This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2015.2406191, IEEE Transactions on Circuits and Systems for Video Technology

Convolutional Fusion Network for Face Verification in the Wild

Chao Xiong, Luoqi Liu, Xiaowei Zhao, Shuicheng Yan, *Senior Member, IEEE,* Tae-Kyun Kim, *Member, IEEE*

***Abstract*—Part-based methods have seen popular applications for face verification in the wild, since they are more robust to local variations in terms of pose, illumination and so on. However, most of the part-based approaches are built on hand-crafted features, which may be not suitable for the specific face verification purpose. In this work, we propose to learn a part-based feature representation under the supervision of face identities through a deep model, which ensures the generated representations are more robust and suitable for face verification. The proposed framework consists of following two deliberate components: a Deep Mixture Model (DMM) to find accurate patch correspondence and a Convolutional Fusion Network (CFN) to extract the part based facial features. Specifically, DMM robustly depicts the spatial-appearance distribution of patch features over the faces via several Gaussian mixtures, which provide more accurate patch correspondence even in the presence of local distortions. Then, DMM only feeds the patches which preserve the identity information to the following CFN. The proposed CFN is a two-layer cascade of Convolutional Neural Networks (CNN): 1) a local layer built on face patches to deal with local variations and 2) a fusion layer integrating the responses from the local layer. CFN jointly learns and fuses multiple local responses to optimize the verification performance. The composite representation obtained possesses certain robustness to pose and illumination variations and shows comparable performance with the state-of-the-arts on two benchmark data sets.**

***Index Terms*—Deep Learning, Part-based Representation, Face Verification, Mixture Model, Feature Learning**

I. Introduction

Face verification aims to distinguish whether two face images belong to the same identity. It has long been an active research problem of computer vision. In particular, face verification under unconstrained settings has received much research attention in recent years. The release of several public data sets, e.g., YouTube Faces Database [1] and Labelled Face in the Wild [2], has greatly boosted the development of face verification techniques.

C. Xiong, X. Zhao and T-K. Kim are with the Department of Electrical and Electronic Engineering, Imperial College, South Kensington Campus, London, UK. E-mail: (chao.xiong10@imperial.ac.uk; x.zhao@imperial.ac.uk; tk.kim@imperial.ac.uk)

L. Liu and S. Yan are with the Department of Electrical and Computer Engineering, National University of Singapore. E-mail: (llq667@gmail.com; eleyans@nus.edu.sg)

Unconstrained photographic conditions bring about various challenges to face verification in the wild. Among them, one prominent challenge is the severe local distortions, such as pose variations, different facial expression. To solve this issue, many state-of-the-art approaches for face verification [3], [4], [5] are built on part-based face representation to take advantages of local representation robustness of local distortions. However, most part-based approaches are built on hand-crafted features, such as the local binary pattern [6], SIFT [7], and Gabor features [8]. Those generic features are not designed specifically for the face verification tasks, and thus suffer from following issues. Firstly of all, some characteristic visual information may be lost in extraction (especially their quantization) stage, which unfortunately cannot be recovered in the later stages. Such information lost may severely damage the face verification performance. Moreover, another weakness of those hand-crafted features is to require faces to be well-aligned, which is considered to be quite challenging to obtain or even not realistic for face images captured in the wild. These issues become even more complicated if various combinations of different features, alignment methods and learning algorithms are considered for choice.

Recently, the well developed deep learning methods propose to solve the above issues by learning the feature representation and classifiers jointly for a specific task, and see great success for various computer vision tasks [9], [10], [11], [12], [13]. Within a general deep neural network, the bottom layers usually extract elementary visual features, e.g., edges or corners, and feed forward the output to the higher layers which then extract higher-level features, such as object parts. The features extracted by the network are optimized in a supervised manner to fit a specified task and bring significant performance boosting. Inspired by the impressive performances, we also propose a deep learning method to solve face verification problem in this work. Although for face verification the part-based approaches have been proven effective with hand-crafted features [5], [3], the power of part based model may be weakened by the improper hand-crafted features, as aforementioned. Therefore, how to learn a suitable local feature representation is a critical problem for face verification, which however has not been explored much yet. Most of the existing deep learning networks [14] [15] [16] aim to learn global features from the full face images, instead of robust local ones as advocated in this work. Moreover, most aforementioned works are built on well-aligned faces, while approaches for

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2015.2406191, IEEE Transactions on Circuits and Systems for Video Technology

