Probabilistic Elastic Matching for Pose Variant Face Verification

Haoxiang Li, Gang Hua
Stevens Institute of Technology
Hoboken, NJ 07030
{hli18, ghua}@stevens.edu

Zhe Lin, Jonathan Brandt, Jianchao Yang
Adobe Systems Inc.
San Jose, CA 95110
{zlin, jbrandt, jiayang}@adobe.com

Abstract

Pose variation remains to be a major challenge for real-world face recognition. We approach this problem through a probabilistic elastic matching method. We take a part based representation by extracting local features (e.g., LBP or SIFT) from densely sampled multi-scale image patches. By augmenting each feature with its location, a Gaussian mixture model (GMM) is trained to capture the spatial-appearance distribution of all face images in the training corpus. Each mixture component of the GMM is confined to be a spherical Gaussian to balance the influence of the appearance and the location terms. Each Gaussian component builds correspondence of a pair of features to be matched between two faces/face tracks. For face verification, we train an SVM on the vector concatenating the difference vectors of all the feature pairs to decide if a pair of faces/face tracks is matched or not. We further propose a joint Bayesian adaptation algorithm to adapt the universally trained GMM to better model the pose variations between the target pair of faces/face tracks, which consistently improves face verification accuracy. Our experiments show that our method outperforms the state-of-the-art in the most restricted protocol on Labeled Face in the Wild (LFW) and the YouTube video face database by a significant margin.

1. Introduction

Face recognition has remained an active research topic in computer vision for decades [17, 32, 31, 22, 2, 11, 1, 21, 30, 14, 18, 29, 7]. In recent years, we have witnessed more and more research efforts on face recognition under uncontrolled settings [21, 14, 18, 29, 7]. Face recognition can be categorized into two tasks: *face identification* and *face verification*. The former attempts to recognize the identity of a probe face based on a set of gallery face images with known identities. The latter tries to arbitrate if a pair of faces is from the same subject or not. In this paper, we address the problem of pose variant face verification in uncontrolled settings.

Among the various visual complications affecting robust face recognition, pose variation is one of the most challenging [30]. Previous work has approached this problem by either exploiting a strong face alignment algorithm [7], or building a robust matching scheme that measures the similarity of faces across different poses [21, 30, 14, 18]. While we have witnessed great progress on face alignment in recent years [4], building a robust face alignment system by itself is a very challenging problem which requires a lot of engineering efforts [4]. As a result, state-of-the-art face alignment systems, even those with published papers, are often not fully accessible to the research community.

Although sharing aligned faces in a carefully crafted benchmark face recognition dataset such as the Labeled Face in the Wild (LFW) [16] partly relieves the issue, it immediately becomes a hurdle when one wants to build an end-to-end functioning system for face recognition. Besides, state-of-the-art face alignment results are still far from perfect, the aligned face may still present a lot of pose variations. Hence, we take the latter approach by designing robust matching schemes for unaligned or roughly aligned pose variant face verification. We believe it is a more fundamental problem as it also addresses the residue pose variations from any state-of-the-art face alignment systems.

We take a part based representation for a single face image or face tracks. Each face image is densely partitioned into overlapping patches at multiple scales, from each of which a local feature such as Local Binary Pattern (LBP) [1] or SIFT [19] is extracted. We augment each local feature with its location in the face image, and hence a face is represented as a bag of spatial-appearance features. To enable robust matching for pose variant face verification, given a set of training images, we firstly build a Gaussian mixture model (GMM) on the spatial-appearance features from all the training images. In speech recognition, such a GMM is also called Universal Background Model (UBM) [13].

To balance the impact of the appearance and spatial location, we further constrain each mixture component of the UBM to be a spherical Gaussian (Section 4.1). When matching two face images for face verification, each component of the GMM model identifies a pair of

1063-6919/13 $26.00 © 2013 IEEE
DOI 10.1109/CVPR.2013.449

appearance features (corresponding to a pair of image patches) from the two face images to be matched (Section 4.2). We concatenate the absolute difference vector of all these feature pairs from all spherical Gaussian components together to form a long difference vector. An SVM classifier is trained on such difference vectors given a set of training matching/non-matching face/face track pairs, which is subsequently used to verify any new face/face track pairs. One important advantage of this matching framework is that it can be used for both image-to-image and video-to-video face verification without any modification.

As we will show in our experiments, the proposed robust matching scheme bridged by the UBM-GMM, namely probabilistic elastic matching (PEM), outperforms the current state-of-the-art performance on both the LFW [16] (working under the most restricted protocol) and the YouTube Video Face Dataset [27] with a significant margin. To make PEM to be adaptive to each pair of faces, we further propose a joint Bayesian adaptation scheme to adapt the UBM-GMM to better fit the features of the pair of faces/face tracks by Bayesian maximum a posteriori parameter estimation (Section 4.3).

We call such an adapted matching algorithm to be adaptive probabilistic elastic matching (APEM). It consistently improves the face verification accuracy over PEM at the cost of additional computation. Our experiments even show that our PEM and APEM algorithms, when applied to face verification with unaligned faces, i.e., raw face images extracted from the Viola-Jones face detector [24], indeed outperforms the state-of-the-art algorithm, such as the bio-inspired V1 features with multiple kernel learning applied to faces aligned with the funneling method [15] under the most restricted protocol in LFW. This provides strong evidence that our proposed PEM and APEM algorithms can better handle pose variations.

