UNIVERSITÀ DEGLI STUDI DI PADOVA

Dipartimento di ingegneria dell'informazione
Corso di Laurea in Ingegneria dell'Automazione

IMPERIAL COLLEGE OF LONDON

Department of Computing

Imperial College London

Dynamical PLDA for face recognition in videos

Laureando:

Alessandro Fabris

Relatori:

Prof. Ruggero Carli

Prof. Stefanos Zafeiriou

A.A. 2014/2015

Padova, 06/10/2015

Abstract

Face recognition is a thriving area of computer vision, with applications spanning the areas of home entertainment, information security, law enforcement and surveillance. In this context, PLDA is a well established generative model that guarantees good discriminability between different individuals and great flexibility, hence being suitable for many different tasks. Within this document we provide a brief primer about face recognition in videos, outlining challenges, applications, different approaches and peculiarities with respect to the domain of still images. We then recall PLDA models, along with their founding ideas and the mathematics necessary to exploit them. Given these premises, we present the first dynamical generalization of PLDA (DPLDA), closely related to Kalman filters, which naturally apply to video sequences. We anticipate a performance improvement for DPLDA in comparison with classical PLDA, providing a theoretical justification in the area of face recognition in videos. Subsequently, we present a robust recognition pipeline, suitable for identity inference in videos under uncontrolled pose, illumination, expression, occlusion and duration. We apply this pipeline to challenging recognition tasks for in-the-wild databases, showing good performance with low- and medium-quality videos of less than 3 seconds. Furthermore, we compare PLDA and DPLDA within this pipeline, showing how DPLDA systematically outperforms PLDA. Finally, we outline future research directions to build on this work and further improve recognition performance.

Contents

1. Introduction

Individual recognition is a fundamental ability for the members of an organized society. The possibility for humans to distinguish between different people is a self-evident prerequisite for the establishment of meaningful interpersonal relationships, which we reasonably overlook and take for granted. At a more or less subconscious level, the human brain has employed face recognition and biometrics since mankind itself exists, most likely with satisfactory results for everyday life applications. However, as human communities grew, purely natural skills became less suitable for recognition tasks, as their scope and difficulty grew proportionally to populations.

Human-to-human recognition is particularly faulty on large-scale applications, wherein a systematic approach is usually beneficial. Especially during the 19th century, cities grew larger and their population changed quicker than ever thanks to migration flows from the countryside and more accessible forms of transportation. Due to these changes, the location-specific knowledge, law enforcement officers relied on to keep crime under control, lost part of its effectiveness. At the same time the institution of recidivism was introduced in most jurisdictional European systems as an a aggravating factor for law violations. These facts concurred in determining the need to establish a systematic way of recording law infringement in conjunction with the identity traits of the offender.

In the 1870s, pioneering French criminologist officer Alphonse Bertillon established a number of combined biometrics techniques − later named "Bertillonage" − that categorized criminals based on available pictures, descriptions and body measurements from a sliding compass. Furthermore he was the first to develop the two-part "mug-shot", consisting of a frontal and a profile picture against a measuring background, still widely used today. Two decades later, in the 1890s, Argentine police official Juan Vucetich successfully introduced fingerprints as a means of identification, which quickly became an effective technique employed all over the world. In addition, in 1936, ophthalmologist Franck Burch proposed to use iris patterns as a means to identify individuals.

It wasn't until the emergence of computers however, that the first automatic face recognition system was deployed by Woodrow Bledsoe and colleagues in 1964. Research and development was mainly funded by governmental agencies since applications such as home entertainment were not in sight yet. A fundamental breakthrough took place in 1987 when Sirovich and Kirby [26] applied Principal Component Analysis to face recognition and showed that less than 100 features properly extracted from a face image can be enough to retain most of its information content, thus opening the way for scalable applications to come in the consequent years.

Nowadays biometrics is ubiquitous and relies on both physiological traits, such as voice timbre and rhythm, and behavioural features such as gait and typing patterns. Face appearance is a fundamental part of biometrics and possibly the one with most applications, spanning the areas of home entertainment, information security, law enforcement and surveillance. This technology is now mature after several decades of research, thus

achieving good performance in terms of recognition accuracy and speed. At the same time it has the inherent advantage of not requiring the collaboration of the individual to be identified − as opposed to fingerprints or iris pattern acquisition − guaranteeing a good trade off between performance and obtrusiveness. Nevertheless permanence is a critical factor, in that face appearance is bound to change over time, thus requiring a continuous and expensive update of databases. Research about face ageing has been carried out (see [23] for a good review), however its impact on the performance of face recognition systems is still widely unknown. For these reasons, face identification is regarded as a mature technology, suitable for many applications, yet still widely researched and susceptible to improvements.

Broadly speaking, face recognition can be divided into 3 categories, depending on the face data available: algorithms based on still image data, those that exploit video sequences with temporal information and those that rely on more elaborate and higher-dimensional information, possibly comprehensive of infrared-imagery and depth maps. The rest of this chapter provides a brief introduction to face recognition in video sequences, highlighting peculiarities, challenges, approaches and applications. For a complete and fairly recent survey about the topic see [3].

1.1 Applications

Up to this point, we have introduced identity inference tasks in generic terms, using the words recognition and identification more or less interchangeably. In this section we briefly characterize four different applications, highlighting their differences.

Face **verification** is a task of authentication whereby a subject claims to be someone. Hence, given the individual's probe video, it has to be compared with a gallery video of the subject he or she claims to be, to determine whether they depict the same person. Within each verification algorithm, a proximity/distance measure is defined to quantify the similarity between two videos. The claims about identity are rejected or accepted according to a threshold value learned for said similarity metrics. We incur in a false positive if a false claim is accepted and a false negative whenever a true claim is rejected. The false accept rate (FAR) is the percentage of false positives and the false reject rate (FRR) is the percentage of false negatives encountered while running an algorithm. Both FAR and FRR provide a measure of the quality of an algorithm. A typical performance metrics for a recognition system is its receiver operating characteristic (ROC), which plots the FAR against the FRR as the similarity threshold varies. More synthetically, the verification rate is often employed, defined as the ratio of correct decisions (true positives and true negatives) divided by the total number of decisions. This metrics is more synthetic, however it is only significant if the number of false and true claims to be verified are the same or at least comparable. An over-trusting verification system, programmed to always output a yes decision, in a world of completely honest people, would achieve a 100 % verification rate, although it should be noted that such a world doesn't exist and such a system is clearly useless.

Closed-set identification requires us to identify the person in a probe video by comparing it against gallery videos of different people and decide for the most similar one. We are guaranteed *a priori* that the subject in the probe video is depicted in at least one gallery video. Performance is commonly measured through the correct identification rate, counting the number of correct identifications divided by their total amount. Unlike verification, identification becomes inherently more difficult as the number of people in the gallery increases.

Open-set identification differs from closed-set identification in that we are not guaranteed that the probe subject appears in the gallery, making it a harder task. The correct identification rate can once more be defined as the number of correct decisions divided by the number of total decisions. This metric may vary significantly for the same algorithm depending on the "openness" of the dataset, which relates to the percentage of probe videos that depict a subject external to the gallery.

Clustering is a problem defined on a collection of videos, which have to be grouped based on the people they depict. The number of people in the videos can be undefined or known *a priori*, the latter case being easier to decide. The correct outcome of a clustering procedure is a partition of the video collection by subject.

In this document we test our novel dynamical model on tasks of verification, closed-set and open-set identification, showing the versatility it has inherited from PLDA while systematically outperforming it.

1.2 Peculiarities and challenges

Inference about identification from still images and from videos share the same problematic factors such as:

- Pose variations: the same subject can be captured from different angles, hence correspondence between pixels and face features is not guaranteed across pictures.
- Illumination variations: pictures can be lit by sources of different magnitude and orientation.
- Expression variations: over 30 facial muscles are responsible for countless configurations (expressions), which alter the basic appearance of a human face.
- Occlusions: when viewing conditions are not controlled, obstacles can come between the subject and the sensor, covering the former during acquisition.
- Motion blur: depending on the exposure time and the relative speed of the subject and the sensor, face pictures may be fuzzy and hard to identify even for humans.

In tackling these nuisances, video-based algorithms can exploit some peculiarities of the video realm. For example, multiple frames are available and the information carried by each of them can be combined in different manners. Depending on the application and its properties, robustness to illumination variations or improved performance through 3D modeling can be achieved taking into account the frame set as a whole. Furthermore, video sequences contain temporal information about the proximity of frames in time and consecutive frames are expected to be more correlated than frames that are far apart in time. Research in face recognition from videos is often based on research about still image identification, exploiting the peculiar advantages of video sequences to increase performance. This thesis is in fact an example of said approach, factoring temporal dependencies into PLDA models, which will be introduced in the next chapter.

1.3 Approaches

The authors of [3] provide an interesting taxonomy for face recognition in videos which we borrow and synthesize in figure 1.1. The classification is determined by how the additional information from multiple frames is exploited. Two essentially different approaches are

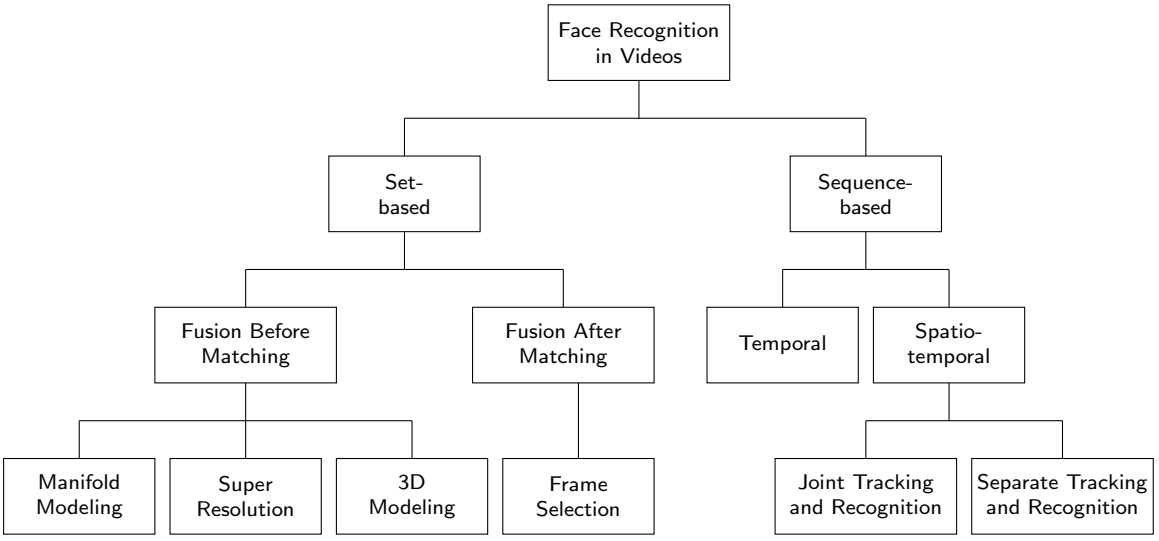

Figure 1.1: Taxonomy for face inference in videos. This figure shows when and how the information from multiple frames is exploited within a recognition pipeline. The procedure proposed in this thesis can be ascribed to Sequence-based, Spatio-temporal, Separate Tracking and Recognition.

set-based and sequence-based. **Set-based** techniques treat a video as a bag-of-images, completely ignoring temporal information. Sequence-based algorithms, on the other hand, exploit the ordering of the frames and their mutual position within a video to draw conclusions about the identity of the depicted subject.

Within the set-based approaches the information from different frames can be fused **before matching** or **after matching**. In the latter case, the matching procedure exploited is essentially one of face-recognition in images, generalized to videos by fusing the image-level results into the video-level domain, for instance through majority voting.

Frame selection allows us to pick a subset of all the video frames which is deemed sufficient to characterize the appearance of a subject. A key idea is that low quality frames and redundant frames can be discarded, improving recognition while compressing the available information. Also, gallery frames could be selected in order to span a wide range of illumination and pose conditions in view of uncontrolled and varying conditions in probe frames to be matched.

In [5], Berrani and Garcia employ robust PCA to select the frames that maximize a robust measure of variance within a low-dimensional base of eigenvectors onto which pixel values are projected. This criterion rules out frames that were captured under peculiar pose/illumination/expression, which lie too far from the others in the new space and are hence considered outliers.

Frame selection generalizes to frame weighting, wherein the importance of each frame in the decision process is determined by a continuous value. In [38] Zhang and Martinez propose a weighted selection scheme that exploits gaussian mixtures to find similarities within video frames, while assigning low weights to low-quality images that might be disruptive for the matching process.

Stallkamp et al. [27] devised a selection method that takes into account two different distance metrics and their combination for frame selection. The distance-to-model metric excludes probe frames whose distance from their closets match in the gallery is larger than a certain threshold, possibly due to its peculiar illumination and pose conditions.

The distance-to-second closest is a measure devised to systematically leave out ambiguous frames whose best and second-best matches are tied, thus extrinsically harder to decide between.

Another key idea for frame selection is to model and encourage intra-class diversity within the gallery, guaranteeing that the available frames within will be representative of an individual under most illumination/pose/expression configurations.

Clustering [13, 33, 34] is often exploited to group frames that share similar configurations. Matching can then be carried out as a subset-to-subset comparison or by selecting one significant exemplar from each cluster, for instance the cluster mean, which is regarded as significant of the whole subset. Exemplar-based approaches only require a limited amount of comparisons yielding a significant computational advantage while retaining most useful information.

In [13] Hadid and Pietikäinen employ a locality preserving LLE to reduce dimensionality while preserving similarity, prior to applying clustering. The cluster centers are regarded as a significant base for intra-class variations and taken into account for robust matching.

Another interesting approach consists of specifically encouraging the selection of high quality frames. Thomas et al. [28] introduce a "faceness" measure produced by the recognition software *FaceIt*. This measure is exploited for different strategies, by ordering video frames according to their faceness and selecting for instance the first N frames, or N evenly spaced frames from the first M with highest faceness.

In [36] Xiong and Jaines introduce an algorithm to systematically select the most suitable frames, based on their intrinsic and extrinsic properties. The intrinsic quality of a face is determined by its orientation, scale and illumination. The extrinsic quality is related to clustering so that a new frame is accepted if, in relation with the other frames previously picked, it achieves limited intra-class variance and increased inter-class distance.

Super-resolution methods tackle the problem of low image quality by incorporating information from multiple frames to reconstruct the high-frequency content lost during acquisition in challenging settings. The problem is typically formulated by modelling an image degradation process that has to be inverted in order to restore the high quality image:

$$\boldsymbol{y}(t) = \boldsymbol{M}(t)\boldsymbol{x}(t) + \boldsymbol{w}(t). \tag{1.1}$$

Here $\boldsymbol{y}(t)$ denotes the observed image sequence up to a time step t, $\boldsymbol{M}(t)$ models the corruption process due to motion blur and sensor limitation, $\boldsymbol{x}(t)$ represents the original, high quality sequence of frames to reconstruct, and $\boldsymbol{w}(t)$ accounts for the remaining sources of noise. The reconstruction is typically carried out by minimizing a cost function consisting of the square reconstruction error plus a regularization term enforcing smoothness and mitigating the influence of noise.

The iterative back-projection algorithm, originally applied to face recognition by Zhou and Bhanu [39], is a classical super-resolution technique, whereby edges are recovered and enhanced by subtracting from the original images a blurred version of themselves. Al-Azzeh et al. [1] applied this procedure in combination with an efficient frequency-based alignment procedure, and completed their recognition pipeline with PCA to test their algorithm in a more scalable setting.

The application of PCA to the recovered images offers no guarantee that the high quality previously recovered will be retained. Furthermore the inversion of matrix $\boldsymbol{M}(t)$ poses a significant computational burden in the pixel domain, whereas performing it in the PCA subspace results in lower computational requirements. For this reason, Gunturk et al. [12] transfer the problem from the pixel domain to the subspace spanned by the principal

components. In the low-dimensional domain, equation (1.1) is further complicated by an error term accounting for PCA reconstruction error.

It should be noted that the taxonomy in figure 1.1 is introduced for exposition convenience and the recognition schemes listed therein are not mutually exclusive. For instance, Jilela and Ross [15] combine super-resolution with frame selection by ruling out the frames corrupted by motion blur. In order to do so, they devise an intra-frame motion parameter β which measures intensity differences between points that are aligned in two consecutive frames. If β is smaller than a certain threshold, no significant movement is assumed to occur between two consecutive frames, which are hence selected for the reconstruction of a high quality image, otherwise they are discarded.

3D Modeling specifically copes with the problem of varying pose and correspondences between pixels and features of the subjects' face. 3D models are often exploited to generate synthetic images of gallery and probe subjects under matching pose and illumination conditions. Park et al. [22], for example, build offline a 3D model for each subject in the gallery. They subsequently exploit an SVM to estimate pose and illumination for the subject in the probe, and accordingly rotate and illuminate each gallery subject in an artificial fashion. Finally a 3D-to-2D projection is carried out, effectively synthesizing videos for each gallery subject under pose and illumination matching the ones of the probe. Alternatively, a 3D model can be learnt from the probe video, which is then rendered under varying modelled conditions corresponding to videos from the gallery.

An alternative to synthesis-driven schemes are model-comparison techniques, whereby 3D models are learnt from both gallery and probe and directly compared. An example of this approach is provided by Liu and Chen [19], who model the human head as a 3D ellipsoid, upon which people's faces are back projected, resulting in a texture map. Said map is divided into deformable patches, each of which corresponds to a specific region of the head. PCA is applied within each patch separately. The distance between 2 face models is then given by the sum of the distances between each pair of patches and exploited for recognition decisions.

Manifold modelling techniques characterize the possible configurations of a human face without explicitly modelling generative processes. The manifold associated to person i can be expressed as

$$\boldsymbol{X}_i = \{c(\boldsymbol{x}_i)\ c \in C\}\,, \tag{1.2}$$

where C is the set of all possible configurations and $c(\boldsymbol{x}_i)$ denotes the appearance of individual i under configuration c. The union of all X_i yields the manifold of all faces.

Pioneering work was carried out in [37], where Yamaguchi et al. use PCA to represent faces of an individual as a linear subspace. The cosine of the principal angle between subspaces serves as a measure of similarity between faces, used in the decision process.

