Face Recognition in Challenging Environments: An Experimental and Reproducible Research Survey

Manuel Günther and Laurent El Shafey and Sébastien Marcel

Abstract One important type of biometric authentication is face recognition, a research area of high popularity with a wide spectrum of approaches that have been proposed in the last few decades. The majority of existing approaches are conceived for or evaluated on constrained still images. However, more recently research interests have shifted towards unconstrained "in-the-wild" still images and videos. To some extent, current state-of-the-art systems are able to cope with variability due to pose, illumination, expression, and size, which represent the challenges in unconstrained face recognition. To date, only few attempts have addressed the problem of face recognition in mobile environment, where high degradation is present during both data acquisition and transmission. This book chapter deals with face recognition in mobile and other challenging environments, where both still images and video sequences are examined. We provide an experimental study of one commercial of-the-shelf and four recent open-source face recognition algorithms, including color-based linear discriminant analysis, local Gabor binary pattern histogram sequences, Gabor grid graphs and inter-session variability modeling. Experiments are performed on several freely available challenging still image and video face databases, including one mobile database, always following the evaluation protocols that are attached to the databases. Finally, we supply an easily extensible open-source toolbox to re-run all the experiments, which includes the modeling techniques, the evaluation protocols and metrics used in the experiments, and provides a detailed description on how to re-generate the results.

Manuel Günther, Idiap Research Institute, Martigny, Switzerland e-mail: manuel.guenther@idiap.ch · Laurent El Shafey, Idiap Research Institute, Martigny, Switzerland e-mail: laurent.el-shafey@idiap.ch · Sébastien Marcel, Idiap Research Institute, Martigny, Switzerland e-mail: marcel@idiap.ch ·

1 Introduction

After the first automatic face recognition algorithms [1, 2] appeared more than three decades ago, this area has attracted many researchers and there has been a huge progress in this field. One of the reasons of its popularity is the broad field of applications of (automatic) face recognition. Due to the availability of mobile camera sensors included into devices such as digital cameras, mobile phones or laptops, new applications of face recognition appeared recently. One such application is the automatic unlocking of the mobile device, when the user is present in front of the camera or screen. Other applications include the recognition of faces in images in order to aid the user categorizing or memorizing people. The particularity of these applications is that imaging conditions are usually uncontrolled and people in the images or videos have different facial expressions, face poses and are possibly partially occluded. In this book chapter we investigate several face recognition algorithms regarding their capability to deal with these kinds of conditions.

Commonly, the face recognition task is composed of several stages. The first stage is face detection, in which location and scale of the face(s) in the image is estimated [3, 4] and the image is geometrically regularized to a fixed image resolution. The regularized face images are then subjected to a photometric enhancement step, which mainly reduces effects of illumination conditions [5, 6]. Then, image features that contain the relevant information needed for face recognition are extracted [7, 8, 9]. Features of some of the images are used to enroll a person-specific template, while the features of the remaining images are used for probing. Based on these extracted features, different face recognition algorithms have been developed during the last decades. They can be classified into two major categories: In the discriminative approach, to which most algorithms belong [8, 10, 11, 12], it is classified whether template and probe belong to the same identity or not. The generative approach [13, 14] computes the probability that a given person could have produced the probe sample.

To evaluate face recognition algorithms, several publicly accessible databases of facial images and videos exist. One important mobile database is MOBIO [15], which contains voice, image and video recordings from mobile phones and laptops. Other unconstrained state-of-the-art face databases are the Labeled Faces in the Wild (LFW) database [16] and the YouTube Faces database [17]. The impact of specific facial appearances such as facial expression, face pose and partial occlusion are investigated based on the Multi-PIE [18] and the small AR face [19] databases. To ensure a fair comparison of face recognition algorithms, image databases are accompanied with evaluation protocols, which all of our experiments follow strictly.

Along with this book chapter, we provide the source code[1] not only for the algorithms, but also for the complete experiments from the raw images or videos to the final evaluation, including the figures and tables that can be found in this chapter. Most of the algorithms use Bob [20], a free signal processing and machine learning

[1] `https://pypi.python.org/pypi/bob.chapter.FRICE`

toolbox for researchers.[2] Some algorithms are taken from the CSU Face Recognition Resources,[3] which provide the baseline algorithms for the Good, the Bad & the Ugly (GBU) face recognition challenge [21, 22]. Finally, all experiments are executed using the FaceRecLib [23],[4] which offers an easy interface to run face recognition experiments either using already implemented face recognition algorithms, or rapidly prototyping novel ideas.

The remaining of this chapter is structured as follows: In sec. 2 we give an overview of related work on face recognition in challenging environments, and a brief survey of reproducible research in biometrics. Sec. 3 describes the databases, the methodology and the results of our face recognition experiments. Finally, Sections 4 and 5 close the paper with a detailed discussion of the tested face recognition algorithms and a conclusion.

2 Related Work

2.1 Reproducible Research in Biometrics

Biometrics research is an interdisciplinary field that combines expertise from several research areas. Examples of these scattered disciplines are: image preprocessing and feature extraction that are from the field of signal and image processing; machine learning, which is required for subspace projections or data modeling; or pattern recognition and distance computations as part of the information theory. Additionally, to make results comparable, a proper implementation of the required evaluation protocols of biometric databases need to be provided. This makes biometrics research a particularly difficult case, especially when comparable results should be provided. Hence, often biometric algorithms are tested only on a few of the available databases. Also, the results of other researchers can not be reproduced since they do not publish all of the meta-parameters of their algorithms. Therefore, survey papers like [24, 25, 26, 27, 28] can only report the results of other researchers, so "it is really difficult to declare a winner algorithm" [24] since "different papers may use different parts of the database for their experiments" [28].

One way of providing comparable results is to apply the concept of reproducible research.[5] A reproducible research paper is comprised of several aspects [29], which makes it possible and easy to exactly reproduce experiments:

- a research publication that describes the work in all relevant details
- the source code to reproduce all results
- the data required to reproduce the results

[2] `http://www.idiap.ch/software/bob`

[3] `http://www.cs.colostate.edu/facerec/algorithms/baselines2011.php`

[4] `http://pypi.python.org/pypi/facereclib`

[5] `http://www.reproducibleresearch.net`

- instructions how to apply the code on the data to replicate the results on the paper

One reason for providing reproducible research, besides making the lives of other researchers easier, is the visibility of the resulting scientific publications. As [30] showed, the average number of citations for papers that provide source-code in the Transactions on Image Processing (TIP) is seven times higher than of papers that do not.

There have been attempts to foment reproducibility of research results in the biometric community with the release of public software [20, 23, 31, 32] and datasets [15, 16, 33, 34]. Various biometric communities organize open challenges [35, 36], for which web-based solutions for data access and result posting are particularly attractive [37]. Some dataset providers also publish an aggregation of the results of different algorithms on their web pages.[6] However, cases where those components are used in a concerted effort to produce a reproducible publication remain rare.

Particularly, two groups of researchers currently try to push forward the reproducibility of biometric recognition experiments. On one hand, OpenBR [32] is an open source C++ library of algorithms to perform biometric recognition experiments. Unfortunately, this library only has a limited set of algorithms and biometric databases, which it can evaluate. On the other hand, the FaceRecLib [23] is an easy-to-use and easy-to-extend Python library that can be used to run complete face recognition experiments on various face image and video databases. Several reproducible research papers based on the FaceRecLib have already been published,[7] using the Python Package Index (PyPI) as a source code distribution portal. All results of the experiments that are reported in this book chapter rely on the FaceRecLib.

Further research on solutions for achieving, distributing and comparing results of biometric experiments in the reproducible research framework is carried out. Currently being under development, the Biometrics Evaluation And Testing (BEAT) platform[8] introduces a biometry-agnostic system for programming algorithms, workflows, running complete evaluations and comparing to other researcher's results only using a web browser.

2.2 Face Recognition in Challenging Environments

For several decades, research on face recognition in controlled environments has been fostered due to its high impact on practical applications such as automatic access or border control, where subjects cooperate with the system. In a study in 2007, it has been shown that automatic face recognition systems in controlled en-

[6] For example, the results on LFW [16] are published under: `http://vis-www.cs.umass.edu/lfw/results.html`

[7] One example for reproducible research based on the FaceRecLib can be found under: `http://pypi.python.org/pypi/xfacereclib.paper.BeFIT2012`

[8] `http://www.beat-eu.org/platform`

vironments can surpass human performance [38], when identities in the images are not previously known to the participants [39].

After having satisfactorily solved face recognition in controlled environments, research interests shifted towards unconstrained environments, where subjects do not cooperate with the face recognition system. Three main directions of applications have arisen: Identifying persons in uncontrolled high quality images to tag private photos with identities using application like Picasa or iPhoto; identifying suspects in low resolution surveillance camera videos; and verifying owners of mobile devices or cars to avoid thefts. Due to the availability of several image and video databases [16, 17] for the first application, research was lead towards this direction. On the other hand, only few databases with surveillance camera [40] or mobile [15, 41] data are available, so this area of face recognition research is still under-developed.

The latest trend for face recognition in uncontrolled environments is the usage of deep convolutional neural networks [42, 43]. Those networks are usually proprietary software and require a huge amount of training data, which is not publicly available and, thus, the reproducibility level of these publications is 0 according to [29]. In [44], Bayesian face recognition [45] is revisited and extended to work with mixtures of Gaussians for both the intrapersonal and the extrapersonal class, using LBP histogram sequences as features. However, they learned their method using training data (PubFig) that overlaps with their test images (LFW), making their experimental results strongly biased. So far, none of these methods is included in our evaluation, though their future integration into the experimental setup is foreseen.

The Point-and-Shoot Face Recognition Challenge (PaSC) [46] investigated five different algorithms on the PaSC data set [41], which contains unconstrained images and videos of indoor and outdoor scenes. The authors of the best performing system [47] claim that their Eigen-PEP approach is naturally robust to pose variations. It would be nice to be able to include their system into our study, but to date we were not able to reimplement their algorithm.

In a study, [48] performed a large scale feature selection to perform unconstrained face recognition. They modeled the low-level feature extraction of the human brain and achieved good results on image pairs with similar pose. However, they found that image pairs with different identities in comparable face pose most often are more similar than images with the same identity but different poses. Hence, those features work well in constrained face recognition, but not as well with unconstrained face image data.

Previous studies [49] have found that Gabor jet and LBP based algorithms are well suited for face recognition in unconstrained environments. Also, color information [22] have shown to contain data useful for face recognition. Furthermore, advanced modeling techniques [50] showed good verification performance on uncontrolled mobile data. Finally, the fusion [51] of several different approaches for unconstrained face recognition was able to outperform single systems.