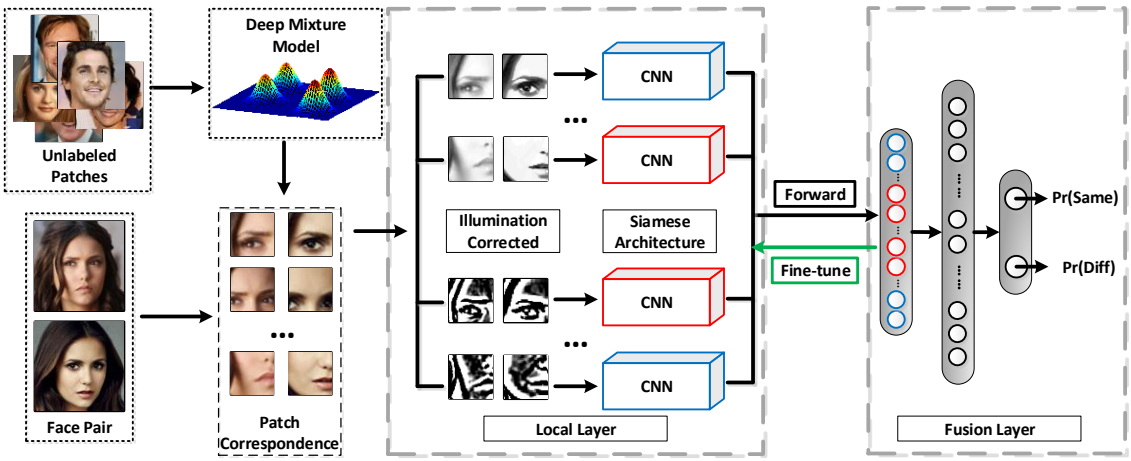


Fig. 1. Flowchart of the proposed framework. A deep mixture model (DMM) is firstly trained with unlabeld local patches to capture the spatial and appearance distribution over faces. For each image pair, a pair of local patches is acquired for each mixture component in DMM with regard to the corresponding responses. The selected patch pairs are then pre-processed with several illumination correction methods and fed into multiple sub-CNNs for supervised pre-training. The pre-trained sub-CNNs are finally fused together with a holistic fusion layer.

verifying faces with natural mis-alignment are still rare.

In this work, we introduce a novel two-stage deep model to automatically learn robust local face representations for face verification in the wild. In contrast to previous works, our proposed model does not require the faces to be well aligned, and deals with the more realistic wild setting where there exists significant mis-alignment between faces. This makes our proposed model more appealing for practical applications. The proposed deep model automatically matches the local face patches via a novel Deep Mixture Model (DMM), and then adopts Convolutional Fusion Network (CFN) to learn a part-based face representation. Benefited from these two stages, the output face representations are more robust to local variations in terms of pose, illumination and so on.

More concretely, the first layer of CFN (local layer) is pre-trained on local patches of different scales, geometric positions and illuminations. The following layer (holistic layer) learns a fully-connected classifier built on the local responses forwarded from the local layer. Conventional CNN assumes that the feature distribution is uniform over the face, thus extracts features with the same convolutional kernels for different face regions. This assumption usually does not hold in practice. In contrast, our network models the explicitly non-stationary feature distribution. Each sub-CNN in the local layer captures features that are specific for patches in the given face regions with the given illumination. Such composite structure leads to representation of tolerance to local distortions, and meanwhile captures the holistic information with the global fusion.

The problem of large pose variations is further addressed via exploring the semantic patch correspondence. Recent works [5], [17] indicate that semantically normalized patches usually improve the performance for face matching problems with various pose. In this paper, a deep mixture model (DMM) is proposed to acquire the patch correspondence. Different from previous approaches relying on manually designed features, both the representation and the mixture component parameters are optimized together by maximizing the posterior probability of the model. With the deep mixture model, patches of highest responses to the same component are taken as matched within each pair of images. The matched pairs are further ranked in terms of their discriminative scores, and those top ranked patches are chosen as the inputs for CFN. The screening process results in higher efficiency of the proposed network while retaining the verification performance.

In general, our contributions can be summarized as follows.

1) We propose a novel way of learning a part-based face representation with Convolutional Fusion Network built on multiple CNN models. Different representations are learnt for different facial regions to adapt to the geometrically non-stationary distribution. The independence leads to a better generalization performance with the holistic fusion.
2) We propose a Deep Mixture Model to obtain the semantic correspondence of patches to handle pose variation. Within the DMM network, the mixture components and the representation are jointly optimized, which is proven to be effective by extensive experiments.
3) We propose a new patch selection procedure to maintain only the discriminative patches for face verification. Such selection largely reduces the number of patches needed in CFN and leads to considerate improvement of accuracy over manually selected approach.

The proposed network is evaluated on two benchmark databases for face verification – YouTube faces database (YTF) and Labelled Face in the Wild (LFW), and achieves competitive results with the state-of-the-arts.

II. Related Work

Due to the enormous number of related topics and space limitation in this paper, we only list the most relevant works in the following two aspects.

A. Part-based Representation for Face Images

Face related tasks have attracted considerable attention due to their application potential. Seeking for a good representation of face images has long been an interesting topic for researchers.

Many methods on face representation [18], [19], [20] have been proposed during the past few decades. These methods can be roughly categorized into holistic and local approaches. Classic works on holistic features, such as Principal Component Analysis [21], are mainly subspace-based approaches that try to represent face images with the subspace basis. Compared with holistic features, local features are more robust and stable to local changes and have been widely used recently. Gabor [8], Local Binary Pattern (LBP) [6] and Bag of Words (BoW) [22] features are classic representations capturing the local information. Gabor feature captures the spatial-frequency information and is found to be robust to the illumination variation. LBP captures contrast information for each pixel by referring to its neighboring points. BoW represents the image as an orderless collection of local features extracted in densely sampled patches.

Part-based face representation [23], [24] is a popular way of capturing the local information and has been successfully applied to facial expression recognition [22], [25], face parsing [26], face identification [27] and face verification [3]. Karan et al. [22] proposed a BoW representation of face images for facial expression recognition. They extracted SIFT descriptors on densely sampled patches of multi-scale and then built the codebook. Luo et al. [26] introduced a hierarchical face parser. The parser combines the results of part detectors and component detectors to transform the face image into a label map. Zhu et al. [27] targeted at face recognition problems with a small number of training samples. They conducted collaborative representation based classification on the face patches and combined the results of all the multi-scale patches.