Other attempts were made in this direction, although linear subspaces struggle to characterize some inherent non-linearities such as illumination variations caused by light from multiple sources.

In [32] Wang et al. model face manifolds as a collection of linear patches, learnt with a method called Maximum Linear Patches. The number of patches is not decided in advance, rather it is determined adaptively. This technique iteratively constructs MLPs until the linearity constraint is broken. The mutual subspace angle is then used to quantify the distance between patches, which are then fused into a global distance metric for two faces.

Synthetic data can be used for manifold learning, as shown by Lina et al. in [18]. A global linear subspace is built through PCA. The available data is clustered according to pose angles and view-specific means and covariance matrices are learnt. First order and second order moments, super-imposed to the global linear subspace, characterize a

view-specific manifold. Artificial transformations such as blurring and slight rotations are used to learn more significant values for mean and covariance. The statistics for untrained poses are learnt through interpolation. Inference is performed based on the Mahalanobis distance between the learnt manifolds.

An interesting approach adopted by Arendjelović and Cipolla [2] is to consider face images as i.i.d. samples drawn from a low-dimensional PDF in the face space, transported by a mapping function onto a manifold in the image space. The authors resort to kernel PCA to unfold the face manifold, allowing them to model non-linearities while retaining computational efficiency. Finally a symmetrized version of the Kullback-Leibner distance (the Resistor-Average distance) is employed to perform inference about identity.

So far we have concentrated on set-based approaches, which combine the information from multiple frames disregarding their order. **Sequence-based** approaches, on the other hand, leverage the temporal information naturally available with videos. Dynamics is useful for face tracking purposes as well as for learning idiosyncratic variations peculiar to each individual. Note that both the notions of tracking and variation require a well defined time ordering of the frames in a video.

Spatio-temporal schemes combine static spatial information with dynamical cues in order to increase the performance of a recognition system. Algorithms ascribed to this category can be further divided, according to whether they perform joint face tracking and recognition or whether they split the two tasks.

Separate tracking and recognition offers the advantage of flexibility in that the tracking and the recognition algorithm can be devised independently of each other. The algorithm proposed in this thesis purely concentrates on the problem of face recognition, exploiting both temporal and spatial information in the process. The algorithm itself requires pre-aligned faces as an input, hence we integrate it into a recognition pipeline, wherein it is preceded by a step of independent face tracking and alignment. For this reason our recognition pipeline can be ascribed to this category of procedures.

A common technique to learn the statistics associated with an individual's face while comprising temporal information are Hidden Markov Models. Based on video frames, which are regarded as the output of an HMM $\Lambda_i = (\boldsymbol{A}_i, \boldsymbol{B}_i, \boldsymbol{\pi}_i)$, the model parameters are learnt through the Viterbi algorithm. A different model Λ_i is associated with each person appearing in the gallery. When a probe sequence $\mathcal{O}$ is available, it is ascribed to the identity i whose HMM Λ_i maximizes the probability of the sequence conditional on the learnt parameters $(\boldsymbol{A}_i, \boldsymbol{B}_i, \boldsymbol{\pi}_i)$. A significant example of this approach was devised by Liu and Cheng [20], who designed an adaptive HMM capable of online learning.

In [11] Gorodnichy resorts to an autoassociative neural network that accumulates information over time by updating its synapses. Based on physiological considerations, the author emphasizes the dynamical nature of human face recognition. Synaptic plasticity helps us to memorize from sequences of subsequent stimuli, rather than from isolated face frames. Analogously the author proposes a scheme to update a neural network's synaptic weights in the presence of frame sequences ordered in time.

Aggarwal et al. [10] propose the first recognition approach based on an autonomous ARMA model. Within their framework, observations correspond to video frames and hidden states are essentially related to pose. Analogously to [20], a separate model is learnt for every individual in the gallery. Finally, principal angles between learnt models are employed as a discrimination metrics.

Feature extraction is commonly carried out at a frame level, exploiting the temporal information in the subsequent steps. Hadid and Pietkäinen [14] introduce the Volume Local Binary Patterns (VLBP), to extract features from 3D spatio-temporal neighbourhoods

comprising neighbouring pixels from neighbouring frames. Recognition is performed by comparing feature histograms of VLBPs and subsequently exploiting an SVM classifier.

Joint tracking and recognition permits a bidirectional flow of information between tracker and recognition system, which can increase the overall performance if properly leveraged. The tracker is likely to return faces that comply with the appearance model employed by the recognition system. Also, information about the pose can be shared seamlessly between the two components of the identification pipeline.

Li et al. [17] propose a unified tracking and recognition scheme which models face appearance as a probabilistic manifold which is approximated by a collection of linear subspaces, each of which is associated with a different pose. The pose information is used by the recognition system to perform identity inference at a frame level. The frame-specific information about identity is fused at a video level with a bayesian approach. At the same time the same pose information from a frame is used for tracking in the next one, wherein the pose is modelled as a gaussian distribution supposed to be centered at its previous value. Connectivity of pose-dependent subspaces is thus modelled with transition matrices that enforce proximity. This scheme is interesting as recognition and tracking share the same appearance model, reducing the misalignments between tracker output and recognition input.

Temporal methods concentrate on the peculiar variational patterns of a face rather than its statical aspect or mean appearance. Temporal face cues cannot be disguised as easily as appearance since they are part of nearly subconscious mechanisms. At the same time, they have been shown to be fundamental for human perception in recognizing familiar faces and, more in general, in complementing statical information in the presence of low quality video sequences.

Chen at al. [7] propose a temporal method based on motion flow fields feature extraction. The motion flow field is a classical dynamical feature related to the temporal evolution of a depicted object. Videos of subjects uttering the same word are acquired in a constrained setting, properly synchronized and landmarked. An optical field is computed for every frame and concatenated in a high-dimensional representation of an individual. PCA and LDA are exploited as a dimension reduction tool yielding individual representations which are compared through their euclidean distance.

Matta and Dugelay [21] propose a system based on piece-wise head motion measurements. The displacement of an individual's eyes, nose and mouth over frame sequences is computed and normalized for robustness with respect to rototranslation and scaling. These features are employed to train individual GMMs which are used as a bayesian classifier.

Benedikt et al [4] concentrate on the temporal data from videos of people uttering the word "Puppy". Faces from different videos are aligned and cropped around the region of the lips. Further dimensionality reduction is achieved through PCA, and Weighted Dynamic Time Warping (WDTW) is employed as a pattern matching measure.

Temporal schemes have shown promising performance and good robustness to potentially deceptive face ornaments, however they remain a vastly unexplored domain less mature than spatio-temporal techniques. In this document we present a novel generative model (DPLDA) that can be ascribed to this latter paradigm.

- In chapter 2 we present a review of PLDA, which constitutes the static foundation of DPLDA, originally conceived for recognition in still images.

- The first part of chapter 3 presents a dynamical generalization of PLDA. The phases of learning and recognition are then described in detail, with the respective implementation details.

- In chapter 4 we introduce the remaining components of our recognition pipeline, and present the results it achieves on different databases for different recognition tasks.
- Conclusions and further directions of research are outlined in chapter 5.

2. PLDA

2.1 A preliminary review of LDA

Linear discriminant analysis (LDA) [9] is the foundation for the generative models described throughout this document. LDA is a classical technique employed in pattern recognition to separate a set of observations into two or more classes. Similarly to PCA, LDA is a deterministic dimensionality reduction tool that specifically aims at retaining most useful information (i.e. the variance) in the original dataset. However, unlike PCA, it is specifically designed to take into account the different classes that training samples are ascribed to, and make sure the data is projected onto some linear subspace that guarantees good discriminability between the different classes.

Intuitively, we would like samples of the same class to look very similar - short distance in the projection subspace - and samples from different classes to be far apart - long distance in the projection subspace. This idea can be formalized by defining the within-class scatter matrix for C different classes

$$\boldsymbol{S}_W = \sum_{j=1}^{C} \frac{1}{N_{c_j}} \sum_{\boldsymbol{x}_i \in c_j} (\boldsymbol{x}_i - \boldsymbol{\mu}(c_j)) (\boldsymbol{x}_i - \boldsymbol{\mu}(c_j))^T , \tag{2.1}$$

where $\boldsymbol{x}_i$ are the observations and $\boldsymbol{\mu}(c_j)$, $j \in (1, C)$ are the centers for each class. $\boldsymbol{S}_W$ captures the variance for data with the same label, whereas the between-class scatter matrix, defined as

$$\boldsymbol{S_b} = \sum_{j=1}^{C} (\boldsymbol{\mu}(c_j) - \boldsymbol{\mu}) (\boldsymbol{\mu}(c_j) - \boldsymbol{\mu})^T , \tag{2.2}$$

measures the overall distance between classes. We then look for the projection matrix $\boldsymbol{W}$ that solves the optimization problem

$$\max \text{Tr}[\boldsymbol{W}^T \boldsymbol{S}_b \boldsymbol{W}] \text{ s.t. } \boldsymbol{W}^T \boldsymbol{S}_W \boldsymbol{W} = 1, \tag{2.3}$$

which yields a new representation for the data, $\boldsymbol{y}_i = \boldsymbol{W}\boldsymbol{x}_i$, featuring both good discrimination power and low dimensionality.

2.2 PLDA: a static model for inference about identity

Probabilistic LDA (PLDA), as detailed in [16], represents a probabilistic, generative framework for inference about identity in still images. Let us assume we have a training set consisting of J images per individual, with a total of I different individuals. Then, according to PLDA, the generative model behind the j-th image of the i-th individual is

$$\boldsymbol{x}_{ij} = \boldsymbol{\mu} + \boldsymbol{F}\boldsymbol{h}_i + \boldsymbol{G}\boldsymbol{w}_{ij} + \boldsymbol{\epsilon}_{ij}. \tag{2.4}$$

Figure 2.1: Generative process. A final image D) can be decomposed into a generic mean face $\boldsymbol{\mu}$ (A), a component from the identity subspace $\boldsymbol{\mu} + \boldsymbol{F}\boldsymbol{h}_i$ (B) and a component from the pose/illumination/expression subspace $\boldsymbol{\mu} + \boldsymbol{G}\boldsymbol{w}_{ij}$ (C).

Here $\boldsymbol{x}_{ij}$ is the image itself, represented by a vector of pixel values. According to the generative model, it depends on $\boldsymbol{h}_i$, which is the latent variable specific to the i-th person, and on $\boldsymbol{w}_{ij}$ which is the latent variable associated with the setting in which the picture was taken. Both $\boldsymbol{h}_i$ and $\boldsymbol{w}_{ij}$ are unobserved random variables. Intuitively, $\boldsymbol{h}_i$ captures the face features of a person that consistently determine his/her look, whereas $\boldsymbol{w}_{ij}$ represents incidental conditions such as pose, illumination and expression that influenced the picture at the moment it was taken.

More explicitly, we can look at the model from a generative point of view, according to which an image is created by taking a realization of random variables $\boldsymbol{h}_i$ and $\boldsymbol{w}_{ij}$, projecting them according to matrices $\boldsymbol{F}$ and $\boldsymbol{G}$, adding the mean vector and a sample from the noise process. The adoption of said model is not motivated by a belief that pictures of individuals are generated in strict compliance to this equation, rather by a combination of simplicity and good discriminatory power. In figure 2.1 we depict a breakdown of a face image into its generative components. Note that the shade in the bottom-right corner is correctly ascribed to $\boldsymbol{G}\boldsymbol{w}_{ij}$ as an incidental illumination condition.

In more rigorous terms, $\boldsymbol{F}$ is a factor matrix whose columns span the between-individual subspace. Each individual is assumed to have a unique position in said subspace, which sets it apart from everyone else and is represented by the hidden variable $\boldsymbol{h}_i$. Analogously, $\boldsymbol{G}$ is a matrix whose columns span the within-individual subspace. $\boldsymbol{w}_{ij}$ represents the position of image $\boldsymbol{x}_{ij}$ therein and is responsible for the differences in photos of the same individual. Vector $\boldsymbol{\mu}$ is simply the mean of all images, which is computed initially for detrending purposes and $\boldsymbol{\epsilon}_{ij}$ is stochastic noise that justifies the observations' distance from the hypothesized model. Said noise is supposed to be a random variable with zero mean and (anisotropic) diagonal covariance matrix. This feature is very important for the discriminative power of the model, allowing it to learn which pixels carry more significant information for the purposes of identity inference. Figure 2.2 portrays different positions in the above-mentioned subspaces, together with the mean face $\boldsymbol{\mu}$ and a sample of $\boldsymbol{\epsilon}_{ij}$.

The prior and conditional distributions associated to model (2.4) are defined as follows:

$$Pr(\boldsymbol{x}_{ij}|\boldsymbol{h}_i, \boldsymbol{w}_{ij}, \theta) = \mathcal{N}(\boldsymbol{x}_{ij}|\boldsymbol{\mu} + \boldsymbol{F}\boldsymbol{h}_i + \boldsymbol{G}\boldsymbol{w}_{ij}, \boldsymbol{\Sigma}), \tag{2.5}$$

$$Pr(\boldsymbol{h}_i) = \mathcal{N}(\boldsymbol{h}_i|\boldsymbol{0}, \boldsymbol{I}), \tag{2.6}$$

$$Pr(\boldsymbol{w}_{ij}) = \mathcal{N}(\boldsymbol{w}_{ij}|\boldsymbol{0}, \boldsymbol{I}), \tag{2.7}$$

where $\mathcal{N}(\boldsymbol{x}|\boldsymbol{a}, \boldsymbol{B})$ indicates a gaussian PDF on random vector $\boldsymbol{x}$ with mean $\boldsymbol{a}$ and variance matrix $\boldsymbol{B}$. Furthermore the set of all the parameters governing the models is defined as $\boldsymbol{\theta} = (\boldsymbol{\mu}, \boldsymbol{F}, \boldsymbol{G}, \boldsymbol{\Sigma})$. The adoption of gaussian PDFs, in conjunction with the linear equation

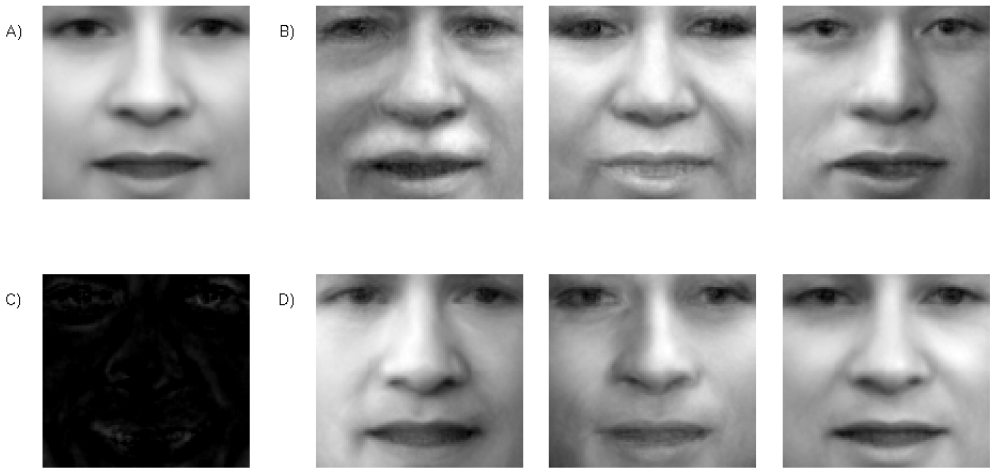

Figure 2.2: Different directions in both subspaces. Some directions in the between-individual subspace (B), capturing the features of specific individuals. Some directions in the within-individual subspace (D), capturing differences in expression and illumination. Overall mean face (A). A sample from noise process $\boldsymbol{\epsilon}_{ij}$ (C)

(2.4), determine nice closed-form solutions for Bayes law computations, which are at the core of generative models.

The term $\boldsymbol{\mu} + \boldsymbol{F}\boldsymbol{h}_i$ accounts for between-individual variance, whereas $\boldsymbol{G}\boldsymbol{w}_{ij}$ is responsible for within-individual variance. From this perspective it is possible to understand the claim that "the relationship between PLDA and standard LDA is similar to that between factor analysis and PCA", formulated by the authors of [16] without any further explanation.

PCA poses the problem of finding a projection matrix from a high-dimensional space to a smaller one, that, given a series of high-dimensional observations, allows us to compress them while retaining the most possible variance. Factor analysis, conversely, is based on a probabilistic framework featuring a low-dimensional (unobserved) latent variable, assumed to be gaussian with zero mean, in analogy with PLDA. Given a set of observations, the procedure aims at finding the projection matrix from the low-dimensional space of the hidden variables to the high-dimensional space of the observations, that maximizes the likelihood of the observed variables under this model. It has been shown in [29] that, as the variance of the latent variable in factor analysis tends to zero, the optimal projection matrix converges to the transpose of the PCA solution, thus giving a precise relationship between the two frameworks. PCA is extremely effective for tasks of unsupervised learning, but it fails to take into account the labels (identities) that come with our training set.

When the goal is to classify data into different classes, LDA is a handy dimensionality reduction tool. In particular, LDA maximizes between-class variance while minimizing within-class variance by finding a suitable projection matrix from the high-dimensional observation space onto a smaller one. In other words, data from the same class is clustered together and kept distant from other classes, thus favouring discrimination. The concepts of between-class and within-class variance entail a notion of labels (classes) that LDA naturally takes into account. PLDA, on the other hand, resembles factor analysis in that it is based on a mapping from a low-dimensional latent space to the high-dimensional data space. Training a PLDA model can be seen as finding the 2 factor matrices $\boldsymbol{F}$ and

$\boldsymbol{G}$ that fit the observations optimally as a projection from the smaller subspace to which the couple $(\boldsymbol{h}_i, \boldsymbol{w}_{ij})$ belongs. Furthermore, just like LDA, PLDA clusters together images $\boldsymbol{x}_{\bar{i}j}$ from the same class $\bar{i}$ by taking into account the respective latent variables $(\boldsymbol{h}_{\bar{i}}, \boldsymbol{w}_{\bar{i}j})$. This clustering is achieved through the hard constraint that $\boldsymbol{h}_{\bar{i}}$ should be the same across all pictures of person $\bar{i}$, thus restricting images of the same individual to vary less in the subspace spanned by the columns of $\boldsymbol{F}$ and $\boldsymbol{G}$. Regardless of this loose explanation, to the best of our knowledge, no precise mathematical characterization relates LDA with PLDA. We end this section with a list of four assumptions that enclose the essence of PLDA:

- **Face images depend on several interacting factors**, such as identity, pose, expression, illumination.
- **Image generation is noisy**, due to both sensor noise and model imprecision.
- **Identity cannot be known exactly:** embracing a bayesian point of view, no attempt will be made to estimate identity variables. However:
- **Recognition tasks do not require identity estimates:** we can decide whether two images share the same identity without explicitly estimating said identity.