Hence, the main contributions of this paper are: 1) we propose to use an universally trained spherical UBM-GMM on spatial-appearance features as a bridge to build invariant feature correspondences through probabilistic elastic matching for both image and video face verification; 2) we show that the joint Bayesian adaptation of the spherical UBM-GMM on the pair of faces/face tracks to be verified can further improve the invariance in matching; and 3) we achieve state-of-the-art face verification accuracy on both LFW (the most restricted protocol in image restricted setting), and the YouTube Faces benchmarks.

2. Related Work

Related works include those adopted UBM-GMM for visual recognition [34, 9, 12, 26], and the current state-of-the-art face verification algorithms on both the LFW [18, 29, 21, 14, 5, 33, 8, 25, 3] and YouTube video face datasets [27]. We briefly discuss them in turn.

The Gaussian mixture model has been widely used for various visual recognition tasks including face recognition [12, 26, 34] and scene recognition [9, 34]. While early works [12, 26] focused on modeling the holistic appearance of the face with GMM, more recent works [34, 9] have largely exploited the bag of local feature representation and use GMM to model the local appearances of the images. These latter works also leveraged the UBM-GMM and Bayesian adaptation paradigm to learn adaptive representations, wherein the super-vector representations are adopted for building the final classification model. While the super-vector representation is related to average pooling scheme, our invariant matching scheme is more similar to the max pooling and the lateral inhibition mechanism found in the visual cortex. Besides, none of these works conducted joint spatial-appearance modeling using spherical Gaussians as the mixture components and their Bayesian adaptation is applied to a single image whereas we conduct a joint Bayesian adaptation on a pair of faces/face tracks to better build the correspondences of the local features in the two face images.

The LFW benchmark has three protocols in the Image-Restricted Training setting for a 10 folds cross validation evaluation. The most restricted protocol does not allow any additional datasets to be used for face alignment, feature extraction, or building the recognition model. The less restricted protocol allows to use additional datasets for face alignment and feature extraction, but not for building the recognition model. While the least restricted protocol allows additional datasets to be exploited for all three tasks. The current state-of-the-art on the most restricted protocol is the work of the bio-inspired V1-like features presented by Pinto et al. [21], which achieved an average accuracy of 0.7935 ± 0.0055[1].

Predominant recent works focused on the less restricted protocol [5, 8, 25] and least restricted protocol [18, 33, 3], which have pushed the recognition accuracy to be as high as 0.9330 ± 0.0128. They all leveraged additional data sources. We focused our experiments on the most restricted protocol on LFW as our interest is the design of a robust matching method for pose variant face verification. Restricting the evaluation to the most restricted protocol enables objective evaluation of the capacity of our proposed approach. Our method only employed simple visual features such as LBP and SIFT. We also observed consistent improvement when fusing the results from these two types of features together, suggesting that we can further improve face verification accuracy from the proposed method by fusing more types of features, or by feature learning, which we leave as our future work.

[1] Pinto et al. [21] used the View 1 of LFW for parameter tuning, which may have partly boosted their accuracy on the View 2 of LFW as the face images in View 1 and View 2 are overlapping.

Wolf *et al*. [27] published a video face verification benchmark, namely YouTube Faces. To date, the state-of-the-art results are reported by the authors, using a method extended from their previous work [29] on image-based face verification. Our proposed approach can be directly applied to video face verification without any modification, which outperformed their method by a significant margin.

3. Spatial-appearance Feature Extraction

For image based face verification, we represent each face image as a bag of spatial-appearance features. As shown in Figure 1, for each face image $\mathcal{F}$, we firstly build a three layer Gaussian image pyramid. Then we densely extract overlapping image patches from each level of the image pyramid. The set of all N patches extracted from face image $\mathcal{F}$ is denoted as $\mathcal{P} = \{p_i\}_{i=1}^{N}$. After that, we extract appearance feature from each image patch p_i which we denote as $\mathbf{a}_{p_i}$. Finally, we augment the appearance feature of each patch p_i with its coordinates $\mathbf{l}_{p_i} = [x\ y]^T$ as its spatial feature. As a result, the final feature representation for patch p_i is a spatial-appearance feature $\mathbf{f}_{p_i} = [{\mathbf{a}_{p_i}}^T, {\mathbf{l}_{p_i}}^T]^T$. The final representation for face image $\mathcal{F}$ is hence an ensemble of these spatial-appearance features, i.e., $\mathbf{f}_{\mathcal{F}} = \{\mathbf{f}_{p_i}\}_{i=1}^{N}$.

In video based face verification, the task is to verify if two tracks of faces are from the same person or not (assuming each track of faces is the face of a single person). We adopt the same bag of spatial-appearance feature representation for a track of faces by repeating the feature extraction pipeline in Figure 1 on each face image in the track. The features extracted from all the face images from a single track are put together to form a larger set of spatial-appearance features to serve as the final representation of a face track. As a result, we take the same kind of feature representation for both image based and video based face verification. Therefore the probabilistic elastic matching method we will introduce in the next section will apply to both image and video based face verification.

4. Probabilistic Elastic Matching

The exact steps of the proposed probabilistic elastic matching method are illustrated in Figure 2. We start by building a GMM from all the spatial-appearance features extracted from face images in the training set. Following the terminology from the speech recognition community [13], we call such a GMM a Universal Background Model (UBM) or UBM-GMM.

