Latent Max-margin Metric Learning for Comparing Video Face Tubes

Gaurav Sharma*
MPI Informatics, Germany
gaurav.sharma@mpi-inf.mpg.de

Patrick Pérez
Technicolor
patrick.perez@technicolor.com

Abstract

Comparing "face tubes" is a key component of modern systems for face biometrics based video analysis and annotation. We present a novel algorithm to learn a distance metric between such spatio-temporal face tubes in videos. The main novelty in the algorithm is based on incorporation of latent variables in a max-margin metric learning framework. The latent formulation allows us to model, and learn metrics to compare faces under different challenging variations in pose, expressions and lighting. We propose a novel dataset named TV Series Face Tubes (TSFT) for evaluating the task. The dataset is collected from 12 different episodes of 8 popular TV series and has 94 subjects with 569 manually annotated face tracks in total. We show quantitatively how incorporating latent variables in max-margin metric learning leads to improvement of current state-of-the-art metric learning methods for the two cases when the testing is done with subjects that were seen during training and when the test subjects were not seen at all during training. We also give results on a challenging benchmark dataset: YouTube faces, and place our algorithm in context w.r.t. existing methods.

1. Introduction

Automatic analysis of faces in digital images and videos is a very important biometrics problem and has attracted much attention in the computer vision community [1, 8, 11, 12, 17, 18, 21, 20, 25, 22, 24, 27, 32, 33, 37, 41, 44]. It has many important applications in recognizing, searching, retrieving and indexing images, including: (i) Surveillance and video archives – find a person in large amounts of videos; (ii) Security – allow access to a person, or not, to a resource; (iii) Consumer databases – find a certain person in private or online image databases like Flickr or Facebook.

Recently, *face verification*, *i.e.* determining if two faces are of the same person or not, has emerged as an important research problem, *e.g.* [12, 18, 20, 27, 32, 33]. The task is set in a supervised learning framework where an annotated set, containing pairs of face images (i) of the same person (taken at different times and conditions) and (ii) of different persons, is provided and the system has to predict if a new test pair (of unseen faces/person(s)) is of the same person or not. Current approaches address this under a metric learning framework [3, 14, 15, 39, 46] (see [5] for a good survey), where a parametrized distance metric function is learnt for comparing face images [18, 27, 33] (with the benchmark Labeled Faces in the Wild (LFW) [20] being a catalyst). Along with such verification task, the learnt distance function can also be used for other applications like identity based linking and clustering of faces.

Similar task of verification but with spatio-temporal tracks of faces in videos, a.k.a. *face tubes*, has also seen some interest, *e.g.* [8, 12, 30, 36, 41, 45]. The task with realistic videos is much more challenging as, while in still

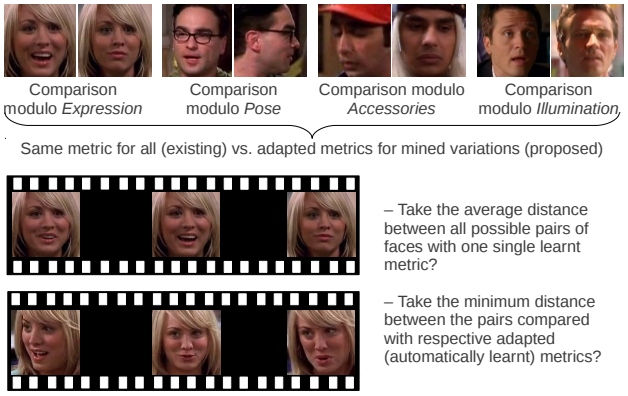

Figure 1. In the context of the important face biometrics based problem of face tube comparison, the traditional methods learn one metric/projection for all types of face appearance variations *e.g.* expression, pose, illumination, and complex real-world combinations thereof (top row). We propose to learn automatically, with a latent variable based formulation, different projections for comparing different combinations of mined variations. As a related question, we investigate how taking an average of distances, using a single metric (traditional methods), between all possible pairs of faces from the two face tubes, compares with taking the minimum distance between faces compared using proposed, variations adapted, metrics.

*GS was with Technicolor where majority of this work was done

image datasets (*e.g.* LFW [20]) the faces are near frontal with good illumination and similar/constrained expressions, the faces in videos have unconstrained pose and expression variations and are taken in diverse and difficult illumination conditions (see Fig. 2). Cinbis *et al.* [12] recently applied the distance metric learned from static face pairs to face tube matching showing that a cast-specific metric performs better than a metric learned from external data. While they addressed the case of unconstrained consumer videos, (i) they used frontal face detectors to construct the evaluation dataset and hence were limited to near frontal poses and (ii) they used static face metric for learning and hence ignored the information that comes from the natural association of faces in face tubes (see §3 for detailed discussion). A recent evaluation [8] of off-the-shelf face matchers shows that the faces with challenging poses, expressions and illuminations are the failure cases of such systems (Fig. 5 in [8]) .