The discussion carried out so far has expanded on the two top assumptions, whereas to better understand the two bottom ones we need to delve deeper into the recognition technique associated to PLDA, which is the subject of the next section.

2.3 Recognition

Suppose, for the moment, that a set of parameters $\boldsymbol{\theta} = (\boldsymbol{\mu}, \boldsymbol{F}, \boldsymbol{G}, \boldsymbol{\Sigma})$ has been learnt according to some optimality criterion. Recognition tasks typically require us to determine which images depict the same person, by exploiting our knowledge about how the pictures were created. Within our framework, recognition implies estimating which images $\boldsymbol{x}_1 \dots \boldsymbol{x}_k$ were generated by the same identity latent variable $\boldsymbol{h}$ in a model governed by parameter set $\boldsymbol{\theta}$. To avoid notational ambiguity, let us define the *process* $\mathcal{P}$ behind a set of images $(\boldsymbol{x}_1 \dots \boldsymbol{x}_N)$ as the ground truth of the people $(\boldsymbol{h}_1 \dots \boldsymbol{h}_N)$ depicted in those pictures. When presented with a task of recognition, or inference about identity, we are given a set of images, with partial knowledge of the process that generated them (the labels of the training set) and we're asked to determine the missing information about said process (the labels of the test set). Note the distinction between our definitions of process and model. The process actually contributes to the generation of the data set, the model is an artificial description we adopt to characterize how this contribution takes place.

Since PLDA is a probabilistic generative model, a natural criterion for recognition is given by Maximum a posteriori (MAP) estimation. Given a set of images $\boldsymbol{x}_{1\dots N}$, we want to find the process $\mathcal{P}^*$ that best explains the observation:

$$\mathcal{P}^* = \arg\max_{\mathcal{P}} Pr(\mathcal{P}|\boldsymbol{x}_{1\dots N}), \tag{2.8}$$

where

$$Pr(\mathcal{P}|\boldsymbol{x}_{1\dots N}) \;\propto\; Pr(\boldsymbol{x}_{1\dots N}|\mathcal{P})P(\mathcal{P}). \tag{2.9}$$

In this regard, the effort lays in quantifying the likelihoods $Pr(\boldsymbol{x}_{1\dots N}|\mathcal{P})$, whereas, for the process priors $P(\mathcal{P})$, uniform probabilities are assumed.

Recognition can be performed for slightly different tasks, among which closed set face identification, open set identification, clustering and verification, as described in section

1.1. One of the advantages of PLDA is that it allows us to tackle all these problems within the same framework. Let us consider, for example, an open set face identification problem, where N images $\boldsymbol{x}_{1,\dots N}$ of N different people represent the gallery and we need to determine whether a probe image $\boldsymbol{x}_p$ shares its identity variable $\boldsymbol{h}$ with one of them. In order to do this, we will compute and compare $N+1$ different likelihoods, conditional on all the possible processes that may have generated the observation.

$$Pr(\boldsymbol{x}_{1,\dots,N,p}|\mathcal{P}_0) = \prod_{n=1}^{N} Pr(\boldsymbol{x}_n)Pr(\boldsymbol{x}_p), \tag{2.10}$$

$$Pr(\boldsymbol{x}_{1,\dots,N,p}|\mathcal{P}_i) = \prod_{n=1,n\neq i}^{N} Pr(\boldsymbol{x}_n)Pr(\boldsymbol{x}_p,\boldsymbol{x}_i), \;\; i = 1\dots N \tag{2.11}$$

Here $\mathcal{P}_0$ denotes the process where the person in the probe image does not appear in the gallery whereas hypotheses $\mathcal{P}_i$ assume that images $(\boldsymbol{x}_p, \boldsymbol{x}_i)$ depict the same person. Terms $Pr(\boldsymbol{x}_n)$ and $Pr(\boldsymbol{x}_p)$ in equation (2.10) represent the likelihood of a single image and can be evaluated according to

$$Pr(\boldsymbol{x}_n) = \iint Pr(\boldsymbol{x}_n, \boldsymbol{h}_n, \boldsymbol{w}_n) d\boldsymbol{h}_n d\boldsymbol{w}_n, \tag{2.12}$$

$$Pr(\boldsymbol{x}_p) = \iint Pr(\boldsymbol{x}_n, \boldsymbol{h}_n, \boldsymbol{w}_n) d\boldsymbol{h}_p d\boldsymbol{w}_p. \tag{2.13}$$

The computation is carried out by comprising latent variables into a joint probability and then marginalizing them out with a bayesian approach. Note this is in perfect agreement with the third and fourth assumptions listed at the end of the previous section. In fact we estimate a PDF for $\boldsymbol{h}$ but we're not interested in the single most plausible value.
$Pr(\boldsymbol{x}_p, \boldsymbol{x}_i)$ represents the likelihood of images $\boldsymbol{x}_p$ and $\boldsymbol{x}_i$ under the hypothesis that they share the same identity hidden variable:

$$Pr(\boldsymbol{x}_p, \boldsymbol{x}_i) = \iiint Pr(\boldsymbol{x}_p, \boldsymbol{x}_i, \boldsymbol{h}_i, \boldsymbol{w}_p, \boldsymbol{w}_i) d\boldsymbol{h}_i d\boldsymbol{w}_p d\boldsymbol{w}_i. \tag{2.14}$$

The constraint of sharing the same identity is expressed by integrating just along $\boldsymbol{h}_i$, implicitly assuming $\boldsymbol{h}_p$ to have the same value. Note, once more, that we integrate along the whole range of possible values of $\boldsymbol{h}_i$, without performing a specific estimate.

Equations (2.12), (2.13), (2.14) can be computed by solving the more general problem of evaluating the likelihood of N images $\boldsymbol{x}_{1\dots N}$ sharing the same identity variable $\boldsymbol{h}$, as explained below. Combining N generative equations we have

$$\begin{bmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_2 \\ \vdots \\ \boldsymbol{x}_N \end{bmatrix} = \begin{bmatrix} \boldsymbol{\mu} \\ \boldsymbol{\mu} \\ \vdots \\ \boldsymbol{\mu} \end{bmatrix} + \begin{bmatrix} \boldsymbol{F} & \boldsymbol{G} & \boldsymbol{0} & \cdots & \boldsymbol{0} \\ \boldsymbol{F} & \boldsymbol{0} & \boldsymbol{G} & \cdots & \boldsymbol{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{F} & \boldsymbol{0} & \boldsymbol{0} & \cdots & \boldsymbol{G} \end{bmatrix} \begin{bmatrix} \boldsymbol{h} \\ \boldsymbol{w}_1 \\ \vdots \\ \boldsymbol{w}_N \end{bmatrix} + \begin{bmatrix} \boldsymbol{\epsilon}_1 \\ \boldsymbol{\epsilon}_2 \\ \vdots \\ \boldsymbol{\epsilon}_N \end{bmatrix}, \tag{2.15}$$

which can be written synthetically as

$$\boldsymbol{x}' = \boldsymbol{\mu}' + \boldsymbol{D}\boldsymbol{y} + \boldsymbol{\epsilon}'. \tag{2.16}$$

The random vectors $\boldsymbol{x}'$ and $\boldsymbol{y}$ are distributed according to the following prior and conditional PDFs:

$$Pr(\boldsymbol{y}) = \mathcal{N}(\boldsymbol{y}|\boldsymbol{0}, \boldsymbol{I}), \tag{2.17}$$

$$Pr(\boldsymbol{x}'|\boldsymbol{y}) = \mathcal{N}(\boldsymbol{x}'|\boldsymbol{\mu}' + \boldsymbol{D}\boldsymbol{y}, \boldsymbol{\Sigma}'), \tag{2.18}$$

where

$$\boldsymbol{\Sigma}' = \begin{bmatrix} \boldsymbol{\Sigma} & \boldsymbol{0} & \cdots & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{\Sigma} & \cdots & \boldsymbol{0} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \boldsymbol{0} & \cdots & \boldsymbol{\Sigma} \end{bmatrix}. \tag{2.19}$$

Note that model (2.16) has the form of a factor analyzer. From the properties of conditional gaussian distributions it's easy to show that

$$Pr(\boldsymbol{x}_{1\ldots N}) = Pr(\boldsymbol{x}') = \mathcal{N}(\boldsymbol{x}'|\boldsymbol{\mu}', \boldsymbol{\Sigma}' + \boldsymbol{D}\boldsymbol{D}^T). \tag{2.20}$$

Having identified this probability distribution, it's now easy to compare a probe image against all the gallery images, evaluate (2.12), (2.13), (2.14) which in turn allow us to compute (2.10) and (2.11) and combining them with the uniform priors over the processes, we can select the most plausible process by picking the option that maximizes the posterior.

2.4 Learning

We postponed to this section a treatment of how the model parameters $\boldsymbol{\theta} = (\boldsymbol{\mu}, \boldsymbol{F}, \boldsymbol{G}, \boldsymbol{\Sigma})$ can be evaluated in order to perform recognition. If we knew the hidden variables $\boldsymbol{h}_i$ and $\boldsymbol{w}_{ij}$, we could employ a maximum-likelihood framework to estimate a sensible value for $\boldsymbol{\theta}$. Likewise, if $\boldsymbol{\theta}$ was known, evaluating the hidden variables would be easy. When presented with a problem like this one, seemingly recursive and unsolvable, a typical solution is resorting to the Expectation-Maximization (EM) algorithm. This procedure allows us to find maximum likelihood solutions in models comprising latent variables, hence it perfectly applies to our case. A formal characterization of the EM algorithm and its properties can be found in appendix A. Here we limit our scope to an intuitive explanation of the steps and the reasoning behind this procedure.

Let us indicate with $\boldsymbol{X}$ the totality of the observations and with $\boldsymbol{Z}$ the underlying latent variables. Our goal is to find the parameter $\boldsymbol{\theta}$ that maximizes

$$p(\boldsymbol{X}|\boldsymbol{\theta}) = \sum_{\boldsymbol{Z}} p(\boldsymbol{X}, \boldsymbol{Z}|\boldsymbol{\theta}), \tag{2.21}$$

or equivalently its logarithm. It often happens that direct optimization on $p(\boldsymbol{X}|\boldsymbol{\theta})$ is a hard problem, whereas optimization on the joint distribution $p(\boldsymbol{X}, \boldsymbol{Z}|\boldsymbol{\theta})$ is considerably easier. To make this claim more tangible, we can relate it to PLDA: within our current framework, direct computation of $p(\boldsymbol{X}|\boldsymbol{\theta})$ implies evaluation of the inverse for covariance matrix in equation (2.20), which takes the form

$$\begin{aligned}(\boldsymbol{\Sigma}' + \boldsymbol{D}\boldsymbol{D}^T)^{-1} = &\begin{bmatrix} (\boldsymbol{\Sigma} + \boldsymbol{G}\boldsymbol{G}^T)^{-1} & & \\ & \ddots & \\ & & (\boldsymbol{\Sigma} + \boldsymbol{G}\boldsymbol{G}^T)^{-1} \end{bmatrix} + \\ &\begin{bmatrix} (\boldsymbol{\Sigma} + \boldsymbol{G}\boldsymbol{G}^T)^{-1}\boldsymbol{F} \\ \vdots \\ (\boldsymbol{\Sigma} + \boldsymbol{G}\boldsymbol{G}^T)^{-1}\boldsymbol{F} \end{bmatrix} (\boldsymbol{I} + J\boldsymbol{F}^T(\boldsymbol{\Sigma} + \boldsymbol{G}\boldsymbol{G}^T)^{-1}\boldsymbol{F}) \begin{bmatrix} (\boldsymbol{\Sigma} + \boldsymbol{G}\boldsymbol{G}^T)^{-1}\boldsymbol{F} \\ \vdots \\ (\boldsymbol{\Sigma} + \boldsymbol{G}\boldsymbol{G}^T)^{-1}\boldsymbol{F} \end{bmatrix}^T,\end{aligned} \tag{2.22}$$

followed by computation of the respective quadratic form and derivation with respect to $\boldsymbol{\theta}$. This is indicative of how hard direct optimization can be. On the other hand, evaluation

of $\log[p(\boldsymbol{X}, \boldsymbol{Z}|\boldsymbol{\theta})]$ is usually simpler, since it can be broken down into the prior and conditional $p(\boldsymbol{Z}|\boldsymbol{\theta})$, $P(\boldsymbol{X}|\boldsymbol{Z}, \boldsymbol{\theta})$, which can be separated additively thanks to the logarithm function. Also, here we see the first advantage of working with gaussian distributions, whose exponentials revert the logarithm, simplifying both out of the equation. To delve into the mathematical details for this example, see subsection 2.4.2.

Our efforts are hence devoted to maximizing the quantity $\log[p(\boldsymbol{X}, \boldsymbol{Z}|\boldsymbol{\theta})]$ with respect to $\boldsymbol{\theta}$. A fundamental problem in this regard is that $\boldsymbol{Z}$ is a latent variable and, as such, unknown. In the absence of certain, deterministic information about $\boldsymbol{Z}$, our next-best option is to evaluate its expectation and regard it as a reliable estimate for its real value. After this heuristic explanation, we can sketch the typical flow of the EM algorithm:

1. Initialize parameter $\boldsymbol{\theta}^{old}$.
2. **E step** - expectation: evaluate $E(\boldsymbol{Z}|\boldsymbol{X}, \boldsymbol{\theta}^{old})$
3. **M step** - maximization: update the value of $\boldsymbol{\theta}$ by computing

$$\boldsymbol{\theta}_{new} = \arg\max_{\boldsymbol{\theta}} Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{old}), \tag{2.23}$$

   where

$$Q(\boldsymbol{\theta}, \boldsymbol{\theta}^{old}) = E_{\boldsymbol{Z}|\boldsymbol{X}, \boldsymbol{\theta}^{old}}[\log(p(\boldsymbol{X}, \boldsymbol{Z}|\boldsymbol{\theta}))] \tag{2.24}$$

   is the expected value with respect to $\boldsymbol{Z}$ of the joint log-probability defined in (2.21), conditioned on the available information ($\boldsymbol{X}$, $\boldsymbol{\theta}^{old}$).
4. Check for convergence

The alternation of E step and M step guarantees to increase the likelihood function $p(\boldsymbol{X}|\boldsymbol{\theta})$ at every iteration (unless we reached a local maximum). A proof of this statement can be found in appendix A. In the next two subsections we provide a sketch of how the concepts we introduced apply to PLDA.

2.4.1 E step

Starting from a sequence of gallery images from the same person i, modeled according to equation

$$\begin{bmatrix} \boldsymbol{x}_{i1} \\ \boldsymbol{x}_{i2} \\ \vdots \\ \boldsymbol{x}_{iN} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\mu} \\ \boldsymbol{\mu} \\ \vdots \\ \boldsymbol{\mu} \end{bmatrix} + \begin{bmatrix} \boldsymbol{F} & \boldsymbol{G} & \boldsymbol{0} & \cdots & \boldsymbol{0} \\ \boldsymbol{F} & \boldsymbol{0} & \boldsymbol{G} & \cdots & \boldsymbol{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{F} & \boldsymbol{0} & \boldsymbol{0} & \cdots & \boldsymbol{G} \end{bmatrix} \begin{bmatrix} \boldsymbol{h}_i \\ \boldsymbol{w}_{i1} \\ \vdots \\ \boldsymbol{w}_{iN} \end{bmatrix} + \begin{bmatrix} \boldsymbol{\epsilon}_{i1} \\ \boldsymbol{\epsilon}_{i2} \\ \vdots \\ \boldsymbol{\epsilon}_{iN} \end{bmatrix}, \tag{2.25}$$

we want to estimate the probability distribution of the respective latent variables, namely the within-individual variable $\boldsymbol{w}_{ij}$, specific to each image, and the identity variable $\boldsymbol{h}_i$, which is shared across the sequence. In a more compact notation, our aim is finding the sufficient statistics for $\boldsymbol{z}_i$ from a model of the form

$$\boldsymbol{x}_i = \boldsymbol{\mu}' + \boldsymbol{D}\boldsymbol{z}_i + \boldsymbol{\epsilon}_i. \tag{2.26}$$

Exploiting the properties of gaussian distributions, it can be shown that

$$P(\boldsymbol{z}_i|\boldsymbol{x}_i) = \mathcal{N}\left(\boldsymbol{z}_i|\ (\boldsymbol{I} + \boldsymbol{D}^T\boldsymbol{\Sigma}'^{-1}\boldsymbol{D})^{-1}\boldsymbol{D}^T\boldsymbol{\Sigma}'^{-1}(\boldsymbol{x}_i - \boldsymbol{\mu}'), (\boldsymbol{I} + \boldsymbol{D}^T\boldsymbol{\Sigma}'^{-1}\boldsymbol{D})^{-1}\right), \tag{2.27}$$

with $\boldsymbol{\Sigma}'$ as defined in (2.19). As can be observed by looking ahead into equations (2.33) - (2.35), the sufficient statistics from this PDF are

$$E(\boldsymbol{z}_i|\boldsymbol{x}_i,\theta) = (\boldsymbol{I} + \boldsymbol{D}^T\Sigma'^{-1}\boldsymbol{D})^{-1}\boldsymbol{D}^T\Sigma'^{-1}(\boldsymbol{x}_i - \boldsymbol{\mu}'), \tag{2.28}$$

$$E(\boldsymbol{z}_i\boldsymbol{z}_i^T|\boldsymbol{x}_i,\boldsymbol{\theta}) = (\boldsymbol{I} + \boldsymbol{D}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{D})^{-1} + E(\boldsymbol{z}_i|\boldsymbol{x}_i,\boldsymbol{\theta})E(\boldsymbol{z}_i|\boldsymbol{x}_i,\boldsymbol{\theta})^T, \tag{2.29}$$

where we made explicit the dependency of the above values on the model parameters $\boldsymbol{\theta}$. In order to compute the distribution of the latent variables $\boldsymbol{z}_i$ associated to every person, the above calculations should be carried out for $i = 1, \ldots, I$.