Given a face/face track pair, both of which are represented as a bag of spatial-appearance features, for each Gaussian component in the UBM-GMM, we look for a pair of features (one from each of the face images/tracks) that induces the highest probability on it. We call such a pair of features a *corresponding feature pair*. We concatenate the absolute difference vectors of all these corresponding feature pairs together to form a long vector, which is subsequently fed into an SVM classifier for prediction.

An additional improvement is to conduct a joint Bayesian adaptation step to adapt the UBM-GMM to the union of the spatial-appearance features from both face images/tracks constrained *a priori* by the parameters of the original UBM-GMM to form a new GMM (A-GMM). Then we could use the A-GMM instead of the UBM-GMM to build the corresponding feature pairs. We call the proposed approach using UBM-GMM to build the corresponding feature pair to be *probabilistic elastic matching* (PEM), and the approach using A-GMM to build the corresponding feature pair to be *adaptive probabilistic elastic matching* (APEM).

We proceed with detailed description of the key steps including the training of the UBM-GMM (Section 4.1), the invariant matching scheme (Section 4.2), and the joint Bayesian adaptation algorithm for the APEM (Section 4.3).

4.1. Training UBM-GMM

As we have mentioned, GMM as UBM is widely used in the area of speech recognition [13]. In our method, to balance the impact of the appearance and spatial location, we confine the UBM to be a GMM with spherical Gaussian components, i.e.,

$$P(\mathbf{f}|\Theta) = \sum_{k=1}^{K} \omega_k \mathcal{G}(\mathbf{f}|\vec{\mu}_k, \sigma_k^2\mathbf{I}), \tag{1}$$

where $\Theta = (\omega_1, \vec{\mu}_1, \sigma_1, \ldots, \omega_K, \vec{\mu}_K, \sigma_K)$; K is the number of Gaussian mixture components; $\mathbf{I}$ is an identity matrix; ω_k is the mixture weight of the k-th Gaussian component; $\mathcal{G}(\mu_k, \sigma_k^2\mathbf{I})$ is a spherical Gaussian with mean μ_k and variance $\sigma_k^2\mathbf{I}$, and $\mathbf{f}$ is an m-dimensional spatial-appearance feature vector *i.e.*, $\mathbf{f} = [\mathbf{a}^T\ \mathbf{l}^T]^T$.

To fit such a UBM-GMM over the training feature set $\chi = \{\mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_M\}$, we resort to the Expectation-Maximization (EM) algorithm to obtain an estimate of the parameters of GMM by maximizing the likelihood $\mathcal{L}$ of the training features χ, formally,

$$\Theta^* = \arg\max_{\Theta} \mathcal{L}(\chi|\Theta) \tag{2}$$

The EM algorithm consists of the **E**-step which computes the expected log-likelihood and the **M**-step which updates parameters to maximize this expected log-likelihood [10]. Specifically, in our UBM-GMM case, in the **E**-step, we calculate

$$n_k = \sum_{i=1}^{M} P(k|\mathbf{f}_i), \tag{3}$$

$$E_k(\mathbf{f}) = \frac{1}{n_k} \sum_{i=1}^{M} P(k|\mathbf{f}_i)\mathbf{f}_i, \tag{4}$$

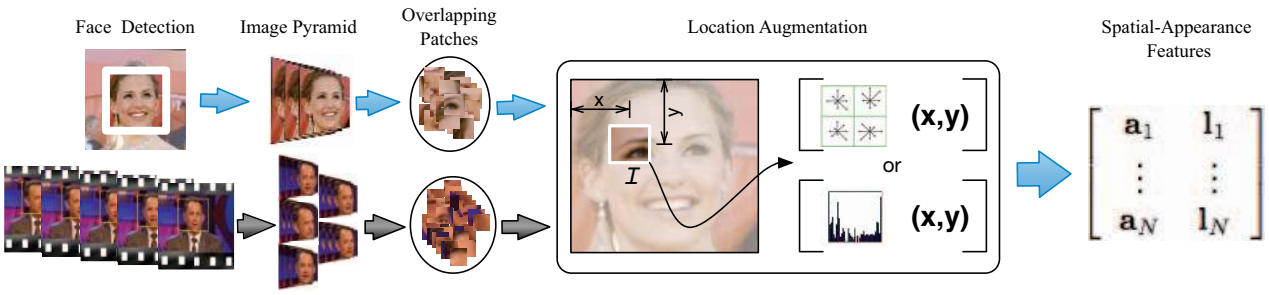


Figure 1. Spatial-appearance feature extraction pipeline.

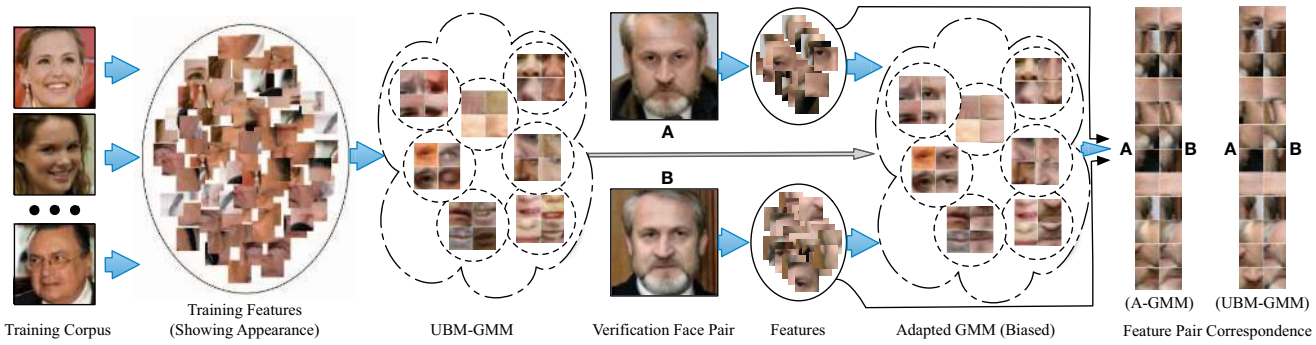


Figure 2. Our pipeline to build feature pair correspondence.

