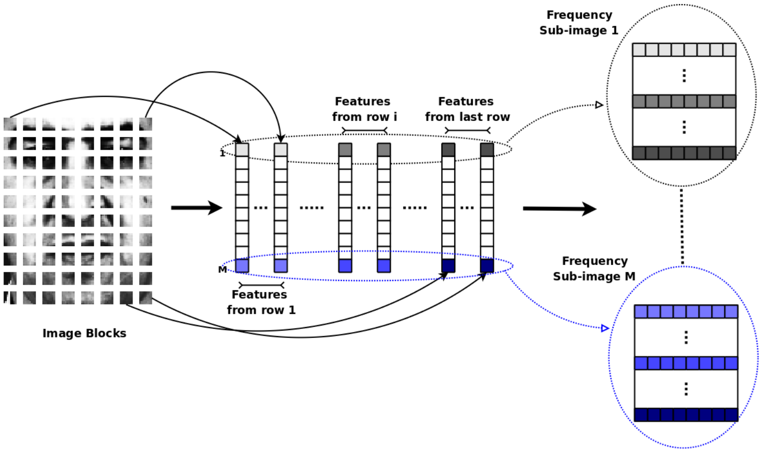

Figure 8: The figure above describes how the face can be decomposed into separate frequency sub-bands (sub-images).

Motivation

To illustrate the differences between the frequency decomposition approach and the full Parts-Based approach the following statements are made. For the Parts-Based approach it is often stated that the face is broken into blocks and the distribution of each block is then modelled [50, 34], however, another stricter statement would be that the frequency information from each block is simultaneously modelled since each dimension of the feature vector represents a different sampling frequency of the DCT. By contrast the frequency decomposition approach separates the frequency information from each local block and forms a feature vector from the resulting frequency sub-images. Many feature vectors are formed from a frequency sub-image and then modelled using background model adaptation, thus, the image is decomposed in both the spatial domain and the frequency domain.

A side effect of working on the frequency sub-images is that the feature vectors formed from these sub-images will retain extra spatial information. This is because the Parts-Based approach obtains an observation from a block, however, the frequency decomposition approach gets the response from each block and then forms a feature vector using responses from several blocks. This means that the feature vectors formed from the frequency sub-images will actually span several blocks when compared to the Parts-Based approach, for instance the feature vector could be formed from a frequency sub-image by spanning an entire row or column of the image.

Feature Extraction

Three methods of forming a feature vector from the frequency sub-images are examined, these are to form a feature vector: (1) across the row of the frequency sub-image (row-based approach), (2) across the column of the frequency sub-image (column-based approach), or (3) from a 4×4 block of the frequency sub-image which is then vectorised (block-based approach). The choice of a 68×68 image results in frequency sub-images of size 16×16 which allows for the fair comparison of the three different feature extraction methods as each method will result in feature vectors of dimension $D = 16$ with $o = 16$ observations from each frequency sub-image. A visualisation of these three methods is provided in Figure 9.

Classifier

Having obtained these feature vectors a classifier is formed using the same background model adaptation approach that was used for the Parts-Based approach [10]. Each local frequency sub-band (k) produces a separate classifier (C_k) and these classifiers are then combined using weighted linear score fusion, $C_{w\_sum} = \sum_{k=1}^{K} \beta_k C_k$. This fusion technique is used as it was shown by Kittler et al. [30] that the sum rule (which is what weighted linear classifier score fusion abstracts to be) is robust to estimation errors. The weights, β_k, for the classifiers are derived using an implementation of linear logistic regression [6].

4.3 Multi-scale Local Binary Pattern Histogram Discriminant Analysis with Score Normalization for robust face recognition (PS_MLBPHLDA_tnorm)

The advance system from University of Surrey first normalizes the face image to a canonical form in which the illumination variations are suppressed. Then the image is represented by the multi-scale local binary pattern histogram discriminative descriptor. This method is optimised for small foot print computer platforms and exhaused by innovative postprocessing. Accordingly, the similarity score of each query image is normalized by the test norm.

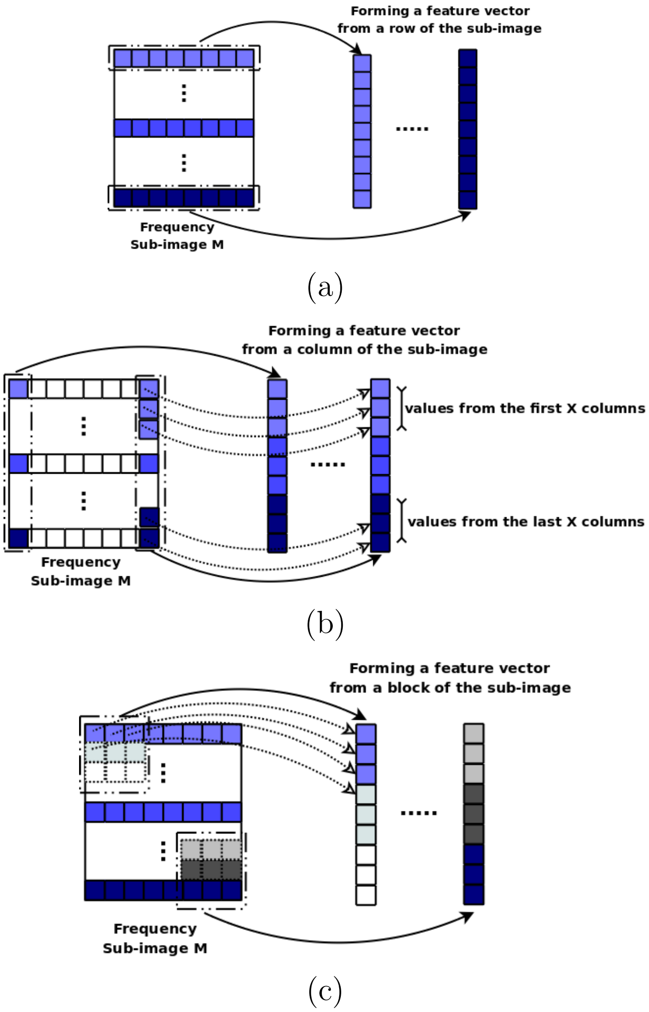
Forming a feature vector
from a row of the sub-image
Frequency
Sub-image M
(a)
Forming a feature vector
from a column of the sub-image
values from the first X columns
values from the last X columns
Frequency
Sub-image M
(b)
Forming a feature vector
from a block of the sub-image
Frequency
Sub-image M
(c)

Figure 9: Forming the feature vectors (a) along the row of the frequency sub-image, (b) along the columns of the frequency sub-image and (c) using blocks of the frequency sub-image.

For a sequence of frames in a video, the final similarity score is the average of frame-based similarities. A brief description of this system is given in following.

4.3.1 Preprocessing sequence approach

In this report, a preprocessing method [54] based on a series of steps presented in Figure 10, designed to reduce the effects of illumination variation, local shadowing and highlights, while still keeping the essential visual appearance information for use in recognition is used.