2.4.2 M step

In the M-step, we aim to update the model parameters $\boldsymbol{\theta} = (\boldsymbol{\mu}, \boldsymbol{F}, \boldsymbol{G}, \boldsymbol{\Sigma})$ in order to maximize the joint likelihood of observations and latent variables; however not knowing the exact value for the latent variables, we will maximize the expected value of the said joint likelihood, leveraging the results from the E-step. Consider the following model for a single image:

$$\boldsymbol{x}_{ij} = \boldsymbol{\mu} + \begin{bmatrix}\boldsymbol{F} & \boldsymbol{G}\end{bmatrix}\begin{bmatrix}\boldsymbol{h}_i \\ \boldsymbol{w}_{ij}\end{bmatrix} + \boldsymbol{\epsilon}_{ij}, \tag{2.30}$$

which can be rewritten as

$$\boldsymbol{x}_{ij} = \boldsymbol{\mu} + \boldsymbol{B}\boldsymbol{z}_{ij} + \boldsymbol{\epsilon}_{ij}. \tag{2.31}$$

Our objective is finding

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} Q\left(\boldsymbol{\theta}, \boldsymbol{\theta}^{old}\right), \tag{2.32}$$

where

$$\begin{aligned}
Q\left(\boldsymbol{\theta}, \boldsymbol{\theta}^{old}\right) &= E_{\boldsymbol{Z}|\boldsymbol{\theta}^{old},\boldsymbol{X}}\left\{\ln Pr(\boldsymbol{X}, \boldsymbol{Z}|\theta)\right\} \\
&= E_{\boldsymbol{Z}|\boldsymbol{\theta}^{old},\boldsymbol{X}}\left\{\ln \prod_{i=1}^{I}\prod_{j=1}^{J} Pr(\boldsymbol{x}_{ij}|\boldsymbol{z}_{ij})Pr(\boldsymbol{z}_{ij})\right\} \\
&= -E_{\boldsymbol{Z}|\boldsymbol{\theta}^{old},\boldsymbol{X}}\left\{\sum_{i=1}^{I}\sum_{j=1}^{J}\left[\frac{1}{2}\ln|\boldsymbol{\Sigma}| + \frac{1}{2}(\boldsymbol{x}_{ij} - \boldsymbol{\mu} - \boldsymbol{B}\boldsymbol{z}_{ij})^T\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}_{ij} - \boldsymbol{\mu} - \boldsymbol{B}\boldsymbol{z}_{ij})\right.\right. \\
&\qquad \left.\left. + \frac{D_h + Dw}{2}\ln 2\pi + \frac{1}{2}\boldsymbol{z}_{ij}^T\boldsymbol{z}_{ij} + \frac{f}{2}\ln 2\pi\right]\right\}
\end{aligned}$$

Here we denote as D_w and D_h the dimensionality of the latent spaces and as f the size of the observations. Note, as anticipated, how the logarithm function in conjunction with gaussian PDFs render the computations simple to handle, allowing for nice closed form solutions. Indeed, by setting to zero the derivatives of Q with respect to $\boldsymbol{\theta}$, we have:

$$\boldsymbol{\mu} = \frac{1}{IJ}\sum_{i=1}^{I}\sum_{j=1}^{J}\boldsymbol{x}_{ij} \tag{2.33}$$

$$\boldsymbol{B} = \left(\sum_{i=1}^{I}\sum_{j=1}^{J}(\boldsymbol{x}_{ij} - \boldsymbol{\mu})E[\boldsymbol{z}_{ij}^T]\right)\left(\sum_{i=1}^{I}\sum_{j=1}^{J}E[\boldsymbol{z}_{ij}\boldsymbol{z}_{ij}^T]\right)^{-1} \tag{2.34}$$

$$\boldsymbol{\Sigma} = \frac{1}{IJ}\sum_{i=1}^{I}\sum_{j=1}^{J}\operatorname{diag}\left\{(\boldsymbol{x}_{ij} - \boldsymbol{\mu})(\boldsymbol{x}_{ij} - \boldsymbol{\mu})^T - 2\boldsymbol{B}E[\boldsymbol{z}_{ij}](\boldsymbol{x}_{ij} - \boldsymbol{\mu})^T + \boldsymbol{B}E[\boldsymbol{z}_{ij}\boldsymbol{z}_{ij}^T]\boldsymbol{B}^T\right\} \tag{2.35}$$

These values are then re-plugged into equations (2.28), (2.29) to update the sufficient statistics, which in turn will allow to update $\boldsymbol{\theta}$, carrying on with this cycle until an appropriate convergence criterion is met.

3. Dynamical PLDA

3.1 PLDA for videos

PLDA has been used for face recognition in still images for a wide variety of databases and different settings. When it comes to applications of inference from videos, it can still be employed as a set-to-set (probe video to gallery video) matching tool that compares all the frames of a probe video against all the frames of a gallery video and determines the overall most likely identity through maximum voting or some similar global metrics. The set-based paradigm allows us to mitigate the effect of noise and non-optimal viewing conditions by smoothing these anomalies across all the available frames. In [33] and [34] this idea is employed by a preliminary pose-dependent clustering and a subsequent matching of same-cluster frames. Different global metrics are explored and compared for the final matching scores.

This approach, despite achieving good results, ignores the temporal information that goes with a video, in particular the continuity that links the frames across time. It's reasonable to expect that two consecutive frames will look fairly similar; it's just as reasonable to impose proximity for the within-individual latent variables in two consecutive frames of the same video. Such property relies on a sequence-based approach, that specifically takes into account and models temporal dependencies and dynamics. We refer to this unprecedented approach as dynamical PLDA (or DPLDA). Throughout this chapter we provide the necessary mathematical framework and the implementation details to perform learning and inference in DPLDA models.

Consider the following formulation:

$$\begin{aligned}
\boldsymbol{x}_{ij}^t &= \boldsymbol{\mu} + \boldsymbol{F}\boldsymbol{h}_i + \boldsymbol{G}\boldsymbol{w}_{ij}^t + \boldsymbol{\epsilon}_{ij}^t, && (3.1)\\
\boldsymbol{w}_{ij}^t &= \boldsymbol{A}_{ij}\boldsymbol{w}_{ij}^{t-1} + \boldsymbol{v}_{ij}^t, && (3.2)\\
\boldsymbol{\epsilon}_{ij}^t &\sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Sigma}), && (3.3)\\
\boldsymbol{v}_{ij}^t &\sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}), && (3.4)\\
\boldsymbol{w}_{ij}^1 &\sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}), && (3.5)\\
\boldsymbol{h}_i &\sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}), && (3.6)
\end{aligned}$$

The notation employed hereafter is explained in table 3.1. Note we didn't introduce it before to avoid confusion due to some overlapping with classical PLDA notation.

Any reader familiar with linear dynamical systems will notice a strong similarity between this model and a discrete-time autonomous LDS. The idea underlying model (3.1) - (3.6) is that the evolution of the frames in a video depicting the face of a person, can be ascribed to the evolution of an underlying latent variable comprising pose, illumination and expression variations. In addition to this, a constant latent variable related to identity influences the aspect of each frame, while remaining the same across all videos of one

Table 3.1: Notation

$\boldsymbol{x}_{ij}^t$	–	Observation relative to the t-th frame from the j-th video of the i-th individual.
$\boldsymbol{\epsilon}_{ij}^t$	–	Observation noise affecting the same frame.
$\boldsymbol{w}_{ij}^t$	–	Private latent variable underlying the observation.
$\boldsymbol{v}_{ij}^t$	–	Process noise affecting the evolution of the private latent variable through time.
$\boldsymbol{h}_i$	–	Public latent variable shared across all videos of person i.
$\boldsymbol{z}_{ij}^t$	–	Concatenation of private and public latent variable underlying a specific frame: $z_{ij}^t = [h_i w_{ij}^t]^T$.
$\boldsymbol{\mu}$	–	Mean value, averaged across all frames of every video of each person.
$\boldsymbol{x}_{ij}$	–	Column vector containing j-th video of person i, concatenating all frames in a single vector.
$\boldsymbol{w}_{ij}$	–	Column vector comprising the private latent variables associated to the j-th video of person i.
$\boldsymbol{x}_i^t$	–	Column vector containing the t-th frame from all the videos of person i, concatenated in a single vector.
$\boldsymbol{w}_i^t$	–	Column vector grouping all the private latent variables of person i at time t.
$\boldsymbol{z}_i^t$	–	Column vector grouping all the latent variables, both private and public, for person i at time t.
$\boldsymbol{x}_i$	–	Column vector with all the frames from all the videos available for person i.
$\boldsymbol{z}_i$	–	Column vector grouping all the latent variables, both private and public, for person i.
$\boldsymbol{X}$	–	Column vector assembling the totality of the observations.
$\boldsymbol{w}$	–	Column vector with all the underlying private latent variables.
$\boldsymbol{h}$	–	Column vector with all the underlying public latent variables.
$\boldsymbol{Z}$	–	Column vector grouping the totality of the latent variables.
f	–	Number of dimensions in the observation space.
D_w	–	Number of dimensions for the private latent space.
D_h	–	Number of dimensions for the public latent space.
I	–	Number of different individuals.
J	–	Number of videos per individual.
T	–	Number of frames per video.

person. Again, note that this model is an artificial description and we don't claim it has any adherence with the physics behind the shooting of a video. However, just like PLDA, it exhibits a good discriminatory power and an intuitive explanation for each parameter while retaining mathematical tractability in terms of learning and inference, to which we will now turn our attention.

3.2 Learning and recognition

The learning phase, in analogy with PLDA, requires the estimation of parameter $\boldsymbol{\theta} = (\boldsymbol{\mu}, \boldsymbol{F}, \boldsymbol{G}, \boldsymbol{\Sigma}, \boldsymbol{A}_{11,\dots,IJ})$. Here matrices $\boldsymbol{A}_{ij}$ represent a novelty specific to dynamic PLDA and they are responsible for the evolution of the private latent variables. For $\boldsymbol{w}_{ij}^t$ we expect slow variations not prone to divergence. Hence we anticipate the dominant eigenvalues of $\boldsymbol{A}_{ij}$ to be positive and slightly less than 1.

As mentioned in section 2.4, the first step to perform learning and later recognition is to consider the joint likelihood of observations and latent variables. To avoid complicating an already convoluted notation, in this section we assume that every individual has exactly J videos in the training set, each consisting of T frames. Given this premise we have:

$$p(\boldsymbol{X}, \boldsymbol{Z}|\theta) = \underbrace{\prod_{i=1}^{I} p(\boldsymbol{x}_i|\boldsymbol{z}_i, \boldsymbol{\theta})}_{P_1} \underbrace{\prod_{i=1}^{I} p(\boldsymbol{z}_i|\boldsymbol{\theta})}_{P_2}$$

$$P_1 = \prod_{i=1}^{I}\prod_{j=1}^{J}\prod_{t=1}^{T} \underbrace{p(\boldsymbol{x}_{ij}^t|\boldsymbol{w}_{ij}^t, \boldsymbol{h}_i, \boldsymbol{\theta})}_{\mathcal{N}(\boldsymbol{x}_{ij}^t|\boldsymbol{\mu}+\boldsymbol{F}\boldsymbol{h}_i+\boldsymbol{G}\boldsymbol{w}_{ij}^t, \boldsymbol{\Sigma})}$$

$$P_2 = \left(\prod_{i=1}^{I} \underbrace{p(\boldsymbol{h}_i|\theta)}_{\mathcal{N}(\boldsymbol{h}_i|\boldsymbol{0},\boldsymbol{I})}\right) \prod_{i=1}^{I}\prod_{j=1}^{J} \underbrace{p(\boldsymbol{w}ij^1|\boldsymbol{\theta})}_{\mathcal{N}(\boldsymbol{w}_{ij}^1|\boldsymbol{0},\boldsymbol{I})} \prod_{t=2}^{T} \underbrace{p(\boldsymbol{w}_{ij}^t|\boldsymbol{w}_{ij}^{t-1}, \boldsymbol{\theta})}_{\mathcal{N}(\boldsymbol{w}_{ij}^t|\boldsymbol{A}_{ij}\boldsymbol{w}_{ij}^{t-1},\boldsymbol{I})}$$

$$\begin{aligned}
\ln p(\boldsymbol{X}, \boldsymbol{Z}|\boldsymbol{\theta}) &= \ln P_1 + \ln P_2 \\
&= \sum_{ijt} \ln \mathcal{N}(\boldsymbol{x}_{ij}^t|\boldsymbol{\mu} + \boldsymbol{F}\boldsymbol{h}_i + \boldsymbol{G}\boldsymbol{w}_{ij}^t, \boldsymbol{\Sigma}) + \\
&\quad \sum_{i} \ln \mathcal{N}(\boldsymbol{h}_i|\boldsymbol{0}, \boldsymbol{I}) + \sum_{ij} \ln \mathcal{N}(\boldsymbol{w}_{ij}^1|\boldsymbol{0}, \boldsymbol{I}) + \\
&\quad \sum_{ij}\sum_{t=2}^{T} \ln \mathcal{N}(\boldsymbol{w}_{ij}^t|\boldsymbol{A}_{ij}\boldsymbol{w}_{ij}^{t-1}, \boldsymbol{I}) \\
&= \sum_{ijt} \left\{ \ln\left(\frac{1}{(2\pi)^{f/2}|\boldsymbol{\Sigma}|^{1/2}}\right) + \right. \\
&\qquad \left. -\frac{1}{2}(\boldsymbol{x}_{ij}^t - \boldsymbol{\mu} + \boldsymbol{F}\boldsymbol{h}_i + \boldsymbol{G}\boldsymbol{w}_{ij}^t)^T \boldsymbol{\Sigma}^{-1}(\boldsymbol{x}_{ij}^t - \boldsymbol{\mu} + \boldsymbol{F}\boldsymbol{h}_i + \boldsymbol{G}\boldsymbol{w}_{ij}^t) \right\} + \\
&\quad \sum_{i} \left\{ \ln\left(\frac{1}{(2\pi)^{D_h/2}}\right) - \frac{1}{2}\boldsymbol{h}_i^T\boldsymbol{h}_i \right\} + \sum_{ij} \left\{ \ln\left(\frac{1}{(2\pi)^{D_w/2}}\right) - \frac{1}{2}\boldsymbol{w}_{ij}^{1T}\boldsymbol{w}_{ij}^1 \right\} + \\
&\quad \sum_{ij}\sum_{t=2}^{T} \left\{ \ln\left(\frac{1}{(2\pi)^{D_w/2}}\right) - \frac{1}{2}(\boldsymbol{w}_{ij}^t - \boldsymbol{A}_{ij}\boldsymbol{w}_{ij}^{t-1})^T(\boldsymbol{w}_{ij}^t - \boldsymbol{A}_{ij}\boldsymbol{w}_{ij}^{t-1}) \right\} \qquad (3.7)
\end{aligned}$$

Just like for PLDA, this procedure is a prerequisite for carrying out learning through the EM algorithm. It is based on splitting the joint likelihood into a prior on the latent variables and a conditional for the observations, merely exploiting Bayes' law. Each of them is further decomposed into the elementary building blocks of model (3.1) - (3.6). These are then additively separated thanks to the logarithmic function, which also allows to further simplify the expression by inverting the exponential from the gaussian distributions. Note that, unsurprisingly, PLDA can be obtained as a special case of DPLDA by choosing $T = 1$, that is to say by considering single frames as videos of unitary length.