In the present paper, we are interested in the difficult and important task of distance metric learning for face tubes under challenging realistic variations in pose, expression, illumination conditions, *etc*. Note that this is different from face recognition; we do not aim at learning person specific characteristics but at learning what makes faces similar or different. We propose a metric learning framework that incorporates and leverages factorization over such variations (and, more realistically, their complex combinations) in the metric as latent variables. This can also be viewed as learning separate metrics for specific combinations of variations. This is in contrast to the existing works, which learn same metric for any and all combinations of variations. Fig. 1 illustrates the point. We show the proposed latent metric can be learnt using an efficient stochastic gradient descent based algorithm. We provide experimental results on the challenging benchmark: YouTube Faces [41], to put our method in the context of existing works on the task of face verification. In addition, we propose a new challenging dataset – TV Series Face Tubes (TSFT) – to evaluate the difficult task of face tube matching in videos with high facial variability such as TV series. We use this dataset to evaluate our method for comparing new face tubes of unseen persons along with the standard *cast specific* setting, *i.e.* comparing face tubes of people already seen at training. Our results on TSFT dataset show that while face verification in static near-frontal faces can be done quite successfully (*e.g.* on LFW[1]), the task of face tubes matching in videos with high facial variability is quite challenging, with room for much progress. We show with quantitative results that the method gives better performance, on TSFT dataset, than strong baselines and the state-of-the-art methods. We find expression variations to be particularly misleading for the system; this problem is amplified in the case of hard datasets with high variability, like TSFT, as the probability of finding another person with a similar expression is very high. As collecting a dataset for sufficiently capturing identity dependent facial expression variations for verification is a very challenging task, we thus identify *expression normalization* as an intriguing problem for face verification.

We now set the context for the work in the next section. We then give the details of the approach (§3) and the proposed dataset (§4), followed by experimental evaluation (§5) and discussions (§6).

[1] http://vis-www.cs.umass.edu/lfw/results.html

2. Related Works and Background

Face verification with still images is a very popular topic of research in computer vision, with the very popular Labeled Faces in the Wild (LFW) benchmark [20] specially catalysing such research. Many recent papers address the problem with novel approaches, *e.g.* Berg and Belhumeur [7] propose a discriminative part based approach, Li *et al.* [26] propose a probabilistic elastic model, Simonyan *et al.* [33] use Fisher vectors [31] with metric learning, Cao *et al.* [10] propose a novel regularization for similarity metric learning, Cui *et al.* [13] propose to fuse many descriptors using multiple metric learning, Sun *et al.* [35] use deep learning, Barkan *et al.* [4] propose a method that uses fast high dimensional vector multiplication and Weng *et al.* [40] use robust feature set matching for partial face recognition. Many of the most competitive approaches on LFW combine many features, *e.g.* [18, 28, 43] and/or use external data, *e.g.* [6, 23]

Metric learning has recently gained much interest, as in [3, 14, 15, 39, 46] (we refer the reader to Bellet *et al.* [5] for an excellent survey) and it has been applied for comparing faces [10, 18, 27, 33] with simple extensions to comparing face tubes [12, 30] as well. Given two face descriptors $\mathbf{v}_i$ and $\mathbf{v}_j$ the task is formulated as learning a Mahalanobis like metric of the form $\tilde{D}^2_M(\mathbf{v}_i, \mathbf{v}_j) = (\mathbf{v}_i - \mathbf{v}_j)^\top M(\mathbf{v}_i - \mathbf{v}_j)$, parametrized by the symmetric positive semi-definite (PSD) matrix M. Various objectives have been proposed to learn M (see [5] for a survey). Since maintaining M as PSD is usually computationally expensive, M is often factorized as $M = L^\top L$. Then the problem can be seen as that of finding a linear subspace, spanned by the rows of L, into which the features are embedded and compared as

$$\tilde{D}^2_M(\mathbf{v}_i, \mathbf{v}_j) = (\mathbf{v}_i - \mathbf{v}_j)^\top L^\top L(\mathbf{v}_i - \mathbf{v}_j) = \|L\mathbf{v}_i - L\mathbf{v}_j\|_2^2. \quad (1)$$

Such formulation of metric learning has been shown to be effective in comparing faces under lighting and expression variations[12, 18, 27]. Although the method is for comparing pairs of face images, it has been used in [12] for comparing face tubes in videos by taking the minimum or average distances between all quadratic number of face pairs for the two tubes. While this was shown to give good results, it might not be optimal, as we discuss next. In a more

recent work [30], it was shown that using strong features the face tube can be represented as a single descriptor vector (*vs*. one vector per face) achieving competitive results, when combined with existing metric learning methods.

Our work is also related to local metric learning methods notably [9] (we encourage the reader to see the references in [9] for more context on local metric learning). They propose to learn metrics by weighting a set of 'basis' metrics based on the location of the test pair vectors. They cluster the data projected into a low dimensional space, and are hence largely generative (although the low dimensional projection is learnt discriminatively). In the proposed method, the metrics and the implicit 'clustering' are automatically learnt in the same optimization.

3. Approach

Motivation. The traditional approach of using face pair metric learning with mininum or average face distances over the face tubes might not be optimal as: (i) A single projection matrix L is used for faces irrespective of their, *i.e*. pose, expression, lighting, *etc.*, which means that faces with different variations are embedded into the same space for comparison; (ii) The learning is not done on face tubes with similar minimum or average distances, but rather on sampled positive and negative pairs of faces, which discards the information present by virtue of different faces being grouped into different tubes; (iii) While some degree of invariance against pose is achieved by aligning the faces [44] or computing descriptors around face landmark detections[12], invariance to other types of variations, especially expression, is not immediate. Even with pose, while current strategies work for near frontal faces (many standard face databases including LFW are constructed by using frontal face detectors), they might, however (a) add errors in facial landmark detections for harder non-frontal faces, which might then propagate to the later parts of the pipeline; (b) still not be optimal to compare aligned frontal images with aligned profile images directly and with the same projection matrix.

Such arguments motivate us to model variations which may be due to (combinations of) pose, illumination, expression, *etc*. as latent variables in metric learning framework which we detail in the following.