This process first applies a gamma correction, which is a nonlinear gray level transformation replacing the pixel value in $\mathbf{I}$ with $\mathbf{I}^{\gamma}$ where $\gamma > 0$. The objective of this process is to enhance the local dynamic range of the image in dark and shadow regions, while

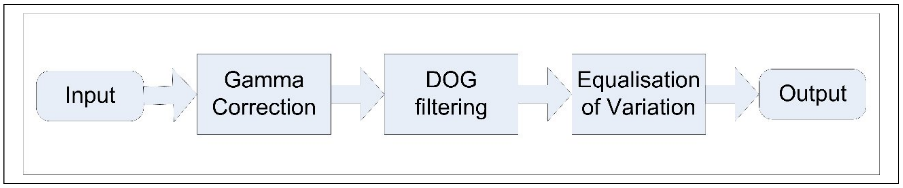

Figure 10: The block diagram of the Preprocessing sequence approach.

suppressing the bright region. In our work, γ is set to 0.2. Then the image is processed by a band-pass filter that is the difference of Gaussian filtering, shown in Equ 21, to remove the influence of intensity gradients such as shading effects, while homomorphic filtering uses the high-pass filter.

$$DoG = (2\pi)^{-\frac{1}{2}}[\sigma_1^{-1}e^{-\frac{x^2+y^2}{(2\sigma_1)^2}} - \sigma_2^{-1}e^{-\frac{x^2+y^2}{(2\sigma_2)^2}}] \tag{21}$$

The reason of choosing the band-pass filter is that it not only suppresses low frequency information caused by illumination gradient, but also reduces the high frequency noise due to aliasing artifacts. In our work, σ_1 is set to 1 and σ_2 is set 2. Then, the two stage contrast equalisation presented in Equ 22 and Equ 23 is employed to further re-normalise the image intensities and standardise the overall contrast.

$$\mathbf{J}(x,y) = \frac{\mathbf{I}(x,y)}{(mean(|\mathbf{I}(x,y)|^a))^{\frac{1}{a}}} \tag{22}$$

$$\widehat{\mathbf{J}}(x,y) = \frac{\mathbf{J}(x,y)}{(mean(min(\tau, |\mathbf{J}(x,y)|)^a))^{\frac{1}{a}}} \tag{23}$$

a, set to 0.1, is used to reduces the influence of large values and τ,set to 10, is a threshold used to truncate large values after the first stage of normalisation. Lastly, a hyperbolic tangent function in Equ 24 is applied to suppress the extreme values and limit the pixel values in normalised image,$\widehat{\mathbf{I}}$, to a range between $-\tau$ and τ

$$\widehat{\mathbf{I}}(x,y) = \tau tanh(\frac{\widehat{\mathbf{J}}(x,y)}{\tau}) \tag{24}$$

4.3.2 Multi-scale Local Binary Pattern Histogram Discriminant Descriptor

The multi-scale local binary pattern representation with Linear Discriminant Analysis, LDA [11] is used in this report. Local binary pattern operators at R scales are first applied to a face image. This generates a grey level code for each pixel at every resolution. The resulting LBP images are cropped to the same size and divided into non-overlapping sub-regions, $\mathbf{M}_0$, $\mathbf{M}_1$,..$\mathbf{M}_{J-1}$. The regional pattern histogram for each scale is computed based on Equ (25)

$$\mathbf{h}_{P,r,j}(i) = \sum_{x',y' \in \mathbf{M}_j} B(LBP_{P,r}(x',y') = i) \quad | \quad i \in [0, L-1], r \in [1, R], j \in [0, J-1],$$
$$B(v) \begin{cases} 1 & \text{when } v \text{ is true} \\ 0 & \text{otherwise} \end{cases} \tag{25}$$

$B(v)$ is a Boolean indicator. The set of histograms computed at different scales for each region, $\mathbf{M}_j$, provides regional information. L is the number of histogram bins. By concatenating these histograms into a single vector, we obtain the final multiresolution regional face descriptor presented in Equ(26)

$$\mathbf{f}_j = [\mathbf{h}_{P,1,j}, \mathbf{h}_{P,2,j}, \cdots, \mathbf{h}_{P,R,j}] \tag{26}$$

This regional facial descriptor can be used to measure the face similarity by fusing the scores of local similarity of the corresponding regional histograms of the pair of images being compared. However, by directly applying the similarity measurement to the multi-scale LBP histogram [41], the performance will be compromised. The reason is that this histogram is of high dimensionality and contains redundant information. By adopting the idea from [4], the dimension of the descriptor can be reduced by employing the principal component analysis (PCA) before LDA. PCA is used to extract the statistically independent information as a prerequisite for LDA to derive discriminative facial features. Thus a regional discriminative facial descriptor, $\mathbf{d}_j$, is defined by projecting the histogram information, $\mathbf{f}_j$, into LDA space $\mathbf{W}_j^{lda}$, i.e.

$$\mathbf{d}_j = (\mathbf{W}_j^{lda})^T \mathbf{f}_j \tag{27}$$

This discriminative descriptor, $\mathbf{d}_j$, gives 4 different levels of locality: 1) the local binary patterns contributing to the histogram contain information at the pixel level, 2) the patterns at each scale are summed over a small region to provide information at a regional level, 3) the regional histograms at different scales are concatenated to produce multiresolution information, 4) the global description of face is established by concatenating the regional discriminative facial descriptors. Our results show that combining Multi-scale Local Binary Pattern Histogram with LDA is more robust in the presence of face mis-alignment and a uncontrolled environment.

4.3.3 Similarity measurement

After projecting the regional histogram into LDA space, the similarity measurement between query image $\mathbf{I_n}$ and the average of m template images, $Sim(\mathbf{I}, \mathbf{I_n})$ is obtained by taking the sum of the normalised correlation between the average of the regional discriminative descriptor $\mathbf{d}_j$ of the template images, and the regional discriminative descriptor $\mathbf{d}'_j$ of probe image respectively which is presented below.

$$Sim(\mathbf{I}, \mathbf{I_n}) = \sum_{j=0}^{J-1} \frac{\mathbf{d}_j \mathbf{d}'_j}{\|\mathbf{d}_j\| \|\mathbf{d}'_j\|} \tag{28}$$

4.3.4 Score Normalisation in each frame

In verification, the similarity score is degraded by many factors, such as a change of pose, illumination, occlusion and the characteristic of different persons enrolled in the system, and it will degrade the system performance as a predefined threshold for making a decision to accept or reject the claimed identity is chosen in an off-line training stage. Interestingly, although client specific thresholds can achieve a better adaptation to class specific distributions, as exemplified by Yang *et al.*'s Z-norm [59]. These methods are not effective when imaging conditions such pose, environment and sensor change. To cope with these problems, we propose to postprocess the similarity scores by test-normalisation(T-norm)[3] because it removes the score variation caused by condition changes. The T-norm is defined as:

$$Norm(\mathbf{I}, \mathbf{I_n}) = \frac{Sim(\mathbf{I}, \mathbf{I_n}) - \mu^C}{\sigma^C} \tag{29}$$

where the parameters, μ^C and σ^C, are the mean and standard deviation of the distribution of the similarity between a cohort impostor templates and an incoming image. Thus, T-norm is a test dependent approach. In this work, the cohort impostor templates are all the subject templates in the enrolment set except the template(s) for the claimed subject during testing and this is called Gallery norm. Lui *et al.* [35] has recently proposed nonlinear T-norm which is mapping the normalised score to the sigmoid function to improve the accuracy of the face verification and our simple T-norm results also show that the performance of our proposed methods mentioned in [12] can be boosted up by over 70%.