$$E_k(\mathbf{f}^T\mathbf{f}) = \frac{1}{n_k}\sum_{i=1}^{M} P(k|\mathbf{f}_i)\mathbf{f}_i^T\mathbf{f}_i, \tag{5}$$

where $P(k|\mathbf{f}_i)$ is defined as

$$P(k|\mathbf{f}_i) = \frac{\omega_k \mathcal{G}(\mathbf{f}_i|\mu_k, \sigma_k^2\mathbf{I})}{\sum_{k\prime=1}^{K}\omega_{k\prime}\mathcal{G}(\mathbf{f}_i|\mu_{k\prime}, \sigma_{k\prime}^2\mathbf{I})} \tag{6}$$

which is the posterior probability that the k-th Gaussian component generated feature $\mathbf{f}_i$.

In the **M**-step, the parameter set Θ is updated as

$$\hat{\omega}_k = \frac{n_k}{M}, \tag{7}$$

$$\hat{\mu}_k = E_k(\mathbf{f}), \tag{8}$$

$$\hat{\sigma}_k^2 = \frac{1}{m}(E_k(\mathbf{f}^T\mathbf{f}) - \hat{\mu}_k^T\hat{\mu}_k). \tag{9}$$

These two steps are iterated until convergence, at which time we obtain the UBM-GMM. Note that variances along different dimensions are indeed taken into consideration through Equation 9.

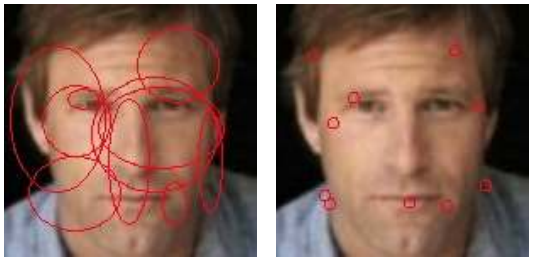
With the location augmented feature, it is a well-recognized problem that the spatial constraint from the augmented $\mathbf{l}$ can be too weak to make a difference if treated in a straightforward manner. This is because, in practice, the dimension m_a of the appearance feature $\mathbf{a}$ can be considerably larger than the dimension of the location feature $\mathbf{l}$ which is $m_l = 2$ in our experiments.

Here we argue and demonstrate that confining each mixture component in GMM to be a spherical Gaussian can handle this issue, as it helps establish a balance between the spatial and appearance constraint. Take the k-th Gaussian component $P(\mathbf{f}|\omega_k, \mu_k, \sigma_k^2\mathbf{I})$ as an example, the probability feature $\mathbf{f}$ over it is

$$\mathcal{G}(\mathbf{f}|\vec{\mu}_k, \sigma_k^2\mathbf{I}) \propto e^{-\frac{\|\mathbf{a}-\vec{\mu}_k^a\|^2}{2\sigma_k^2}} e^{-\frac{\|\mathbf{l}-\vec{\mu}_k^l\|^2}{2\sigma_k^2}}, \tag{10}$$

where $\vec{\mu}_k^a$ and $\vec{\mu}_k^l$ are the appearance and location part of $\vec{\mu}_k$, respectively, such that $\vec{\mu}_k = [\vec{\mu}_k^{aT}, \vec{\mu}_k^{lT}]^T$. As shown in Equation 10, the spherical Gaussian on the spatial-appearance model can be regarded as the product of two equal variance Gaussian distribution over two Euclidean distances produced by the appearance and location, respectively. As long as the ranges of the two Euclidean distances are matched, the influence of these two Gaussians will be balanced. This can be easily achieved by normalizing the appearance and the location part of the spatial-appearance feature in an appropriate way, such as scaling $\mathbf{a}$ to be unit vector and keeping every element of $\mathbf{l}$ has a value between 0 and 1.

As illustrated in Figure 3, without confining the mixture components to be spherical Gaussians, the spatial constraint introduced from $\mathbf{l}$ is so weak that the spatial spanning of Gaussian components are highly overlapped, which could not help build correct feature correspondences in the invariant matching stage. In contrast, the spatial variances of



(a) UBM - normal Gaussians (b) UBM - spherical Gaussians

Figure 3. Spatial distribution of 10 selected Gaussian components in the UBM over a face. Each red ellipse (or circle) stands for a Gaussian component. The center and span show mean and variance of the spatial part of the Gaussian component.

spherical Gaussian components are more localized, which could tolerate pose variations more appropriately.

Note that if the UBM-GMM is with normal Gaussian components, one can not address this issue by scaling $\mathbf{a}$. This can be observed by checking the equations in the EM algorithm: if $\mathbf{a}$ is scaled, the corresponding means and covariances will be scaled proportionally. Then the probability of $\mathbf{f}$ over each of the Gaussian components will be scaled in the same way. As a result, $P(k|\mathbf{f}_i)$ is unchanged (Equation 6), which means the EM estimates will undesirably remain the same – it only scales the mean and variance estimates. This is not able to help balance the influence of the appearance and the location.