There have also been some recent works with part-based representation on face verification, which refreshed the state-of-the-art performance, especially for unconstrained face verification in the wild. To name a few, Li et al. [5] built a Gaussian Mixture Model (GMM) in terms of both appearance and spatial information to discover the correspondence between the patches in pair. The model is trained with LBP and SIFT features extracted from densely sampled patches. Their approach improved the state-of-the-art performance by around 4% on LFW with the most strict setting. In [3], Fisher Vector (FV), a typical descriptor for object recognition, was applied on LFW, and improved the performance further. FV in their work is built on SIFT feature extracted from the patches scanned densely through the images.

The aforementioned methods extract the same features from the different facial parts. However, we consider the feature distribution is not stationary over the whole face in this paper, and the learnt filters are different for different face regions. Without the hand-crafted features as in mentioned works, the proposed fusion network learns the feature representation automatically with direct guidance of the face identification.

B. Deep Learning

The breakthrough by Hinton and Salakhutdinov [28] triggered the enthusiasm for deep learning in both academia and industry. By stacking multiple non-linear layers, deep neural networks are able to extract more abstract features automatically than the hand-crafted features.

Over the past few years, such a deep structure has been successfully applied in many computer vision fields [9], [10], [11], [13], [12]. To name just a few, Krizhevsky et al. [9] won the ImageNet contest in 2012 by training deep CNNs fine-tuned with multiple GPUs. Sun et al. [12] proposed a three-level cascade of convolutional networks for facial keypoints detection and outperformed the state-of-the-art methods in both detection accuracy and reliability. Ouyang and Wang [29] proposed joint deep learning framework to address pedestrian detection. Feature extraction, deformation handling and occlusion handling are incorporated in a unified framework and achieves the best performance on the Caltech dataset.

Several recent works also apply deep learning to face verification task. Huang et al. [11] developed a convolutional Restricted Boltzman Machine (RBM) and evaluated it on the LFW-a database (with face alignment). The proposed method achieves comparable result to those with hand-crafted features. Chopra et al. [30] defined a mapping from input space to the target space to approximate the semantic distance in the original space. The mapping is learned with two symmetric neural networks that share the same weights to tackle face verification problem. Liao et al. [31] proposed a three-layered hierarchy without explicit detection and alignment stages in testing. However, these networks are trained with full face images only and do not specifically handle local variations. Different from the aforementioned papers, our network learns a composite representation from both the holistic faces and local patches by integrating the responses of discriminative local sub-nets.

A gradual increase in the amount of data significantly improves the verification accuracy of deep models. Sun et al. [14] learnt a set of high-level features through a multi-class identification task. The network is trained on pre-defined face patches based on the landmark positions. The performance is further improved by Sun et al. [15], in which the network is trained by jointly optimizing the identification and verification objectives. Taigman et al. [16] introduced the largest facial dataset to-date, which is used to learn an effective representation. The learnt presentation is directly applied on LFW and achieves close accuracy to that of human beings. The above deep networks are trained

with an assumption that face images are well aligned. In contrast, the proposed framework is learnt with the existence of mis-alignment. To handle such mis-alignment, a deep mixture model network is proposed to capture the spatial-appearance distribution over faces. The DMM network automatically retrieves the patch correspondence, which is proven to be effective for unconstrained face verification.

III. Convolutional Fusion Network

Most state-of-the-art approaches evaluated on benchmark datasets for face verification are built on hand-crafted features [5], [3], [32]. Instead, we address the problem of face verification in the wild by learning a part-based face representation automatically with deep convolutional neural network (CNN). Conventional CNN is built by stacking multiple convolutional layers and pooling layers. The cascade of convolution-pooling structure provides certain robustness to shifting and rotation variations. However, the final features captured are mainly holistic. Compared with holistic features, local features are more robust to local facial distortions which are common in face images in the wild. Thus, we aim at designing a network capturing both holistic and local facial properties. Introducing local information to CNN enables the network to learn a more diverse and complex presentation and leads to potential improvement.

Accordingly, the proposed Convolutional Fusion Network, illustrated in Fig. 1, has a structure of two layers – the local layer and the fusion layer. The local layer is composed of several parallel sub-CNNs corresponding to the local face patches (the full-face images are resized and treated the same as local patches) , and thus captures features with regard to the local variations. The fusion layer contains a fully-connected layer followed by a softmax classifier. It integrates the local responses to acquire a holistic view of the original image. Sub-CNNs are pre-trained separately to guarantee a certain level of independence. Such independence leads to a mutual complementary ability among sub-CNNs, resulting in a considerate improvement with fusion layer.

Illumination is also a significant factor degrading the performance of unconstrained face verification. Hua and Akbarzadeh [33] included the illumination pre-processing step and reported a considerate performance improvement. In this paper, the face images are pre-processed with several standard illumination correction methods. The local patch are then cropped from lighting-corrected images, and passed to corresponding sub-CNNs.

We denote the output of sub-CNN i as $h^{(i)}(\cdot)$, and the forward propagation of the final fusion layer can be represented as

$$y = softmax(\sum_{i=1}^{N} \boldsymbol{W}_f^{(i)} \cdot h^{(i)}(\cdot) + b_f), \tag{1}$$

where $\boldsymbol{W}_f^{(i)}$ and b_f are the corresponding weights and bias in the fusion layer, and N is the number of sub-CNNs.