However, so far no reproducible study has been performed that analyzes face recognition algorithms according to their behavior in presence of (uncontrolled) illumination, facial expression, face pose and partial occlusions. The reproducibility of the present study is guaranteed due to the availability of the data and the algo-

rithms, as well as the evaluation protocols and methods. Furthermore, a properly documented script that shows, how to regenerate all results, is provided.

3 Experiments

This section provides an overview of our experimental evaluation. The employed algorithms are explained and the evaluated databases are presented, including a brief description of the databases and evaluation metrics. After optimizing the configurations of the algorithms, the performance of the algorithms under three different sets of experiments are evaluated. First, the dependence on the single variations facial expression, face pose and partial occlusions is investigated. Second, the performance in an uncontrolled image database is evaluated and the extensibility to video face recognition is tested. Finally, the results of the algorithms on a mobile image and video database are reported.

3.1 Face Recognition Techniques

The face recognition algorithms that we test in our evaluation are recent open-source approaches to still image face recognition. All algorithms are adapted to process several images for template enrollment and for probing. Additionally, several image preprocessing techniques are evaluated.

The implementation of the preprocessing techniques and three of the face recognition algorithms relies on the open source toolbox Bob [20], which provides functionality in a research-friendly Python environment and implements identified bottlenecks in C++. One algorithm is taken from the CSU face recognition resources [22], which is completely implemented in Python. To test the advantage of commercial systems over the open-source approaches, additionally one Commercial Off-The-Shelf (COTS) algorithm is investigated. In our experiments, the evaluation of video data is performed by sub-sampling the frames of the videos and providing the algorithms with several images per video.

Though we run several of face recognition algorithms, there is a common execution order to perform a face recognition experiment. Given a raw image or video from a certain database, the first stage is to detect the face, remove the background information and geometrically normalize the face. Throughout our experiments, for image databases, we use the hand-labeled annotations provided by the databases to geometrically normalize the face, while for video databases we detect the faces [52] and eye locations [53] in each used frame. The aligned face image is further processed using some preprocessing technique, usually to attenuate the effects of illumination.

In the next step, features are extracted from the preprocessed images. Features from one or more images of one identity are used to enroll a template of the person,

(a) Orig **(b)** None **(c)** HEQ **(d)** SQI **(e)** T&T **(f)** LBP

Fig. 1: IMAGE PREPROCESSING TECHNIQUES. *This figure shows the effect of different image preprocessing techniques on the (a) original image: (b) no preprocessing, (c) histogram equalization, (d) self quotient image, (e) Tan & Triggs algorithm and (f) LBP feature extraction.*

and several of those templates are used as a gallery. These templates are compared with probe features of other images or videos, and a similarity score is computed for each template/probe pair. Since face recognition algorithms are usually bound to a specific type of features, we present both the feature extraction and the modeling and comparison techniques together as combines algorithms.

Finally, the scores are evaluated to compute the final performance measure, using one of the evaluation metrics defined in sec. 3.2.1.

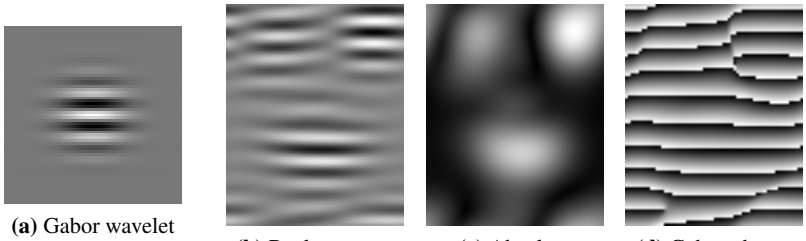
3.1.1 Image Preprocessing

Before a preprocessing technique is applied, the image is converted to gray scale and aligned. This implies that the image is geometrically normalized such that the left and right eyes are located at specific locations in the aligned image, e. g., $\mathbf{a}_l = (48, 16)^\top$ and $\mathbf{a}_r = (15, 16)^\top$, and the image is cut to a resolution of, e. g., 64×80 pixels. Fig. 1(b) shows the result of the alignment of the image shown in fig. 1(a).

To reduce the impact of illumination, we test four different preprocessing techniques, which are always executed on the aligned image. The first algorithm is Histogram Equalization (HEQ) [54]. Second, we investigate the Self Quotient Image (SQI) algorithm [55]. Third, we examine the multistage preprocessing technique (T&T) as presented by Tan and Triggs [6]. Finally, we examine a preprocessing technique [5] based on Local Binary Patterns (LBP). Examples of preprocessed images can be found in fig. 1.

3.1.2 Linear Discriminant Analysis on Color Channels

An extension of Linear Discriminant Analysis (LDA) to the two color channels I-chrominance and the Red channel (LDA-IR) has been proposed in [22]. After a geometric normalization of the face, the raw pixels are concatenated to form a one-dimensional feature vector. A PCA+LDA transformation matrix, which is a combination of the Principal Component Analysis (PCA) and LDA projection, is



(a) Gabor wavelet **(b)** Real response **(c)** Absolute response **(d)** Gabor phases

Fig. 2: GABOR WAVELET TRANSFORM. *This figure displays the (b) real part, (c) absolute values, and (d) Gabor phases of the convolution of the image from fig. 1(b) with the (a) Gabor wavelet.*

computed independently for both color channels. Each channel is projected into its corresponding subspace, and both projected vectors are concatenated to form the final feature vector.

In the template enrollment step, all enrollment features are simply stored. Since none of the other algorithms are allowed to use cohort data for score normalization, we decided to disable[9] the cohort normalization usually applied in [22]. This transforms the distance function between a template and a probe feature into a simple Euclidean distance. The final score is empirically found to be the minimum distance value.

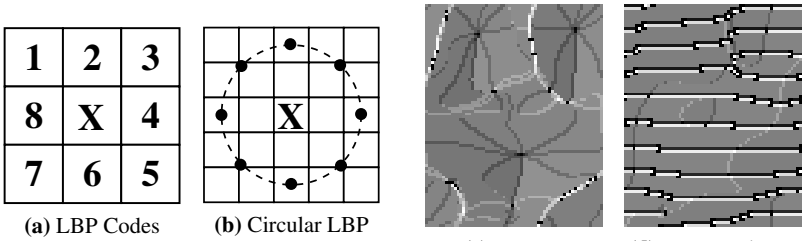
LDA-IR is the only examined algorithm that incorporates color information into the face recognition process. Therefore, it cannot be combined with the preprocessing techniques defined in sec. 3.1.1 Hence, image alignment and feature extraction rely on the original implementation of the LDA-IR algorithm.

3.1.3 Gabor Grid Graphs

The idea of the Graphs algorithm relies on a Gabor wavelet transform [56, 57]. The preprocessed image is transformed using a family of $j = 1, \ldots, 40$ complex-valued Gabor wavelets, which is divided into the common set of 8 orientations and 5 scales [56]. The result of the Gabor transform are 40 complex-valued image planes in the resolution of the preprocessed image. Commonly, each complex-valued plane is represented by absolute values and phases. The transform process for a single Gabor wavelet is visualized in fig. 2.

From these complex planes, grid graphs of Gabor jets are extracted. A Gabor jet is a local texture feature, which is generated by concatenating the responses of

[9] To avoid misunderstandings, we do not use the name CohortLDA as in [22], but we stick to the old name of the algorithm (LDA-IR).



(a) LBP Codes **(b)** Circular LBP **(c)** LBP on absolute values **(d)** LBP on phases

Fig. 3: LOCAL GABOR BINARY PATTERNS. *This figure displays the generation process of (a) LBP codes and (b) the circular $LBP^{u2}_{8,2}$ operator. Additionally, the results of the $LBP^{u2}_{8,2}$ operator on (c) the absolute Gabor wavelet responses and (d) the Gabor phases (as given in fig. 2(c) and fig. 2(d)) are shown.*

all Gabor wavelets at a certain offset-position in the image. As shown by [58], it is beneficial to normalize the absolute values in a Gabor jet to unit Euclidean length.

In our implementation, the bunch graph [56] concept is used for template enrollment. For each node position, the Gabor jets from all enrollment graphs are stored. For the comparison of template and probe, we investigate several local and global scoring strategies. Each strategy relies on a comparison of Gabor jets, which employs one of several Gabor jet similarity function [8, 56, 58]. In the optimal strategy (see sec. 3.3), an average of the local maximum of similarities is computed, using a similarity function partially based on Gabor phases [8].

3.1.4 Local Gabor Binary Pattern Histogram Sequences

In the Local Gabor Binary Pattern Histogram Sequences (LGBPHS) [59], three different approaches of face recognition are combined. First, the preprocessed image is Gabor wavelet transformed [56], which leads to 40 complex-valued representations of the images. Then, Local Binary Patterns (LBP's) [60] are extracted from the absolute and the phase part [59]. An LBP is generated by comparing the gray value of a pixel with the gray values of its neighbors, resulting in a binary representation with discrete values between 0 and 255. The extraction process of LBPs from Gabor wavelet responses is illustrated in fig. 3. Different LBP variants like circular or uniform patterns [61] are evaluated.

In order to obtain local features, these image planes are split into possibly overlapping image blocks [62]. As each bit of the LBP code is similarly important, these codes cannot be compared with a simple distance function. Instead, LBP codes are collected in histograms, one for each block and each Gabor wavelet. Concatenating all these histograms into one histogram sequence ends up in a huge feature vector, which is called the Extended Local Gabor Binary Pattern Histogram Sequence (ELGBPHS) [59].

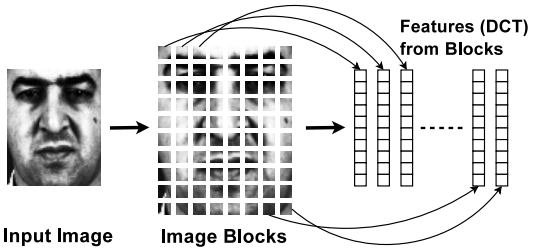


Fig. 4: DCT FEATURE EXTRACTION. *This figure shows the computation of parts-based features by decomposing an image into a set of blocks and extracting DCT features from each block.*

To enroll a template from several images, we decided to compute the average over histogram sequences (which results in non-integral numbers in the histograms). Finally, template and probe features can be compared using dedicated histogram similarity measures such as histogram intersection, the χ^2 distance or the Kullback-Leibler divergence.

3.1.5 Inter-Session Variability Modeling

An alternative to previously detailed discriminative approaches to automatic face recognition is to describe the face of a person by a generative model. The idea is to extract local features from the image of a subject's face before modeling the distribution of these features with a Gaussian Mixture Model (GMM) [7, 63], instead of concatenating them as usually done in discriminative approaches.

Parts-based features [7] are extracted by decomposing preprocessed images into overlapping blocks. A 2D Discrete Cosine Transform (DCT) is applied to each block before extracting the lowest-frequency DCT coefficients. These coefficients are used to build the descriptor of a given block, after applying proper pre- and post-processing of each block [13]. This feature extraction process is detailed in fig. 4.