3.2.1 M step

This phase is very similar to its analogous for PLDA. The main difference is the need to estimate matrix A and to take into account temporal dependencies. In order to evaluate the parameters $\boldsymbol{\theta} = (\boldsymbol{\mu}, \boldsymbol{F}, \boldsymbol{G}, \boldsymbol{\Sigma}, \boldsymbol{A}_{ij})$, we take the expected value of (3.7) with respect to the PDF $p(\boldsymbol{h}_i \boldsymbol{w}_{ij}^t | x)$:

$$\begin{aligned} E\left[\ln p(\boldsymbol{X}, \boldsymbol{Z}|\boldsymbol{\theta})\right] = -\sum_{ijt} \Bigg\{ & \frac{1}{2} \ln |\boldsymbol{\Sigma}| + \frac{1}{2} (\boldsymbol{x}_{ij}^t - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{x}_{ij}^t - \boldsymbol{\mu}) + \frac{1}{2} E\left[\boldsymbol{z}_{ij}^{tT} \boldsymbol{B}^T \boldsymbol{\Sigma}^{-1} \boldsymbol{B} \boldsymbol{z}_{ij}^t\right] + \\ & - E\left[\boldsymbol{z}_{ij}^{tT} \boldsymbol{B}^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{x}_{ij}^t - \boldsymbol{\mu})\right] \Bigg\} \\ & - \sum_{ij} \sum_{t=2}^{T} \left\{ \frac{1}{2} E\left[\boldsymbol{w}_{ij}^{t-1\ T} \boldsymbol{A}_{ij}^T \boldsymbol{A}_{ij} \boldsymbol{w}_{ij}^{t-1}\right] - E\left[\boldsymbol{w}_{ij}^{t-1\ T} \boldsymbol{A}_{ij}^T \boldsymbol{w}_{ij}^t\right] \right\} \quad + K, \end{aligned} \tag{3.8}$$

where K includes all the terms independent of $\boldsymbol{\theta}$ and we have defined

$$\begin{aligned} \boldsymbol{B} &= \begin{bmatrix} \boldsymbol{F} & \boldsymbol{G} \end{bmatrix}, \\ \boldsymbol{z}_{ij}^t &= \begin{bmatrix} \boldsymbol{h}_i \\ \boldsymbol{w}_{ij}^t \end{bmatrix}. \end{aligned}$$

Just like in section 2 we compute the derivatives of expression (3.8) with respect to each parameter. By setting them to zero we have:

$$\boldsymbol{\mu} = \frac{1}{IJT} \sum_{ijt} \boldsymbol{x}_{ij}^t, \tag{3.9}$$

$$\boldsymbol{B} = \left(\sum_{ijt} (\boldsymbol{x}_{ij}^t - \boldsymbol{\mu}) E[\boldsymbol{z}_{ij}^{tT}] \right) \left(\sum_{ijt} E[\boldsymbol{z}_{ij}^t \boldsymbol{z}_{ij}^{tT}] \right)^{-1}, \tag{3.10}$$

$$\boldsymbol{\Sigma} = \frac{1}{IJT} \sum_{ijt} \operatorname{diag} \left\{ (\boldsymbol{x}_{ij}^t - \boldsymbol{\mu})(\boldsymbol{x}_{ij}^t - \boldsymbol{\mu})^T - 2\boldsymbol{B} E[\boldsymbol{z}_{ij}^t](\boldsymbol{x}_{ij}^t - \boldsymbol{\mu})^T + \boldsymbol{B} E[\boldsymbol{z}_{ij}^t \boldsymbol{z}_{ij}^{tT}] \boldsymbol{B}^T \right\}. \tag{3.11}$$

Note the complete analogy with equations (2.33) - (2.35), the only difference laying in the summation over index t. Maximization of (3.8) with respect to A_{ij} yields

$$\begin{aligned} \frac{\partial}{\partial \boldsymbol{A}_{ij}} \quad & \Bigg\{ - \sum_{ij} \sum_{t=2}^{T} \frac{1}{2} tr\left[\boldsymbol{A}_{ij}^T \boldsymbol{A}_{ij} \operatorname{Var}\left(\boldsymbol{w}_{ij}^{t-1} \right) \right] + \frac{1}{2} E\left[\boldsymbol{w}_{ij}^{t-1} \right]^T \boldsymbol{A}_{ij}^T \boldsymbol{A}_{ij} E\left[\boldsymbol{w}_{ij}^{t-1} \right] \\ & - tr\left[\boldsymbol{A}_{ij} \operatorname{Cov}(\boldsymbol{w}_{ij}^{t-1}, \boldsymbol{w}_{ij}^t) \right] - E\left[\boldsymbol{w}_{ij}^{t-1} \right]^T \boldsymbol{A}_{ij}^T \left[\boldsymbol{w}_{ij}^t \right] \Bigg\} = 0, \end{aligned}$$

where we have used

$$E\left[\boldsymbol{x}^T\boldsymbol{A}\boldsymbol{y}\right] = tr\left[\boldsymbol{A}^T \operatorname{Cov}(\boldsymbol{x},\boldsymbol{y})\right] + E(\boldsymbol{x})^T\boldsymbol{A}E(\boldsymbol{y}). \tag{3.12}$$

Hence

$$\sum_{t=2}^{T}\left\{\boldsymbol{A}_{ij}\operatorname{Var}\left(\boldsymbol{w}_{ij}^{t-1}\right) - \operatorname{Cov}\left(\boldsymbol{w}_{ij}^{t-1},\boldsymbol{w}_{ij}^{t}\right)^T + E\left[\boldsymbol{w}_{ij}^{t}\right]E\left[\boldsymbol{w}_{ij}^{t-1}\right]^T(\boldsymbol{A}_{ij}-1)\right\} = 0$$

$$\boldsymbol{A}_{ij} = \left(\sum_{t=2}^{T} E\left[\boldsymbol{w}_{ij}^{t}\boldsymbol{w}_{ij}^{t-1\,T}\right]\right)\left(\sum_{t=2}^{T} E\left[\boldsymbol{w}_{ij}^{t-1}\boldsymbol{w}_{ij}^{t-1\,T}\right]\right)^{-1} \tag{3.13}$$

As a special case, if we imposed the same state evolution matrix for each $\boldsymbol{w}_{ij}$, that is $\boldsymbol{A}_{ij} = \boldsymbol{A}$, we would get

$$\boldsymbol{A} = \left(\sum_{ij}\sum_{t=2}^{T} E\left[\boldsymbol{w}_{ij}^{t}\boldsymbol{w}_{ij}^{t-1\,T}\right]\right)\left(\sum_{ij}\sum_{t=2}^{T} E\left[\boldsymbol{w}_{ij}^{t-1}\boldsymbol{w}_{ij}^{t-1\,T}\right]\right)^{-1}. \tag{3.14}$$

3.2.2 E step

In this phase we need to compute the first and second order moments in formulas (3.9), (3.10),(3.11) and (3.13), where the expectation is conditional on all the available observations $\boldsymbol{X}$. As already pointed out, equations (3.1) - (3.6) describe a system largely similar to an autonomous LDS. As i and j vary, a number IJ of such systems is spanned. The main difference with a classical LDS, however, is the presence of a latent variable h_i shared across all the J videos of person i. The most intuitive solution to this problem is to combine the observations and the latent variables for each video in analogy with (2.25), resulting in the augmented system

$$\begin{bmatrix}\boldsymbol{x}_{i1}^t\\ \boldsymbol{x}_{i2}^t\\ \vdots\\ \boldsymbol{x}_{iJ}^t\end{bmatrix} = \begin{bmatrix}\boldsymbol{\mu}\\ \boldsymbol{\mu}\\ \vdots\\ \boldsymbol{\mu}\end{bmatrix} + \begin{bmatrix}\boldsymbol{F} & \boldsymbol{G} & \boldsymbol{0} & \cdots & \boldsymbol{0}\\ \boldsymbol{F} & \boldsymbol{0} & \boldsymbol{G} & \cdots & \boldsymbol{0}\\ \vdots & \vdots & \vdots & \ddots & \vdots\\ \boldsymbol{F} & \boldsymbol{0} & \boldsymbol{0} & \cdots & \boldsymbol{G}\end{bmatrix}\begin{bmatrix}\boldsymbol{h}_i\\ \boldsymbol{w}_{i1}^t\\ \vdots\\ \boldsymbol{w}_{iJ}^t\end{bmatrix} + \begin{bmatrix}\boldsymbol{\epsilon}_{i1}^t\\ \boldsymbol{\epsilon}_{i2}^t\\ \vdots\\ \boldsymbol{\epsilon}_{iJ}^t\end{bmatrix}, \tag{3.15}$$

$$\begin{bmatrix}\boldsymbol{h}_i\\ \boldsymbol{w}_{i1}^t\\ \vdots\\ \boldsymbol{w}_{iJ}^t\end{bmatrix} = \begin{bmatrix}\boldsymbol{I} & & & & \\ & \boldsymbol{A}_{i1} & & & \\ & & \boldsymbol{A}_{i2} & & \\ & & & \ddots & \\ & & & & \boldsymbol{A}_{1J}\end{bmatrix}\begin{bmatrix}\boldsymbol{h}_i\\ \boldsymbol{w}_{i1}^{t-1}\\ \vdots\\ \boldsymbol{w}_{iJ}^{t-1}\end{bmatrix} + \begin{bmatrix}0\\ \boldsymbol{v}_{i1}^t\\ \vdots\\ \boldsymbol{v}_{iJ}^t\end{bmatrix}, \tag{3.16}$$

which can be re-written in a more compact form as

$$\begin{cases}\boldsymbol{x}_i^t = \bar{\boldsymbol{C}}\boldsymbol{z}_i^t + \bar{\boldsymbol{\mu}} + \boldsymbol{\epsilon}_i^t\\ \boldsymbol{z}_i^t = \bar{\boldsymbol{A}}_i\boldsymbol{z}_i^{t-1} + \boldsymbol{v}_i^t\end{cases}, \tag{3.17}$$

where

$$\boldsymbol{\epsilon}_i^t \sim \mathcal{N}(\boldsymbol{0},\bar{\boldsymbol{\Sigma}}), \qquad \bar{\boldsymbol{\Sigma}} = \begin{bmatrix}\boldsymbol{\Sigma} & & & \\ & \boldsymbol{\Sigma} & & \\ & & \ddots & \\ & & & \boldsymbol{\Sigma}\end{bmatrix} \tag{3.18}$$

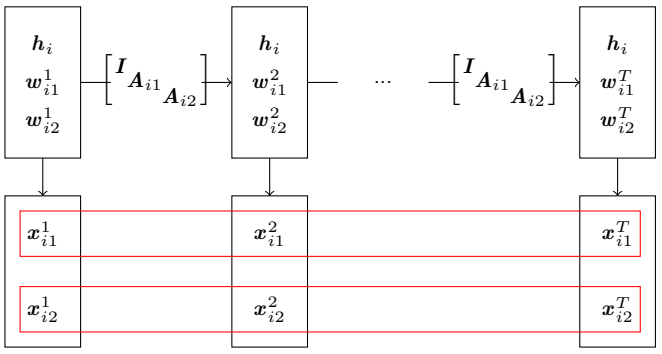

Figure 3.1: Schematic for the LDS. The arrows depict the flow of information as each latent variable is realized and influences both its future value and the current state. The frames from the same video are surrounded by red rectangles.

$$\boldsymbol{v}_i^t \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{\Gamma}), \qquad \boldsymbol{\Gamma} = \begin{bmatrix} \boldsymbol{0} & & & \\ & \boldsymbol{I} & & \\ & & \ddots & \\ & & & \boldsymbol{I} \end{bmatrix} \tag{3.19}$$

and

$$\begin{aligned} &\boldsymbol{x}_i^t,\ \bar{\boldsymbol{\mu}},\ \boldsymbol{\epsilon}_i^t \in \mathbb{R}^{Jfx1}, \\ &\boldsymbol{z}_i^t,\ \boldsymbol{v}_i^t \in \mathbb{R}^{D_h+JD_w}, \\ &\bar{\boldsymbol{C}} \in \mathbb{R}^{Jfx(D_h+JD_w)}, \\ &\bar{\boldsymbol{A}}_i \in \mathbb{R}^{(D_h+JD_w)x(D_h+JD_w)}. \end{aligned}$$

This way we successfully put system (3.17) in a canonical LDS form, while guaranteeing that the original properties desired for the model are retained. In this respect, note that that we are forcing the identity variable $\boldsymbol{h}_i$ to be shared for all the videos of person i while remaining the same across all frames. The private variable $\boldsymbol{w}_{ij}^t$ is instead subject to the typical dynamics of a linear system. Figure 3.1 depicts a schematic for the model with $J = 2$. Here, much like in a graphical model, the arrows provide information about the conditional dependencies in effect. The red rectangles enclose all the frames from a single video.

It should be observed that reducing our model to a canonical LDS came at the cost of multiplying the size of the observations and the hidden state by a factor of J, which results in an undesirable computational complexity for Kalman filtering, increasing with the cube of J. In section 3.3.1 we provide a less intuitive yet equivalent formulation which circumvents this issue. Hereafter, borrowing from the control systems literature, we will use the word "state" as a synonym for latent variable.

The advantage of dealing with a canonical LDS is the chance of resorting to the related, well established estimate techniques, epitomized by the Kalman filter. In the presence of jointly gaussian observations and latent variables, the Kalman filter gives unbiased conditional estimates for the state while keeping at a minimum the mean square of the prediction error. Also, it scales well with T since its computational complexity is linear with respect to the length of the filtered sequence. For a thorough introduction to Kalman filtering, [24] and [6] are helpful references. From the latter we borrow the notational

convention, in recalling the filtering equations:

$$\begin{aligned}
\boldsymbol{m}_i^t &= \bar{\boldsymbol{A}}_i \boldsymbol{m}_i^{t-1} + \boldsymbol{K}_i^t(\boldsymbol{x}_i^t - \bar{\boldsymbol{C}}\bar{\boldsymbol{A}}_i \boldsymbol{m}_i^{t-1}), && (3.20)\\
\boldsymbol{V}_i^t &= (\boldsymbol{I} - \boldsymbol{K}^t \bar{\boldsymbol{C}})\boldsymbol{P}_i^{t-1}, && (3.21)\\
\boldsymbol{P}_i^{t-1} &= \bar{\boldsymbol{A}}_i \boldsymbol{V}_i^{t-1} \bar{\boldsymbol{A}}^T + \boldsymbol{\Gamma}, && (3.22)\\
\boldsymbol{K}_i^t &= \boldsymbol{P}_i^{t-1} \bar{\boldsymbol{C}}^T (\boldsymbol{C}\boldsymbol{P}_i^{t-1}\boldsymbol{C}^T + \boldsymbol{\Sigma})^{-1}, && (3.23)
\end{aligned}$$

with first update

$$\begin{aligned}
\boldsymbol{m}_i^1 &= \boldsymbol{m}_i^0 + \boldsymbol{K}_i^1(\boldsymbol{x}_i^1 - \bar{\boldsymbol{C}}\boldsymbol{m}_i^0), && (3.24)\\
\boldsymbol{V}_i^1 &= (\boldsymbol{I} - \boldsymbol{K}^1 \bar{\boldsymbol{C}})\boldsymbol{P}_i^0, && (3.25)
\end{aligned}$$

and initial conditions, resulting from priors (3.5) - (3.6), given by

$$\begin{aligned}
\boldsymbol{m}_i^0 &= [\boldsymbol{0}, \boldsymbol{0}, \ldots, \boldsymbol{0}]^T, && (3.26)\\
\boldsymbol{P}_i^0 &= \begin{bmatrix} \boldsymbol{I} & & & \\ & \boldsymbol{I} & & \\ & & \ddots & \\ & & & \boldsymbol{I} \end{bmatrix}. && (3.27)
\end{aligned}$$

These equations allow us to compute the filtering probabilities

$$P(\boldsymbol{z}_i^t | \boldsymbol{x}_i^1, \ldots, \boldsymbol{x}_i^t) = \mathcal{N}(\boldsymbol{m}_i^t, \boldsymbol{V}_i^t), \tag{3.28}$$

and the conditionals

$$\begin{aligned}
c_i^t &= P(\boldsymbol{x}_i^t | \boldsymbol{x}_i^1, \ldots, \boldsymbol{x}_i^{t-1}) = \mathcal{N}(\boldsymbol{x}_i^t | \bar{\boldsymbol{C}}\bar{\boldsymbol{A}}_i \boldsymbol{m}_i^{t-1}, \boldsymbol{\Sigma} + \bar{\boldsymbol{C}}\boldsymbol{P}_i^{t-1}\bar{\boldsymbol{C}}^T), && (3.29)\\
c_i^1 &= P(\boldsymbol{x}_i^1) = \mathcal{N}(\boldsymbol{x}_i^1 | \bar{\boldsymbol{C}}\boldsymbol{m}_i^0, \boldsymbol{\Sigma} + \bar{\boldsymbol{C}}\boldsymbol{P}_i^0\bar{\boldsymbol{C}}^T) && (3.30)
\end{aligned}$$

Note that the probability (3.28) is for $\boldsymbol{z}_i^t$ conditional on the observations up to the same time step t, whereas for the E step we want to condition on all the available information $\boldsymbol{X}$ up to the final time step T. To take into account the totality of the information, the filtering phase is followed by Kalman smoothing, which is a backward propagation of information that can be formulated as follows:

$$\begin{aligned}
\widehat{\boldsymbol{m}}_i^T &= \boldsymbol{m}_i^T, && (3.31)\\
\widehat{\boldsymbol{V}}_i^T &= \boldsymbol{V}_i^T, && (3.32)\\
\widehat{\boldsymbol{m}}_i^t &= \boldsymbol{m}_i^t + \boldsymbol{J}_i^t(\widehat{\boldsymbol{m}}_i^{t+1} - \bar{\boldsymbol{A}}_i \boldsymbol{m}_i^t), && (3.33)\\
\widehat{\boldsymbol{V}}_i^t &= \boldsymbol{V}_i^t + \boldsymbol{J}_i^t(\widehat{\boldsymbol{V}}_i^{t+1} - \boldsymbol{P}_i^t)\boldsymbol{J}_i^{t\,T}, && (3.34)\\
\boldsymbol{J}_i^t &= \boldsymbol{V}_i^t \bar{\boldsymbol{A}}_i^T (\boldsymbol{P}_i^t)^{-1}, && (3.35)
\end{aligned}$$

yielding the smoothing probabilities

$$P(\boldsymbol{z}_i^t | \boldsymbol{X}) = \mathcal{N}(\boldsymbol{z}_i^t | \widehat{\boldsymbol{m}}_i^t, \widehat{\boldsymbol{V}}_i^t), \tag{3.36}$$

and second order moments

$$\mathrm{Cov}\left[\boldsymbol{z}_i^{t-1}, \boldsymbol{z}_i^t | \boldsymbol{X}\right] = \boldsymbol{J}_i^{t-1} \widehat{\boldsymbol{V}}_i^t. \tag{3.37}$$

Remembering the definition of $\boldsymbol{z}_i^t = \left[\boldsymbol{h}_i, \boldsymbol{w}_{i1}^t, \ldots, \boldsymbol{w}_{iJ}^t\right]^T$, from the first and second order moments in formulas (3.33), (3.34), (3.36), we can extract the sufficient statistics necessary for the M step.

3.2.3 Recognition

As in chapter 2, recognition problems can be solved, provided we have an algorithm to estimate the likelihood that J videos share the same identity. Namely we have to estimate $P(\boldsymbol{x}_i) = P(\boldsymbol{x}_i^1, \dots, \boldsymbol{x}_i^T)$, where $\boldsymbol{x}_i^t$ is defined according to (3.17) as the column vector containing all the pixels of all videos of person i at time t. Resorting to equation (3.29), said likelihood can be computed through Kalman filtering as

$$P(\boldsymbol{x}_i) = \prod_{t=1}^{T} c_i^t. \tag{3.38}$$

3.3 Implementation Details

3.3.1 Video concatenation

As mentioned in the previous section, augmenting the state and the observation vectors according to model (3.15) becomes more and more inefficient as the number of gallery videos per individual J increases. Typically, the bottleneck for Kalman filters is represented by the inversion of matrix $(\boldsymbol{C}\boldsymbol{P}_i^{t-1}\boldsymbol{C}^T + \boldsymbol{\Sigma})$ in equation (3.23), which for the above mentioned model would be of size Jf x Jf. Its inversion would require a number of operations proportional to the cube of Jf. A more desirable model would allow us to transfer across videos the information learnt about the identity latent variable $\boldsymbol{h}_i$, while maintaining a linear complexity with respect to J. Another limitation of the above formulation is that it does not easily generalize to the case of videos of different length.