A. Siamese Architecture

Each sub-CNN in CFN has a composite structure of two identical sub-networks as illustrated in Fig. 2. Such a structure is termed as Siamese Architecture in [30], [13]. The two networks share the same weights, and define a mapping from the input feature space to a low-dimensional space where faces are close in terms of L_1 distance if they are of the same identity.

Each sub-network in the composite structure is a Convolutional Neural Network, for which we follow the standard configuration in [9]. Each CNN contains two convolution layers C_1 and C_2, each of which is followed by a max-pooling layer. The output of convolutional layer is passed through a non-linear activation function before being forwarded to the pooling layer. In our networks, we use rectified linear unit (ReLu). And the forward function can be represented as

$$h(\boldsymbol{x}_i) = max(0, \boldsymbol{W}_c^T \boldsymbol{x}_i + b_c), \tag{2}$$

where $\boldsymbol{W}_c$ and b_c represent the weight and bias of the corresponding convolutional layer. The last layer before softmax is a mapping layer L consisting of two fully-connected linear layers. And the output of this linear layer is the final representation for each face pair and can be computed as

$$L(\boldsymbol{x}_i) = ||g(\boldsymbol{F}_i^{(1)}) - g(\boldsymbol{F}_i^{(2)})||_1, \tag{3}$$

where $g(\cdot)$ represents the mapping from the input space to the final feature space, and $\boldsymbol{F}_i^{(1)}$ and $\boldsymbol{F}_i^{(2)}$ are the two faces in a pair.

The output of $L(\boldsymbol{x}_i)$ is finally forwarded to a softmax layer denoted as S. As a binary classification problem, the learnable weight of S is a two column vector $\boldsymbol{w}_s = \{\boldsymbol{w}_s^{(1)}, \boldsymbol{w}_s^{(2)}\}$. The posterior probability of $\boldsymbol{x}_i$ labeled as y_i is

$$Pr(y = y_i|\boldsymbol{w}_s, b_s, \boldsymbol{x}_i) = \frac{exp(-\boldsymbol{w}_s^{(y_i)} \cdot L(\boldsymbol{x}_i) + b_s)}{\sum_{j=1}^{2} exp(-\boldsymbol{w}_s^{(j)} \cdot L(\boldsymbol{x}_i) + b_s)}. \tag{4}$$

Accordingly, the cost function is formulated as follows

$$\mathcal{L} = -\sum_{i=1}^{n} \log\ Pr(y = y_i|\boldsymbol{w}_s, b_s, \boldsymbol{x}_i). \tag{5}$$

IV. Pose-invariant Patch Selection

To acquire the local information, sub-CNNs of CFN are pre-trained on the discriminative facial parts, and thus the selection of patches will largely affect the performance. A typical part-based approach is built on patches that are densely sampled with overlap as in [5], [3]. Intuitively, we can generate patches following the same strategy. However, there are mainly two reasons prohibiting us from doing so. First, such an approach will generate a huge network with an unaffordable number of sub-CNNs since each local patch requires one sub-network in Fig. 2. The unaffordable computation cost makes it infeasible to adopt this approach.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2015.2406191, IEEE Transactions on Circuits and Systems for Video Technology

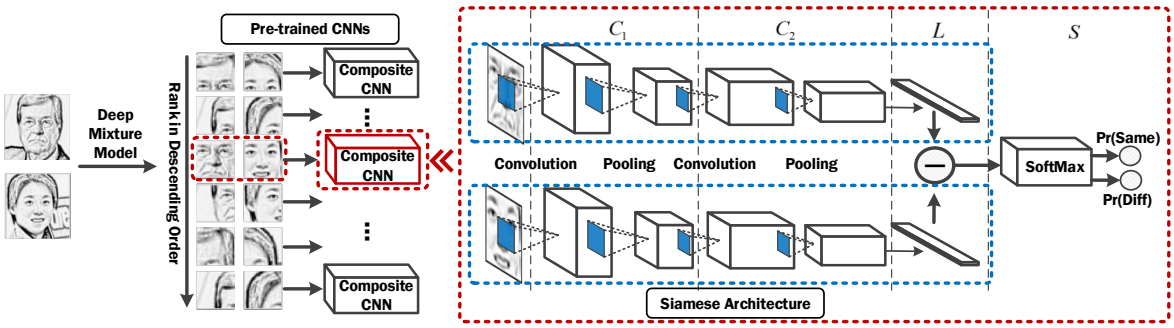


Fig. 2. Siamese architecture. Each sub-CNN corresponding to a local support patch is composed of two identical CNNs that share the same weights. Such identical CNNs define a mapping from the input space to a space for a better similarity measurement.