The distribution of the features for a given identity are modeled by a GMM with several multivariate Gaussian components [64]. To overcome the issue of limited enrollment data, first a Universal Background Model (UBM) is estimated as a prior [64], which is later adapted to the enrollment samples of a person using a Maximum A Posteriori (MAP) estimation [65]. It has been shown that such an approach offers descent performance with a reasonable complexity [66].

In the context of a GMM-based system, Inter-Session Variability (ISV) modeling [67] is a technique that has been successfully employed for face recognition [50, 68]. In ISV, it is assumed that within-person variation is contained in a linear subspace and by adding a corresponding offset to the GMM means describing each sample. A template is enrolled by suppressing those session-dependent components from the feature vectors and yielding the true session-independent person-specific template GMM.

To compare the template GMM with probe features, a two-fold similarity measure is applied. First, the session-dependent offset is estimated for the probe sample. Since the session-offset is estimated at both enrollment and probing time, it significantly reduces the impact of within-person variation. Second, the log-likelihood ratio score is computed by comparing the probe features both to the template GMM as well as to the UBM. A more detailed description of this algorithm can be found in [67, 50].

3.1.6 Commercial Of-The-Shelf Algorithm

We obtained a Commercial Of-The-Shelf (COTS) face recognition system[10] with a C++ interface for algorithms used in several steps in the face recognition tool chain. Obviously, no detailed information of the employed algorithms is known. We wrote a Python interface for a small subset of this functionality that allowed us to run the COTS algorithms in the FaceRecLib. Particularly, we implemented bindings for functions to extract features, to enroll a template from several features, and to compute scores given one template and one probe feature.

Although the C++ interface of COTS provides functionality for face and eye landmark detection, we rely on the same data as in the other experiments as detailed below. Particularly, we use hand-labeled eye locations in the image experiments, and our face detection and landmark localization algorithm in the video experiments. The first reason is that we want to assure that all algorithms see exactly the same data, and secondly some of the faces in the MOBIO database are not found correctly by the COTS face detection algorithm.

3.2 Databases and Evaluation Protocols

To guarantee a fair comparison of algorithms, it is required that all algorithms are provided with the same image data for training and enrollment, and the same pairs of template and probe data are evaluated. This is achieved by defining evaluation protocols, which might either be biased, i. e., (partially) having the data of the same identities in the training and the test set, or unbiased by splitting the identities between the sets. For all databases used in this book chapter, we provide an implementation of the protocols, a more complete list of implemented database interfaces is given on the Bob web page.[11]

[10] The COTS vendor requested to stay anonymous.

[11] `http://github.com/idiap/bob/wiki/Packages`

3.2.1 Evaluation Metrics

The evaluation protocols of all databases used in our evaluation define a verification scenario. Several evaluation measures exist, which are all built on top of the False Acceptance Rate (FAR) and the False Rejection Rate (FRR). To compute these rates, the scores are split into genuine scores s_{gen}, which result from comparing template and probe from the same person, and impostor scores s_{imp}, where template and probe of different identities are compared [69]. FAR and FRR are defined over a certain threshold θ:

$$\text{FAR}(\theta) = \frac{\left|\left\{s_{\text{imp}} \mid s_{\text{imp}} \geq \theta\right\}\right|}{\left|\left\{s_{\text{imp}}\right\}\right|} \qquad \text{FRR}(\theta) = \frac{\left|\left\{s_{\text{gen}} \mid s_{\text{gen}} < \theta\right\}\right|}{\left|\left\{s_{\text{gen}}\right\}\right|} \tag{1}$$

In most of the evaluated protocols, the data is split in three sets: a training set, a development set and an evaluation set. Scores and FAR/FRR are computed for both the development and the evaluation set independently. Then, a threshold θ^* is obtained based on the intersection point of FAR and FRR curves of the development set. This threshold is used to compute the Equal Error Rate (EER) on the development set and the Half Total Error Rate (HTER) on the evaluation set:

$$\text{EER} = \frac{\text{FAR}_{\text{dev}}(\theta^*) + \text{FRR}_{\text{dev}}(\theta^*)}{2} \quad \text{HTER} = \frac{\text{FAR}_{\text{eval}}(\theta^*) + \text{FRR}_{\text{eval}}(\theta^*)}{2} \tag{2}$$

There are two databases, for which a different evaluation protocol is provided, i. e., LFW and YouTube (see sec. 3.2.2). In the protocol, pairs of images or videos are specified, for which a score should be computed. In our case, we always choose the first image or video of the pair for template enrollment and the second as probe. In both databases, the subjects are split into 10 different subsets, so-called `folds`. In each `fold`, 300 (LFW) or 250 (YouTube) genuine pairs and the same amount of impostor pairs exist. For each `fold`, the Classification Success (CS) is computed:

$$\text{CS} = \frac{\left|\left\{s_{\text{gen}} \mid s_{\text{gen}} \geq \theta^*\right\}\right| + \left|\left\{s_{\text{imp}} \mid s_{\text{imp}} < \theta^*\right\}\right|}{\left|\left\{s_{\text{gen}}\right\}\right| + \left|\left\{s_{\text{imp}}\right\}\right|} \tag{3}$$

We use our own implementation this 10-fold protocol, which provides an additional development set, from which the threshold θ^* in eq. (3) is estimated. For each of the 10 experiments, 7 `folds` are used for training, the development set is built from 2 `folds`, and the last `fold` is employed to compute the CS. Finally, as required by [16], the mean and the standard deviation of the CSs over all 10 experiments is reported. For the LFW database, we chose the unrestricted configuration [16] since the identity information is required by some algorithms, which is forbidden to be used in the image-restricted training set. However, none of our algorithms is provided with additional external training data.

Protocol	Training set		Development set			Evaluation set			Remark
	ident.	files	templ.	enroll	probe	templ.	enroll	probe	
Multi-PIE *(images)*			*all template/probe pairs*						[18]
U	208	9785	64	64	4864	65	65	4940	controlled non-frontal illumination
E		1095			576			585	controlled facial expressions
P		7725			3328			3380	face poses in 15% yaw angles
XM2VTS *(images)*			*all template/probe pairs*						[33]
darkened	200	600	200	600	800	200	600	800	non-frontal illumination
AR face *(images)*			*all template/probe pairs*						[19]
illumination	50	329	43	86	258	43	86	258	controlled non-frontal illumination
occlusion		827			172			172	sunglasses and scarfs
both		827			344			344	illumination and occlusion
BANCA *(images)*			*selected template/probe pairs*						[70]
P	30	300	26	130	2370	26	130	2370	diverse illumination
LFW *(images)*			*selected template/probe pairs*						[16]
foldX[12]	4024	9263	913	913	915	456	456	458	uncontrolled images
MOBIO *(images/videos)*			*all template/probe pairs*						[15]
male	50	9600	24	120	2520	38	190	3990	mobile recordings of men
female			18	90	1890	20	100	2100	mobile recordings of women
YouTube *(videos)*			*selected template/probe pairs*						[17]
foldX[12]	1013	2288	500	500	490	250	250	245	uncontrolled videos

Table 1: DATABASES AND PROTOCOLS. *This table lists the evaluation protocols of the databases used in our experiments. For the training set, the number of training identities and training files is given. For both the development and the evaluation set, the number of templates, the number of enrollment files and the number of probe files is provided.*

3.2.2 Databases

This section specifies the image and video databases including their evaluation protocols, which are used in our experiments. An overview of the databases and protocols is given in tab. 1.

The CMU Multi-PIE database [18] consists of 755,370 images shot in 4 different sessions from 337 subjects. We generated and published several unbiased face verification protocols, all of which are split up into a training, a development and an evaluation set. The training set is composed of 208 individuals, while the size of development set (64 identities) and evaluation set (65 identities) is almost equal. In each protocol, a single image per person with neutral facial expression, neutral illumination and frontal pose are selected for template enrollment. The probe sets contain images with either non-frontal illumination (protocol U), facial expressions (protocol E) or face poses (protocol P).

[12] In total, 10 folds (fold1 to fold10) exist in the LFW and YouTube protocols, here we provide average counts.

XM2VTS [33] is a comparably small database of 295 subjects. We use only the `darkened` protocol in our image preprocessing experiments, which includes non-frontally illuminated images. The particularity of the `darkened` protocol is that the training and development set consists of well-illuminated images, while the evaluation set consists of non-frontally illuminated ones. The enrollment of a template is performed with 3 images per person, whereas 4 probe files per identity are used to compute the scores. The training set consists of exactly the same images as used for template enrollment [33], making the protocol biased.

The AR face database [19] contains 3312 images[13] from 76 male and 60 female identities taken in two sessions. Facial images in this database include three variations: facial expressions, strong controlled illumination, and occlusions with sunglasses and scarfs. We have created and published several unbiased verification protocols for this database, splitting up the identities into 50 training subjects (28 men and 22 women) and each 43 persons (24 male and 19 female) in the development and evaluation set. For template enrollment, we use those two images per identity that have neutral illumination, neutral expression and no occlusion. The protocols `occlusion`, `illumination` and `both` test the specific image variations that are defined in the database, i. e., probe images have either partially occluded faces, non-frontal illumination, or both occlusion and illumination. The training set for the `illumination` protocol is comprised of images with illumination variations only, whereas in the training sets for `occlusion` and `both`, occluded faces are additionally included.

Originally, in BANCA [70] video and audio recordings of 52 persons were captured for each 4 different languages, where the participants were asked to utter prompted sequences. Recordings were taken in 12 different sessions. In each session, every subject generated two videos, one true genuine access and one informed impostor access. From each of these videos, 5 images and one audio signal were extracted. However, only the English language was made available [70], together with several unbiased open set verification protocols. We here take only the most challenging protocol `P`, in which templates are enrolled from 5 controlled images, while the system is probed with controlled, degraded and adverse images. Two particularities of this database are that it is small, e. g., the training set consists of only 300 images and that the number of 2340 genuine and 3120 impostor scores is balanced.

One of the most popular image databases is the Labeled Faces in the Wild (LFW) database [16]. It contains 13,233 face images from 5749 celebrities, which were downloaded from the internet and labeled with the name of the celebrity. In most images, faces in close-to-frontal poses with good illumination are shown, some examples are given in fig. 8(a). In fact, there is an ongoing discussion if the LFW data set is fully representative for unconstrained face recognition [48]. In this work, we use the images aligned by the funneling algorithm [71]. The database owners do

[13] The website `http://www2.ece.ohio-state.edu/~aleix/ARdatabase.html` reports more than 4000 images, but we could not reach the controller of the database to clarify the difference.

not provide the eye locations for the images, but we rely on publicly available[14] automatically extracted annotations [72].