4.4 Evaluation

To be consistent with the evaluation of the baseline algorithms reported in D3.2, BANCA video with P protocol is used to test the performance of automated face verification and the Detection Error Trade-off (DET) curve is used to compare the performance.

Experiment setup of PS_MLBPHLDA_tnorm

In this experiment, the face images are using the IDIAP baseline face detector called a Cascaded Boosted LBP-based face detector[22]. In the enrolment stage, the average face descriptor is extracted from the video as a subject template for the histogram-based representation, while a set of face descriptors derived from several videos is used to create a subject template for the LDA representation. In the test stage, the face descriptor in each frame is extracted and compared with the enrolled subject templates to derive the similarity measurement. The similarity of video is the average of the similarities computed for each frame.

4.5 Results

A comparison of all systems using DET and Expected Performance Curves are shown in Figures 12 and 13, respectively. It clearly shows that both advanced systems are better than those baseline systems reported in D3.2. In contrast to IDIAP advanced system, PS_MLBPHLDA system performance without score normalisation is significantly better. In this report, the best method of the advanced systems, achieved 4% in average of EER of g1 and g2 at $\beta = 1$, is PS_MLBPHLDA_tnorm which is at least 4 times better than others. The main reason for the superior performance is the photometric normalisation and the multiresolution framework with the test normalisation providing the robust solution for the change of conditions. In [12], PS_MLBPHLDA_tnorm was also tested on BANCA image database with P protocols and got 2.11% in average of HTER of g1 and g2 at $\beta = 1$. The performance in video database is worse than that in image database because most of the images from video cropped by the face detector, shown in 11, are misaligned. In other words, the performance of PS_MLBPHLDA_tnorm can further be improved by replacing more powerful face detection and/or alignment methods.

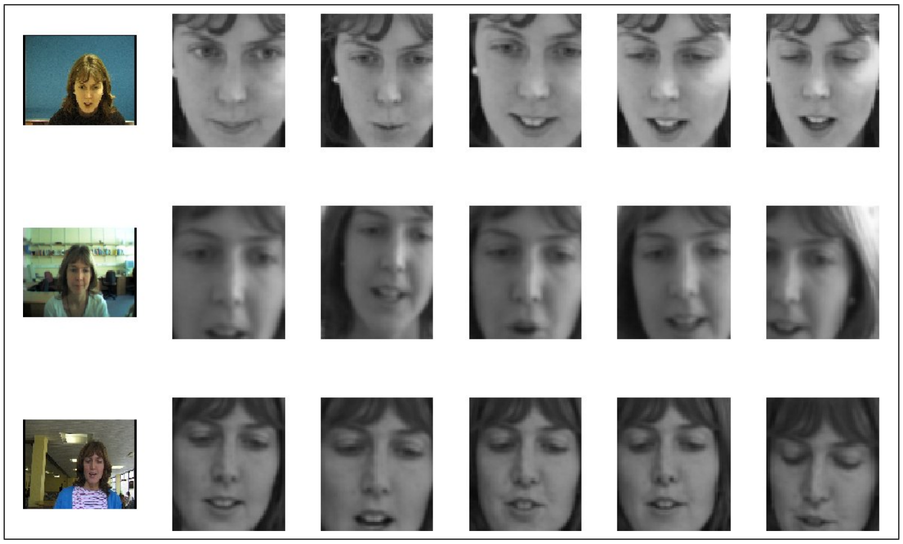

Figure 11: Images from BANCA video cropped by the face detector [22] under Controlled, Degraded and Adverse Scenarios, presented in row 1, 2 and 3 repectively

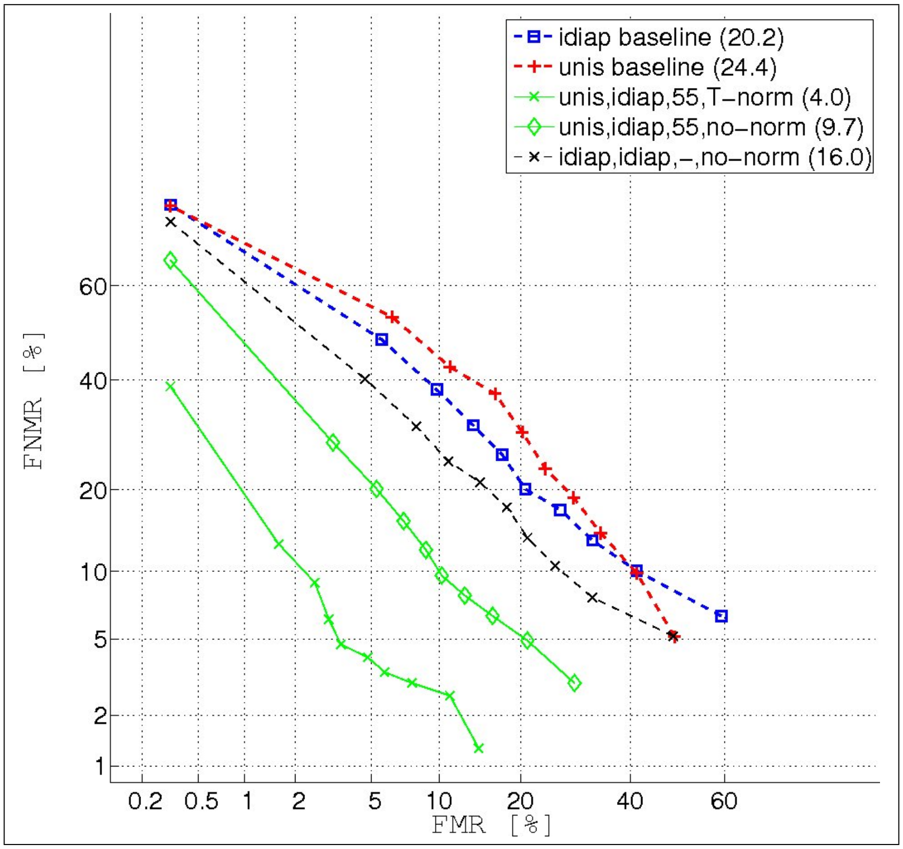

Figure 12: The average of DET curves of the evaluated systems computed on the g1 and g2 of the BANCA video according to the P protocol.