Consider the model in figure 3.2, alternative to the one previously discussed in figure 3.1. According to this new setting, depicting the case $J = 2$, videos of the same person can be joined end-to front, avoiding the inefficient state and observation augmentation. This is shown by the rectangles in red, which again delimit the frames within the same video. Exploiting this idea, frames from videos of the same person are concatenated in one single macro-video of length JT. Crucially, in correspondence to the transition from one video to the next, the matrix responsible for the evolution of the latent variables changes. More specifically, its lower diagonal block, the one responsible for the evolution of the private latent variable, is equal to $\boldsymbol{A}_{i1}$ for the whole duration of the first video. When the video ends, it becomes zero for one iteration, making sure that the prior for the private latent variable $\boldsymbol{w}_{i2}^1$ is set to a gaussian with zero mean and unit covariance, blocking the transfer of information from video to video. For the next iteration, i.e. for the transition from the first to the second frame of video 2, the lower diagonal block takes the value $\boldsymbol{A}_{i2}$, and retains it for the whole duration of the second video. If a third video was available, another

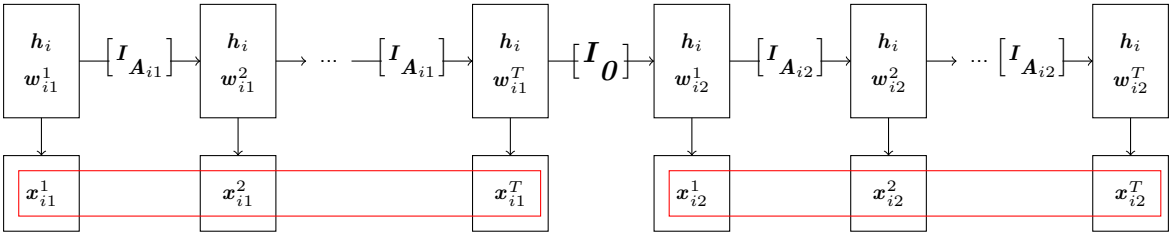

Figure 3.2: Schematic for the LDS. Videos of the same person are joined end-to-front in a single macro video. In correspondence of a transition from a video to the next one, the state evolution matrix changes.

"zero transition" would be present between video 2 and video 3. At the same time, the upper diagonal block, responsible for the evolution of the public latent variable $\boldsymbol{h}_i$, remains fixed at I. This effectively forces $\boldsymbol{h}_i$ to be constant across all videos and all frames of the same individual and allows us to transfer the related information across videos.

To show the equivalence between this formulation and the one featuring the augmented state, we have to prove that the time-varying model resulting from the different state transition matrices described above, applied to the macro-video, enforces the desired priors $\boldsymbol{w}_{ij}^1 \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$. For $j = 1$ this is trivial, since $\boldsymbol{w}_{ij}^1$ underlays the first frame of the macro-video, for which such prior is imposed as usual. For $j > 1$, model (3.17)-(3.19) still holds, after redefining

$$\bar{\boldsymbol{C}} = [\boldsymbol{F}, \boldsymbol{G}], \tag{3.39}$$

$$\boldsymbol{\Gamma} = \begin{bmatrix} \boldsymbol{0} & \\ & \boldsymbol{I} \end{bmatrix} \tag{3.40}$$

$$\boldsymbol{z}_i^t = [\boldsymbol{h}_i,\, \boldsymbol{w}_i^t]^T, \tag{3.41}$$

$$\boldsymbol{w}_i^t = \boldsymbol{w}_{ij}^k, \tag{3.42}$$

$$\boldsymbol{x}_i^t = \boldsymbol{x}_{ij}^k, \tag{3.43}$$

where j, k are defined as

$$j = (t-1)/T + 1,$$

$$k = (t-1) \;\%\; T \;\; + 1.$$

Here '/' denotes the integer division and '%' its remainder. Finally, state matrix $\boldsymbol{A}_i$ varies as discussed above, according to figure 3.2.

Given said definitions, let us take as an example $j = 2$, without any loss of generality. We want to estimate

$$P(\boldsymbol{z}_i^{T+1} | \boldsymbol{x}_i^1, \ldots, \boldsymbol{x}_i^{T+1}) = P\left(\begin{bmatrix} \boldsymbol{h}_i \\ \boldsymbol{w}_{i2}^1 \end{bmatrix} | \boldsymbol{x}_{i1}^1, \ldots, \boldsymbol{x}_{i1}^T, \boldsymbol{x}_{i2}^1 \right) = \mathcal{N}(\boldsymbol{m}_i^{T+1},\, \boldsymbol{V}_i^{T+1}) \tag{3.44}$$

and look into the PDF of its lower half $\boldsymbol{w}_{i2}^1$. Notice that the current state matrix

$$\boldsymbol{A}_i^{T+1} = \begin{bmatrix} \boldsymbol{I} & \\ & \boldsymbol{0} \end{bmatrix} \tag{3.45}$$

directly affects the update of $\boldsymbol{P}_i^T$ and $\boldsymbol{m}_i^{T+1}$ as follows

$$\boldsymbol{P}_i^T = \begin{bmatrix} \boldsymbol{I} & \\ & \boldsymbol{0} \end{bmatrix} \boldsymbol{V}_i^T \begin{bmatrix} \boldsymbol{I} & \\ & \boldsymbol{0} \end{bmatrix} + \begin{bmatrix} \boldsymbol{0} & \\ & \boldsymbol{I} \end{bmatrix} = \begin{bmatrix} \boldsymbol{V}_{i,h}^T & \\ & \boldsymbol{I} \end{bmatrix} \tag{3.46}$$

$$\begin{aligned} \boldsymbol{m}_i^{T+1} &= \begin{bmatrix} \boldsymbol{I} & \\ & \boldsymbol{0} \end{bmatrix} \boldsymbol{m}_i^T + \boldsymbol{K}_i^{T+1} \left(\boldsymbol{x}_i^{T+1} - [\boldsymbol{F}\, \boldsymbol{G}] \begin{bmatrix} \boldsymbol{I} & \\ & \boldsymbol{0} \end{bmatrix} \boldsymbol{m}_i^T \right) \\ &= \begin{bmatrix} \bar{\boldsymbol{h}}_i^T \\ 0 \end{bmatrix} + \boldsymbol{K}_i^{T+1} \left(\boldsymbol{x}_i^{T+1} - \begin{bmatrix} \boldsymbol{F} \bar{\boldsymbol{h}}_i^T \\ \boldsymbol{0} \end{bmatrix} \right), \end{aligned} \tag{3.47}$$

where we have called $\bar{\boldsymbol{h}}_i^T$, $\boldsymbol{V}_{i,h}^T$ the estimated mean and covariance for $\boldsymbol{h}_i$ conditional on the information up to time step T. Comparing these results with (3.24) - (3.27), it should be clear that this model behaves exactly as desired. That is to say, in correspondence to

a transition from the first video to the second, it behaves like a freshly started Kalman filter with prior beliefs,

$$\boldsymbol{P}_i^0 = \begin{bmatrix} \boldsymbol{V}_{i,h}^T & \\ & \boldsymbol{I} \end{bmatrix}, \tag{3.48}$$

$$\boldsymbol{m}_i^0 = \begin{bmatrix} \bar{\boldsymbol{h}}_i^T \\ \boldsymbol{0} \end{bmatrix}, \tag{3.49}$$

thus allowing for an unaltered transfer of information about $\boldsymbol{h}_i$ while resetting the information about $\boldsymbol{w}_{i2}$ just like a prior $\boldsymbol{w}_{i2}^1 \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$. Equivalently it can be said that, under the model just discussed,

$$P(\boldsymbol{w}_{i2}^1 | \boldsymbol{x}_{i1}^1, \ldots, \boldsymbol{x}_{i1}^T, \boldsymbol{x}_{i2}^1) = P(\boldsymbol{w}_{i2}^1 | \boldsymbol{x}_{i2}^1). \tag{3.50}$$

3.3.2 Offline computations

By inspecting the filtering and smoothing equations (3.20) - (3.23) and (3.33) - (3.35), one should notice that the heaviest computations can be carried out offline, since the measurements $\boldsymbol{X}$ only affect the estimates of the expectations. This is especially advantageous in the case of a unique state matrix shared among all videos, that is $\boldsymbol{A}_{ij} = \boldsymbol{A}$. In this case, the most demanding operation, the inversion of matrices $(\boldsymbol{C}\boldsymbol{P}_i^{t-1}\boldsymbol{C}^T + \boldsymbol{\Sigma})$, can be carried out once and for all videos preliminarily, thus significantly alleviating the computational burden associated with DPLDA.

In addition, the inversion of matrix $(\boldsymbol{C}\boldsymbol{P}_i^{t-1}\boldsymbol{C}^T + \boldsymbol{\Sigma})$ can be carried out efficiently exploiting Woodbury's formula

$$(\boldsymbol{C}\boldsymbol{P}_i^{t-1}\boldsymbol{C}^T + \boldsymbol{\Sigma})^{-1} = \boldsymbol{\Sigma}^{-1} - \boldsymbol{\Sigma}^{-1}(\boldsymbol{C}(\boldsymbol{C}^T\boldsymbol{\Sigma}^{-1}\boldsymbol{C} + [\boldsymbol{P}_i^{t-1}]^{-1})^{-1}\boldsymbol{C}^T)\boldsymbol{\Sigma}^{-1}, \tag{3.51}$$

wherein the inversion of $\boldsymbol{\Sigma}$ is simple to compute due to its diagonal structure.

3.3.3 Inference

In order to decide the most likely identity for a probe video, in principle we can estimate the likelihood that it shares its identity with all the subjects in the gallery, according to (3.38) and pick the person that maximizes it. This would imply concatenating the probe video to all the gallery videos of a subject as discussed in subsection 3.3.1, and performing Kalman filtering on this macro video. Suppose that the probe video consists of T frames and that the gallery is made up of I_{gal} individuals each of which is depicted in J different videos of length T. To try out all the possible combinations one would have to run I_{gal} Kalman filters of length $(J+1)T$.

A more efficient approach consists in running offline a Kalman filter of length JT on each gallery subject, consequently storing the learnt values for the public latent variable. As soon as a probe video is available for identification, I_{gal} Kalman filters of length T can be run along its frames utilizing a different initial state estimate $\boldsymbol{m}_i^0$ in each of the runs, corresponding to the values learnt for $\boldsymbol{h}_i$ from each different gallery individual which had previously been stored. Supposing I_{probe} different videos need to be identified, this simple trick reduces the computational requirements from $O(I_{gal}I_{probe}(J+1)T)$ to $O(I_{gal}JT + I_{probe}I_{gal}T)$.

From a probabilistic perspective this approach basically consists in recognizing that the joint probability of gallery and probe videos can be split as

$$P(\boldsymbol{x}_{i1} \ldots \boldsymbol{x}_{iJ}\boldsymbol{x}_{probe}) = P(\boldsymbol{x}_{i1} \ldots \boldsymbol{x}_{iJ})P(\boldsymbol{x}_{probe} | \boldsymbol{x}_{i1} \ldots \boldsymbol{x}_{iJ}) \tag{3.52}$$

into a marginal probability for the gallery videos alone and a conditional probability for the probe based on the information learnt from the gallery videos. These 2 probabilities can be assessed using two separate Kalman filters.

Note that the evaluation of said probabilities entails the evaluation of a multivariate gaussian PDF, thus requiring the computation of the determinant for matrix $(\boldsymbol{C}\boldsymbol{P}_i^{t-1}\boldsymbol{C}^T + \boldsymbol{\Sigma})$. Consider for instance a video consisting of 90x90 pixel frames, which can be concatenated in a single feature vector of size $f = 8100$. The covariance matrices $(\boldsymbol{C}\boldsymbol{P}_i^{t-1}\boldsymbol{C}^T + \boldsymbol{\Sigma})$ will thus have size 8100x8100. With double precision, if the geometric mean of its eigenvalues falls outside the interval $[0.91, 1.09]$, direct evaluation of the matrix's determinant will result in underflow or overflow. For this reason, said computation is carried out in the domain of logarithms. By exploiting the positive definiteness of the covariance matrix, we can resort to a Cholesky decomposition to obtain its eigenvalues and compute the logarithm of its determinant l as follows. Given the Cholesky decomposition

$$(\boldsymbol{C}\boldsymbol{P}_i^{t-1}\boldsymbol{C}^T + \boldsymbol{\Sigma}) = \boldsymbol{U}\boldsymbol{U}^T,$$

then

$$l = 2\sum_{k=1}^{f} \log(\boldsymbol{U}_{kk}),$$

where $\boldsymbol{U}_{kk}$ denotes the k-th element on the diagonal of matrix $\boldsymbol{U}$. With this simple idea we can circumvent the problem of underflow and overflow in an efficient manner while exploiting the LDSs by direct comparison of different log-likelihoods.

4. Pipeline and simulations

The models presented so far are applicable to face images in ideal viewing conditions, wherein faces have been located, cropped and appear under a frontal pose. Furthermore, although model noise is accounted for with a gaussian term, it is unrealistic to assume that disturbances such as occlusions and different illuminations will always conform to our statistical convenience. For this reason, we introduce a recognition pipeline wherein DPLDA represents the third and last stage. In order to obtain a good input for DPLDA, we employ a first stage for face detection and frontalization, followed by one of feature extraction. The first stage makes sure that pixel locations can be consistently treated as specific to the same area of the face, e.g. guaranteeing that pixel (40,32) always lays on the tip of the nose. The second stage extracts from each frontalized image a representation that is more suitable for face recognition, guaranteeing a higher robustness margin to pixel noise. Finally, DPLDA takes care of the recognition task, with improved performance thanks to the high quality features resulting from the combination of the two previous stages.

4.1 Face frontalization

The problems of face detection and alignment are an essential prerequisite for face recognition, guaranteeing the subsequent steps of a recognition pipeline to work on pixel values consistently related to the same areas of the face. For this purpose we adopt the FAR framework, as detailed in [25]. This technique performs joint face landmark detection and frontalization reconstruction, employing a limited amount of training data from frontal-pose faces, without any need to resort to popular yet computationally expensive 3D face models.

A pivotal idea for this technique is the consideration that the rank for a frontal image (expressed as a matrix) is smaller than the rank from the same face in any other pose, due to the natural symmetry of the human face. Specifically, the problem of face frontalization can be viewed as that of finding the warping of a random-pose face image that gives the closest result to a frontal image of the same face. In this respect, we define $\boldsymbol{x}(\boldsymbol{p}) = \boldsymbol{x}(\mathcal{W}(\boldsymbol{p}))$ the warping of an image, dependent on warping coordinates $\boldsymbol{p}$ and denote by $\boldsymbol{X} = \mathcal{R}_{m\times n}(\boldsymbol{x})$ the reshape operator that acts on a vector yielding an n by m matrix.

The idea of a low rank representation for a frontal pose face is then coupled with a belief that only a limited amount of pixels will be corrupted by non-gaussian noise, giving rise to a sparsity constraint on the error matrix. Overall, a warped corrupted image can be expressed as

$$\boldsymbol{X}(\boldsymbol{p}) = \boldsymbol{L} + \boldsymbol{E} = \sum_{i=1}^{k} \mathcal{R}_{m\times n}(\boldsymbol{u}_i)c_i + \boldsymbol{E}. \tag{4.1}$$

Here $\boldsymbol{E}$ represents a sparse error matrix and $\boldsymbol{L} = \sum_{i=1}^{k} \mathcal{R}_{m\times n}(\boldsymbol{u}_i)c_i$ is the clean frontal-

Figure 4.1: Frontalized faces. Frames from our database of celebrities downloaded from Youtube.

ization of a face image, laying on the low dimensional subspace spanned by the basis $< \boldsymbol{u}_i >$, $i = 1 \dots k$. By imposing the low rank and sparsity constraints the following optimization problem arises:

$$\begin{aligned} &\underset{\boldsymbol{L},\boldsymbol{E},c,\Delta p}{\arg\min} \quad \|\boldsymbol{L}\|_* + \lambda \|\boldsymbol{E}\|_1 \\ &\text{s.t.} \qquad\quad \boldsymbol{X}(\boldsymbol{p}) = \boldsymbol{L} + \boldsymbol{E}, \quad \boldsymbol{L} = \sum_{i=1}^{k} \mathcal{R}_{m \times n}(\boldsymbol{u}_i) c_i. \end{aligned} \tag{4.2}$$

Here $\|\|_*$ denotes the nuclear norm and $\|\|_1$ the l-1 norm, which are employed as convex approximations of rank and cardinality respectively. Optimization problem (4.2) is far from trivial, mainly due to its non-linearity with respect to $\boldsymbol{p}$ and is solved with an augmented lagrangian. Figure 4.1 shows the result of the frontalization process on a private in-the-wild database we assembled from Youtube.

4.2 Feature extraction

So far we dealt with (grey scale) images in their "natural" representation, i.e. as a sequence of pixel values ranging from a minimum (black) to a maximum (white). However, this representation is not robust to affine transformation, differences in illumination, occlusions and, in general, noise. Feature extraction is the branch of machine learning that deals with the research of ways to describe the data, that prove to be particularly suitable for a task of interest, in environments with different extents of non-ideality. In our case, we would like to extract features from images which allow us a good performance for tasks of face recognition in videos, exhibiting robustness to non-gaussian noise. An increasing amount of alternative representations have been proposed and studied in computer vision. Some of them are directly derived from pixel values, such as Local Binary Patterns, others act on different domains, such as Histogram of Oriented Gradients (HOG) or the popular Scale-Invariant Feature Transform (SIFT) both requiring the computation of image gradients.

As we test our algorithm on videos from different and challenging databases, we expect the respective frames to be far from ideal. On the contrary, significant pixel noise is encountered and has to be dealt with by our identification procedure. For this reason we introduce this stage within our pipeline, wherein we choose a robust feature representation for the frontalized images.

Image Gradient Orientation (IGO) methods, as described in [31], represent an interesting alternative to pixel-based features and were shown to outperform other popular descriptors, such as LBP and Gabor features, in tasks of face recognition. In this section we describe the intuition behind it and sketch a proof of its suitability for face recognition in pre-aligned images.