Proposed Method. Given spatio-temporal face tubes, which may be obtained by using face detection and/or tracking technologies, we are interested in learning a distance function for comparing them. Denoting $\mathbf{s} = [\mathbf{s}_{1:N_s}] \in \mathbb{R}^{d\times N_s}$ a face tube (with N_s faces, each represented with d dimensional feature), we propose to compare two face tubes with following distance:

$$D^2(\mathbf{s},\mathbf{t}) = \min_{(\ell,p,q)} \|L_\ell(\mathbf{s}_p - \mathbf{t}_q)\|_2^2, \tag{2}$$

Algorithm 1. SGD based learning algorithm.

Input: Annotated training pairs $\mathcal{T} = \{(\mathbf{s},\mathbf{t},y_{st})\}$, rate ($r$) and number of stochastic updates (`niters`)
Initialize: $b, \{L_\ell\}_{\ell=1}^k$
for $i = 1,\ldots,$`niters` **do**
  Randomly sample a training pair $(\mathbf{s},\mathbf{t})$ from $\mathcal{T}$
  Randomly sub-sample each tube to length (up to) m
  Compute (ℓ,p,q) in (2) over this pair of tubes
  $L_\ell \leftarrow L_\ell - r\nabla_{L_\ell}\mathcal{L}_{st}$
  $b \leftarrow b - r\nabla_b\mathcal{L}_{st}$
end for
Output: $b, \{L_\ell\}_{\ell=1}^k$

where, $(\ell,p,q) \in [\![1,k]\!]\times[\![1,N_s]\!]\times[\![1,N_t]\!]$ are latent variables. Matrix $L_\ell \in \mathbb{R}^{d'\times d}, d' \ll d$, defines the linear embedding associated to a specific combination of facial variations and p,q specify the frames in the corresponding face tubes with features $\mathbf{s}_p$ and $\mathbf{t}_q \in \mathbb{R}^d$. Each L_ℓ can be thought of as a projection in a subspace where a certain combination of face-variations may be compared *e.g*. smiling-surprised, frontal-profile.

Given set $\mathcal{T} = \{(\mathbf{s},\mathbf{t},y_{st})\}$ of annotated tube pairs, with $y_{st} = 1$ for face tubes of same person and $y_{st} = -1$ of different persons, we learn the projection matrices L_ℓ's by minimizing the hinge loss

$$\mathcal{L} = \sum_{\mathcal{T}} \underbrace{\max[0, 1 - y_{st}(b - D^2(\mathbf{s},\mathbf{t}))]}_{\mathcal{L}_{st}}, \tag{3}$$

w.r.t. b and $\{L_\ell\}_{\ell=1}^k$.

We perform the optimization with a stochastic gradient descent algorithm (Alg. 1) using the subgradients w.r.t. L_ℓ,

$$\nabla_{L_\ell}\mathcal{L}_{st} = \begin{cases} 0, & \text{if } y_{st}(b - D^2(\mathbf{s},\mathbf{t})) > 1 \\ 2y_{st}L_\ell(\mathbf{s}_p - \mathbf{t}_q)(\mathbf{s}_p - \mathbf{t}_q)^\top, & \text{otherwise,} \end{cases} \tag{4}$$

where the latent variables (p,q,ℓ) are obtained for the current pair $(\mathbf{s},\mathbf{t})$ using Eq. 2, and that w.r.t. b,

$$\nabla_b\mathcal{L}_{st} = \begin{cases} 0, & \text{if } y_{st}(b - D^2(\mathbf{s},\mathbf{t})) > 1 \\ -y_{st} & \text{otherwise.} \end{cases} \tag{5}$$

We note here, that we do not specify the kind of variations and related clustering (for different ℓ's) we want to have–there are no explicit variation level annotations such as 'these people are smiling' or 'are in profile pose'. The factorizations over the variations is thus learnt automatically within a discriminative learning framework.

The lack of explicit regularization in the optimization objective (loss in Eq. 3) is compensated by (i) fixing the dimension of the projected space (d'), thereby limiting the rank of the learned metric [12, 27, 30] and (ii) a combination of low learning rate and fixed number of iterations (inspired by the experiments with SGD for classification [2]).

While doing the stochastic updates, to generate much larger number of training points and to get smoother estimates of the projection matrices, at each stochastic step we not only randomly sample an annotated tube pair but also sample (up to) a fixed number (m) of images from each of the tubes. This allows us to construct many more virtual training tube pairs (especially positive pairs which are usually relatively few), which is important to help the algorithm update often all of the L_ℓ matrices and provide a smoother estimation.

4. TV Series Face Tubes (TSFT) Dataset

Context. We are specifically interested in face tube matching algorithms in videos with very high variations in expression, pose, lighting, *etc*. and their combinations. Some face video datasets for evaluating face tracking and recognition exist, *e.g*. [17, 22, 24, 25, 34, 41], however they are either recorded with coorperative subjects in limited background/lighting variations or are based on near-frontal faces only. Also, many of them are generated automatically (with face detector) with a post processing step for eliminating duplicates and false positives and hence are dependent on and are limited by the statistics of the face detector used. While Sivic *et al*. [34] worked with TV series video including profile faces as well, the task they addressed was of learning character specific classifiers. The data they made publicly available is from two episodes of a single TV series with a relatively small number of subjects.

In a recent evaluation of commercial off-the-shelf face matchers applied to videos [8], it was found that faces with extreme face pose and illumination conditions were not enrolled by available systems (Fig. 5 in [8]) – we are interested in such high variability scenarios. To the best of our knowledge, a suitable publicly available large dataset to evaluate face tube matching algorithms in the presence of challenging high variations in expressions, pose, illuminations and their complex combinations, did not exist at the time of submission of this paper.