The MOBIO database [15] consists of video data of 150 people taken with mobile devices like mobile phones or a laptop, we here use only the mobile phone data. For each person, 12 sessions were recorded. The faces visible in these recordings differ in facial expression, pose, illumination conditions, and sometimes parts of the face are not captured by the device. Along with the MOBIO database, two gender-specific unbiased evaluation protocols `female` and `male` are provided, where exclusively female or male images are compared. In these protocols, 5 recordings per identity are used to enroll a template, and all probe files are tested against all templates of the same gender. The training set consists of 9600 recordings from 13 females and 37 males. In our experiments, we solely perform gender-independent training. The development set contains 18 female and 24 male identities, which are probed with 1890 or 2520 recordings, respectively. The evaluation set embraces 20 female and 38 male identities, using 2100 or 3990 probe files, respectively. For the MOBIO image database, one image was extracted from each video recording by choosing a single frame after approximately one second of video run time, and the eye centers were labeled by hand.

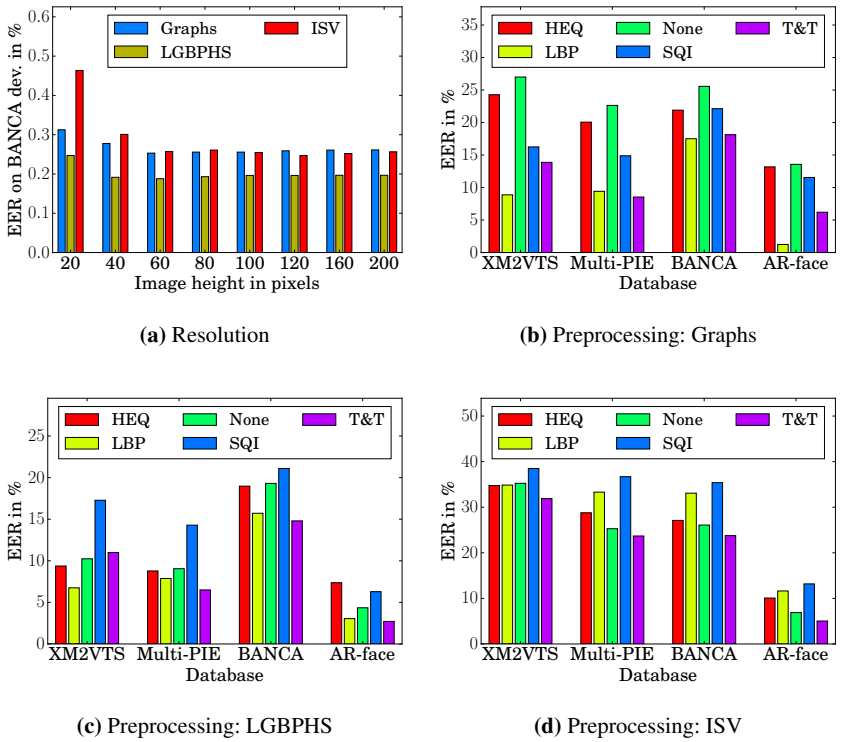
The YouTube Faces database [17] contains a collection of 3425 videos of 1595 celebrities collected from the YouTube video portal, showing faces in several poses and with good illumination. The length of a video sequence varies between around 50 to 6000 frames. Although the YouTube database is accompanied by bounding boxes that were detected for each frame in each of the videos, and pre-cropped frames that were aligned with the help of detected facial landmarks [17], we rely on our own face detector and landmark localization algorithm to align faces in all (used) frames.

3.3 Configuration Optimization

Any face recognition algorithm has several intrinsic meta-parameters, which we refer to as the algorithm configuration. Examples of such parameters are the number, resolution and overlap of blocks in the LGBPHS and the ISV algorithms, or the Gabor jet similarity metric used in the Graphs algorithm. To be as fair as possible, we optimize the configurations of all of the algorithms taken from Bob [20] independently. We do not optimize the configuration of LDA-IR since the configuration has been optimized already — though to another database — and defining new color transformations is out of the scope of this work.

We chose the BANCA database with protocol `P` to perform the optimization experiments since the database is small, but still quite challenging and focused on semi-frontal facial images as they occur in unconstrained or mobile databases. According to the designated use of the evaluation protocol, we optimize the algorithm

[14] `http://lear.inrialpes.fr/people/guillaumin/data.php`



(a) Resolution

(b) Preprocessing: Graphs

(c) Preprocessing: LGBPHS

(d) Preprocessing: ISV

Fig. 5: CONFIGURATION OPTIMIZATION. *This figure displays the results of image resolution and the preprocessing tests of the configuration optimization steps for the algorithms.*

configurations using the development set of BANCA. It should be noted that the goal of this study is to provide a replicable evaluation of a range of state-of-the-art face recognition algorithms for research to built upon. It is **not** the goal of this study to demonstrate the superiority of a single best face recognition algorithm.

One important aspect of face recognition is the resolution of the facial image and its content. Interestingly, there are only few publications, e. g., [9, 49, 58] that pay attention to this aspect, but rather every researcher uses his or her own image resolution. Hence, the first set of experiments that we conduct is to find out, which image resolution is best suited for face recognition. We execute all algorithms with configurations that we have set according to literature. We selected several different image resolutions, ranging from height 20 to 200 pixels, always keeping an aspect ratio of $4:5$ and the eye locations at the same relative coordinates. Also, configuration parameters that are sensitive to the image resolution are adapted accordingly. Note that we do not include LDA-IR in the image resolution evaluation since changing the parametrization of this algorithm in its original implementation is highly complex.

The resulting EER on protocol `P` of the BANCA development set are given in fig. 5(a). Interestingly, the results of most of the algorithms are very stable for any image resolution that is at least 32×40 pixels, which corresponds to an inter-eye-distance of 16 pixels. Only for resolutions smaller than that, results degrade. ISV and Graphs require resolutions that are a bit higher, but also these algorithms settle around 100 pixels image height. Since there is not much difference between the resolutions greater than 32×40 pixels, we choose to stick at the resolution 64×80 as used in many of our previous publications [14, 23, 68, 73] for the rest of our experiments.

One severe issue in automatic face recognition is uncontrolled or strong illumination. Several image preprocessing techniques that should reduce the impact of illumination in face recognition have been proposed (see sec. 3.1.1). Unfortunately, in literature there is no comprehensive analysis of image preprocessing techniques for face recognition, but each researcher uses a single preferred technique, if any.

To evaluate the preprocessing techniques, we execute them on three databases with challenging controlled illumination conditions: the XM2VTS database (protocol `darkened`), the Multi-PIE database (protocol `U`) and the AR face database (protocol `illumination`). Finally, we test the techniques on a database with uncontrolled illumination, for which we again select BANCA (protocol `P`). The results of the preprocessing test can be observed in fig. 5(b) -(d). Apparently, the preferred preprocessing technique differs between face recognition algorithms. However, there is an overall trend for the LBP-based and the Tan & Triggs preprocessing techniques, while histogram equalization and self quotient image do not perform as well and, obviously, neither executing no preprocessing technique at all.

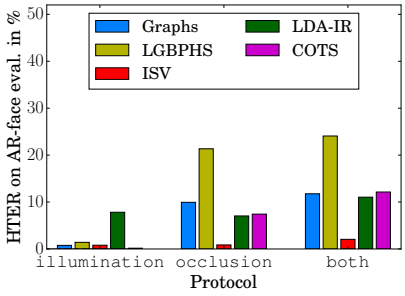
For each of the algorithms, we chose the best performing preprocessing technique for our following experiments, which is Tan & Triggs for LGBPHS and ISV, and the LBP-based preprocessing for Graphs.

After finding a suitable image resolution and the optimal image preprocessing technique for each algorithm, we optimize their configurations independently. Due to the partially large number of configuration parameters to be optimized, we performed optimization in several steps. Each step groups together configuration parameters that might influence each other. Due to a limited space in this book chapter, the detailed description of each of the steps can be found only in the source code package, including a detailed description of the configuration parameters. In the subsequent experiments we run all algorithms with the configurations optimized to the BANCA database.

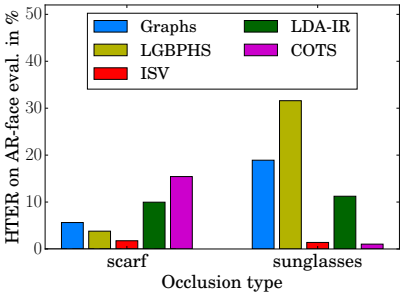
3.4 Face Variations

In this section, we test the optimized face recognition algorithms against several variations that influence recognition. We now also integrate the LDA-IR and the COTS algorithms into our experiments.

(a) Examples images of AR face: template, `illumination`, `occlusion`, `both`



(b) Effect of illumination and occlusion



(c) Effect of different occlusion types

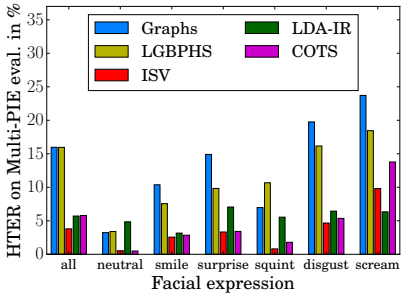
Fig. 6: PARTIAL OCCLUSIONS. *This figure shows examples of illumination and occlusion, and the effect of partial occlusions of the face on the different face recognition algorithms.*

One aspect of automatic face recognition in mobile environments is the partial occlusion of faces. Two prominent occlusions are scarfs covering the lower part of faces in winter and sunglasses as they are worn during summer. Example images of these occlusions can be found in fig. 6(a). One database that provides images with exactly these two types of occlusions is the AR face database, i. e., in the protocols `occlusion` and `both`. Fig. 6(b) contain the results of the occlusion experiments. As a baseline for this database we selected the protocol `illumination`,[15] on which all algorithms perform nicely. We only observed slight problems of LDA-IR, either with strong illumination or with occluded faces in the training set. When occlusions come into play, the Gabor wavelet based algorithms and the COTS suffer a severe drop in performance, while ISV results remain stable and LDA-IR results seem to be less affected by occlusion than by illumination. Having a closer look by separating between the two occlusion types (cf. fig. 6(c)), scarfs and sunglasses seem to have different impacts. While people wearing a scarf that covers approximately half of the face can still reasonably well be recognized, sunglasses completely break down the Graphs and LGBPHS systems. Interestingly, the COTS results show exactly the opposite behavior, whereas ISV and LDA-IR can handle both types of occlusions similarly well. In [74] it was found that the eye region contains most discriminative information. Our results approve these findings for some face recognition algorithms, but we clearly show that they cannot be generalized to all of them.