4.2. Invariant Matching

After we obtained the K-components UBM-GMM trained over a set of m-dimensional spatial-appearance features, we exploit it to form an elastic matching scheme in the form of a $D = m \times K$ dimensional long difference vector for a pair of face images/tracks.

Formally, we present a face/face track $\mathcal{F}$ as a bag of spatial-appearance features $\mathbf{f}_\mathcal{F} = \{\mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_N\}$. First we let each Gaussian component $(\omega_k, \mathcal{G}_k(\vec{\mu}_k, \sigma_k^2\mathcal{I}))$ commit one feature $f_{g_k(\mathcal{F})}$ from $\mathbf{f}_\mathcal{F}$, such that

$$g_k(\mathcal{F}) = \arg\max_i \omega_k \mathcal{G}(\mathbf{f}_i|\vec{\mu}_k, \sigma_k^2\mathbf{I}). \quad (11)$$

The face/face track $\mathcal{F}$ is then represented as a sequence of K m-dimensional features, i.e, $[\mathbf{f}_{g_1}\ \mathbf{f}_{g_2}\ \ldots\ \mathbf{f}_{g_K}]$. After this stage, given the i-th faces/face tracks pair ($\mathcal{F}$ and $\mathcal{F}'$), the difference vector is a concatenated vector, i.e.,

$$\mathbf{d}_i = [\Delta\mathbf{a}_{g_1}\ \Delta\mathbf{a}_{g_2}\ \ldots\ \Delta\mathbf{a}_{g_K}]^T, \quad (12)$$

where $\Delta\mathbf{a}_{g_k} = |\mathbf{a}_{g_k(\mathcal{F})} - \mathbf{a}_{g_k(\mathcal{F}')}|^T$, which serves as the final matching feature vector of a pair of faces/face tracks for face verification. Note in this final representation, we focus on the appearance differences since the spatial component is already taken into consideration when we build the corresponding feature pairs. The way we build correspondence from the spatial-appearance GMM model is motivated by and related to max pooling and the lateral inhibition mechanism in receptive fields, both have been proven to be beneficial when building visual representations.

A kernel SVM classifier, i.e.,

$$f(\mathbf{d}) = \sum_{i=1}^{\mathcal{V}} \alpha_i k(\mathbf{d}_i, \mathbf{d}) + b, \quad (13)$$

is then trained over C training difference vectors $\{\mathbf{d}_1, \mathbf{d}_2, \ldots, \mathbf{d}_C\}$ with the Gaussian Radial Basis Function (RBF) kernel, i.e. ,

$$k(\mathbf{d}_i, \mathbf{d}_j) = exp(-\gamma||\mathbf{d}_i - \mathbf{d}_j||^2), \gamma > 0, \quad (14)$$

where $i, j = 1, \ldots, C$. Given the difference vector $\mathbf{d}_t$ of a testing face/face track pair, the SVM predicts its label. We employed the LibSVM [6] to train the SVM classifier. We call the matching algorithm presented in this section to be *probabilistic elastic matching* (PEM).

4.3. Joint Bayesian Model Adaptation

Prior work applying GMMs with Bayesian adaptation to visual recognition [34, 9] has operated either at the class level or at the image level. To make the matching process adaptive for each face/face track pair, we propose a joint Bayesian adaptation on the union of the bag of spatial-appearance features from the faces/face tracks pair. In the joint adaptation process, the parameters of the UBM-GMM build the prior distribution for the parameters of the jointly adapted GMM under a Bayesian maximum a posteriori (MAP) framework.

We denote the UBM parameter set as Θ_b and parameter set of the GMM after joint adaptation as Θ_p, where $\Theta_x = \{\omega_{x_1}, \vec{\mu}_{x_1}, \sigma_{x_1}, \ldots, \omega_{x_K}, \vec{\mu}_{x_K}, \sigma_{x_K}\}, x = \{b, p\}$. Given a face/face track pair $\mathcal{Q}$ and $\mathcal{S}$, the adaptive GMM is trained over the joint feature set $\chi_p = \{\mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_P\}$ which is the union of feature sets of $\mathcal{Q}$ and $\mathcal{S}$ as χ_q and χ_s, where $|\chi_p| = |\chi_q| + |\chi_s|$. Upon χ_p a MAP estimate for Θ_p can be obtained by maximizing the log-likelihood $\mathcal{L}(\Theta_p)$, i.e.,

$$\mathcal{L}(\Theta_p) = lnP(\chi_p|\Theta_p) + lnP(\Theta_p|\Theta_b). \quad (15)$$

The conjugate prior distribution of Θ_p is composed from the UBM-GMM parameter Θ_b [9, 34, 10], i.e.,

$$(\omega_{p_1}, \ldots, \omega_{p_K}) \sim Dir(T\omega_{b_1}, \ldots, T\omega_{b_K}), \quad (16)$$

$$\mu_{p_k} \sim \mathcal{N}(\vec{\mu}_{b_k}, \sigma_{b_k}^2/\gamma). \quad (17)$$

The prior distribution over the mixture weights is a Dirichlet distribution. The parameter T can be interpreted as the count of features introduced by the UBM-GMM. The prior distribution for mean μ_{p_k} is a spherical Gaussian distribution with variance smoothed by parameter γ. We can

(a) Feature correspondences built through UBM-GMM

(b) Feature correspondences built through A-GMM

Figure 4. In both figures, the row above shows local patches from face A shown in Figure 2, while the bottom ones are from face B. Each column shows a pair of features captured by one Gaussian component in the GMM.

also use a Normal-Wishart distribution over the variance as in [9, 10]. However, in order to stabilize the adapted GMM, we confined the adapted variance to be the same as that of the UBM-GMM, i.e, $\sigma^2_{p_k} = \sigma^2_{b_k}$.