Proposed Dataset. We propose a novel dataset of *manually annotated* face tubes in popular TV series videos–TV Series Face Tubes (TSFT) dataset[2]. The dataset captures the many challenging variations present specifically in the case of videos. Tab. 1 gives the statistics of the dataset, Fig. 2 shows faces of the subjects from the dataset. Face tracks of 94 subjects who vary in age, build, race and sex were manually annotated in 12 episodes of 8 different series. Every tenth frame of the tubes was manually marked with a bounding box covering the face of the person and the intermediate bounding boxes were linearly interpolated. The boxes were expanded to make them square and padded with black pixels when they went out of the image. In total there are more than 32,000 faces in the dataset. Fig. 3 shows the distribution of track lengths and face sizes for the dataset. The average track duration is a bit over 2.2 seconds (55 frames) and the average face size is 121×121 pixels. Each face tube has labels assigned manually for the associated character in the series and the actor playing the character. The series and the scenes within, where the tracks are marked, are of highly diverse nature, they occur indoors (*e.g*. home, office, bar) and outdoors (*e.g*. playing field, street, market) with very challenging facial expressions, head motions, hence pose changes, and lighting conditions. There are a total of 2005 positive pairs and order of 100k negative pairs. Hence the dataset is large and very challenging for studying the problem of comparing face tubes in videos.

[2]The dataset is publicly available, please contact the authors

Proposed evaluation. We propose two evaluation settings with provided training and testing splits of the dataset.

(i) *Known Persons.* This is the classic cast-specific metric [12] evaluation where there is at least one training face tube for each person present in the test set. To generate the train and test sets for this scenario, we randomly selected at least 50% face tubes of each subject in the dataset as training examples while keeping the rest as testing examples. This gave us all 94 subjects in the train set, with 780 positive and 75k negative tube pairs, and 85 subjects in the test set, with 189 positive and 19k negative tube pairs.

(ii) *Unknown Persons.* In this setting the training and testing are done on face tubes of different subjects, *i.e*. the subjects in the test set were never seen during training. To generate the train and test sets for this scenario, we randomly selected 80% of the subjects for training and kept the rest for testing. This gave us 75 subjects with 1590 positive and about 1 million negative tube pairs for training and 19 subjects with 414 positive and 6k negative tube pairs for testing.

To evaluate the performance of the methods we report the average precision (AP) for the different methods on the respective test sets. If a method involves a random component we suggest running the method 10 times and reporting the mean and standard deviation of the AP for the 10 runs.

5. Experimental Results

5.1. TV Series Face Tubes (TSFT) dataset

Image representation. Recent works on face verification [11, 21, 37] showed that local pattern features are powerful facial descriptors. We thus use local binary pattern (LBP) [1] as our base features. We extract LBP in 3×3 circular pixel neighborhoods (with the diagonal pixels bilinearly interpolated) and use the uniform LBP patterns, *i.e*. patterns with at most two bitwise transitions from 0 to 1, or vice versa, when the bit pattern is seen as circular. We extract such LBP densely at every pixel at 3 scales with face image resized to 120×120, 80×80 and 60×60 pixels. We

Figure 2. Example faces of the 94 subjects from the proposed TV Series Face Tubes (TSFT) dataset. The dataset has diverse set of subjects (age, gender, race, sex) who appear in different lighting conditions (home, office, bar, field, inside car, at day, at night) and have varied pose, expressions and motions.

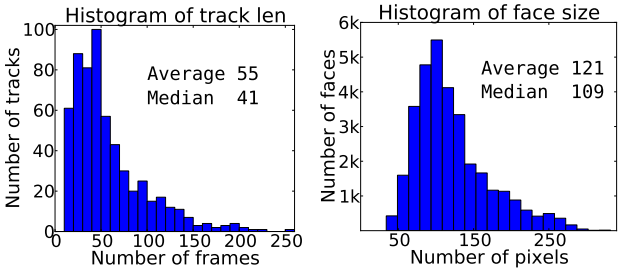

Figure 3. Histogram of track lengths and face sizes in the proposed TV Series Face Tubes (TSFT) dataset.

Sl.	Series	Sub	Tubes	S/E*	+ pairs	- pairs
1.	BBT*	10	71	1/1, 1/6	315	2871
2.	Buffy	11	69	5/2	195	2610
3.	Castle	12	97	3/1	439	5192
4.	Dexter	12	78	1/1,2/24	267	3348
5.	Homeland	11	74	1/1	269	3044
6.	Mad Men	12	70	1/2,2/13	200	2685
7.	Numb3rs	11	54	1/1	128	1613
8.	Weeds	15	76	1/1,1/6	192	3118
-	Total	94	569	-	2005	24681

Table 1. Statistics for the TV Series Face Tubes (TSFT) dataset. (*BBT = The Big Bang Theory, *S/E = Season/Episode).

make spatial histograms of the LBP features with cell size of 10×10 pixels and concatenate all the (ℓ_1 normalized) cell histograms for all the scales to give a final vector of $(12^2 + 8^2 + 6^2) \times 59 = 14,396$ dimensions for each face image.

Baseline method (PCA+CS). Recently Chen *et al*. [11] and Hussain *et al*. [21] showed that projecting the local pattern histograms on to their PCA basis and then using cosine similarity with the projected vectors is a very strong baseline, which competes with many recently proposed supervised methods [11, 21]. As a baseline: (i) We do ℓ_1 normalization followed by elementwise square root normalization of the local pattern histograms. This corresponds to the non-linear Helinger kernel map [38] and was recently shown to give very good performance for facial analysis tasks [32]. With this kernel map, the Euclidean distances between mapped vectors correspond to the Bhattacharyya distance between the probability distributions represented by the original distances. (ii) We project these high dimensional vectors onto their low rank PCA basis. Similar to Hussain *et al*. [21], we found that with such PCA based compression of vectors the performance practically stays the same upto a certain dimension and then falls as we go below. We found, with preliminary experiments, that 1000 PCA dimensions (almost similar to [21] who suggested 900) as a good operating point. (iii) Finally we follow recent works [11, 21] and use cosine similarity (CS), *i.e*. dot product divided by norms of the vectors, as the comparison measure. In practice, we ℓ_2 normalize the PCA projected vectors and then compare them using Euclidean distance, achieving the same effect.