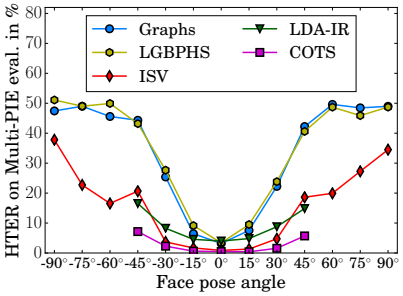
[15] To be comparable to the `occlusion` and `both` protocols, the same training set, i.e., including occluded faces, was also used in the `illumination` protocol.

(a) Examples of facial expressions: neutral, smile, surprise, squint, disgust, scream

(b) Examples of face poses from left to right profile



(c) Facial expressions



(d) Face poses

Fig. 7: FACIAL EXPRESSIONS AND POSES. *This figure shows the examples and the effect of facial expressions and face pose on the different face recognition algorithms.*

Another aspect that an automatic face recognition system must deal with is facial expression. To test the algorithms against various facial expressions, we selected the protocol `E` of the Multi-PIE database, which includes images with strongly pronounced expressions (see fig. 7(a)). The results of the experiments are shown in fig. 7(c). Interestingly, it can be observed that facial expressions are not handled satisfactorily by most algorithms. While neutral faces are recognized quite well by all algorithms, other expressions influence most of the algorithms severely. One exception is ISV, which seems to be stable against mild facial expressions and is still very good in presence of extreme expressions like surprise and disgust. Facial expressions are also handled well by LDA-IR, it is able to outperform ISV on screaming faces. Again, variations in the mouth region (as in the scream expression) perturb COTS more than variations in the eyes region.

Note that these two aspects of face recognition were tested in [75], where it was shown that faces with facial expressions or occlusions (in the `accessories` protocol of [75]) were more difficult to identify by all the algorithms they tested. However, we are not aware of any scientific publication, where a detailed analysis of types of facial expressions or occlusions was performed.

To test how the algorithms perform on non-frontal images, we execute them on protocol `P` of the Multi-PIE database. Similar to all other protocols we evaluate in this paper, the template enrollment is done using frontal images, while now probe images are taken from left profile to right profile in steps of 15° (see fig. 7(b) for

examples). The hand-labeled eye positions are used for the image alignment step, as long as both eyes are visible in the image, i. e., for images with a rotation less or equal to $\pm 45^\circ$. In the profile and near-profile cases, images are aligned according to the eye and mouth positions. In fig. 7(d) verification performance is plotted for each of the tested poses independently, though the algorithms are trained using images from all poses together. It can be observed that close-to-frontal poses up to $\pm 15^\circ$ can be handled by most algorithms, the performance order of the algorithms is similar to what we obtained before. For rotations greater than $\pm 45^\circ$, the verification performance of the algorithms that do not make use of the training data, i. e., LGBPHS and Graphs is around chance level. The algorithms that can handle rotations between $\pm 30^\circ$ and $\pm 60^\circ$ better are ISV, LDA-IR and COTS. Anyways, none of the tested algorithms can be used to identify profile faces, i. e., with rotations larger than $\pm 60^\circ$. Unfortunately, we could run LDA-IR and COTS experiments only on near-frontal faces since we could not provide the eye and mouth positions, which are required for profile image alignment, to the LDA-IR or COTS algorithms. For the same reason, the results of LDA-IR in fig. 7(d) are advantageously biased because the training set does not contain any profile images, i. e., with a rotation greater than $\pm 45^\circ$.

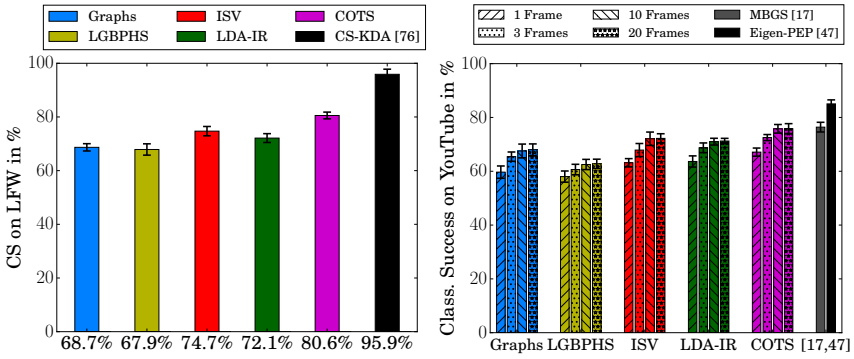
3.5 Unconstrained Image and Video Databases

Now, we evaluate the face recognition algorithms on more challenging unconstrained facial image and video database, i. e., LFW and YouTube, using the 10-fold evaluation protocols proposed by both databases. Each fold is evaluated separately, which includes a separate training for ISV and LDA-IR for each fold, and a separate decision threshold (cf. sec. 3.2.1) which is computed for the development set that we have defined for each fold.

Fig. 8(b) displays the average classification rates as well as the standard deviations over the 10 different folds of the LFW protocol [16]. Of the tested algorithms, the commercial COTS system was able to outperform all open source algorithms by a relatively high margin. With 74.7% classification success ISV is the best performing open source algorithm on this database, followed by LDA-IR. Also Graphs and LGBPHS perform almost as well, though they do not make use of the training data. However, none of the algorithms is able to reach the best performance [76] reported on the LFW website, which is given in the last column of fig. 8(b). Reasons are that our algorithms are not adapted to LFW, no external training data is used, no algorithm fusion is applied, we use a tight crop of the face (cf. [77]) and, finally, our decision threshold is computed on an independent development set for each fold, which makes our results completely unbiased, but which is not enforced by the LFW protocol.

One way to improve face recognition algorithms is to exploit video information as soon as it is available. To see, whether selecting more frames improves verification, we choose 1, 3, 10 and 20 frames from the videos of the YouTube faces database and feed them to our face recognition systems, which are tuned to work

(a) Examples of LFW



(b) LFW **(c)** YouTube

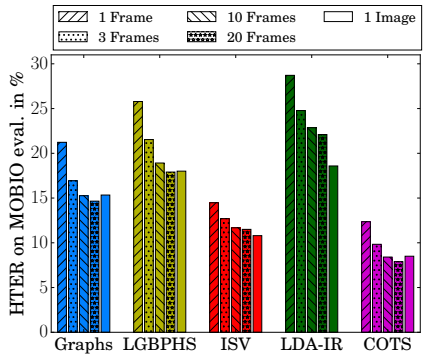
Fig. 8: LFW AND YOUTUBE. *This figure shows the average classification success and the standard deviation over the 10 folds for the experiments on the LFW and YouTube databases.*

with several images for template enrollment and for probing. These frames are taken such that they are distributed equally over the whole video sequence, and no further frame selection strategy is applied. Since there are no hand-labeled eye annotations available, we perform a face detection based on boosted LBP features [52] and a landmark localization algorithm [53] to detect the eye landmarks automatically.

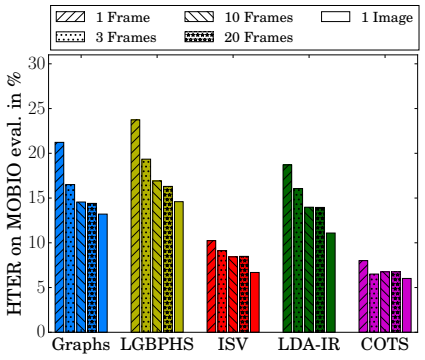
Fig. 8(c) shows the results of the experiments for the five evaluated algorithms. Apparently, increasing the number of frames also increases the recognition accuracy, though results settle after approximately 10 frames per video. Since the YouTube database contains several non-frontal face video recordings, and COTS has shown to be quite stable against those variations, it comes with no surprise that COTS performed best in our experiments. Of the tested open source systems, once more ISV is able to outperform the other three, but only slightly. Particularly, LDA-IR is able to compete. The most drastic improvement was gained by Graphs (+8.3%), where a strategy to incorporate several frames based on a local maximum is used, ISV (+9%), where the probability of the joint distributions of features from several frames are modeled, and COTS (+8.7%), which seems to provide a proper enrollment strategy. With the simple averaging strategy of LGBPHS (+4.8%), we are not able to exploit many frames that well, and the maximum score strategy of LDA-IR (+7.6%) lies in-between.

For the YouTube database, we also provide the best performing systems from [17] and [47], both of which exploit all frames of all videos. The first is taken using the Matched Background Similarity (MBGS), where samples from a cohort set are exploited and the computation of a discriminative classifier is required for each template and for each probe video. The second reported algorithm uses the Probabilistic

(a) Examples of MOBIO images



(b) `female`



(c) `male`

Fig. 9: MOBIO. *This figure displays examples of images in the MOBIO database and the results of the experiments for the two protocols* `female` *and* `male`*, with varying numbers of frames of the videos, and using the hand-labeled images.*

Elastic Part (PEP) algorithm, which is claimed to be robust against pose by modeling a GMM on SIFT features, reducing the dimensionality of the features using PCA and using a Bayesian classifier [78] for scoring. Though none of our algorithms can reach these baselines, we need to point out that: First, we do not include any cohort information into the classification process. Second, we exploit only up to 20 frames, not the whole video. Third, the image cropping used in the recognition experiments in [17] included more information (such as hair, headdresses or clothes), which has been shown to be able to help recognizing people [79], whereas our face cropping solely focuses on the inner facial area. Fourth, the Eigen-PEP algorithm is directly developed to solve video-to-video face recognition, whereas our algorithms were mainly developed for still image comparison. Finally, the configuration parameters for MBGS and Eigen-PEP were optimized for the YouTube database, while our algorithms were used with a configuration that was not adapted to YouTube.

3.6 Mobile Image and Video Database

The mobile database that we use in our experiments is the MOBIO database [15]. Though MOBIO was taken with hand-held mobile devices, the faces are usually in high resolution, mostly in a close-to-frontal pose, and degradation caused by motion blur is limited. However, illumination conditions in the videos are very diverse, and due to the fact that identities were talking during the recordings, a variety of

facial expressions is embedded in the frames. For the readers to get a picture of the variability of the MOBIO database, some of the images of one identity are shown in fig. 9(a).

As before, we choose 1, 3, 10 and 20 frames from the video sequences to see the impact on the face recognition algorithms. For each frame, the face is detected and the eye positions are localized automatically. Fig. 9 shows the HTER computed on the evaluation set of the MOBIO database for both protocols `female` and `male`. As before, the COTS results outperform all other algorithms on both protocols, followed by ISV, Graphs and LGBPHS. The LDA-IR results on `female` are the worst, while for `male` LDA-IR ranges third. Apparently, incorporating the information from several frames improves the recognition accuracy, drastically for Graphs, LGBPHS and LDA-IR, and moderately for ISV and COTS. Keeping in mind that each template is enrolled from 5 recordings, ISV and COTS already perform well using a single frame per video, while the other three algorithms gain more by exploiting several frames. All in all, when using 20 frames per video, in total features from 100 frames are incorporated in one enrolled template.