With these priors, the parameters of the adapted GMM can be estimated by a Bayesian EM algorithm [9, 34, 10], i.e., in the **E**-step, we calculate

$$n_k = \sum_{i=1}^{P} P(k|\mathbf{f}_i), \tag{18}$$

$$E_k(\mathbf{f}) = \frac{1}{n_k} \sum_{i=1}^{P} P(k|\mathbf{f}_i)\mathbf{f}_i, \tag{19}$$

where

$$P(k|\mathbf{f}_i) = \frac{\omega_{p_k}\mathcal{G}(\mathbf{f}_i|\mu_{p_k}, \sigma^2_{p_k})}{\sum_{k\prime=1}^{K}\omega_{p_{k\prime}}\mathcal{G}(\mathbf{f}_i|\mu_{p_{k\prime}}, \sigma^2_{p_{k\prime}})}, \tag{20}$$

and in **M**-step, we update Θ_p as

$$\hat{\omega}_{p_k} = \alpha\frac{n_k}{N} + (1-\alpha)\omega_{b_k}, \tag{21}$$

$$\hat{\mu}_{p_k} = \beta_k E_k(\mathbf{f}) + (1-\beta_k)\vec{\mu}_{b_k}, \tag{22}$$

where

$$\alpha = N/(N+T),\ \beta_k = n_k/(n_k+\gamma). \tag{23}$$

After we obtain the adapted GMM given a pair of faces/face tracks, we conduct APEM to build the difference vector for invariant matching. We could observe A-GMM improves some feature correspondences as shown in Figure 4, such as the 10th and the last column.

5. Multiple Feature Fusion

In visual recognition, different kinds of multiple feature fusion techniques are widely adopted [7, 21]. In this paper, we augment our PEM/APEM by a simple multiple feature post-fusion framework to combine the effectiveness of different features using a linear SVM.

To post-fuse multiple features, we repeat the proposed pipeline over all face/face track pairs using D types of different local features to obtain D confidence scores for each face/face track pair p_i as a score vector

$$\mathbf{s}_i = [s_{i_1}\ s_{i_2}\ \ldots s_{i_D}], \tag{24}$$

where s_{i_d} denotes the score assigned by the classifier using the d-th type of feature. Over all C training score vectors $\{\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_C\}$ and their labels, we train a linear SVM classifier to predict the label for a testing score vector $\mathbf{s}_t$ of a face/face track pair. Such a simple scheme proved to be very effective in our experiments. We note here that more advanced method such as multiple kernel learning (MKL) similar to what has been adopted in [21] can also be adopted, but we observe no performance difference when compared with our simple fusion scheme with a linear SVM.

6. Experimental Evaluation

Extensive experiments are performed over two challenging datasets, Labeled Face in the Wild (LFW) [16] and YouTube Faces Database [27].

6.1. Labeled Faces in the Wild

Labeled Faces in the Wild (LFW) [16] dataset is designed to address the unconstrained face verification problem. This challenging dataset contains more than 13,000 images from 5749 people. In general there are two training methods over LFW, *image-restricted* method and *image-unrestricted* method. By design, *image-restricted* paradigm does not allow experimenters to use the name of a person to infer two face images are matched or non-matched, while in the *image-unrestricted* paradigm experimenters may form as many matched or non-matched face pairs as desired for training. Over LFW, researchers are expected to explicitly state the training method they used and report performance over 10-folds cross-validation. In our experiments, we followed the *most restricted* protocol, in which detected faces are aligned with the funneling method [15].

6.1.1 Baseline Algorithm

To better investigate our PEM/APEM approach to pose variant face verification, we introduce a baseline algorithm that shows how well a trivial location-based feature pair matching scheme performs. The baseline algorithm provides a basis of comparison to evaluate the effectiveness of building feature pair correspondences bridged by UBM-GMM or adapted GMM. Formally, $\mathcal{F}$ and $\mathcal{F}'$ are representations of two faces, both have N features, i.e., $\mathcal{F} = \{\mathbf{f}_1\ \ldots \mathbf{f}_N\}$ and $\mathcal{F}' = \{\mathbf{f}'_1\ \ldots \mathbf{f}'_N\}$, where $\mathbf{f}_n$ and $\mathbf{f}'_n$ are two spatial-appearance feature from the n-th local patch at the same location. Similar to Section 4.2, the concatenated difference vector between faces $\mathcal{F}$ and $\mathcal{F}'$ is $d(\mathcal{F}, \mathcal{F}') = [|\mathbf{f}_1 - \mathbf{f}'_1|^T \ldots |\mathbf{f}_N - \mathbf{f}'_N|^T]^T$. Then we train an SVM classifier over

Table 1. Performance comparison on the most restricted LFW

Algorithm	Accuracy $\pm$ Error(%)
Nowak[20]	73.93 ± 0.49
Hybrid descriptor-based[28]	78.47 ± 0.51
V1/MKL[21]	79.35 ± 0.55
Baseline (fusion)	77.30 ± 1.59
PEM (LBP)	81.10 ± 1.71
PEM (SIFT)	81.38 ± 0.98
PEM (fusion)	82.93 ± 1.18
APEM (LBP)	81.97 ± 1.90
APEM (SIFT)	81.88 ± 0.94
APEM (fusion)	$\mathbf{84.08 \pm 1.20}$
APEM (fusion, unaligned)	81.70 ± 1.78

training difference vectors to predict if a testing face/face track pair is matched.