Consider a set of images $\{\boldsymbol{I}_i\}$. Let us compute the respective gradients and indicate by $\{\boldsymbol{\Phi}_i\}$, their orientation, which we normalize so that $\boldsymbol{\Phi}_i \in [0, 2\pi)$. Taking two different images, we denote their orientation difference as

$$\Delta(\boldsymbol{\Phi}_{ij}) = \boldsymbol{\Phi}_i - \boldsymbol{\Phi}_j. \tag{4.3}$$

Let us define as $\boldsymbol{P}$ the indices corresponding to the image support, arranged in lexicographic order and as $N(\mathcal{P})$ the cardinality of the set. We can then give the following

Definition Two images $\boldsymbol{I}_i$ $\boldsymbol{I}_j$ are pixel-wise dissimilar if $\forall k \in \mathcal{P},\ \Delta\boldsymbol{\Phi}_{ij}(k) \sim \mathcal{U}[0, 2\pi)$.

The above condition can be verified with a statistical significance test in order to determine whether 2 images follow the definition or not. In the domain of gradient orientations, let us define the cosine-based correlation between two images as follows:

$$s(\boldsymbol{\phi}_i, \boldsymbol{\phi}_j) = \sum_{k\in\mathcal{P}} \cos[\Delta\boldsymbol{\Phi}_{ij}(k)] = cN(\mathcal{P}), \tag{4.4}$$

where $c \in [-1, 1]$ is the coefficient providing a measurement of the spatial correlation between images $\boldsymbol{I}_i$ and $\boldsymbol{I}_j$. Note for instance that for highly correlated images $\Delta\boldsymbol{\Phi}_{ij}(k) \approx 0$ hence $c \to 1$.

As we will show, the correlation measurement (4.4) is robust with respect to outliers. In this regard, suppose we can partition the image support $\mathcal{P}$ into a subset $\mathcal{P}_2$ of pixels corrupted by outliers (within either image) and a subset $\mathcal{P}_1$ of ideally noise-less pixels (in both images). A robust correlation measurement between two such images should disregard $\mathcal{P}_2$ and be well approximated by

$$s_1(\boldsymbol{\phi}_i, \boldsymbol{\phi}_j) = \sum_{k\in\mathcal{P}_1} \cos[\Delta\boldsymbol{\Phi}_{ij}(k)] = c_1 N(\mathcal{P}). \tag{4.5}$$

To show that indeed this property holds, we assume that in $\mathcal{P}_2$ the images are pixel-wise dissimilar according to the definition given above. This is a reasonable assumption which proves to be well supported by empirical verification and makes the following theorem meaningful for our analysis:

Theorem Let $u(.)$ be a random process and $u(t) \sim \mathcal{U}[0, 2\pi)$ then:

- $E[\int_{\mathcal{X}} \cos u(t)dt] = 0$ for any non-empty interval $\mathcal{X} \in \mathbb{R}$.
- If $u(.)$ is mean ergodic, then $\int_{\mathcal{X}} \cos u(t)dt = 0$.

We also employ the following approximation

$$\int_{\mathcal{X}} \cos[\Delta\boldsymbol{\Phi}_{ij}(t)]dt \simeq \sum_{k\in\mathcal{P}} \cos[\Delta\boldsymbol{\Phi}_{ij}(k)], \tag{4.6}$$

where the integrand in the left hand side is defined in a continuous domain with some notational abuse. By applying the theorem above to the subset of pixels $\mathcal{P}_2$ corrupted by outliers, we have

$$s_2(\boldsymbol{\phi}_i, \boldsymbol{\phi}_j) = \sum_{k\in\mathcal{P}_2} \cos[\Delta\boldsymbol{\Phi}_{ij}(k)] \simeq 0, \tag{4.7}$$

which in turn yields

$$s(\boldsymbol{\phi}_i, \boldsymbol{\phi}_j) = s_1(\boldsymbol{\phi}_i, \boldsymbol{\phi}_j), \tag{4.8}$$

completing our sketch of proof for the robustness of correlation measurement (4.4) with respect to outliers.

The considerations above allow us to define an interesting distance measure:

$$d^2(\boldsymbol{\phi}_i, \boldsymbol{\phi}_j) = \sum_{k \in \mathcal{P}} \{1 - \cos[\Delta \boldsymbol{\Phi}_{ij}(k)], \} \tag{4.9}$$

which inherits the robustness properties of (4.4). Through basic manipulations, said distance can be rewritten as

$$\begin{aligned} d^2(\boldsymbol{\phi}_i, \boldsymbol{\phi}_j) &= \frac{1}{2} \sum_{k \in \mathcal{P}} \underbrace{\{\cos^2(\boldsymbol{\phi}_i) + \sin^2(\boldsymbol{\phi}_i) + \cos^2(\boldsymbol{\phi}_j) + \sin^2(\boldsymbol{\phi}_j) +}_{2} & (4.10) \\ & -2\sin(\boldsymbol{\phi}_i(k))\sin(\boldsymbol{\phi}_j(k)) - 2\cos(\boldsymbol{\phi}_i(k))\cos(\boldsymbol{\phi}_j(k))\} & (4.11) \\ &= \frac{1}{2} \sum_{k \in \mathcal{P}} (\cos(\boldsymbol{\phi}_i) - \cos(\boldsymbol{\phi}_j))^2 + (\sin(\boldsymbol{\phi}_i) - \sin(\boldsymbol{\phi}_j))^2 & (4.12) \\ &= \frac{1}{2} \left\| e^{j\boldsymbol{\phi}_i} - e^{j\boldsymbol{\phi}_j} \right\|^2 & (4.13) \end{aligned}$$

To recapitulate, computing the orientation of gradient for an image, and picking its complex exponential as a new representation,

$$\boldsymbol{z}_i = e^{j\boldsymbol{\phi}_i}, \tag{4.14}$$

provides a set of features for which the typical l_2-distance is robust with respect to outliers in the original domain, i.e. the domain of pixel values.

Furthermore, PCA can be carried out in the domain of the transformed data $\boldsymbol{z}_i$ by looking for a set of orthonormal vectors $\boldsymbol{U} = [\boldsymbol{u}_1 | \ldots | \boldsymbol{u}_K]$ which minimizes

$$\| \boldsymbol{Z} - \boldsymbol{U}\boldsymbol{U}^* \boldsymbol{Z} \|_F^2 , \tag{4.15}$$

where $\boldsymbol{U}^*$ denotes the conjugate transpose of $\boldsymbol{U}$ and $\boldsymbol{Z}$ is the matrix with the available measurements in its columns . This procedure is connected to the version of robust PCA introduced in [30] and yields a subspace of small dimensionality onto which to project the data while retaining most of its "real" variance, and disregarding the variance deriving from outliers. Without going into excessive detail, this desirable property is inherited from the norm minimized in (4.15), computed in the gradient orientation domain, which we have shown to be stable in the presence of corrupted pixel values.

4.3 Simulations

We evaluate the proposed model on a set of experiments comprising synthetic and real data.

4.3.1 Synthetic data

As a motivational synthetic experiment, we create some artificial data according to model (3.1)-(3.6). The data, 400-dimensional, is folded into 20 x 20 pixel images for an easier visualization. The real values for the factor matrices $\boldsymbol{F}$ and $\boldsymbol{G}$, depicted in the first row

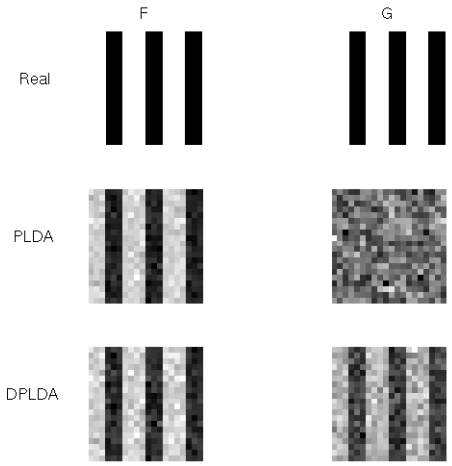

Figure 4.2: Private and shared factor matrices learnt by PLDA and DPLDA compared with the original values.

of figure 4.2, are a series of ones and zeros alternated, so as to create eight strips in the resulting images. Note that in this specific case $\boldsymbol{F} = \boldsymbol{G}$. For simplicity, the observation noise process is made isotropic with covariance matrix $\boldsymbol{\Sigma} = \sigma \boldsymbol{I}$. The matrix $\boldsymbol{A}$, associated with the linear evolution of the within-individual latent variable, is set to $\boldsymbol{A} = 0.98$, mimicking the smooth variations encountered in real videos.

We then assess the performance of PLDA and DPLDA by learning the necessary parameters for both models, and consequently exploit them for recognition. As argued at the beginning of section 3.1, DPLDA will enforce a proximity constraint on subsequent estimates of the within-individual latent variable $\boldsymbol{w}_{ij}^{t-1}$, $\boldsymbol{w}_{ij}^{t}$, due to the dynamical model at its core. PLDA, on the other hand is a static model, and as such it won't take into account any temporal information, so that $\boldsymbol{w}_{ij}^{t}$ is supposed to have zero mean, instead of $\boldsymbol{A}\boldsymbol{w}_{ij}^{t-1}$. As a result, we expect DPLDA to have a better performance in terms of parameter reconstruction, and consequently of recognition performance.

In our synthetic experiment we observe that, if the observation variance σ is kept low, both models learn matrices $\boldsymbol{F}$ and $\boldsymbol{G}$ that are very similar to the original versions, consequently performing well in recognition. As the variance increases, however, PLDA's performance worsens quickly, until the estimate for matrix $\boldsymbol{G}$ deteriorates completely, as depicted in the second row of figure 4.2. The model thus fails to ascribe the observations to either private or shared latent space in the right proportion, which in turn causes a drop in recognition performance. DPLDA, on the other hand, when dealing with the same extent of noise, still manages to reconstruct successfully both factor matrices and retains a good accuracy, as depicted in the third row of figure 4.2.

4.3.2 Real data

Verification

As a first experiment with real data, we compare the performance of classical LDA, PLDA and DPLDA in a verification task on the YouTube Faces database [35]. The data consists of 3,425 videos of 1,595 different people, all of which were downloaded from Youtube. An average of 2.15 videos are available per subject (ranging from 1 to 6) with a mean duration

Method	Accuracy ± SE
LDA	0.723 ± 0.0054
PLDA	0.830 ± 0.0091
DPLDA	0.845 ± 0.0065

Table 4.1: YTF verification. Average accuracy and standard error.

of 181 frames.

A benchmark protocol is defined for verification tasks within the database. More precisely, 5000 pairs of videos were selected, half of them depicting the same person, the other half depicting different people. The pairs are further divided into 10 splits, onto which verification has to be performed separately, exploiting the information from the other splits. The restricted protocol only allows access to this information, whereas the unrestricted protocol allows to incorporate information about the identity of the subjects during the training procedure. Since we're testing class-based methods derived from LDA, we resort to the latter protocol.

We select all the people with 4 or more videos and exploit them for training. However the training set is made split-dependent by erasing from it all videos of all the people that appear in any of the 500 pairs of the split itself. This way we can carry out a more significant test, less prone to overfitting and more indicative of the real performance achieved by each method. Typically the training set for each split consists of roughly 200 different identities, with 4-6 videos each. Again, none of the people depicted in the training set used for a split appear in any video of the respective split.

We consider the first 60 frames for the videos in the database; for each frame we obtain a frontalized version through the FAR algorithm. Consequently, we crop the image from the eyebrows to the lower lip with a bounding box of 65 x 61 pixels. We then perform whitened IGO-PCA on this rectangle of pixel values, reducing dimensionality to 1000 features, which proved to be the lowest size to yield good performance on this database.

As a final step of the verification pipeline we interchange LDA, PLDA and DPLDA, keeping all the other steps identical. For LDA, we compute the distance between all the frames of the first video in a pair from all the frames in the second video of the same pair. Their average is regarded as a video-to-video distance. When considering a specific split, we incorporate the same/not-same information from the remaining 9 splits to learn a linear distance-based classifier, which we then apply to the distances in the current split, determining the final confirm/reject choice.

For PLDA we adopt a similar fusion metrics, applying it to both the sum of the likelihoods of two frames taken separately − one from video 1, one from video 2 − and to the joint likelihood under the hypothesis that they share the same identity. We take their difference as a sufficient statistics for which we learn a confirm/reject threshold, based on the remaining 9 splits, and apply it to the current one.

DPLDA was applied in its most basic version, with $\boldsymbol{A}_{ij} = \boldsymbol{A}$, i.e. estimating a single state evolution matrix supposed valid for every video. Thresholding is performed analogously to PLDA and no score fusion is necessary since the algorithm acts directly on whole videos instead of single frames. Separate results are computed for each split, and finally the 10 verification rates are averaged. We repeated this experiment 5 times, and the mean values are reported in table 4.1.

Note that both DPLDA and PLDA perform significantly better than simple LDA. Also, despite the simplification $\boldsymbol{A}_{ij} = \boldsymbol{A}$, DPLDA outperforms PLDA by more than 1%. This is

a first important confirmation that explicitly modelling the dynamics for the private latent variable in video sequences, even with a simple linear evolution dependent on a matrix shared among all the videos, can improve the discriminatory power of PLDA models in a significant and challenging setting.

Identification

We assembled an "in-the-wild" database consisting of 250 videos of 50 people, each depicted in exactly 5 videos. The videos, featuring famous people not present in the YTF database, were downloaded from Youtube at 24fps in medium quality. For each video we manually checked the identity of the person depicted and the quality of the whole sequence, in order to guarantee the presence of significant temporal information to exploit, and avoid still-image slide-shows. Exploiting the pipeline described above, we test PLDA and DPLDA for closed-set and open-set identification on this database.

5 analogous closed-set identification tasks are carried out by picking one of the 5 videos for each person to incorporate in the probe and utilizing the remaining 4 for training. Fixing the size of the shared subspace to 50, we let the size of the private subspace vary from 1 to 50. The 5 tests are repeated 3 times each, with the average results depicted in figure 4.3. For $\mathrm{Dim}(\boldsymbol{G}) < 10$, PDLA performs better than DPLDA, however the latter exhibits a steeper performance increase as $\mathrm{Dim}(G)$ becomes larger. For $\mathrm{Dim}(\boldsymbol{G}) > 15$, DPLDA consistently outperforms PLDA by roughly 2%.

For open set recognition, we keep the same setting, simply adding a variable number of external videos to the probe. For convenience we picked them from the YFT database, with which our database has no overlap whatsoever. As a performance measurement we pick the generalized identification accuracy, by simply contemplating an additional external class for which no training data is available. We test for different levels of "openness" by letting the percentage of external videos in the probe vary from 17 % to 66 %. For every probe video, we compute the conditional likelihood of it sharing its identity with all the individuals in the gallery, storing the maximum value, along with the marginal likelihood of the probe video alone. The probe video is labelled as external if the difference between conditional and marginal likelihoods is smaller than a specific threshold, otherwise it is given the label that maximizes the conditional likelihood. No specific threshold is learnt, rather we depict the accuracy rate as a function of the threshold values for both PLDA and DPLDA in figure 4.3. Note that the x axis scale has been normalized so that its extrema actually represent the minimum and maximum values measured for the difference between conditional and marginal likelihood on any probe video. In other words, the graphs are scaled and aligned so that outside the depicted threshold values they are completely flat.

When the threshold is at a minimum, all videos are labelled as internal, hence the advantage of DPLDA due to better performance in closed-set identification. As the threshold increases, more and more videos are rejected as external, until all of them are when the normalized threshold is equal to 1. To this extremum, PLDA and DPLDA have an identical performance in terms of accuracy, which is equal to the percentage of external videos in the probe. For intermediate threshold values, the advantage of DPLDA over PLDA in terms of accuracy is quite evident, thus confirming again the suitability of the proposed method for different task of identity inference in video sequences.

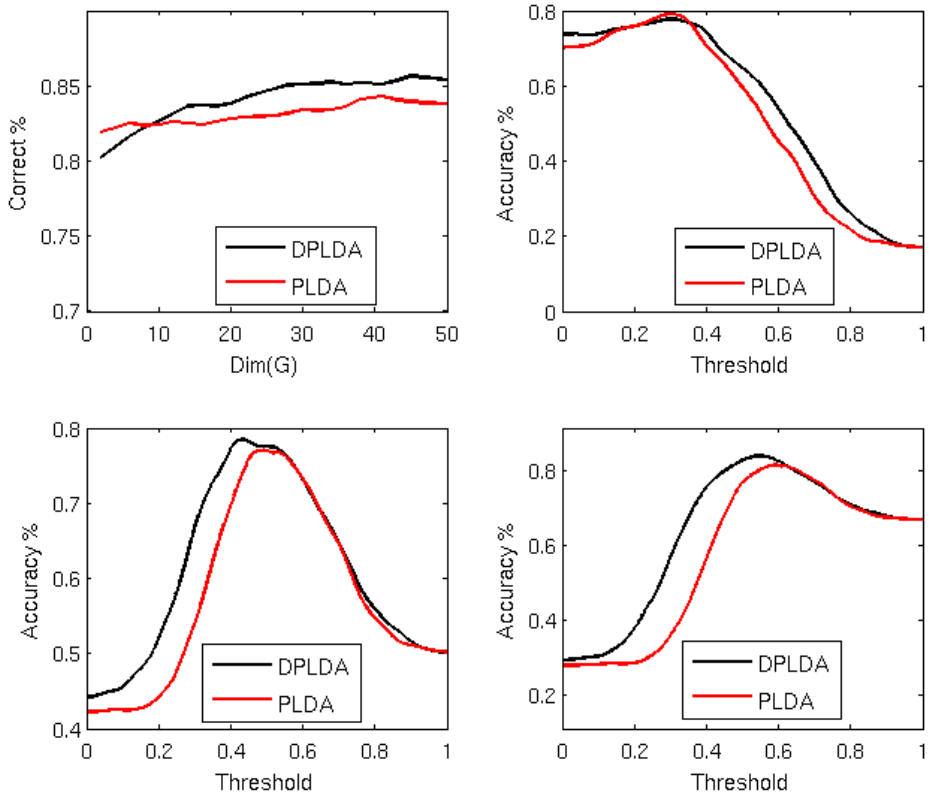

Figure 4.3: Closed- and Open-set identification on our database. (Top-left) Comparison of Closed-set identification accuracy as the private latent space dimensionality increases. (Top-right), (Bottom-left), (Bottom-right) Comparison of Open-set identification accuracy dependent on threshold value. The percentage of external videos are 17%, 50% and 66% respectively.