From fig. 9 can be observed that females are more difficult to verify than males, particularly for ISV and LDA-IR. This finding complies with other face verification experiments performed on this database [13, 35]. This might be due to the fact that the MOBIO training set (as well as the development and evaluation sets) has a bias towards males. While for Graphs, which does not depend on the training set, similar results for both males and females are generated, both ISV and LDA-IR follow the bias of the database and perform better on males than on females.

As the MOBIO database also provides images with hand-labeled eye coordinates, we can directly compare the impact of properly located eye positions against an off-the-shelf face detector [52] and landmark localization algorithm [53]. The results of the hand-labeled images from the MOBIO database are given in the last columns of each plot in fig. 9. Apparently, using hand-labeled eye positions rather than automatically detected faces works best for all algorithms. Even when exploiting 20 frames of a video, the results cannot reach the verification accuracy of a single hand-labeled image per video, except for Graphs, LGBPHS and COTS on the `female` protocol. There are several possible reasons for this. First, some faces in the MOBIO database are not completely contained in the image and, thus, the face detector usually returns a bounding box that is smaller than the actual face. Sometimes, due to strong illumination conditions, no face is found at all, and the extracted region contains image background. Second, the landmark localization might not be perfect, which is known to drop recognition accuracy [80]. And third, the hand-labeled images were selected such that the faces are mostly frontal with a neutral expression, while no such selection is done in the video frames.

4 Discussion

4.1 Algorithm Complexity

After executing all these experiments and showing the verification performances of the algorithms under several conditions and for various image and video databases, we want to discuss other properties of the algorithms.

4.1.1 Algorithm Execution Time

To be usable in real-world applications, the algorithms should be able to run in a reasonable amount of time. The execution times of all tested algorithms were measured in a test run on the protocol `P` of the BANCA database. Particularly, the training for the feature extraction, the computation of the projection matrix and the training of the enrollment is executed using 300 training files, while feature extraction and projection are performed on 6020 images. During enrollment, 52 templates are generated, each using the features of 5 images. Finally, 5460 scores are computed in the scoring step. In any case, we do not take into account the time for accessing the data on hard disk, but we only measure the real execution time of the algorithms. Hence, the actual processing time might increase due to hard disk or network latencies.

In tab. 2(a) it can be observed that the execution time of the algorithms differ substantially. For the simple color-based algorithm LDA-IR, which is based on a pure Python implementation, the training of the projection matrix finished after a couple of seconds, while the feature projection takes most of the time, here around four minutes. Enrollment is almost instantaneous since it just needs to store all features, and the scoring is also very fast. The extraction of Gabor graphs takes a little bit more time, while the enrollment of the templates is, again, instantaneous. The scoring is longer since computing the similarity measure requires a higher computational effort. The LGBPHS feature extraction needs a huge amount of time as the features themselves are huge and, hence, we chose a compressed format to store the histograms. This decreases the size of the LGBPHS feature vector (though tab. 2(b) shows that LGBPHS features still are longest), but complicates the feature extraction and the template enrollment, and also the scoring time is affected. The longest training and projection time is needed by ISV. During training, the distribution of the mixture of Gaussians and the linear subspace of the ISV algorithm are estimated — both procedures rely on computationally intensive iterative processes. Furthermore, the long projection time can be explained by its complexity, where sufficient statistics of the samples given the Gaussian mixture model are first computed, before being used to estimate session offsets. Finally, the scoring time is comparably short since most of the time consuming estimations are cached in the projection and enrollment steps.

Algorithm	Graphs	LGBPHS	ISV	LDA-IR	COTS
Training	—	—	1.8 h	6.8 s	—
Extraction	2.0 m	4.1 h	4.6 m	—	23.5 m
Projection	—	—	3.5 h	4.3 m	—
Enrollment	4.5 s	1.8 m	38.6 s	0.9 s	1.4 s
Scoring	39.4 s	25.5 s	7.5 s	6.1 s	11.5 s
total	2.7 m	4.2 h	5.5 h	4.6 m	23.7 m

(a) Execution time

Algorithm	Graphs	LGBPHS	ISV	LDA-IR	COTS
Model	—	—	29 MB	6.6 MB	???
Feature	160 kB	≈3 MB	1.4 MB	3.9 kB	4.5 kB
Projected	—	—	800 kB	—	—
Template	800 kB	≈9 MB	300 kB	12 kB	22.5 kB

(b) Memory requirements

Table 2: TIME AND MEMORY PROPERTIES. *This table gives an overview of the execution time that specific parts of the algorithms need and the size of the produced elements on hard disk. The times are measured on a 3.4 GHz Intel i7 processor with 16 GB of RAM, executing experiments on both development and evaluation set of the BANCA database.*

4.1.2 Memory Requirements

Tab. 2(b) displays the memory requirements of the objects produced during the execution of the algorithms. Except for LDA-IR and COTS, all elements are stored in double precision, i. e., with 8 bytes for each number. Depending on the complexity of the algorithms, the size of the features and templates differ slightly. In any case, the trained model needs to be stored to be able to use these technologies in a real word application, which might be problematic, e. g., on mobile devices with limited memory.

The lowest memory consumption is achieved by the LDA-IR algorithm, except that it needs to load the trained model once. Please note that these values are estimates since the format, which is stored, is unknown.[16] The size of the features and templates of COTS is clearly optimized, and a binary format is used to store them. However, there is no detailed information about the trained model of COTS. The size of the Gabor graphs is also relatively small, though the enrolled templates enlarges since all 5 feature vectors are stored. For LGBPHS, the feature and template sizes are much higher. Please note that the sizes of the LGBPHS feature vectors and enrolled templates differ slightly because we use a compressed format to store the histograms. Still, feature vectors and templates of this size make it difficult to use

[16] We just use the `pickle` module of Python to store the LDA-IR data. Tab. 2(b) shows the resulting file size on disk.

this algorithm in a real world application, at least with the configuration that we optimized in sec. 3.3. Finally, the size of the ISV projection matrix and the projected features are comparably high, while the enrolled template is relatively small. This is an advantage over having large templates since in a face recognition application, usually many templates are stored, but only few probe images need to be processed at a time.

4.2 About this Evaluation

Of course, an evaluative survey of face recognition algorithms as we provide in this book chapter cannot cover the full range of all recently developed face recognition algorithms including all their variations, and we might have omitted some aspects of face recognition. We know that this book chapter does not answer the question: What is the best face recognition algorithm? Nonetheless, we hope to provide some insights about advantages and drawbacks of the algorithms that we tested and also some hints, which algorithms are well suited under different circumstances.

4.2.1 What we Missed

Though we could not test all state-of-the-art face recognition algorithms, we tried to find a good compromise, which algorithms to test and which to leave out, and we are sorry if we do not evaluate the algorithm of your choice. Also, we executed algorithms only like they are reported in literature. Theoretically, we could have tried ISV modeling of Gabor jets, LGBPHS features on color image, etc., the range of possible tests is unlimited.

One aspect of biometric recognition is score normalization using an image cohort. For example, *ZT-norm* [81] has been shown lately [13] to be very effective and able to improve face verification drastically. Also the fusion of several algorithms [51] outperforms single algorithms. In this work, we do not perform any score normalization, and no fusion system is studied.

For the image databases, we used hand-labeled eye locations to align the faces, particularly during the evaluation of different face variations in sec. 3.4. From the results of the experiments on the MOBIO database, we assume that fully-automatic face recognition algorithms produce different results, especially as faces might not be detected correctly in presence of expressions, occlusions or non-frontal pose.

For video face recognition, we used a simple approach to select the frames. We did not apply any quality measure of the images, e. g., assessing motion blur, focus or other quality degradations of videos that present challenges in mobile video face recognition. Also, no sequence-based approaches [82] were tested, which exploit different kind of information from video sequences than simple frames.

We tried to make the comparison of the face recognition systems as fair as possible. We optimized the configurations of most algorithms to a certain image database.

Only LDA-IR was optimized to another database [22] and we did not touch this configurations in our experiments. This biases the algorithms towards different image variations, but still we think we could show the trends of the algorithms. Also, the optimization was done in several steps using discrete sets of configuration parameters. A joint optimization strategy with continuous parameters could have resulted in a slightly better performance on BANCA.

We intentionally optimized the configurations on one database and kept them stable during all subsequent tests. Therefore, the results on the other databases are not optimal. Certainly, the optimization of the configuration parameters to the each evaluated database would have improved the performance, though it is not clear, how high the gain would have been.

4.2.2 What we Achieved

Nevertheless, the contribution of this book chapter is — to our best knowledge — unique. We perform the first reproducible and extensible evaluative survey of face recognition algorithms that is completely based on open source software, freely available tools and packages and no additional commercial software needs to be bought to run the experiments. All experiments can be rerun and all results (including the figures and tables from this book chapter) can be regenerated by other researchers, simply by invoking a short sequence of commands, which are documented in the software package.

Utilizing these commands ourselves, we executed several recent open-source face recognition algorithms, optimized their configurations and tested them on various image and video databases. Additionally, we included one commercial of-the-shelf face recognition algorithm into our investigations. To be able to reproduce the figures from this paper, we provide the score files obtained with this algorithm for download.[17] Our experiments showed the impact of different image variations like illumination, expression, pose and occlusion on those algorithms, and we reported the performance on the LFW and YouTube databases. Finally, we showed that running video face recognition in mobile devices need to be improved by using face detectors and facial feature localizers specialized for mobile environments.

Since the implementation of the evaluation protocols is time consuming and error prone, many researchers rely on results generated on small image databases using their own protocols, which makes their results incomparable to the results of other researchers [24, 28]. In the source code that we provide [20, 23] evaluation protocols for several publicly available image and video databases are already implemented, and changing the database or the protocol is as easy as changing one command line parameter. Additionally, the same software package also allows to prototype new ideas, test combinations of these ideas with existing code, run face recognition experiments and evaluate the results of these experiments. Since the evaluation is always executed identically, results are directly comparable, throughout.

[17] `http://www.idiap.ch/resource/biometric`

With this software package, we want to encourage researchers to run face recognition experiments in a comparable way. Using Python and the Python Package Index (PyPI) it is easily possible for researchers to provide their source code for interested people to regenerate their results. A nice side effect of publishing source code together with scientific paper lies in the fact [30] that papers with source code are cited on average 7 times more than papers without. The software package that we distribute with this book chapter is one example of how to provide source code and reproducible experiments to other researchers.