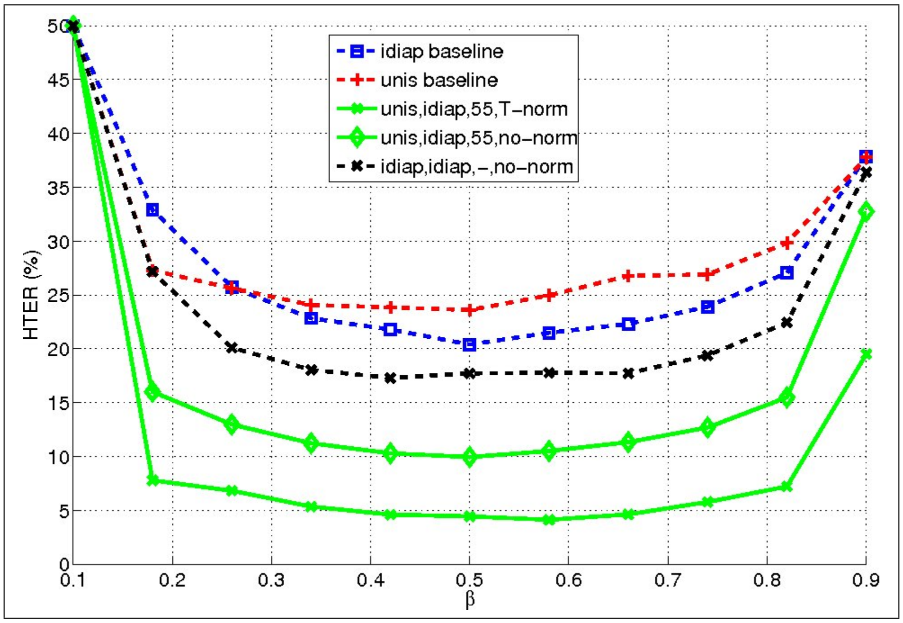

Figure 13: The average of DET curves of the evaluated systems computed on the g1 and g2 of the BANCA video according to the P protocol.

5 Speaker Verification

Both advanced speaker recognition systems from LIA and BUT were based on Factor Analysis (FA), which is nowadays the core technique of all state-of-the-art speaker recognition systems. The following section 5.1 introduces Joint FA from the theoretical point of view. Sections 5.2 and 5.3 then present the details of implementation of LIA and BUT verification systems. Section 5.4 summarizes the results obtained on BANCA and discusses them.

5.1 Joint Factor Analysis

This section is based on the introductory JFA chapter from JHU 2008 report [7]. Joint factor analysis (JFA) is a two-level generative model of how different speakers produce speech and how their (remotely) observed speech may differ on different occasions (or *sessions*). The hidden deep level is the *joint factor analysis* part that models the generation of speaker-and-session-dependent GMMs. The output level is the GMM generated by the hidden level, which in turn generates the sequence of feature vectors of a given session.

The GMM part needs no further introduction. As is customary in speaker recognition, all of the GMMs differ only in the mean vectors of the components [45]. The component weights and the variances are the same for all speakers and sessions. The session-dependent GMM component means are modeled as:

$$\mathbf{M}_{ki} = \mathbf{m}_k + \mathbf{U}_k \mathbf{x}_i + \mathbf{V}_k \mathbf{y}_{s(i)} + \mathbf{D}_k \vec{z}_{\,ks(i)} \tag{30}$$

Here the *indices* are: k for the GMM component; i for the session; and $s(i)$ for the speaker in session i. The *system hyperparameters* are:

$\mathbf{m}_k$, speaker-and-session-independent mean vector;
$\mathbf{U}_k$, rectangular *channel-factor loading matrix*;
$\mathbf{V}_k$, rectangular *speaker-factor loading matrix*;
$\mathbf{D}_k$, diagonal *speaker-residual scaling matrix*;

The *hidden speaker and session variables* are:

$\mathbf{x}_i$, session-dependent vector of *channel-factors*;
$\mathbf{y}_s$, speaker-dependent vector of *speaker-factors*;
$\vec{z}_{\,ks}$, speaker-and-component-dependent vector of *speaker-residuals.*

Standard normal distributions are used as a prior for all of these hidden variables.

5.1.1 Supervector model

We can summarize our JFA model by stacking component-dependent hyperparameters into larger matrices:

$$\mathbf{V} = \begin{bmatrix} \mathbf{V}_1 \\ \mathbf{V}_2 \\ \vdots \end{bmatrix}, \qquad \mathbf{U} = \begin{bmatrix} \mathbf{U}_1 \\ \mathbf{U}_2 \\ \vdots \end{bmatrix}, \qquad \mathbf{D} = \begin{bmatrix} \mathbf{D}_1 & \mathbf{0} & \cdots \\ \mathbf{0} & \mathbf{D}_2 & \cdots \\ \vdots & \vdots & \ddots \end{bmatrix} \tag{31}$$

We refer to $\mathbf{V}$ as the *eigenvoice matrix*; to $\mathbf{U}$ as the *eigenchannel matrix*; and $\mathbf{D}$ as the *residual scaling matrix*. By also stacking component-dependent vectors into larger vectors, which we shall refer to as *supervectors*:

$$\mathbf{M}_i = \begin{bmatrix} \mathbf{M}_{1i} \\ \mathbf{M}_{2i} \\ \vdots \end{bmatrix} \qquad \mathbf{m} = \begin{bmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \\ \vdots \end{bmatrix} \qquad \mathbf{z}_s = \begin{bmatrix} \vec{z}_{1s} \\ \vec{z}_{2s} \\ \vdots \end{bmatrix}, \tag{32}$$

the JFA model can be expressed succinctly in supervector form as

$$\mathbf{M}_i = \mathbf{m} + \mathbf{U}\mathbf{x}_i + \mathbf{V}\mathbf{y}_{s(i)} + \mathbf{D}\mathbf{z}_{s(i)} \tag{33}$$

5.1.2 Generative ML training

In this section we give a rough summary of how the hyperparameters of a JFA system may be trained. The steps are as follows:

1. Train the universal background model (UBM) on a large selection of development data, possibly on all of it. The UBM is a GMM trained by maximum likelihood (ML), via appropriate initialization followed by multiple iterations of the EM algorithm. The UBM essentially provides the following functionality:

   - Its component means are a good choice to use for the speaker-and-session-independent supervector $\mathbf{m}$; and its variances and weights are a good choice to use for all speaker-and-session-dependent GMM variances and weights.
   - It parametrizes a computationally efficient approximation to all GMM log-likelihoods, used during training and operation of the JFA system. Specifically, all GMM log-likelihoods are approximated by the EM-algorithm auxiliary function [39], often denoted 'Q-function' in the literature. Informally, given some GMM, we approximate $\log p(\text{data}|\text{GMM}) \approx Q(\text{UBM}, \text{GMM}, \text{data})$. All further processing makes use of this approximation.

2. Train the eigenvoice matrix $\mathbf{V}$ with an EM algorithm designed to optimize a maximum likelihood criterion over a database of as many speakers as possible. Pool multiple sessions per speaker, to attenuate intersession variation.

3. Given $\mathbf{V}$ as obtained above and with $\mathbf{D}$ temporarily set to zero, train the eigenchannel matrix $\mathbf{U}$ with a similar EM algorithm, over a database that has multiple sessions per speaker. This data should be rich in channel variation. The *Mixer Databases* are very good for this purpose.