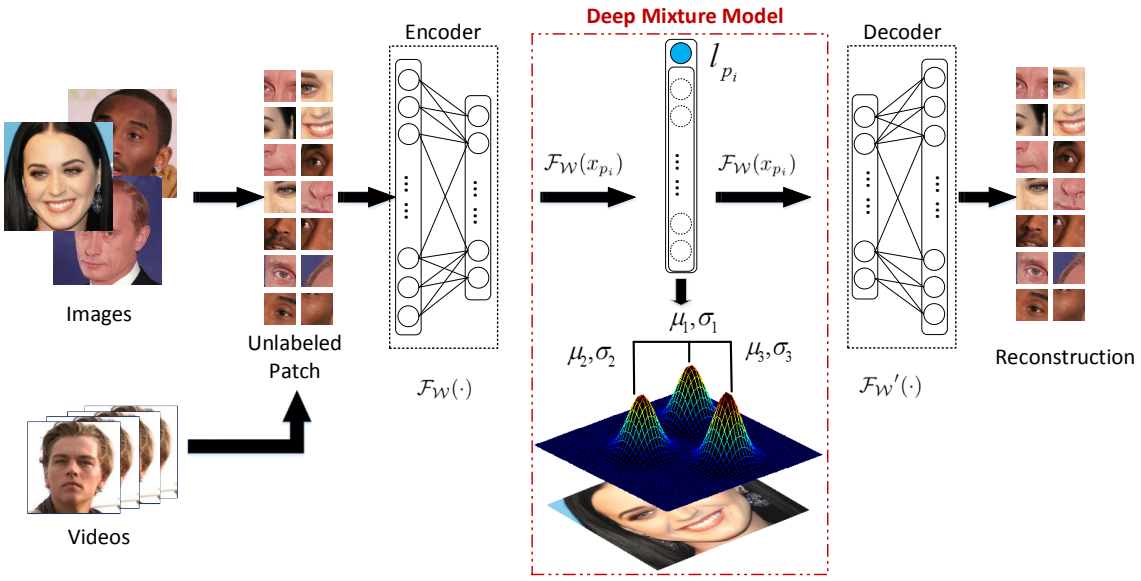


Fig. 3. DMM network structure. The proposed network is of an encoder-decoder structure similar to Autoencoder and is trained with unlabeled patches extracted from input images or videos. The encoded features are augmented with the corresponding location vectors and applied to train the mixture model. The mixture component and the encoding function are jointly learnt within the unified framework.

Second, large networks are difficult to train even if we ignore the computation cost. With too many parameters to learn, it is hard for the network to converge. Moreover, the optimization of the deep network is non-convex, and thus sensitive to the initialization of parameters. It easily falls into the "basin" of poor local minimum without a proper initialization.

Another way to utilize local information is to extract patches with regard to the key facial landmarks, such as eyes, nose, mouth, etc. This kind of approaches largely relies on the precision of landmark detectors. However, the unconstrained photography conditions still remain challenging for most existing landmark detectors. Moreover, accurate landmark detectors usually demand a large set of outside training samples, which are not always available. Thus, this strategy is prohibited for some datasets in the wild, e.g., LFW under the most restricted condition.

Our approach is built on the assumption that the face images are captured in the wild and no accurate landmarks are available, and thus the faces are only roughly aligned. The pose variation has proven to be an important factor impacting the face recognition accuracy. We propose to learn a Deep Mixture Model (DMM) to capture the spatial-appearance distribution over faces. By learning the mixture components, the correspondences of local patches are acquired to address the mis-matching brought by pose difference. Different from APEM [5], our deep network learns both the representation for appearance and the mixture components jointly without reference to any manually designed features.

A. Deep Mixture Model

Given a set of unlabeled images, we divide each image into multiple overlapped grids. The image set then can be represented as a collection of local patches $\{\boldsymbol{p}_1, \boldsymbol{p}_2, ..., \boldsymbol{p}_N\}$. Each local patch $\boldsymbol{p}_i$ is represented as a

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2015.2406191, IEEE Transactions on Circuits and Systems for Video Technology

spatial and appearance pair $[\boldsymbol{x}_{p_i}, \boldsymbol{l}_{p_i}]^T$, where $\boldsymbol{x}_{p_i}$ is the raw-pixel representation and $\boldsymbol{l}_{p_i}$ (each element is in $[0, 1]$) is the normalized location vector .

Different from most existing works for learning a mixture model, our approach does not rely on hand-crafted features. Instead, the representation is learnt together with the mixture components. Similar to Autoencoder, the DMM network contains an encoder and a decoder as shown in Fig. 3. The encoder maps the high-dimension data to a low-dimension code, and the decoder recovers the original input from the compressed code. In this work, the encoder is of a two layered structure: 800 hidden units for the first layer and 200 hidden units for the second layer. The decoder has a symmetric structure to the encoder. Also, the encoder and decoder have "tied" weights, i.e. the weight matrix for the decoder layer is the transpose of that for the corresponding decoder layer. The "encoded" feature is forward into the third layer, i.e. the mixture layer. The mixture layer is composed of multiple branches, each of which corresponds to a mixture component. The output of each component sub-net is the probability of certain patch committed to the corresponding component.