We stress again that this is a very strong baseline [11, 21]. We denote this baseline as PCA+CS in the following.

Compared method: Supervised Metric Learning (ML). Supervised metric learning is currently one of the state-of-the-art methods for face verification tasks [18, 27, 33]. As discussed above in §2, the main idea is to learn a Mahalanobis like metric – which is itself decomposed into a product of a low rank matrix with its transpose, effectively making it a low dimensional projection learning problem. Different losses such as hinge [33] and logistic [18, 27] have been optimized giving good results.

We compare with such supervised metric learning

Known Persons					Unknown Persons				
Method	Dist.	Dim.	k	Avg. Prec.	Method	Dist.	Dim.	k	Avg. Prec.
Chance	-	14396	-	1.0	Chance	-	14396	-	6.3
PCA+CS [11, 21]	min	1000	-	11.8	PCA+CS [11, 21]	min	1000	-	32.8
	avg	1000	-	4.0		avg	1000	-	21.1
ML [12, 27, 33]	min	64	-	17.7 ± 0.3	ML [12, 27, 33]	min	64	-	36.3 ± 0.7
	avg	64	-	10.7 ± 1.1		avg	64	-	32.7 ± 1.1
ML [12, 27, 33]	min	128	-	17.0 ± 0.4	ML [12, 27, 33]	min	128	-	$\mathbf{36.4 \pm 0.1}$
	avg	128	-	6.5 ± 0.2		avg	128	-	27.9 ± 0.2
Latent ML	-	64×2	2	$\mathbf{18.3 \pm 0.4}$	Latent ML	-	64×2	2	35.4 ± 1.4
ML [12, 27, 33]	min	192	-	14.7 ± 0.7	ML [12, 27, 33]	min	192	-	34.7 ± 0.5
	avg	192	-	5.3 ± 0.3		avg	192	-	24.2 ± 0.4
Latent ML	-	64×3	3	$\mathbf{18.1 \pm 0.8}$	Latent ML	-	64×3	3	$\mathbf{36.0 \pm 1.8}$
ML [12, 27, 33]	min	256	-	13.7 ± 0.1	ML [12, 27, 33]	min	256	-	32.2 ± 0.6
	avg	256	-	4.3 ± 0.1		avg	256	-	21.7 ± 0.6
Latent ML	-	64×4	4	$\mathbf{18.1 \pm 0.7}$	Latent ML	-	64×4	4	$\mathbf{37.0 \pm 1.6}$
ML [12, 27, 33]	min	320	-	11.2 ± 0.3	ML [12, 27, 33]	min	320	-	31.8 ± 0.6
	avg	320	-	3.6 ± 0.1		avg	320	-	19.9 ± 0.6
Latent ML	-	64×5	5	$\mathbf{18.2 \pm 0.6}$	Latent ML	-	64×5	5	$\mathbf{37.5 \pm 1.4}$
ML [12, 27, 33]	min	384	-	9.7 ± 0.5	ML [12, 27, 33]	min	384	-	29.5 ± 0.6
	avg	384	-	3.2 ± 0.1		avg	384	-	17.3 ± 0.4
Latent ML	-	64×6	6	$\mathbf{18.4 \pm 0.6}$	Latent ML	-	64×6	6	$\mathbf{38.8 \pm 1.0}$

Table 2. Results of the various experiments on the TV Series Face Tubes (TSFT) dataset. See §5 for a detailed discussion.

method by optimizing the hinge loss. Essentially we optimize the proposed objective (Eq. 3) with $k = N_s = N_t = 1$. We use a stochastic gradient descent algorithm [33] with a fixed number of one million iterations generating face pairs by first randomly sampling an annotated face tube pair and then randomly sampling one face each from the two face tubes. Once we learn a metric, we compare face tubes using either minimum or average distance [12], *i.e*. take the minimum (resp. average) distance over all possible face pairs from the two tubes. We denote this method as ML in the following.

For such supervised learning methods, and similarly for the proposed metric learning with latent variables, two strategies have been used in the literature. Either, use a first compression method, like PCA, and then learn again another low dimensional projection with the PCA reduced vectors [11]. Or, learn the low dimensional projection directly with the original high-dimensional vectors [33]. We found, with preliminary experiments, that reducing the dimension using PCA and then learning the metric leads to similar performance as the second method while being faster. Hence, we follow the second approach and map the features first by PCA and then learn the metric for both proposed and compared methods.

Initialization. Methods with latent variables are generally based on non-convex optimizations and thus proper initialization is important for them. The experiments that follow require initialization for the supervised metric learning as well as our proposed latent metric learning. We follow recent work [33] for the compared supervised metric learning (ML), and initialize the projection matrix with low rank PCA basis, corresponding to the largest eigenvalues, whitened by dividing each of the PCA vectors with the square root of the corresponding eigenvalue.

We tried different strategies for initializing the projection matrices for the proposed latent metric learning. First, we did an unsupervised clustering of faces using k-means and used the clusters to initialize the projection matrices as in the case with ML above. The intuition is that the clustering will bring the faces that are similar in feature space, but (possibly) not of the same person, closer and then the projection matrices will specialize in better separating the confusing cases in each cluster. However, we found that this initialization doesn't work well in practice. Instead, we found that randomly selecting a small number (1500) of training vectors and initializing the projection matrix with the low rank whitened PCA matrix (as above for the supervised ML) of these vectors gave good results. We thus follow this initialization strategy.