4. Finally (and optionally), train $\mathbf{D}$, with a similar EM-algorithm, on some held-out data.

5.1.3 JFA operation

When used operationally, the steps performed by a JFA system to score a given *trial*, composed of one *train* and one *test* segment, can be described as follows:

1. Use the JFA model (33) and the *train* segment to make a MAP[1] point-estimate of the target speaker model. That is, the hidden variables $\mathbf{x}$, $\mathbf{y}$, and (optionally) $\mathbf{z}$ are jointly estimated. Then $\mathbf{x}$ is discarded and $\mathbf{M}$ is denoted the *target speaker model.* Note that this model now has a unspecified parameter $\mathbf{x}$, because its value in a test segment will be different from its value in the train segment. This uncertainty is modeled by the standard normal prior over $\mathbf{x}$.

2. Compute an approximation to the log-likelihood of the target speaker model, given the *test* segment data, $\log p(\text{test segment}|\mathbf{M})$. Good approximations to use here include [24]:

   - The Q-function approximation, where the unknown nuisance variable $\mathbf{x}$ is integrated out, see [28], equation 19.
   - A linear simplification to the Q-function, where a MAP point-estimate of $\mathbf{x}$ is used. For computational efficiency $\mathbf{x}$ is estimated relative to the UBM, i.e. with $\mathbf{y} = \mathbf{0}$ and $\mathbf{z} = \mathbf{0}$.

3. Compute the same approximation to the UBM log-likelihood, i.e. with $\mathbf{y} = \mathbf{0}$ and $\mathbf{z} = \mathbf{0}$. The *raw score* (or raw log-likelihood-ratio) is now the difference between the target model log-likelihood and the UBM log-likelihood.

4. Normalize the raw score by applying the following in order: (i) divide by the number of test frames, (ii) z-norm, (iii) t-norm.

5.1.4 Gender dependency

JFA systems benefit from gender-dependent components:

- Some, like the CRIM system at NIST SRE'08, are trained from the UBM onwards on gender-dependent data. This gives independent male and female systems, which can be used respectively for all-male or all-female trials.

- Others, like the BUT system at NIST SRE'08, are trained on mixed data, but then use gender-dependent ZT-norm cohorts.

[1] MAP denotes *maximum a-posteriori.* The likelihood used here is the Q-function approximation and the prior is the standard normal distributions over the hidden variables.

5.2 LIA advanced system

The LIA advanced system is based on the Factor Analysis model decomposition. It consists in an estimation of two components: speaker and channel. It corresponds to the same approach than JFA without the eigen-voices concept. It is presented in 5.2.3.

Our advanced system is based on the system used for the NIST SRE 2008 evaluation campaign. A full description of LIA system can be found in [37].

5.2.1 Feature extraction, frame selection

The LIA feature vector is composed of 50 coefficients including 19 linear frequency cepstral coefficients (LFCC), their first derivative, their 11 first second derivatives and the delta-energy. A 24 filter bank coefficients is first computed over 20ms Hamming windowed frames with a shift of 10ms. The bandwidth is limited to the 300-3400Hz range.

Energy coefficients are normalized using a mean removal and a variance reduction. After, they are used to train a three-components GMM, which is used to select high-level energy frames (only about 30% of the frames are kept during NIST SRE 2006 and 2008 evaluations). 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 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 normalization are computed file by file using previously selected frames.

5.2.2 Universal background models

Two Universal Background Models (UBM) are used: one for males and the second for females. These two GMMs are trained on Fisher English Training Speech Part 1 (LDC:LDC2004S13), and consists of about 20 millions of speech frames (about 55^h).

The gender-dependent UBM are composed of 512 Gaussian components with diagonal covariance matrices. The UBM parameters estimation is done it two passes:

- first, several pass of EM algorithm are done using only 10% of frames selected randomly at each new iteration;

- the two last iterations are made with the entire signal.

During all the process, a variance flooring is applied.

5.2.3 Factor analysis

The standard JFA, presented in 5.1, assumes that channel factor and speaker factor can be separated. Our system is based on the Lattent Factor Analysis (LFA) and considers only the channel factor (corresponding to the **U** matrix in equation 30). In this case, the LFA models can be written as:

$$\mathbf{M}_{ki} = \mathbf{m}_k + \mathbf{U}_k \mathbf{x}_i + \mathbf{D}_k \vec{z}_{\ ks(i)} \tag{34}$$

For the advanced system, we work on the **U** matrix training/adaptation. In the baseline system this matrix was the one trained for NIST SRE 2008 evaluation. Regarding to [38], we worked on the general statistic to train the lattent variable of the **U** matrix. These are the zero order and first order statistics with respect to the UBM model.

Let $\mathbf{N}_{(h,s)}$ be the vector containing the zero order speaker and session dependent statistics:

$$\mathbf{N}_{(h,s)}[g] = \sum_{t \in (h,s)} \gamma_g(t), \tag{35}$$

where $\gamma_g(t)$ is the *a posteriori* probability of Gaussian g for the observation t. In the equation, $\sum_{t \in (h,s)}$ means the sum over all frames belonging the session h of speaker s.

Let $\mathbf{X}_{(h,s)}$ be the vector containing the first order speaker and session dependent statistics. The dimensions of $\mathbf{X}_{(h,s)}$ is equal to $M \times D$:

$$\{\mathbf{X}_{(h,s)}\}_{[g]} = \sum_{t \in (h,s)} \gamma_g(t) \cdot t \tag{36}$$

A good estimation of the **U** matrix parameters needs a large amount of data from several speakers and several sessions. It is not easy to respect theses constraints especially in the case of small database as BANCA. That's why we introduce this method to adapt the **U** matrix parameters to a specific context from a matrix train in another one.

Working in the statistic level, we now define these parameters as:

$$N_{(h,s)}[g]^{adv} = \alpha \times N_{(h,s)}[g]^{NIST} + (1-\alpha) \times N_{(h,s)}[g]^{BANCA} \tag{37}$$

and

$$\{\mathbf{X}_{(h,s)}\}_{[g]}{}^{adv} = \alpha \times \{\mathbf{X}_{(h,s)}\}_{[g]}{}^{NIST} + (1-\alpha) \times \{\mathbf{X}_{(h,s)}\}_{[g]}{}^{BANCA} \tag{38}$$

where α can be set into the range $[0;1]$.

Then the **U** matrix parameters can be estimated as described in [38].

5.2.4 Normalization

In order to have a simpler system and as there is not enought data into BANCA database, we decided to not perform any score normalization. But if needed, the system can still be improved performing a score normalization (Tnorm, Znorm, ZTnorm, . . .).

5.2.5 Results

LIA results for NIST 2008[2] data are presented in Tables 3. During the NIST SRE 2008, LIA concentrated on the task Det6, Det7 and Det8.

	det6	det7	det8
DCF	4.28	1.64	1.50
EER	7.66%	3.34%	2.79%

Table 3: Summary of LIA results on NIST SRE 2008 evaluation campaign. The first line contains DCF results and the second line the EER.

5.3 BUT advanced system

BUT advanced system is based on its submission for NIST 2008 SRE evaluation – a full JFA system. This section is based on our consolidated system description presented at Interspeech 2009 [8].

5.3.1 Feature extraction, segmentation

Features are derived with classical analysis window of 20 ms with shift of 10 ms: short time gaussianized MFCC 19 + energy augmented with their delta and double delta coefficients, making 60 dimensional feature vector. The system is therefore called "FA-MFCC20⇒60". Short-time gaussianization uses a window of 300 frames (3 sec).