4.2.3 What we Found

We have tested four recent open source and one commercial face recognition algorithms on several image databases and with different image variations. In most of the tests we have found that:

1. ISV, the generative approach that models a face as the distribution of facial features, outperforms the other algorithms, sometimes by far. Unfortunately, quite a long time for the (offline) training and template enrollment, and also for the (online) feature extraction is needed by this algorithm.
2. Color information, as used by LDA-IR, can be very helpful, especially when the texture itself is degraded due to low resolution, difficult facial expressions, occlusions or pose. However, uncontrolled or strong illumination seems to have a strong effect on this algorithm.
3. Image preprocessing plays an important role, and the preferred preprocessing technique differs for each face recognition algorithm. Sometimes, the best preprocessing technique even changes from database to database. Interestingly, algorithms work with many image resolutions — as far as it exceeds a lower limit of approximately 16 pixels inter-eye-distance.
4. Images with strong or uncontrolled illumination conditions are handled better by algorithms using Gabor wavelets. Furthermore, a proper use of Gabor phases improves the performance of these algorithms. In this study, we used two methods that do not include any training. We assume that these methods can be improved by incorporating knowledge from the training set using machine learning techniques.
5. None of the algorithms is able to handle non-frontal pose, even if all poses have been available during training. The direct comparison of features from different poses seems not to be possible with the discriminative algorithms, and similar problems have been observed even in the generative approach. Hence, we believe that different kinds of methods need to be invented, e. g., [44, 83] showed promising approaches to the pose problem.
6. When multiple frames are available for template enrollment or probing, the ISV algorithm, which directly incorporates multiple images, and the Graphs algorithm, which used a local scoring strategy, are able to exploit these data better than the other algorithms that use only simple scoring strategies like computing the average histogram or maximum similarity. However, the extension of

image-based face recognition algorithms towards videos is inferior to algorithms particularly designed for video-to-video face recognition [47].
7. Face detection and facial landmark localization in video sequences play important roles in video face recognition. Particularly for mobile devices, face detectors need to be able to stably detect faces that are only partially visible in the frames.
8. Besides few exceptions, the best results are obtained by the COTS algorithm. Apparently, the gap between academic research and the commercial application of face recognition algorithms still exists.

5 Conclusion

In this book chapter we presented the first evaluative, reproducible and extensible study of four recent open source and one commercial of-the-shelf face recognition algorithms. We briefly described the employed face recognition algorithms including several image preprocessing techniques. The implementations for most of the algorithms were taken from the open source software library Bob [20], while one algorithm stems from the Colorado State University toolkit [22].

The first evaluation that we performed assessed, which image resolution is required for the different algorithms to run properly. After selecting a proper image resolution, we evaluated the performance of the algorithms under several different image preprocessing technique on some image databases with difficult illumination and selected the most appropriate preprocessing for each face recognition algorithm. Subsequently, we optimized the configurations of most algorithms to the BANCA database, leaving the already optimized configuration of the CSU algorithm untouched. We tested the algorithm performance with regard to different image variations like facial expressions, partial occlusions and non-frontal poses. Then, we selected a challenging image and a challenging video database and ran the algorithms on them. Afterward, we examined the performance of the algorithms in the MOBIO database, using both the images with hand-labeled eye positions and the video sequences. Finally, we discussed a number of attributes of the algorithms that might limit their usability in mobile applications.

A short summary of the evaluation could be that there is not a single algorithm that works best in all cases and for all applications. Nevertheless, there are some favorites. Gabor wavelet based algorithms are well suited in difficult illumination conditions and were average in the other tests we performed. Still there is room for improvement of these algorithms since the ones we have tested in this work do not make use of the training set. The only algorithm in our test that used color information, i. e., LDA-IR works very well under several circumstances, especially when the image conditions are rather poor and algorithms cannot rely on facial features any more. The generative algorithm ISV performed best in most of the tests, but has the drawback of a very long execution time and high memory usage and cannot be used, e. g., in mobile devices with limited capacities and real-time

demands. Finally, the commercial algorithm worked best in most of our evaluations, particularly when face poses are non-frontal.

One important aspect of this evaluation is that we provide the source code for each of the experiments, including all image and video database interfaces, all preprocessing techniques, all feature extractors, all recognition algorithms and all evaluation scripts. Therefore, all experiments can be rerun and all figures can be recreated by anybody that has access to the raw image data. Additionally, we want to motivate other researchers to use our source code to run their own face recognition experiments since the software is designed to be easy to handle, easy to extend and to produce comparable results. We furthermore want to encourage researchers to publish the source code of their algorithms in order to build a strong community that can finally answer research questions that are still unsolved.

Acknowledgements This evaluation has received funding from the European Community's FP7 under grant agreements 238803 (BBfor2: `bbfor2.net`) and 284989 (BEAT: `beat-eu.org`). This work is based on open source software provided by the Idiap Research Institute and the Colorado State University. The authors want to thank all contributors of the software for their great work.

References

1. T. Kanade. *Picture Processing System by Computer Complex and Recognition of Human Faces*. PhD thesis, Kyoto University, Japan, 1973.
2. L. Sirovich and M. Kirby. Low-dimensional procedure for the characterization of human faces. *Journal of the Optical Society of America A*, 4(3):519–524, 1987.
3. H. A. Rowley, S. Baluja, and T. Kanade. Rotation invariant neural network-based face detection. In *CVPR*, pages 38–44. Springer, 1998.
4. P. Viola and M. Jones. Robust real-time object detection. *International Journal of Computer Vision*, 57(2):137–154, 2002.
5. G. Heusch, Y. Rodriguez, and S. Marcel. Local binary patterns as an image preprocessing for face authentication. In *FG*, pages 9–14. IEEE Computer Society, 2006.
6. X. Tan and B. Triggs. Enhanced local texture feature sets for face recognition under difficult lighting conditions. *Transactions on Image Processing*, 19(6):1635–1650, 2010.
7. C. Sanderson and K. K. Paliwal. Fast features for face authentication under illumination direction changes. *Pattern Recognition Letters*, 24(14):2409–2419, 2003.
8. M. Günther, D. Haufe, and R. P. Würtz. Face recognition with disparity corrected Gabor phase differences. In *ICANN*, volume 7552 of *LNCS*, pages 411–418. Springer, 2012.
9. B. Zhang, S. Shan, X. Chen, and W. Gao. Histogram of Gabor phase patterns (HGPP): A novel object representation approach for face recognition. *Transactions on Image Processing*, 16(1):57–68, 2007.
10. W. Zhao, A. Krishnaswamy, R. Chellappa, D. L. Swets, and J. Weng. Discriminant analysis of principal components for face recognition. In *Face Recognition: From Theory to Applications*, pages 73–85. Springer, 1998.
11. W. Gao, B. Cao, S. Shan, D. Zhou, X. Zhang, and D. Zhao. The CAS-PEAL large-scale Chinese face database and baseline evaluations. Technical report, Joint Research & Development Laboratory for Face Recognition, Chinese Academy of Sciences, 2004.
12. W. Zhang, S. Shan, W. Gao, X. Chen, and H. Zhang. Local Gabor binary pattern histogram sequence (LGBPHS): A novel non-statistical model for face representation and recognition. In *ICCV*, volume 1, pages 786–791. IEEE Computer Society, 2005.

13. R. Wallace, M. McLaren, C. McCool, and S. Marcel. Cross-pollination of normalization techniques from speaker to face authentication using Gaussian mixture models. *Transactions on Information Forensics and Security*, 7(2):553–562, 2012.
14. L. El Shafey, C. McCool, R. Wallace, and S. Marcel. A scalable formulation of probabilistic linear discriminant analysis: Applied to face recognition. *Transactions on Pattern Analysis and Machine Intelligence*, 35(7):1788–1794, 2013.
15. C. McCool et al. Bi-modal person recognition on a mobile phone: using mobile phone data. In *ICME Workshop on Hot Topics in Mobile Multimedia*, pages 635–640. IEEE Computer Society, 2012.
16. G. B. Huang, M. Ramesh, T. Berg, and E. G. Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical report, University of Massachusetts, Amherst, 2007.
17. L. Wolf, T. Hassner, and I. Maoz. Face recognition in unconstrained videos with matched background similarity. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2011.
18. R. Gross, I. Matthews, J. Cohn, T. Kanade, and S. Baker. Multi-PIE. *Image and Vision Computing*, 28(5):807–813, 2010.
19. A. Martínez and R. Benavente. The AR face database. Technical Report 24, Computer Vision Center, 1998.
20. A. Anjos, L. El Shafey, R. Wallace, M. Günther, C. McCool, and S. Marcel. Bob: a free signal processing and machine learning toolbox for researchers. In *ACM-MM*, pages 1449–1452. ACM press, 2012.
21. P. J. Phillips, J. R. Beveridge, B. A. Draper, G. Givens, A. J. O'Toole, D. S. Bolme, J. Dunlop, Y. M. Lui, H. Sahibzada, and S. Weimer. An introduction to the good, the bad, & the ugly face recognition challenge problem. In *FG*, pages 346–353. IEEE Computer Society, 2011.
22. Y. M. Lui, D. S. Bolme, P. J. Phillips, J. R. Beveridge, and B. A. Draper. Preliminary studies on the good, the bad, and the ugly face recognition challenge problem. In *CVPR Workshops*, pages 9–16. IEEE Computer Society, 2012.
23. M. Günther, R. Wallace, and S. Marcel. An open source framework for standardized comparisons of face recognition algorithms. In *ECCV. Workshops and Demonstrations*, volume 7585 of *LNCS*, pages 547–556. Springer, 2012.
24. X. Tan, S. Chen, Z.-H. Zhou, and F. Zhang. Face recognition from a single image per person: A survey. *Pattern Recognition*, 39:1725–1745, 2006.
25. Á. Serrano, I. Martín de Diego, C. Conde, and E. Cabello. Recent advances in face biometrics with Gabor wavelets: A review. *Pattern Recognition Letters*, 31(5):372–381, 2010.
26. D. Huang, C. Shan, M. Ardabilian, Y. Wang, and L. Chen. Local binary patterns and its application to facial image analysis : A survey. *Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 41(6):765–781, 2011.
27. R. Jafri and H. R. Arabnia. A survey of face recognition techniques. *Journal of Information Processing Systems*, 5(2):41–68, 2009.
28. L. Shen and L. Bai. A review on Gabor wavelets for face recognition. *Pattern Analysis and Applications*, 9(2):273–292, 2006.
29. P. Vandewalle, J. Kovacevic, and M. Vetterli. Reproducible research in signal processing - what, why, and how. *IEEE Signal Processing Magazine*, 26, 2009.
30. P. Vandewalle. Code sharing is associated with research impact in image processing. *Computing in Science and Engineering*, 14(4):42–47, 2012.
31. K. Ko. User's guide to NIST biometric image software (NBIS). Technical report, NIST Interagency/Internal Report (NISTIR) - 7392, January 2007.
32. J. C. Klontz, B. F. Klare, S. Klum, A. K. Jain, and M. J. Burge. Open source biometric recognition. In *IEEE International Conference on Biometrics: Theory, Applications and Systems (BTAS)*, pages 1–8, Sept 2013.
33. K. Messer, J. Matas, J. Kittler, J. Luettin, and G. Maitre. XM2VTSDB: The extended M2VTS database. In *AVBPA*, pages 72–77. LNCS, 1999.
34. A. Martin, M. Przybocki, and J. P. Campbell. *The NIST Speaker Recognition Evaluation Program*, chapter 8. Springer, 2005.