Performance and comparison with existing methods. Tab. 2 gives the performances of the proposed method along with compared methods. On the *Known persons* experiments (while training and test face tubes are from different scenes/episodes all the test subjects were seen on training) the baseline of PCA projection to 1000 dimensions and then comparison with cosine similarity (CS) improves the 1.0 chance performance to 11.8 while compressing the features by 14×. Adding supervision and learning the metric by optimizing the hinge loss (ML), for differnt values of the projection dimension, improves this to up to 17.7 reducing the

#negs→	1k	2.5k	5k	10k	15k	20k
Chance	15.9	7.0	3.6	1.9	1.2	1.0
ML	53.2	40.8	28.9	25.0	18.9	17.6
LatML	**55.2**	**42.1**	**31.6**	**25.9**	**22.2**	**18.3**

Table 3. Performance of methods, chance, baseline and proposed ($k = 2$), with different number of negative test pairs.

size of the vector from 14396 to 64. We see that increasing the projection dimension for ML leads to overfitting and the performance drops above $d' = 128$. Finally, upon using the proposed latent max-margin metric learning we improve the performance to 18.4 while avoiding overfitting. For similar amount of compression (*cf.* $d' = 128$ for ML), the proposed method achieves 18.3 (with $k = 2$ and $d' = 64$). Thus the proposed addition of latent variables in the standard metric learning formulation improves the performance for the task of *Known persons* verification.

It is interesting to note that the results using the two types of tube distances, *i.e*. minimum and average (take the minimum/average distance between all possible face pairs from the two tubes) are different from recent previous works [12]. While Cinbis *et al*. [12] reported that average distance works better than the minimum distance on their dataset, we find that the opposite holds on the proposed dataset. We conjecture that this might be due to some overfitting of their method on the limited negative pairs as (i) the dataset was generated from one series only so the subjects were fewer and (ii) they used automatic negative pair generation by using co-occurrence arguments, *i.e*. face tubes which appear together are of different person, which leads to negative pairs with reduced diversity in appearances. Hence in their case they reported that some test negatives also have small distances which suggests possible overfitting on the negative set. However, in the present case the negatives are relatively much more diverse than the postives as, while the positive pairs come from the same series (albeit from different diverse scenes and from different episodes also) the negatives can come from completely different filming conditions of different series. Hence, in the present case the minimum distance works better, *i.e*. the system chooses to predict based on the best matching pair of faces for two test tubes.

In the more challenging and realistic situation of 'Unknown person' testing (no test subject was seen during training) the proposed method again shows improvements. The PCA+CS method improves the chance performance of 6.3 to 32.8 while the ML method improves it further to 36.4 ($d' = 128$). The proposed latent metric learning improves the performance to up to 38.8 (for $d' = 64$ and $k = 6$).

The performances for the two evaluation scenarios should be seen relative to the chance performances. While, for the proposed method, the chance performance improves by about 18× (from 1.0 to 18.4) in the *Known person* evaluation, the same improvement is only about 6× (6.3 to 38.8) in the *Unknown person* setting underlying the much more challenging nature of the later.

Experiments with different number of negatives. Not surprisingly, the absolute magnitude of the AP depends on the random chance performance or the relative number of positives and negatives in the test set. When we vary the number of negatives (by random subsampling, for the *Known persons* evaluation) we see that the AP decreases with increasing number of negative examples (Tab. 3). With a chance performance around the same as the *Unknown persons* evaluation, the AP rises to 42.1 (cf. 38.8 for *Unknown person*), while when we take the maximum number of negatives available (around 20k) the AP is more than 2× lower. The probability of finding another person with almost same kind of expression and other variations increases with the possible number of negatives (see qualitative results below). We stress here that in a real-world system the number of negatives will far exceed the number of positives (as the number of images of the same person will be much lower that the total number of images in the database) and hence the expected performances will be much lower. Hence, we conclude that the task of face verification, especially in the presence of high facial variations, is very challenging with a large room for improvement.

Qualitative results. Fig. 4 shows some of the top false positives for the proposed method. As seen in the figure, in video databases it is much more likely to have two different persons with very similar expressions or in very similar poses and illumination conditions which makes it much more challenging. Upon visual inspection, we conclude that usually a combination of many factors, including expression, pose and illumination, contribute to the confusion. However, the high probability of the presence of very similar expressions seems to be a recurrent reason. We also attempted to visualize the implicit clustering obtained but we didn't get easily semantically interpretable results. As the initialization is done randomly, the mined variations seemed to be non-trivial combinations of expression, pose, illumination, *etc*.

5.2. YouTube Faces (YTF) dataset

We now give results on a standard challenging benchmark of video face verification: YouTube Faces (YTF) dataset [41]. YTF dataset contains 3425 unconstrained videos of 1595 celebrities, downloaded automatically from YouTube. The benchmark provides detected and aligned faces and has a standard evaluation protocol. It is divided into 10 splits, each split containing, randomly selected, 250 positive and 250 negative pairs of face tracks. The performance is reported as the average over 10 disjoint runs, where in each run 1 fold is used for testing and the rest 9

Figure 4. Typical false positive face pairs, selected from test face tube pairs, with the proposed latent metric learning algorithm.