Speech/silence segmentation is performed by our Hungarian phone recognizer [51], where all phoneme classes are linked to 'speech' class.

5.3.2 Universal background models

Two universal background models (UBMs) are trained on Switchboard II Phases 2 and 3, Switchboard Cellular Parts 1 and 2, and NIST SRE 2004 and 2005 telephone data. In total, there were 16307 recordings (574 hours) from 1307 female speakers and 13229 recordings (442 hours) from 1011 male speakers. Two gender-dependent UBMs with 2048 Gaussians were trained. We used 20 iterations of EM algorithm for up to 256 Gaussians and 25 iterations for 512 and more. No variance flooring was used.

5.3.3 Joint factor analysis

The Joint factor analysis (JFA) system closely follows the description of "Large Factor Analysis model" in Patrick Kenny's paper [29] and in section 5.1.

[2] http://www.itl.nist.gov/iad/mig/tests/sre/2008/

Two gender-dependent UBMs are used to collect zero and first order statistic for training two gender-dependent JFA systems. The mean **m** of JFA equation was set to the UBM mean and, in contrary to [29], it was never re-trained. The super-vector of variances (diagonal of $\mathbf{\Sigma}$ from [29]) is also set to UBM values and not re-trained in the training of JFA.

First, for each JFA system, 300 eigenvoices are trained on the same data as UBM, although only speakers with more than 8 recordings were considered here. For the estimated eigenvoices, MAP estimates of speaker factors are obtained and fixed for the following training of eigenchannels. A set of 100 eigenchannels is trained on NIST SRE 2004 and 2005 telephone data (5029 and 4187 recordings of 376 females and 294 males speaker respectively). Another set of 100 eigenchannels is trained on SRE 2005 auxiliary microphone data (1619 and 1322 recordings of 52 females and 45 males speaker respectively). Both sets are concatenated. In contrary to Kenny's paper [29], the diagonal matrix describing the remaining speaker super-vector variability (matrix **D** in JFA equation) is estimated on top of eigenvoices and eigenchannels. A disjoint set of NIST SRE 2004 speakers with less than 8 recordings (277 and 82 recordings of 44 females and 13 males speaker respectively) is used for this purpose and MAP estimates of speaker and channel factors are fixed for estimating the diagonal matrix. To obtain speaker models, MAP estimates of all the factors are estimated on enrollment segments using Gauss-Seidel-like iterative method [57]. Unlike Kenny [29], we use only MAP estimates (not posterior distribution) of channel factors and standard 10-best Expected Log Likelihood Ratio for scoring.

5.3.4 Normalization

Scores are normalized using zt-norm. We used 221 females and 149 males z-norm segments, 200 females and 159 males t-norm models, together 729 segments derived each from one speaker of NIST SRE 2004 and 2005 data. Experiments have shown that in contrary to simple eigenchannel adaptation where we obtained only small improvement from zt-norm, for JFA system, gender-dependent zt-norm is crucial for good performance.

5.3.5 Results

The results for NIST 2006[3] and 2008[4] data are summarized in Tables 4 and 5. For NIST 2006 data, the condition that corresponds the most to MOBIO channels is "mic-mic", while for NIST 2008 data, we should look to det2 "interview speech from the same microphone type in training and test" for comparable results. Note however, that NIST enrollment and test sessions contain much more data (around 2.5 minutes) than BANCA utterances (around 10 seconds).

[3] `http://www.itl.nist.gov/iad/mig/tests/sre/2006/`

[4] `http://www.itl.nist.gov/iad/mig/tests/sre/2008/`

system	det1 - all trials			
	phn-phn	phn-mic	mic-phn	mic-mic
FA-MFCC20⇒60	1.34	1.27	1.71	2.89
	2.60	2.86	4.03	5.20

Table 4: Summary of BUT results on 2006 data. The first line contains 100×DCF (decision cost function), the second line EER in [%].

system-metric	det1	det2	det3	det4	det5	det6	det7	det8
FA-MFCC20⇒60	4.01	1.00	3.97	3.00	3.09	2.95	1.40	1.38
	8.11	2.73	8.00	7.50	7.13	5.71	2.85	2.79

Table 5: Summary of BUT results on 2008 data. The first line contains 100×DCF, the second line EER in [%].

5.4 Tests on BANCA

The experiments with target data were performed on the BANCA database using the experimental protocol described in [36]. The results are directly comparable to the ones presented with the baseline systems in [5].

5.4.1 LIA tests

The UBM training requires a lot of data not available in the BANCA database, that's why the UBM trained for NIST SRE 2008 was used. More precisely, the two universal models are used: one for male and one for female. The UBM to use is automatically chosen thanks to a gender detection system.

As BANCA is composed of two data set, when the tests are performed on $G1$ set, the $\mathbf{U}$ is adapted using the data available into $G2$; and vice-versa. The results are presented into Table 6. With an EER lower than 3.5%, we can noticed that our results are close to these obtained during NIST SRE 2008 campaign.

5.4.2 BUT tests

As BANCA does not offer sufficient number of speaker for the training of UBM and/or speaker or channel factors, the NIST 2008 system, as it was described in section 5.3 was used, with the following differences:

1. the system was gender-independent (one UBM was used).

	G1 subset	G2 subset
EER	3.32%	3.42%

Table 6: LIA results on BANCA. No normalization are performed. The numbers are equal error rates (EER).

	G1 subset	G2 subset
ZT-norm	7.10	5.33
No ZT-norm	6.61	7.49
Mean ZT-norm	3.65	4.93

Table 7: BUT results on BANCA with different score normalizations. The numbers are equal error rates (EER).

N	EER	EER-meanzt
none	5.53	-
5	13.04	7.29
10	8.80	6.93
5	6.76	6.60
20	5.50	6.24
30	4.26	5.97
40	3.97	5.88
50	3.77	5.81
70	3.40	5.81
100	3.11	5.74
150	3.03	5.67

Table 8: Dependence of system performance on the number of speakers in z- and t-norm cohorts. N is the number of speakers per gender and normalization cohort, so that the real number for an experiment is $N \times 4$. The first column with results stands for standard zt-norm while in the second, only mean normalization was done.

2. the zt-norm speakers were selected from the opposite part of the database (from G2 for G1 and vice-versa).

The first results (see the 1^{st} line of Table 7) were not satisfactory, therefore we started to investigate the possible causes of the problem.

As we have already pointed out in [8], the normalization is crucial for correct performance of any JFA-based system. Note that BUT system uses a gender-dependent normalization. We have therefore looked at the numbers of available z/t-norm speakers. For BANCA, where each part of the database includes only 26 speakers, this number is only 6: we need to separate genders and also z-norm and t-norm speakers, which makes up $6 \times 2 \times 2 = 24$ speakers. In our NIST experiments, the numbers of speakers were substantially higher, in the order of hundreds. We have therefore investigated in the importance of number of z- and t-norm speakers on NIST 2006 data.

The results in Table 8 reveal that using only a small number of speakers not only does not help, but on contrary, can hurt system performance. We have therefore run the system without any normalization and found mixed results – improved performance on G1 subset of BANCA, but a hit on part G2, see the 2^{nd} line in Table 7.