Assume the encoding function defined by the deep network is $\mathcal{F}(\cdot; \boldsymbol{W}_e, \boldsymbol{b}_e)$, where $\boldsymbol{W}_e$ and $\boldsymbol{b}_e$ stand for the encoding weight and bias. By augmenting the compressed code and the location vector, the combined spatial-appearance feature is represented as $\boldsymbol{f}_{p_i} = [\mathcal{F}(\boldsymbol{x}_{p_i}; \boldsymbol{W}_e, \boldsymbol{b}_e)^T, \boldsymbol{l}_{p_i}^T]^T$, which is then forwarded to the following mixture layer. We formulate the deep mixture model in terms of Gaussian components as follows,

$$Pr(\boldsymbol{f}_{p_i}|\boldsymbol{\theta}) = \sum_{j=1}^{C} \omega_j \cdot \mathcal{N}(\boldsymbol{f}_{p_i}|\boldsymbol{\mu}_j, \sigma_j), \tag{6}$$

where $\boldsymbol{\theta} = \{\boldsymbol{\mu}_i, \sigma_i | i = 1, 2, ..., C\}$, and $\boldsymbol{\mu}_i$ and σ_i are the mean and variance of the i-th component. $\mathcal{N}(\cdot)$ represents a normal distribution for the component with corresponding mixture weight w_i.

The DMM network is optimized by minimizing the following cost function

$$\begin{aligned}\mathcal{L}(\mathcal{W}, \boldsymbol{b}, \boldsymbol{\theta}) \quad = \quad & -\sum_{i=1}^{N} ln(P(\boldsymbol{f}_{p_i}|\boldsymbol{\theta})) \\ & -\sum_{i=1}^{N} ln \frac{\max_j \mathcal{N}(\boldsymbol{f}_{p_i}|\boldsymbol{\mu}_j, \sigma_j)}{\sum_{j=1}^{C} \mathcal{N}(\boldsymbol{f}_{p_i}|\boldsymbol{\mu}_j, \sigma_j)} \\ & +\sum_{i=1}^{N} \alpha ||\boldsymbol{x}_{p_i} - \boldsymbol{x}'_{p_i}||^2, \end{aligned} \tag{7}$$

where α is a parameter controlling the contribution scale of the third term, and $\boldsymbol{x}'_{p_i}$ is the reconstruction of $\boldsymbol{x}_{p_i}$ and is computed as

$$\boldsymbol{x}'_{p_i} = \mathcal{F}'(\mathcal{F}(x_{p_i}; \boldsymbol{W}_e, \boldsymbol{b}_e); \boldsymbol{W}'_e, \boldsymbol{b}'_e), \tag{8}$$

where $\mathcal{F}'(\cdot; \boldsymbol{W}'_e, \boldsymbol{b}'_e)$ is the decoding function with the corresponding decoder weight $\boldsymbol{W}'_e$ and bias $\boldsymbol{b}'_e$.

The cost in Eqn. 7 is defined based on considerations on the following three aspects. Same as the standard Gaussian Mixture Model, the first term is defined as the log likelihood function. For the second term, the proposed DMM aims to regularize that the spatial-appearance components correspond to different semantic facial parts, such as eyes, nose, etc. In other words, the learnt mixture components are expected to follow a spatially scattering distribution. Therefore, we introduce the second term to constrain that each sample is only committed to one component and its contribution to other components are neglectable. It is also important to note that, in DMM, the encoding of patches is jointly optimized with the component parameters. Directly optimizing with regard to the first and second terms will result in an undesired global minimum where both $\boldsymbol{W}_e$ and $\boldsymbol{b}_e$ are all zero for the encoder. Therefore, the third term is introduced to penalize the construction error such that the representations of face patches are not mapped into the undesirable all-zero space.

The mixture parameters are only present in the mixture layer, and thus are independent of the reconstruction error. Accordingly, $\boldsymbol{\mu_k}$ and σ_k can be updated directly as follows.

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial \boldsymbol{\mu}_k} = (-\frac{w_k}{P(\boldsymbol{f}_{p_i}|\boldsymbol{\theta})} + \frac{1}{\sum_{j=1}^{C} \mathcal{N}(\boldsymbol{f}_{p_i}|\boldsymbol{\mu}_j, \sigma_j)} \\ - \frac{\mathbb{1}_{j=k}}{\max_j \mathcal{N}(\boldsymbol{f}_{p_i}|\boldsymbol{\mu}_j, \sigma_j)}) \cdot \frac{\partial \mathcal{N}(\boldsymbol{f}_{p_i}|\boldsymbol{\mu}_k, \sigma_k)}{\partial \boldsymbol{\mu}_k}, \end{aligned} \tag{9}$$

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial \sigma_k} = (-\frac{w_k}{P(\boldsymbol{f}_{p_i}|\boldsymbol{\theta})} + \frac{1}{\sum_{j=1}^{C} \mathcal{N}(\boldsymbol{f}_{p_i}|\boldsymbol{\mu}_j, \sigma_j)} \\ - \frac{\mathbb{1}_{j=k}}{\max_j \mathcal{N}(\boldsymbol{f}_{p_i}|\boldsymbol{\mu}_j, \sigma_j)}) \cdot \frac{\partial \mathcal{N}(\boldsymbol{f}_{p_i}|\boldsymbol{\mu}_k, \sigma_k)}{\partial \sigma_k}. \end{aligned} \tag{10}$$

The optimization of $\boldsymbol{W}$ and $\boldsymbol{b}$ can be easily achieved with the standard back-propagation algorithm.

B. Local Patch Matching

The acquired DMM reflects the distribution of spatial and appearance feature over the faces. By assigned each face patch to its "Nearest" mixture component, we are able to cluster the patches in terms of the encoded similarity. Within each face pair, face patches with the maximal responses to the same mixture component are considered as matched. Therefore, the number of components determines the number of sub-nets that need to be pre-trained. Large number of chosen patches will result in a huge computation cost. Instead, we consider that not all the patches will contribute to the final verification problem. Therefore, it is desirable to retain only those discriminative patches without impacting the generalized performance.