Method	AUC	EER
Random chance	50.0	50.0
ML (avg dist) [12, 27, 33]	85.9	22.5
ML (min dist) [12, 27, 33]	83.4	24.6
Proposed Latent ML	86.0	22.6
MBGS [41]	82.6	25.3
MBGS & SVM$\ominus$ [45]	86.9	21.2
FV (base method) [30]	-	16.2
DeepFace-single [36]	96.3	8.6

Table 4. Performance of the proposed method vs. baselines (top) and some existing method (bottom) on YTF dataset (see §5.2).

folds are used for training. We report the Area under the Receiver Operating Characteristic curve (AUC) and the Receiver Operating Characteristic Equal Error Rate (EER).

We use the three types of Local Binary Pattern (LBP) features provided by the authors *i.e.* LBP [29], Center-Symmetric LBP (CSLBP) [19] and Four-Patch LBP [42]. We follow similar setting as for the TSFT dataset and first project the concatenated features to 1000 dimensions with PCA and then learn the lower dimensional embeddings.

Tab. 4 gives the result of the baseline ML, proposed method and some existing state-of-the-art methods. The results obtained with the proposed method are better than the dataset creators' [41] results (86.0 vs 82.6 AUC) and are competitive w.r.t. one of their recent works (86.9 AUC) [45] based on a more complicated learning setup. This validates the implementation of our model w.r.t. the existing art, with similar features. State-of-the-art on this benchmark is primarily approached by engineering strong features [30] or by using large amounts of external data [36] and is hence not directly comparable to our results.

For the baseline method, the average distance performs better than minimum distance for the baseline ML (85.9 vs 83.4 AUC). The proposed method performs better than the baseline with minimum distance (86.0 vs 83.4 AUC), however, it performs similar to the baseline ML with average distance (+0.1 AUC and -0.1 EER). The main challenges of this dataset are filming conditions, video quality and motion blur. Pose robustness seems to be largely corrected by using alignment. First, the faces are outputs of a face detector and hence are constrained enough for reasonable landmark detections and second, empirically we observed a big gain (26 vs. 22.5 EER) when using the features computed on aligned versions of the face images *vs*. those on unaligned versions (both provided with the dataset). The main challenges thus remaining are related largely to the quality of videos, where averaging the distances seems to smooth out the noise (*e.g.* in [36] as well). This also seems to be supported, perhaps surprisingly, by recent results where features were pooled/averaged for all the frames of the video together into one descriptor [30]. Opportunistically choosing the minimum best distance between two faces of the two face tubes seems to be dominated by noise leading to the relatively lower performance of the baseline ML with minimum distance. This loss is largely recovered by using proposed latent ML which, however, is not able to surpass the averaged distance.

6. Discussion and conclusion

The task of video face track verification is an important and challenging face based biometrics problem. In the experiments reported, the performances were found to be far from saturated on the proposed challenging dataset. We note here that the performance depends on the random chance performance; in reality the number of negatives far exceeds the number of positives and the evaluation should mimic such challenging scenario.

We believe that a principle challenge for the task is that due to expression. In a large pool of negative candidates, it is highly probable to find another person with similar expression. Since annotating a large number of expression diverse faces will be quite a challenge, *expression normalization* would be a critical problem. The solutions could be inspired by the recent work using '3D frontalization' for face verification [36] and also the methods used for facial re-enactment and performance transfer *e.g.* [16].

To conclude, we presented a metric learning algorithm incorporating latent variables. We showed results on the popular benchmark YouTube faces where the model performs competitively w.r.t. current art using similar features. We also proposed a challenging dataset–TV Series Face Tubes (TSFT)–with manual annotations for 569 face tubes of 94 different subjects appearing in 8 popular TV series. We showed that the method improves upon the current state-of-the-art metric learning algorithms on TSFT. The dataset is available upon request.

Acknowledgement. This work was partly supported by the European integrated project AXES.

References

[1] T. Ahonen, A. Hadid, and M. Pietikainen. Face description with local binary patterns: Application to face recognition. *PAMI*, 28(12):2037–2041, 2006. 1, 4

[2] Z. Akata, F. Perronnin, Z. Harchaoui, and C. Schmid. Good practice in large-scale learning for image classification. *PAMI*, 36(3):507–520, 2013. 3

[3] A. Bar-Hillel, T. Hertz, N. Shental, D. Weinshall, and G. Ridgeway. Learning a Mahalanobis metric from equivalence constraints. *JMLR*, 6(6):937–965, 2005. 1, 2

[4] O. Barkan, J. Weill, L. Wolf, and H. Aronowitz. Fast high dimensional vector multiplication face recognition. In *ICCV*, 2013. 2

[5] A. Bellet, A. Habrard, and M. Sebban. A survey on metric learning for feature vectors and structured data. *arXiv.org*, 2013. 1, 2

[6] T. Berg and P. N. Belhumeur. Tom-vs-pete classifiers and identity-preserving alignment for face verification. In *BMVC*, 2012. 2

[7] T. Berg and P. N. Belhumeur. POOF: Part-based one-vs.-one features for fine-grained categorization, face verification, and attribute estimation. In *CVPR*, 2013. 2

[8] L. Best-Rowden, B. Klare, J. Klontz, and A. K. Jain. Video-to-video face matching: Establishing a baseline for unconstrained face recognition. In *Biometrics: Theory, Applications and Systems*, 2013. 1, 2, 4

[9] J. Bohné, Y. Ying, S. Gentric, and M. Pontil. Large margin local metric learning. In *ECCV*, 2014. 3

[10] Q. Cao, Y. Ying, and P. Li. Similarity metric learning for face recognition. In *ICCV*, 2013. 2

[11] D. Chen, X. Cao, F. Wen, and J. Sun. Blessing of dimensionality: High-dimensional feature and its efficient compression for face verification. In *CVPR*, 2013. 1, 4, 5, 6