Finally, we hypothesized that the normalization can still be beneficial if only the mean of impostor distribution is estimated and subtracted, and no division by the standard deviation is performed, as this estimation can be more sensitive to the limited amount of speakers. While testing this technique on NIST data (right column in Table 8), we have found that this technique under-performs the classical zt-normalization for usual high numbers of speaker, but that it is less sensitive for limited amounts of normalization speakers. The final result obtained on BANCA (3^{rd} line in Table 7) clearly show superior performance of mean-zt-norm on this task.

5.4.3 Conclusions

Both advanced systems are based on Factor Analysis. The LIA one is based on the Lattent Factor Analysis which assumed that it's not possible to extract the speaker channel which is not separable from the rest of the speech signal. BUT system, for its part, is based on the Joint Factor Analysis.

It is important to notice that whatever the Factor Analysis version used, both outperform the baseline system. JFA based system seems to require a score normalization post-processing but a particular attention needs to be paid. Mean-zt-norm brought improvement for limited number of speakers in the z- and t-norm cohorts, but we are aware of ad-hoc nature of this approach.

6 Summary

This document presented the advanced unimodal algorithms for face and speech authentication developed in the MOBIO project. The developed algorithms were compared against each other and against the baseline systems described in D3.2 [5].

In face detection and face point localisation, the performance of the baseline algorithms was already high. Some improvement over the baseline methods in performance was achieved as shown by the experiments. Still, there was no method that would clearly outperform the others in all the cases.

In face verification, clear improvement over the baseline algorithms were achieved by the advanced verification methods. In particular, the PS_MLBPHLDA_tnorm showed very good verification performance, demonstrating the importance of score normalization in the authentication task. This method achieved a half total error rate of 2.11 % on the P protocol of BANCA dataset, which is among the best values reported for this dataset.

Both the advanced speech verification systems were using the Factor Analysis technique to model the variability in speech samples. A clear improvement over the baseline systems was demonstrated, both on the NIST and BANCA speech data. On BANCA dataset, the GMM-LFA system achieved equal error rate of 3.37 % and the GMM-JFA system equal error rate of 4.29 %.

Acknowledgements

The authors would like to thank Tim Cootes (UMAN), Phil Tresadern (UMAN), Sebastien Marcel (IDIAP), Chris McCool (IDIAP), Matti Pietikäinen (UOULU), Abdenour Hadid (UOULU), Josef Kittler (UNIS), Norman Poh (UNIS), Chi-Ho Chan (UNIS), Honza Cernocky (BUT), Pavel Matejka (BUT), Ondrej Glembek (BUT), Christophe Levy (LIA), Anthony Larcher (LIA), Driss Matrouf (LIA) and Jean-Francois Bonastre (LIA) for their contributions to and review of this report.

References

[1] Timo Ahonen, Abdenour Hadid, and Matti Pietikäinen. Face description with local binary patterns: Application to face recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(12):2037–2041, 2006.

[2] Timo Ahonen, Esa Rahtu, Ville Ojansivu, and Janne Heikkilä. Recognition of blurred faces using local phase quantization. In *ICPR*, pages 1–4, 2008.

[3] Roland Auckenthaler, Michael Carey, and Harvey Lloyd-Thomas. Score normalization for text-independent speaker verification systems. *Digital Signal Processing*, 10(1-3):42 – 54, 2000.

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

[5] Harish Bhaskar and Philip A. Tresadern. D3.2: Report on the description and evaluation of baseline algorithms for unimodal authentication. Technical report, The MOBIO project, 2008.

[6] N. Brummer. Tools for fusion and calibration of automatic speaker detection systems. *http://www.dsp.sun.ac.za/~nbrummer/focal/index.htm*, 2005.

[7] L. Burget, N. Brümmer, D. Reynolds, P. Kenny, J. Pelecanos, R. Vogt, F. Castaldo, N. Dehak, R. Dehak, O. Glembek, Z. N. Karam, J. Noecker Jr., E. Na, C. C. Costin, V. Hubeika, S. Kajarekar, N., and J. Černocký. Robust speaker recognition over varying channels, report from jhu workshop 2008. Technical report, Johns Hopkins University, Baltimore, MD, 2008.

[8] L. Burget, M. Fapso, V. Hubeika, O. Glembek, M. Karafiat, M. Kockmann, P. Matejka, P. Schwarz, and J. Cernocky. But system for nist 2008 speaker recognition evaluation. In *Proc. Interspeech 2009*, pages 2335–2338, 2009.

[9] N.J. Butko and J.R. Movellan. Optimal scanning for faster object detection. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009)*, pages 2751–2758, 2009.

[10] F. Cardinaux, C. Sanderson, and S. Marcel. Comparison of mlp and gmm classifiers for face verification on xm2vts. *International Conference on Audio- and Video-based Biometric Person Authentication*, pages 1058–1059, 2003.

[11] 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.

[12] Chi Ho CHAN, Josef Kittler, Norman Poh, Timo Ahonen, and Matti Pietikainen. (multiscale) local phase quantisation histogram discriminant analysis with score normalisation for robust face recognition. In *ICCVWS*, 2009.

[13] Jie Chen, Xilin Chen, Jie Yang, Shiguang Shan, Ruiping Wang, and Wen Gao. Optimization of a training set for more robust face detection. *Pattern Recognition*, 42(11):2828 – 2840, 2009.

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

[15] J. Coughlan and S. Ferreira. Finding deformable shapes using loopy belief propagation. In *Proc. European Conf. on Computer Vision*, volume 3, pages 453–468, 2002.

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

[17] Augusto Destrero, Christine Mol, Francesca Odone, and Alessandro Verri. A regularized framework for feature selection in face detection and authentication. *Int. J. Comput. Vision*, 83(2):164–177, 2009.

[18] 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.

[19] H.K. Ekenel, J. Stallkamp, H. Gao, M. Fischer, and R. Stiefelhagen. Face recognition for smart interactions. In *Multimedia and Expo, 2007 IEEE International Conference on*, pages 1007–1010, 2-5 July 2007.

[20] P. Felzenszwalb and D. Huttenlocher. Pictorial structures for object recognition. *Int. J. Comput. Vis.*, 61(1):55–79, January 2005.

[21] B. Froba and A. Ernst. Face detection with the modified census transform. *IEEE International Conference on Automatic Face and Gesture Recognition*, pages 91–96, 2004.

[22] B. Froba and A. Ernst. Face detection with the modified census transform. In *AFGR*, pages 91–96, 2004.

[23] Andrea Frome, German Cheung, Ahmad Abdulkader, Marco Zennaro, Bo Wu, Alessandro Bissacco, Hartwig Adam, Hartmut Neven, and Luc Vincent. Large-scale privacy protection in google street view. In *Proc. IEEE International Conference on Computer Vision (ICCV 2009)*, 2009.

[24] O. Glembek, L. Burget, N. Dehak, N. Brummer, and P. Kenny. Comparison of scoring methods used in speaker recognition with joint factor analysis. In *Proc. ICASSP 2009*, 2009.