This task can be interpreted as a feature selection problem [34], [35], which selects a subset of features while preserving or improving the discriminative ability of the classifier. Suppose we are given n training samples $\{(\boldsymbol{x}_1, y_1), ..., (\boldsymbol{x}_n, y_n)\}$, where $\boldsymbol{x}_i \in D^d$ and $y_i \in \{-1, +1\}$ is the label of $\boldsymbol{x}_i$. For face verification, the training samples are

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2015.2406191, IEEE Transactions on Circuits and Systems for Video Technology

Fig. 4. Convolutional kernels computed. Each block corresponds to a selected patch with its learnt convolutional kernels in the first layer. Clearly, the learnt kernels are different for different facial patches.

given in pairs. The task is to tell whether or not the paired samples (probe and gallery) are of the same identity. We denote $\boldsymbol{F}_i^{(1)}$ and $\boldsymbol{F}_i^{(2)}$ as the first and second face in the i-th pair. The input vector for the feature selection process is computed by $\boldsymbol{x}_i = |\boldsymbol{F}_i^{(1)} - \boldsymbol{F}_i^{(2)}|$, where $|\cdot|$ computes the element-wise absolute value.

In both [35] and [34], an indicator vector $\boldsymbol{\delta} = \{\delta_1, ..., \delta_d\} \in \{0,1\}^d$ is introduced to define whether a certain feature $\boldsymbol{x}_i^{(j)}$ is selected, i.e. $\delta_j = 1$ indicates it is a "support feature". Instead of finding the pixel-wise discriminative features as in [34], [35], we aims to select the discriminative patches. With the learnt C-component DMM, each face pair is represented as a concatenated vector $\boldsymbol{A}_i = \{\boldsymbol{p}_i^{(1)}, \boldsymbol{p}_i^{(2)}, ..., \boldsymbol{p}_i^{(C)}\}$.

$$\boldsymbol{p}_i^j = \arg\max_{\boldsymbol{p}_k} \left(\mathcal{N}\left(\boldsymbol{f}_{p_k}|\boldsymbol{\mu}_j, \sigma_j\right)\right) \quad \forall \boldsymbol{p}_k \in \boldsymbol{x}_i, \tag{11}$$

Accordingly, the weight vector of SVM is divided as $\boldsymbol{w} = \{\boldsymbol{w}^{(1)}, ..., \boldsymbol{w}^{(C)}\}^T$. In this work, we simplify the problem by eliminating the indicator vector. Now the problem is transformed into a classic SVM issue. The classifier is

$$f(\boldsymbol{A}_i) = \boldsymbol{w}^T \boldsymbol{A}_i + b, \tag{12}$$

where b is the bias.

Note that a pixel in the original image may be included in multiple patches. By minimizing the L_2 term $||\boldsymbol{w}||^2$ in the cost function, the corresponding duplicate pixels are assigned with the same weight if no individual normalization within each patch. Therefore, the discriminative scores of the duplicates in different patches are consistent. We define the discriminative score as the overall contribution of pixels within the patch to the decision boundary. The discriminative score $S^{(i)}$ of patch $\boldsymbol{p}^{(i)}$ is computed as

$$S^{(i)} = ||\boldsymbol{w}^{(i)}||_1. \tag{13}$$

Patches are then sorted in terms of the corresponding discriminative scores, and the top K patches are chosen as support patches.

Support patches tend to be those containing key facial components closely related to face identification, such as eyes and forehead. While, least informative patches include little information on either the outline of faces or key facial landmarks.

V. Training the Networks

The whole framework can be largely divided into two parts: 1) Deep Mixture Model to find the patch correspondence and 2) Convolutional Fusion Network for face verification. Both networks are large and hard to train directly without getting stuck at undesired local minimum. Erhan et al. [36] mentioned that pre-training provides a prior knowledge that can help reduce the strong dependencies between parameters across layers and locates the network in a region within the parameter space, such that a better optimum is found for the training criterion. We include some details on the training strategies for both networks as follows.

DMM. An initial representation is essential to avoid undesired clustering performance for appearance-wise DMM. This paper follows standard unsupervised pre-training methods used for Autoencoder. The network is pre-trained layer-by-layer with regard to the squared reconstruction error, i.e. the third term in Eqn. 7. For training the DMM network, we also need proper initialization for the location vectors. The location related part in $\boldsymbol{\mu}_i$ is initialized randomly with regard to a uniform distribution over $[0, 1]$. Moreover, for the starting 5 iterations, the encoder parameters ($\boldsymbol{W}_e$ and $\boldsymbol{b}_e$ in the 1st and 2nd layer) are not updated. In such a way, we acquire a proper geometric initialization for the mixture components.