6.1.2 Settings

In our experiments, images are center cropped to 150x150 before feature extraction. As shown in Figure 1, SIFT and LBP features are extracted over each scale for a 3-scale Gaussian image pyramid with scaling factor 0.9. SIFT features are extracted from patches from a 8x8 sliding window with 4-pixel spacing, and LBP features[2] are extracted from a 32x32 sliding window with 4-pixel spacing. After that, the appearance feature is augmented by the coordinates of the patch center to build the spatial-appearance feature vector. Over all training features, we trained a UBM-GMM of 1024 spherical Gaussian components for PEM. For APEM, given a pair of face images, all features in the joint feature set are utilized for joint adaptation. After calculating matching difference vectors, we trained an SVM classifier using RBF kernel for classification. We followed the standard 10-folds cross-validation over View 2 to report our performance, and we never use the View 1 dataset.

6.1.3 Results

As shown in Table 1 and Figure 5, our methods outperformed the state-of-the-art by a considerable margin. We demonstrated the effectiveness of the invariant matching by comparing with the baseline and we also observed joint Bayesian adaptation and multiple features fusion bring consistent improvements. Furthermore, our approach on unaligned faces [16], which are the outputs of the Viola-Jones face detector, even outperformed state-of-the-art methods with faces aligned by the funneling method.

6.2. YouTube Faces Dataset

This work is a general framework which can handle both image and video based face verification without modification. Wolf *et al*. [27] published YouTube Faces Dataset

[2]The LBP feature is constructed in a part-based scheme by partitioning each window uniformly into 16 8x8 cells and concatenating 16 58-dimensional uniform LBP histogram [23] calculated in each cell to form the 928-dimensional LBP feature.

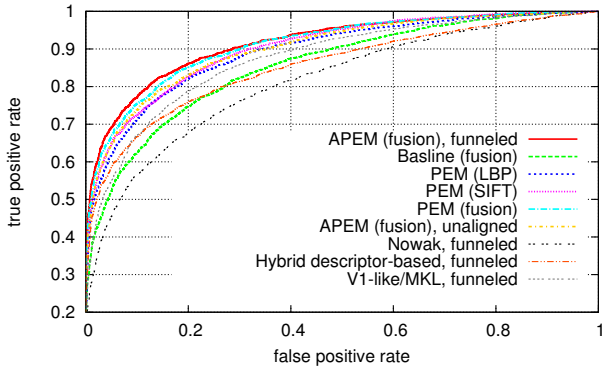


Figure 5. Performance comparison on the most restricted LFW.

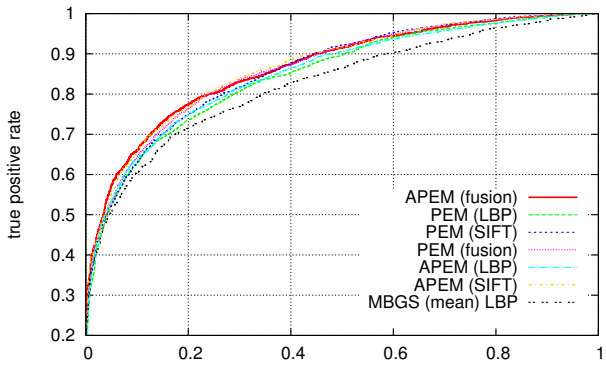


Figure 6. Performance comparison over YouTube Faces

(YTFaces) for studying the problem of unconstrained face recognition in videos. The dataset contains 3,425 videos of 1,595 different people. On average, a face track from a video clip consists of 181.3 frames of faces. Faces are detected by the Viola-Jones detector and aligned by fixing the coordinates of automatically detected facial feature points [27]. Protocols are similar to LFW, for the same purpose, we focus on the restricted video face verification paradigm. To date, the state-of-art performance is published by the authors using Matched Background Similarity (MBGS) algorithm with LBP feature.

6.2.1 Settings

In the video faces experiments, each image frame is center cropped to 100x100 before feature extraction. Then features are extracted in the same way in Section 6.1.2 for each frame, except that both SIFT and LBP are extracted with 8-pixels spacing. On average, more than 40000 features are extracted from one face track. In the stage of joint Bayesian adaptation, to ease the computational intensity, 1000 out of 40000 features are sampled randomly from each face track to be combined into the joint feature set.

6.2.2 Results

As shown in table 2 and figure 6, our method outperformed the state-of-the-art algorithm by a significant margin. Even

Table 2. Performance comparison over YouTube Faces

Algorithm	Accuracy $\pm$ Error(%)
MBGS[27]	76.4 ± 1.8
PEM (LBP)	76.82 ± 1.60
PEM (SIFT)	77.52 ± 2.06
PEM (fusion)	78.36 ± 1.69
APEM (LBP)	77.44 ± 1.46
APEM (SIFT)	78.54 ± 1.42
APEM (fusion)	$\mathbf{79.06 \pm 1.51}$

without feature fusion, PEM with LBP features already have comparable performance with slightly better accuracy over MBGS.