[25] Ralph Gross and Vladimir Brajovic. An image preprocessing algorithm for illumination invariant face recognition. In *4th International Conference on Audio- and Video-Based Biometric Person Authentication (AVBPA)*. Springer, June 2003.

[26] L. Gu, E. P. Xing, and T. Kanade. Learning GMRF structures for spatial priors. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2007.

[27] Guillaume Heusch, Yann Rodriguez, and Sebastien Marcel. Local binary patterns as an image preprocessing for face authentication. In *FGR '06: Proceedings of the 7th International Conference on Automatic Face and Gesture Recognition*, pages 9–14, Washington, DC, USA, 2006. IEEE Computer Society.

[28] P. Kenny, G. Boulianne, P. Oullet, and P. Dumouchel. Joint factor analysis versus eigenchannes in speaker recognition. *IEEE Transactions on Audio, Speech, and Language Processing*, 15(7):2072–2084, 2007.

[29] P. Kenny, P. Ouellet, N. Dehak, V. Gupta, and P. Dumouchel. A Study of Inter-Speaker Variability in Speaker Verification. *IEEE Trans. Audio, Speech and Language Processing*, 16(5):980–988, July 2008.

[30] J. Kittler, M. Hatef, R. P. W. Duin, and J. Matas. On combining classifiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20:226–239, 1998.

[31] V. Kolmogorov. Convergent tree-reweighted message passing for energy minimization. *IEEE Trans. Pattern Anal. Mach. Intell.*, 28(10):1568–1583, 2006.

[32] C. Lee and J. Gauvain. *Bayesian adaptive learning and MAP estimation of HMM*, pages 83–107. Kluwer Academic Publishers, Boston, Massachusetts, USA, 1996.

[33] L. Liang, F. Wen, Y.-Q. Xu, X. Tang, and H.-Y. Shum. Accurate face alignment using shape constrained Markov network. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition*, 2006.

[34] S. Lucey and T. Chen. A gmm parts based face representation for improved verification through relevance adaptation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 855–861, 2004.

[35] Yui Man Lui, J. Ross Beveridge, Bruce A. Draper, and Michael Kirby. Image-set matching using a geodesic distance and cohort normalization. In *FG*, pages 1–6, 2008.

[36] P. Matejka and J. Cernocky. D7.1: Planning of evaluation campaigns. Technical report, The MOBIO project, 2008.

[37] Driss Matrouf, Jean-Francois Bonastre, Corinne Fredouille, Anthony Larcher, Salah Mezaache, Mitchell McLaren, and Fernando Huenupan. LIA GMM-SVM system description: NIST SRE. In *NIST SRE*, Montreal (Canada), april 2008.

[38] Driss Matrouf, Nicolas Scheffer, Benoit Fauve, and Jean-Francois Bonastre. A Straightforward and Efficient Implementation of the Factor Analysis Model for Speaker Verification. In *Interspeech*, Pittsburgh (USA), 2007.

[39] T. Minka. Expectation-maximization as lower bound maximization. Technical report, Microsoft, 1998.

[40] A. Nefian and Monson H. Hayes III. Hidden markov models for face recognition. *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech, and Signal Processing*, 5:2721–2724, 1998.

[41] Timo Ojala, Matti Pietikainen, and Topi Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987, 2002.

[42] Ville Ojansivu and Janne Heikkilä. Blur insensitive texture classification using local phase quantization. In *International Conference on Image and Signal Processing (ICISP 2008)*, pages 236–243, 2008.

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

[44] D. Reynolds. Comparison of background normalization methods for text-independent speaker verification. *Proc. European Conference on Speech Communication and Technology (Eurospeech)*, 2:963–966, 1997.

[45] D. Reynolds, T. Quatieri, and R. Dunn. Speaker verification using adapted Gaussian mixture models. *Digital Signal Processing*, 10(1/2/3):19–41, 2000.

[46] Yann Rodriguez. *Face Detection and Verification using Local Binary Patterns.* PhD thesis, Ecole Polytechnique Fdrale de Lausanne, 2006.

[47] Henry A. Rowley, Shumeet Baluja, and Takeo Kanade. Neural network-based face detection. *IEEE Transactions On Pattern Analysis and Machine intelligence*, 20:23–38, 1998.

[48] F. Samaria and F. Fallside. Face identification and feature extraction using hidden markov models. *Image Processing: Theory and Applications*, pages 295–298, 1993.

[49] F. Samaria and S. Young. Hmm-based architecture for face identification. *Image and Vision Computing*, 12(8):537–543, 1994.

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

[51] P. Schwarz, P. Matejka, and J. Cernocky. Hierarchical structures of neural networks for phoneme recognition. In *Proc. ICASSP 2006*, Toulouse, France, May 2006.

[52] Hiromasa Takatsuka, Masayuki Tanaka, and Masatoshi Okutomi. Spatial merging for face detection. In *SICE-ICASE International Joint Conference*, pages 5587–5592, 2006.

[53] Hiromasa Takatsuka, Masayuki Tanaka, and Masatoshi Okutomi. Distribution-based face detection using calibrated boosted cascade classifier. In *ICIAP '07: Proceedings of the 14th International Conference on Image Analysis and Processing*, pages 351–356, 2007.

[54] Xiaoyang Tan and Bill Triggs. Enhanced local texture feature sets for face recognition under difficult lighting conditions. In Shaohua Kevin Zhou, Wenyi Zhao, Xiaoou Tang, and Shaogang Gong, editors, *AMFG*, volume 4778 of *Lecture Notes in Computer Science*, pages 168–182. Springer, 2007.

[55] P. A. Tresadern, H. Bhaskar, S. A. Adeshina, C. J. Taylor, and T. F. Cootes. Combining local and global shape models for deformable object matching. In *Proc. British Machine Vision Conf.*, September 2009.

[56] Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1:511–518, 2001.

[57] R. Vogt and S. Sridharan. Explicit modelling of session variability for speaker verification. *Computer Speech & Language*, 22(1):17–38, 2008.

[58] Haitao Wang, Stan Z. Li, and Yangsheng Wang. Face recognition under varying lighting conditions using self quotient image. In *FGR*, pages 819–824, 2004.

[59] Fei Yang, Shiguang Shan, Bingpeng Ma, Xilin Chen, and Wen Gao. Using score normalization to solve the score variation problem in face authentication. In *IWBRS*, pages 31–38, 2005.

[60] B. Zhang, S. Shan, X. Chen, and W. Gao. Histogram of gabor phase patterns (hgpp): A novel object representation approach for face recognition. *Image Processing, IEEE Transactions on*, 16(1):57–68, Jan. 2007.

[61] Lun Zhang, Rufeng Chu, Shiming Xiang, ShengCai Liao, and Stan Z. Li. Face detection based on multi-block lbp representation. In *Advances in Biometrics, International Conference, ICB 2007, Seoul, Korea, August 27-29, 2007, Proceedings*, volume 4642 of *Lecture Notes in Computer Science*, pages 11–18. Springer, 2007.

[62] Xuan Zou, J. Kittler, and K. Messer. Illumination invariant face recognition: A survey. In *Biometrics: Theory, Applications, and Systems, 2007. BTAS 2007. First IEEE International Conference on*, pages 1–8, Sept. 2007.