[12] R. G. Cinbis, J. Verbeek, and C. Schmid. Unsupervised metric learning for face identification in TV video. In *ICCV*, 2011. 1, 2, 3, 4, 6, 7, 8

[13] Z. Cui, W. Li, D. Xu, S. Shan, and X. Chen. Fusing robust face region descriptors via multiple metric learning for face recognition in the wild. In *CVPR*, 2013. 2

[14] J. V. Davis, B. Kulis, P. Jain, S. Sra, and I. S. Dhillon. Information-theoretic metric learning. In *ICML*, 2007. 1, 2

[15] A. Frome, Y. Singer, F. Sha, and J. Malik. Learning globally-consistent local distance functions for shape-based image retrieval and classification. In *CVPR*, 2007. 1, 2

[16] P. Garrido, L. Valgaerts, O. Rehmsen, T. Thormaehlen, P. Perez, and C. Theobalt. Automatic face reenactment. In *CVPR*, 2014. 8

[17] R. Goh, L. Liu, X. Liu, and T. Chen. The CMU Face In Action (FIA) database. In *ICCV Workshops*, 2005. 1, 4

[18] M. Guillaumin, J. Verbeek, and C. Schmid. Is that you? Metric learning approaches for face identification. In *ICCV*, 2009. 1, 2, 5

[19] M. Heikkila, M. Pietikainen, and C. Schmid. Description of interest regions with center-symmetric local binary patterns. In *ICVGIP*, 2006. 8

[20] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst, 2007. 1, 2

[21] S. Hussain, T. Napoleon, and F. Jurie. Face recognition using local quantized patterns. In *BMVC*, 2012. 1, 4, 5, 6

[22] M. Kim, S. Kumar, V. Pavlovic, and H. Rowley. Face tracking and recognition with visual constraints in real-world videos. In *CVPR*, 2008. 1, 4

[23] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar. Attribute and simile classifiers for face verification. In *ICCV*, 2009. 2

[24] K.-C. Lee, J. Ho, M.-H. Yang, and D. Kriegman. Visual tracking and recognition using probabilistic appearance manifolds. *CVIU*, 99(3):303–331, 2005. 1, 4

[25] K.-C. Lee and D. Kriegman. Online learning of probabilistic appearance manifolds for video-based recognition and tracking. In *CVPR*, 2005. 1, 4

[26] H. Li, G. Hua, Z. Lin, J. Brandt, and J. Yang. Probabilistic elastic matching for pose variant face verification. In *CVPR*, 2013. 2

[27] A. Mignon and F. Jurie. PCCA: A new approach for distance learning from sparse pairwise constraints. In *CVPR*, 2012. 1, 2, 3, 5, 6, 8

[28] H. V. Nguyen and L. Bai. Cosine similarity metric learning for face verification. In *ACCV*, 2010. 2

[29] T. Ojala, M. Pietikainen, and D. Harwood. A comparative study of texture measures with classification based on feature distributions. *Pattern Recognition*, 29(1), 1996. 8

[30] O. M. Parkhi, K. Simonyan, A. Vedaldi, and A. Zisserman. A compact and discriminative face track descriptor. In *CVPR*, 2014. 1, 2, 3, 8

[31] J. Sanchez, F. Perronnin, T. Mensink, and J. Verbeek. Image classification with the fisher vector: Theory and practice. *IJCV*, 105(3):222–245, 2013. 2

[32] G. Sharma, S. ul Hussain, and F. Jurie. Local higher-order statistics (LHS) for texture categorization and facial analysis. In *ECCV*, 2012. 1, 5

[33] K. Simonyan, O. M. Parkhi, A. Vedaldi, and A. Zisserman. Fisher vector faces in the wild. In *BMVC*, 2013. 1, 2, 5, 6, 8

[34] J. Sivic, M. Everingham, and A. Zisserman. "Who are you?" – learning person specific classifiers from video. In *CVPR*, 2009. 4

[35] Y. Sun, X. Wang, and X. Tang. Hybrid deep learning for face verification. In *ICCV*, 2013. 2

[36] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In *CVPR*, 2014. 1, 8

[37] X. Tan and B. Triggs. Enhanced local texture feature sets for face recognition under difficult lighting conditions. *TIP*, 19(6):1635–1650, 2010. 1, 4

[38] A. Vedaldi and A. Zisserman. Efficient additive kernels via explicit feature maps. *PAMI*, 34(3):480–492, 2012. 5

[39] K. Weinberger, J. Blitzer, and L. Saul. Distance metric learning for large margin nearest neighbor classification. In *NIPS*, 2006. 1, 2

[40] R. Weng, J. Lu, J. Hu, G. Yang, and Y.-P. Tan. Robust feature set matching for partial face recognition. In *ICCV*, December 2013. 2

[41] L. Wolf, T. Hassner, and I. Maoz. Face recognition in unconstrained videos with matched background similarity. In *CVPR*, 2011. 1, 2, 4, 7, 8

[42] L. Wolf, T. Hassner, and Y. Taigman. Descriptor based methods in the wild. In *ECCV Workshops*, 2008. 8

[43] L. Wolf, T. Hassner, and Y. Taigman. Similarity scores based on background samples. In *ACCV*, 2009. 2

[44] L. Wolf, T. Hassner, and Y. Taigman. Effective unconstrained face recognition by combining multiple descriptors and learned background statistics. *PAMI*, 33(10):1978–1990, 2011. 1, 3

[45] L. Wolf and N. Levy. The svm-minus similarity score for video face recognition. In *CVPR*, 2013. 1, 8

[46] E. P. Xing, A. Y. Ng, M. I. Jordan, and S. Russell. Distance metric learning with application to clustering with side-information. In *NIPS*, 2003. 1, 2