35. M. Günther et al. The 2013 face recognition evaluation in mobile environment. In *The 6th IAPR International Conference on Biometrics*, June 2013.
36. E. Khoury et al. The 2013 speaker recognition evaluation in mobile environment. In *The 6th IAPR International Conference on Biometrics*, June 2013.
37. D. Bansé, G. R. Doddington, D. Garcia-Romero, J. J. Godfrey, C. S. Greenberg, A. F. Martin, A. McCree, M. Przybocki, and D. A. Reynolds. Summary and initial results of the 2013-2014 speaker recognition i-vector machine learning challenge. In *Fifteenth Annual Conference of the International Speech Communication Association*, 2014.
38. A. J. O'Toole, P. J. Phillips, F. Jiang, J. Ayyad, N. Penard, and H. Abdi. Face recognition algorithms surpass humans matching faces over changes in illumination. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29:1642–1646, 2007.
39. A. M. Burton, S. Wilson, M. Cowan, and V. Bruce. Face recognition in poor-quality video: Evidence from security surveillance. *Psychological Science*, 10(3):243248, 1999.
40. M. Grgic, K. Delac, and S. Grgic. SCface–surveillance cameras face database. *Multimedia Tools and Applications*, 51(3):863–879, 2011.
41. J. R. Beveridge, P. J. Phillips, D. S. Bolme, B. A. Draper, G .H. Givens, Y. M. Lui, M. N. Teli, H. Zhang, W. T. Scruggs, K. W. Bowyer, P. J. Flynn, and S. Cheng. The challenge of face recognition from digital point-and-shoot cameras. In *Biometrics: Theory, Applications and Systems (BTAS), 2013 IEEE Sixth International Conference on*, pages 1–8, 2013.
42. Y. Taigman, M. Yang, M.'A. Ranzato, and L. Wolf. DeepFace: Closing the gap to human-level performance in face verification. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.
43. Y. Sun, X. Wang, and X. Tang. Deeply learned face representations are sparse, selective, and robust. *CoRR*, 2014.
44. C. Lu and X. Tang. Learning the face prior for Bayesian face recognition. In *Computer Vision ECCV 2014*, volume 8692 of *Lecture Notes in Computer Science*. Springer International Publishing, 2014.
45. B. Moghaddam, W. Wahid, and A. Pentland. Beyond eigenfaces: Probabilistic matching for face recognition. In *FG*, pages 30–35. IEEE Computer Society, 1998.
46. J. R Beveridge, H. Zhang, Y. Flynn, P. J.and Lee, V. E. Liong, J. Lu, M. de Assis Angeloni, T. de Freitas Pereira, H. Li, Hua G., V. Struc, J. Krizaj, and P. J. Phillips. The IJCB 2014 PaSC video face and person recognition competition. In *IEEE International Joint Conference on Biometrics IJCB*, pages 1–8, 2014.
47. H. Li, G. Hua, X. Shen, Z. Lin, and J. Brandt. Eigen-PEP for video face recognition. In *Asian Conference on Computer Vision (ACCV)*, 2014.
48. D. Cox and N. Pinto. Beyond simple features: A large-scale feature search approach to unconstrained face recognition. In *Automatic Face Gesture Recognition and Workshops (FG 2011), 2011 IEEE International Conference on*, pages 8–15, March 2011.
49. J. Ruiz-del Solar, R. Verschae, and M. Correa. Recognition of faces in unconstrained environments: A comparative study. *EURASIP Journal on Advances in Signal Processing*, 2009(1), 2009.
50. C. McCool, R. Wallace, M. McLaren, L. El Shafey, and S. Marcel. Session variability modelling for face authentication. *IET Biometrics*, 2(3):117–129, 2013.
51. E. Khoury, M. Günther, L. El Shafey, and S. Marcel. On the improvements of uni-modal and bi-modal fusions of speaker and face recognition for mobile biometrics. In *Biometric Technologies in Forensic Science*, October 2013.
52. C. Atanasoaei. *Multivariate Boosting with Look-up Tables for Face Processing*. PhD thesis, EPFL, 2012.
53. M. Uřičář, V. Franc, and V. Hlaváč. Detector of facial landmarks learned by the structured output SVM. In G. Csurka and J. Braz, editors, *VISAPP '12: Proceedings of the 7th International Conference on Computer Vision Theory and Applications*, volume 1, pages 547–556. SciTePress, 2012.
54. K. Ramírez-Gutiérrez, D. Cruz-Pérez, and H. Pérez-Meana. Face recognition and verification using histogram equalization. In *ACS*, pages 85–89. WSEAS, 2010.

55. H. Wang, S. Z. Li, and Y. Wang. Face recognition under varying lighting conditions using self quotient image. In *FG*, pages 819–824. IEEE Computer Society, 2004.
56. L. Wiskott, J.-M. Fellous, N. Krüger, and C. van der Malsburg. Face recognition by elastic bunch graph matching. *Transactions on Pattern Analysis and Machine Intelligence*, 19:775–779, 1997.
57. M. Günther. *Statistical Gabor Graph Based Techniques for the Detection, Recognition, Classification, and Visualization of Human Faces*. PhD thesis, Institut für Neuroinformatik, Technische Universität Ilmenau, Germany, 2011.
58. D. González Jiménez, M. Bicego, J. W. H. Tangelder, B. A. M. Schouten, O. O. Ambekar, J. Alba-Castro, E. Grosso, and M. Tistarelli. Distance measures for Gabor jets-based face authentication: A comparative evaluation. In *ICB*, pages 474–483. Springer, 2007.
59. W. Zhang, S. Shan, L. Qing, X. Chen, and W. Gao. Are Gabor phases really useless for face recognition? *Pattern Analysis & Applications*, 12:301–307, 2009.
60. T. Ojala, M. Pietikäinen, and D. Harwood. A comparative study of texture measures with classification based on featured distributions. *Pattern Recognition*, 29(1):51–59, 1996.
61. T. Ojala, M. Pietikäinen, and T. Mäenpää. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *Transactions on Pattern Analysis and Machine Intelligence*, 24(7):971–987, 2002.
62. T. Ahonen, A. Hadid, and M. Pietikainen. Face recognition with local binary patterns. In *ECCV*, pages 469–481. Springer, 2004.
63. F. Cardinaux, C. Sanderson, and S. Marcel. Comparison of MLP and GMM classifiers for face verification on XM2VTS. In *AVBPA*, volume 2688 of *LNCS*, pages 911–920. Springer, 2003.
64. D. A. Reynolds, T. F. Quatieri, and R. B. Dunn. Speaker verification using adapted Gaussian mixture models. *Digital Signal Processing*, 10(1-3):19–41, 2000.
65. J.-L. Gauvain and C.-H. Lee. Maximum a posteriori estimation for multivariate Gaussian mixture observations of Markov chains. *Transactions on Speech and Audio Processing*, 2(2):291–298, 1994.
66. F. Cardinaux, C. Sanderson, and S. Bengio. User authentication via adapted statistical models of face images. *Transactions on Signal Processing*, 54(1):361–373, 2006.
67. R. J. Vogt and S. Sridharan. Explicit modelling of session variability for speaker verification. *Computer Speech & Language*, 22(1):17–38, 2008.
68. R. Wallace, M. McLaren, C. McCool, and S. Marcel. Inter-session variability modelling and joint factor analysis for face authentication. In *IJCB*, pages 1–8. IEEE, 2011.
69. A. K. Jain, P. Flynn, and A. A. Ross. *Handbook of Biometrics*. Springer, 2008.
70. E. Bailly-Bailliére et al. The BANCA database and evaluation protocol. In *AVBPA*, volume 2688 of *LNCS*, pages 625–638. SPIE, 2003.
71. G. B. Huang, V. Jain, and E. G. Learned-Miller. Unsupervised joint alignment of complex images. In *ICCV*, pages 1–8. IEEE, 2007.
72. M. Guillaumin, J. Verbeek, and C. Schmid. Is that you? Metric learning approaches for face identification. In *ICCV*, pages 498–505. IEEE, 2009.
73. E. Khoury, L. El Shafey, C. McCool, M. Günther, and S. Marcel. Bi-modal biometric authentication on mobile phones in challenging conditions. *Image and Vision Computing*, 2013.
74. O. Ocegueda, S. K. Shah, and I. A. Kakadiaris. Which parts of the face give out your identity? In *CVPR*, pages 641–648. IEEE Computer Society, 2011.
75. W. Gao, B. Cao, S. Shan, X. Chen, D. Zhou, X. Zhang, and D. Zhao. The CAS-PEAL large-scale Chinese face database and baseline evaluations. *Systems, Man and Cybernetics, Part A: Systems and Humans*, 38:149–161, 2008.
76. S. R. Arashloo and J. Kittler. Class-specific kernel fusion of multiple descriptors for face verification using multiscale binarised statistical image features. *IEEE Transactions on Information Forensics and Security*, 9(12):2100–2109, 2014.
77. N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar. Attribute and simile classifiers for face verification. In *IEEE International Conference on Computer Vision (ICCV)*, Oct 2009.

78. D. Chen, X. Cao, L. Wang, F. Wen, and J. Sun. Bayesian face revisited: A joint formulation. In *Proceedings of the 12th European Conference on Computer Vision - Volume Part III*, pages 566–579, 2012.
79. E. Khoury, C. Senac, and P. Joly. Face-and-clothing based people clustering in video content. In *Proceedings of the International Conference on Multimedia Information Retrieval*, MIR '10, pages 295–304, New York, NY, USA, 2010. ACM.
80. A. Dutta, M. Günther, L. El Shafey, S. Marcel, R. Veldhuis, and L. Spreeuwers. Impact of eye detection error on face recognition performance. *IET Biometrics*, 2014.
81. R. Auckenthaler, M. Carey, and H. Lloyd-Thomas. Score normalization for text-independent speaker verification systems. *Digital Signal Processing*, 10(1):42–54, 2000.
82. J. R. Barr, K. W. Bowyer, P. J. Flynn, and S. Biswas. Face recognition from video: a review. *International Journal of Pattern Recognition and Artificial Intelligence*, 26(5), 2012.
83. M. K. Müller, M. Tremer, C. Bodenstein, and R. P. Würtz. Learning invariant face recognition from examples. *Neural Networks*, 41:137–146, 2013.