7. Conclusion

In this paper, we proposed a probabilistic elastic matching algorithm with an additional joint Bayesian adaptation component as a general framework for both image and video based face verification. Extensive experiments were performed in which PEM/APEM showed superior performances over state-of-the-art methods on two standard face verification benchmark datasets, most restricted LFW and restricted Youtube Faces dataset.

Acknowledgement

This work is partly supported by GH's start-up funds from Stevens Institute of Technology and a Collaboration Research Gifts from Adobe System Incorporated.

References

[1] T. Ahonen, A. Hadid, and M. Pietikainen. Face recognition with local binary patterns. In *ECCV*, 2004.

[2] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman. Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection. *T-PAMI*, 1997.

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

[4] X. Cao, Y. Wei, F. Wen, and J. Sun. Face alignment by explicit shape regression. In *CVPR*, 2012.

[5] Z. Cao, Q. Yin, X. Tang, and J. Sun. Face recognition with learning-based descriptor. In *CVPR*, 2010.

[6] C.-C. Chang and C.-J. Lin. LIBSVM: A library for support vector machines. *ACM T-IST*, 2011.

[7] D. Chen, X. Cao, L. Wang, F. Wen, and J. Sun. Bayesian face revisited: A joint formulation. In *ECCV*, 2012.

[8] D. Cox and N. Pinto. Beyond simple features: A large-scale feature search approach to unconstrained face recognition. In *FGR*, 2011.

[9] M. Dixit, N. Rasiwasia, and N. Vasconcelos. Adapted gaussian models for image classification. In *CVPR*, 2011.

[10] J.-L. Gauvain and C.-H. Lee. Maximum a posteriori estimation for multivariate gaussian mixture observations of markov chains. *T-SAP*, 1994.

[11] A. S. Georghiades, P. N. Belhumeur, and D. J. Kriegman. From few to many: Illumination cone models for face recognition under variable lighting and pose. *T-PAMI*, 2001.

[12] R. Gross, J. Yang, and A. Waibel. Growing gaussian mixture models for pose invariant face recognition. *ICPR*, 2000.

[13] T. Hasan and J. Hansen. A study on universal background model training in speaker verification. *Audio, Speech, and Language Processing, IEEE Transactions on*, 2011.

[14] G. Hua and A. Akbarzadeh. A robust elastic and partial matching metric for face recognition. In *ICCV*, 2009.

[15] G. Huang, V. Jain, and E. Learned-Miller. Unsupervised joint alignment of complex images. In *ICCV*, 2007.

[16] G. B. Huang, M. Mattar, T. Berg, and E. Learned-Miller. Labeled Faces in the Wild: A Database forStudying Face Recognition in Unconstrained Environments. In *Faces in Real-Life Images Workshop in ECCV*, 2008.

[17] V. Jain, A. Ferencz, and E. Learned-miller. Discriminative training of hyper-feature models for object identification. In *BMVC*, pages 357–366, 2006.

[18] N. Kumar, A. Berg, P. N. Belhumeur, and S. Nayar. Describable visual attributes for face verification and image search. *T-PAMI*, 2011.

[19] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 2004.

[20] E. Nowak and F. Jurie. Learning visual similarity measures for comparing never seen objects. In *CVPR*, 2007.

[21] N. Pinto, J. J. DiCarlo, and D. D. Cox. How far can you get with a modern face recognition test set using only simple features? In *CVPR*, 2009.

[22] M. A. Turk and A. P. Pentland. Face recognition using eigenfaces. In *CVPR*, 1991.

[23] A. Vedaldi and B. Fulkerson. Vlfeat: an open and portable library of computer vision algorithms. In *ACM Multimedia*, 2010.

[24] P. Viola and M. J. Jones. Robust real-time face detection. *IJCV*, 2004.

[25] F. Wang and L. J. Guibas. Supervised earth mover's distance learning and its computer vision applications. In *ECCV*, 2012.

[26] X. Wang and X. Tang. Bayesian face recognition based on gaussian mixture models. In *ICPR*, 2004.

[27] L. Wolf, T. Hassner, and I. Maoz. Face recognition in unconstrained videos with matched background similarity. In *CVPR*, 2011.

[28] L. Wolf, T. Hassner, and Y. Taigman. Descriptor based methods in the wild. In *Faces in Real-Life Images Workshop in ECCV*, 2008.

[29] L. Wolf, T. Hassner, and Y. Taigman. Effective unconstrained face recognition by combining multiple descriptors and learned background statistics. *T-PAMI*, 2011.

[30] J. Wright and G. Hua. Implicit elastic matching with randomized projections for pose-variant face recognition. In *CVPR*, 2009.

[31] S. Yan, M. Liu, and T. Huang. Extracting age information from local spatially flexible patches. In *ICASSP*, 2008.

[32] S. Yan, X. Zhou, M. Liu, M. Hasegawa-Johnson, and T. Huang. Regression from patch-kernel. In *CVPR*, 2008.

[33] Q. Yin, X. Tang, and J. Sun. An associate-predict model for face recognition. In *CVPR*, 2011.

[34] X. Zhou, N. Cui, Z. Li, F. Liang, and T. Huang. Hierarchical gaussianization for image classification. In *ICCV*, 2009.