5. Conclusion

An increasing interest has been devoted to face recognition in the last few decades, starting from tasks on still images, progressively generalized to video sequences. Despite the maturity of this technology, there is still a variety of applications susceptible to improvement. Challenges arise especially in the context of non-cooperative video recording, such as ubiquitous CCTV cameras, whereby one or more factors such as quality, pose, illumination, occlusions and duration are uncontrolled.

In this document we focused our attention on in-the-wild databases with uncontrolled acquisition. We introduced a 3-stage pipeline featuring frontalization, feature extraction and recognition to handle this challenging setting. For the recognition stage we chose PLDA for its inherent flexibility, which allowed us to handle the classical tasks of verification and closed-set identification, but also clustering and open-set identification. The proposed recognition pipeline achieved a good performance on the YTF database and on a privately assembled one, only exploiting the information in the first 3 seconds of each video.

Furthermore we leveraged a sequence-based, spatio-temporal approach to propose the first dynamical version of PLDA. We outlined its theoretical advantage over classical PLDA and proved it in three tasks on two different databases, both featuring challenging conditions.

The systematic advantage of DPLDA emerged despite the simple setting we adopted, forcing all videos to share the same matrix for state dynamics ($\boldsymbol{A}_{ij} = \boldsymbol{A}$). A richer model, featuring different evolution matrices, is likely to improve performance even further, allowing us to take into account the peculiar dynamics of each video. However, the inversion of the innovation covariance matrices from the Kalman filters represents a bottleneck, hindering the adoption of more complex, performance-improving models. An interesting idea would be to explore approximate approaches to Kalman filtering, such as the conjugate gradient, to circumvent this issue.

A. The EM Algorithm

The EM algorithm [8] is a powerful statistical tool to perform maximum-likelihood estimates for parameters in models comprising latent variables. It was formalized in a classical paper by Dempster et al. in 1977. The authors recognized similar procedures had been proposed before, as solutions to specific statistical models, which, however, lacked a unifying framework. Notable examples are the Baum-Welch algorithm and K-means clustering.

In this appendix we complement the intuitive explanation provided in section 2.4 with a formal proof of convergence. More specifically, when looking for maximum-likelihood solutions, the aim is to maximize the likelihood of the observations, or equivalently its logarithm

$$\ln P(\boldsymbol{X}|\boldsymbol{\theta}) = \ln \left[\sum_{Z} p(\boldsymbol{X}, \boldsymbol{Z}|\boldsymbol{\theta}) \right], \tag{A.1}$$

where $\boldsymbol{Z}$ groups all the latent variables in the model. For any PDF $q(\boldsymbol{Z})$ defined over the latent variables the following decomposition holds:

$$\ln P(\boldsymbol{X}|\boldsymbol{\theta}) = \mathcal{L}(q, \boldsymbol{\theta}) + KL(q||p). \tag{A.2}$$

Here $KL(q||p)$ is the Kullback-Leibner divergence, defined as

$$KL(q||p) = \sum_{\boldsymbol{Z}} q(\boldsymbol{Z}) \ln \left\{ \frac{q(\boldsymbol{Z})}{p(\boldsymbol{Z}|\boldsymbol{X}, \boldsymbol{\theta})} \right\}, \tag{A.3}$$

which measures the difference between $q(\boldsymbol{Z})$ and $p(\boldsymbol{Z}|\boldsymbol{X}, \boldsymbol{\theta})$. As a metric for the distance between PDFs, it is always non-negative, equal to zero if an only only if $q(\boldsymbol{Z}) = p(\boldsymbol{Z}|\boldsymbol{X}, \boldsymbol{\theta})$.

$\mathcal{L}(q, \boldsymbol{\theta})$ is a functional of $q(\boldsymbol{Z})$ and a function of $\boldsymbol{\theta}$ defined as

$$\mathcal{L}(q, \boldsymbol{\theta}) = \sum_{\boldsymbol{Z}} q(\boldsymbol{Z}) \ln \left\{ \frac{p(\boldsymbol{X}, \boldsymbol{Z}|\boldsymbol{\theta})}{q(\boldsymbol{Z})} \right\}, \tag{A.4}$$

representing a lower bound on the log-likelihood. As we will show, iterative optimization of $\mathcal{L}(q, \boldsymbol{\theta})$ results in the maximization of the log-likelihood, hence in a ML solution.

In the E-step we optimize $\mathcal{L}(q, \boldsymbol{\theta}^{old})$ with respect to $q(\boldsymbol{Z})$ while holding parameter $\boldsymbol{\theta}$ fixed to a previous estimate $\boldsymbol{\theta}^{old}$. Since $\ln p(\boldsymbol{X}|\boldsymbol{\theta})$ does not depend on $q(\boldsymbol{Z})$, the largest values of $\mathcal{L}(q, \boldsymbol{\theta}^{old})$ will occur when the Kullback-Leibner divergence is at its smallest, which happens for $q(\boldsymbol{Z}) = p(\boldsymbol{Z}|\boldsymbol{X}, \boldsymbol{\theta}^{old})$.

During the M-step, $q(\boldsymbol{Z})$ is held fixed and $\mathcal{L}(q, \boldsymbol{\theta})$ is maximized with respect to $\boldsymbol{\theta}$. Unless $\mathcal{L}(q, \boldsymbol{\theta})$ has a maximum in $\boldsymbol{\theta}^{old}$, the new optimum $\boldsymbol{\theta}^{new}$ will result in an increase of $\mathcal{L}(q, \boldsymbol{\theta})$ and an even larger increase of the log-likelihood, due to the fact that $KL(q||p) = KL(p(\boldsymbol{Z}|\boldsymbol{X}, \boldsymbol{\theta}^{old})||p(\boldsymbol{Z}|\boldsymbol{X}, \boldsymbol{\theta}^{new}))$ will become positive.

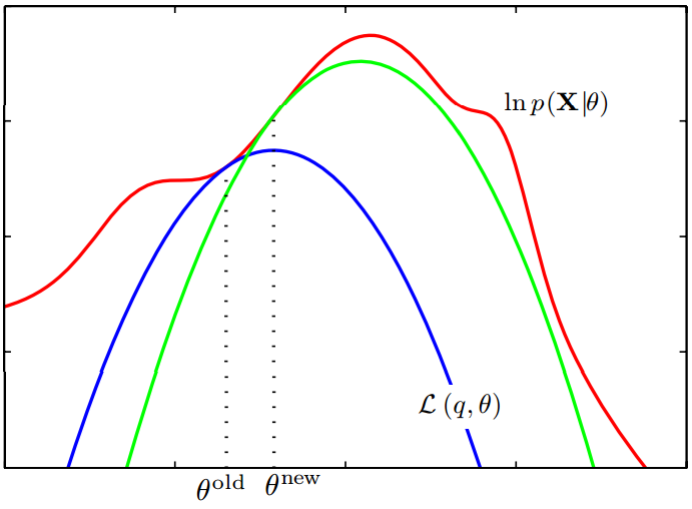

Figure A.1: Graphical depiction of the iterative optimization carried out through the EM algorithm.

Furthermore, note that the following decomposition holds

$$\begin{aligned}\mathcal{L}(q,\boldsymbol{\theta}) &= \sum_{\boldsymbol{Z}} p(\boldsymbol{Z}|\boldsymbol{X},\boldsymbol{\theta}^{old}) \ln p(\boldsymbol{X},\boldsymbol{Z}|\boldsymbol{\theta}) - \sum_{\boldsymbol{Z}} p(\boldsymbol{Z}|\boldsymbol{X},\boldsymbol{\theta}^{old}) \ln p(\boldsymbol{Z}|\boldsymbol{X},\boldsymbol{\theta}) \\ &= Q(\boldsymbol{\theta},\boldsymbol{\theta}^{old}) + \text{const}, \end{aligned} \tag{A.5}$$

where the constant is the entropy of the PDF $q(\boldsymbol{Z})$ and is independent of $\boldsymbol{\theta}$. To summarize, maximization (2.23) is equivalent to optimizing $\mathcal{L}(q,\boldsymbol{\theta})$, which, in alternation with the evaluation of $p(\boldsymbol{Z}|\boldsymbol{X},\boldsymbol{\theta})$, guarantees a continuous increase in the likelihood of the observations until convergence to a local maximum.

This process is depicted in figure A.1, taken from [6] where we suppose to start from an initial parameter value $\boldsymbol{\theta}^{old}$. The curve in red represents the log-likelihood to be maximized, the curves in blue and green is the auxiliary functional $\mathcal{L}(q,\boldsymbol{\theta})$ we optimize instead. In the first E-step, we compute $p(\boldsymbol{Z}|\boldsymbol{X},\boldsymbol{\theta})$, giving rise to $\mathcal{L}(q,\boldsymbol{\theta})$ which equals the log-likelihood at $\boldsymbol{\theta}^{old}$. The lower bound $\mathcal{L}(q,\boldsymbol{\theta})$ is then maximized with respect to $\boldsymbol{\theta}$ during the M-step, which gives a new parameter value $\boldsymbol{\theta}^{new}$, and yields an improvement in the log-likelihood. In the subsequent E-step, $p(\boldsymbol{Z}|\boldsymbol{X},\boldsymbol{\theta})$ is evaluated for the new set of parameters $\boldsymbol{\theta}^{new}$, again providing a convex lower bound represented by the green curve.

Acknowledgments

I am pleased to thank my official supervisor from Imperial College, Stefanos Zafeirou, and my unofficial supervisor, Mihalis Nicolau, for their patience, their help, their guidance.

Nice things should not be left unspoken, the following should be uttered in Italian.

Ringrazio il mio relatore patavino, Ruggero Carli, per la sua disponibilità e i suoi preziosi consigli.

Un sentito ringraziamento va ai miei amici e conterranei Giorgia, Irene, Ama, l'Ale, Prince, Gaspa, Chicco, Checco, Cant. Grazie per esserci stati, perché ci siete e ci sarete. Grazie perché rendete casa un posto meraviglioso a cui fare ritorno.

Un grazie ai *Sattoni* perché cambiano un po' ogni anno, ma in fondo non cambiano mai. Grazie per le risate, le pallavolate, per la quiete e le monete. È vero.

Ringrazio Paola, che mi ha cresciuto, educato, amato. Grazie per avermi trattato come un figlio con discrezione ed affetto ineguagliabili.

Un grazie immenso va a mio padre, Ennio, mio primo sostenitore da sempre. Grazie per tutto quello che hai fatto, grazie per gli abbracci, per i consigli, per le sberle meritate. Grazie per aver fatto del tuo meglio per insegnarmi a stare al mondo; il risultato, colui che scrive, spero non sia così male.

Da ultimo ringrazio mia sorella, Olga, che mi ha preso per mano nel giorno della mia nascita e da allora non ha mai lasciato andare. Grazie perché sei un modello, una madre, una sorellina, un'amica. Grazie perché in me c'è così tanto di te, che la mia vita senza te proprio non riesco a immaginarla.

Bibliography

[1] M. Al-Azzeh, A. Eleyan, and H. Demirel. Pca-based face recognition from video using super-resolution. *Proc. 2008 International Symposium on Computer and Information Sciences*, pages 1 – 4, 2008.

[2] O. Arandjelović and R. Cipolla. Probabilistic models for inference about identity. *Image and Vision Computing*, 24:639 – 647, 2006.

[3] J. R. Barr, K. W. B. P. J. FLynn, and S. Biswas. Face recognition from video: a review. *International Journal of Pattern Recognition and Artificial Intelligence*, 2012.

[4] L. Benedikt, D. Cosker, P. Rosin, and D. Marshall. 3d facial gestures in biometrics: from feasibility study to application. *Proc. 2008 IEEE International Conference on Biometrics: Theory, Applications and Systems*, 34:1 – 6, 2008.

[5] S. Berrani and C. Garcia. Enhancing face recognition from video sequences using robust statistics. *Proc. 2005 IEEE Conference on Advanced Video and Signal Based Surveillance*, 34:324 —- 329.

[6] C. Bishop. *Pattern Recognition and Machine Learning*. Information Science and Statististics. Springer, 2006.

[7] L. Chen, H. Liao, and J. Lin. Person identification using facial motion. *Proc. 2001 International Conference on Image Processing*, 2:677 – 680, 2001.

[8] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39(1):1 – 38, 1977.

[9] R. A. Fisher. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7(2):179 – 188, 1936. ISSN 2050-1439. doi: 10.1111/j.1469-1809.1936.tb02137.x. URL `http://dx.doi.org/10.1111/j.1469-1809.1936.tb02137.x`.

[10] A. C. G. Aggarwal and R. Chellappa. A system identification approach for video-based face recognition. *Proc. 2004 International Conference on Pattern Recognition*, 4:175 – 178, 2004.

[11] D. Gorodnichy. Video-based framework for face recognition in video. *Proc. 2005 Canadian Conference on Computer and Robot Vision*, 34:330 – 338, 2005.

[12] B. Gunturk, A. Batur, Y. Altunbasak, M. Hayes, and R. Mersereau. Eigenfacedomain super-resolution for face recognition. *IEEE Transactions on Image Processing*, pages 587 – 606, 2003.

[13] A. Hadid and M. Pietikäinen. Selecting models from videos for appearance-based face recognition. *Proc. 17th International Conference on Pattern Recognition*, 1:304 – 308, 2004.

[14] A. Hadid and M. Pietikäinen. Combining appearance and motion for face and gender recognition from videos. *IEEE Transactions on Pattern Recognition and Machine Intelligence*, 42:2818 – 2827, 2009.

[15] R. R. Jillela and A. Ross. Adaptive frame selection for improved face recognition in low-resolution videos. *Proc. 2009 International Joint Conference on Neural Networks*, pages 2835 – 2841, 2009.

[16] P. Li, Y. Fu, U. Mohammed, J. H. Elder, and S. J. Price. Probabilistic models for inference about identity. *IEEE Transactions on Pattern Recognition and Machine Intelligence*, 34:144 – 157, 2012.

[17] Y. Li, S. Gong, and H. Liddell. Constructing facial identity surfaces for recognition. *International Journal of Computer Vision*, 53:71 – 92, 2003.

[18] Lina, T. Takahash, I. Ide, and H. Murase. Incremental unsupervised-learning of appearance manifold with view-dependent covariance matrix for face recognition from video sequences. *IEICE Transactions on Information and Systems*, 34:642 – 652, 2009.

[19] X. Liu and T. Chen. Face mosaicing for pose robust video-based recognition. *Proc. of the 8th Asian Conference on Computer Vision*, pages 662 – 671, 2007.

[20] X. Liu and T. Cheng. Video-based face recognition using adaptive hidden markov models. *Proc. 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1:340 – 345, 2003.

[21] F. Matta and J.-L. Dugelay. Video face recognition: A physiological and behavioural multimodal approach. *Proc. 2007 IEEE International Conference on Image Processing*, pages 497 – 500, 2007.

[22] U. Park, H. Chen, and A. Jain. 3d model-assisted face recognition in video. *Proc. 2005 Canadian Conference on Computer and Robot Vision*, pages 322 – 329, 2005.

[23] A. Ramanathan, R. Chellappa, and S. Biswas. Computational methods for modeling facial aging: A survey. *Journal of Visual Languages and Computing*, 20:131 – 144, 2009.

[24] M. Ribeiro. Kalman and extended Kalman filters: Concept, derivation and properties. Technical report., Institute for Systems and Robotics, Lisboa, 2004.

[25] C. Sagonas, Y. Panagakis, S. Zafeiriou, and M. Pantic. Face frontalization for alignment and recognition. (to appear).

[26] L. Sirovich and M. Kirby. Low-dimensional procedure for characterization of human faces. *Journal of Optical Society of America*, 3:519 – 524, 1987.

[27] J. Stallkamp, H. K. Ekenel, and R. Stiefelhagen. Video-based face recognition on real-world data. *Proc. 2007 IEEE International Conference on Computer Vision*, 206:1 – 8.

[28] D. Thomas, K. W. Bowyer, and P. J. Flynn. Multi-frame approaches to improve face recognition. *Proc. 2007 IEEE Workshop on Motion and Video Computing*, page 19, 2007.

[29] M. E. Tipping and C. M. Bishop. Probabilistic principal component analysis. *Journal of the Royal Statistical Society, Series B*, 61:611 – 622, 1999.

[30] F. D. L. Torre and M. Black. A framework for robust subspace learning. *IJCV 54*, pages 117 – 142, 2003.

[31] G. Tzimiropoulos, S. Zafeiriou, and M. Pantic. Subspace learning from image gradient orientations. *IEEE TPAMI*, (to appear).

[32] R. Wang, S. Shan, X. Chen, and W. Gao. Manifold-manifold distance with application to face recognition based on image set. *Proc. 2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1 – 8, 2008.

[33] M. E. Wibowo, D. Tjondronegoro, and V. Chandran. Probabilistic matching of image sets for video-based face recgonition. *Digital Image Computing Techniques and Applications (DICTA)*, pages 1 – 6, 2012.

[34] M. E. Wibowo, D. Tjondronegoro, L. Zhang, and I. Himawan. Heteroscedastic probabilistic linear discriminant analysis for manifold learning in video-based face recognition. *Applications of Computer Vision*, pages 46 – 52, 2013.

[35] L. Wolf, T. Hassner, and I. Maoz. Face recognition in unconstrained videos with matched background similarity. *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[36] Q. Xiong and C. Jaynes. Mugshot database acquisition in video surveillance networks using incremental auto-clustering quality measures. *Proc. 2003 IEEE Conference on Advanced Video and Signal Based Surveillance*, page 191.

[37] O. Yamaguchi, K. Fukui, and K. Maeda. Face recognition using temporal image sequence. *Proc. 1998 IEEE International Conference on Automatic Face and Gesture Recognition*, pages 318 – 323, 1998.

[38] Y. Zhang and A. M. Martinez. A weighted probabilistic approach to face recognition from multiple images and video sequences. *Image and Vision Computing*, 24:626 — 638, 2006.

[39] X. Zhou and B. Bhanu. Human recognition based on face profiles in video. *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, page 15, 2005.