CFN. Convolutional Fusion Network is initialized with the supervised pre-training. Selecting local patches can be viewed as a way of obtaining a good prior for the later fine-tuning stage. The pair of local patches shares the same label as the full-face pair, i.e. patches generated from the "matched" face pairs are also labeled as "matched". Therefore, each sub-CNN in the local layer can be pre-trained with the label information. After the supervised pre-training, the outputs of all the sub-CNNs are concatenated as a super-vector for each face instance, which is then fed forward to the fusion layer. A universal fine-tuning is then applied with back propagation through the whole network.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2015.2406191, IEEE Transactions on Circuits and Systems for Video Technology

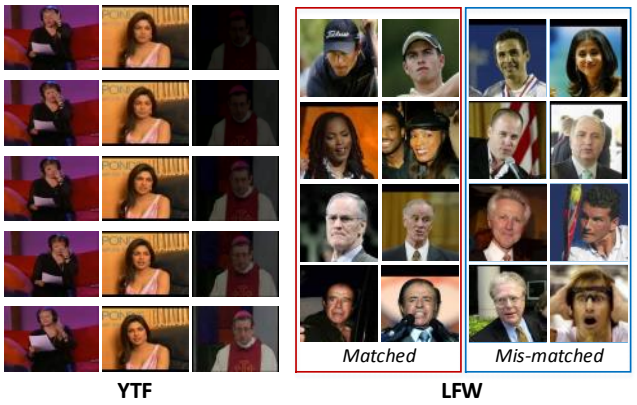


Fig. 5. Examples from YTF (left) and LFW (right). Both datasets include variations on pose, illumination and facial expressions that has large influence on the matching performance. Moreover, occlusion, frame blur and scene transition, which are common in videos, make YTF even more challenging.

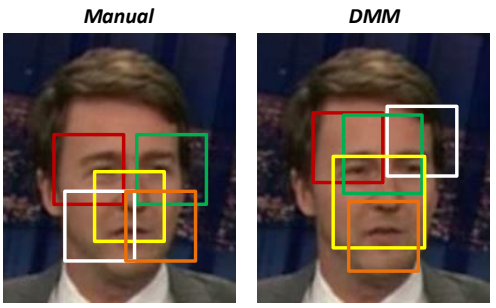


Fig. 6. Illustration on manual patches (Left) and DMM patches (Right). Since faces are aligned roughly, we extract patches around eyes, nose and mouth corners with fixed locations. For DMM, the locations are learnt automatically w.r.t the spatial-appearance distribution. Compared with manual approach, DMM demonstrates a better tolerance to pose changes.

Experiments show that the final fusion stage results in a considerate performance improvement.

VI. Experiments

The proposed network is aimed at face verification under the unconstrained conditions with considerate variations on pose and illumination. Extensive experiments are conducted on two benchmark datasets for face verification in the wild – YouTube Faces Dataset (YTF) and Labeled Face in the Wild (LFW). Examples of YTF and LFW is shown in Fig. 5. The results are compared with several state-of-the-art approaches.

A. YouTube Faces Database

YTF is a dataset designed for studying the problem of unconstrained face verification in videos. YTF contains $3,425$ unconstrained videos of $1,595$ celebrities. In the standard protocol, the evaluation set is composed of $5,000$ pre-defined video pairs and is divided into 10 mutually exclusive folds. The average verification accuracy of 10 folds is reported for comparison.

1) Experiment Settings: We address the problem of verification of two face videos as the matching problem of two sets of frames. Specifically speaking, 20 frames are drawn randomly from each video within the pair to generate 20 frame pairs. The average matching score of the 20 frame pairs is taken as the matching score of the corresponding video pair. In the following experiments, we directly take the roughly aligned faces provided. Within each frame, the face is cropped from the center down-scaled by 2.2 and is of size 144×144. The face images are then processed with two common illumination correction methods – Histogram Equalization (HE) and Local Ternary Pattern (LTP) [18]. For LTP, the gamma parameter is set as 0.2, and the sigma values for inner and outer Gaussian filter are set as 0.2 and 1, respectively. Together with RGB images, three copies of each images are adopted as inputs.

Pre-processed face images are scanned by sliding windows of size 40×40 and 60×60. The corresponding sliding strides are 20 and 30 pixels, respectively. Thus, we extract 44 local patches in each face image. These patches are resized to 32×32, and used as inputs of the DMM network.

CFN Structure. The whole network contains 18 sub-nets of Siamese Architecture in the local layer and a linear layer followed by a softmax layer in the fusion layer. Each sub-network i has a four-layer structure consisting of two convolutional layers $C_1^{(i)}$ and $C_2^{(i)}$, one linear layer $L^{(i)}$ and one softmax layer $S^{(i)}$. $C_1^{(i)}$ contains 40 convolutional kernels with size 7×7, and $C_2^{(i)}$ has 40 kernels of size 5×5, and $L^{(i)}$ has 100 neurons. Both convolutional layers are followed by max-pooling of shape 2×2 with pooling stride 2×2.

Examples of learnt convolution kernels are shown in Fig. 4. The convolutional kernels are learnt to reflect the discriminative information for the given local regions. For patches with complex facial structure (Full Face and patch 2), there are more high frequency kernels. While, for less complex patches (Patch 3, 4 and 5), the learnt kernels are mostly edge-like filters.

To further reduce over-fitting, drop-out [37] is applied on each layer of sub-CNNs, except for the softmax layer. The drop-out rate is 0.2 for convolutional layers $C_1^{(i)}$, $C_2^{(i)}$ and the linear layers $L^{(i)}$. We also include random noises in the input images, and the corruption probability of a single pixel is 0.1.

2) Comparison with the State-of-the-arts: The proposed approach, i.e. *DMM+CFN(3)*, is compared with several existing works reported on YTF in table I. Moreover, we include the results of four approaches related to our method for self comparison.

CNN_Single shows the result of single CNN trained only with the full face images. *CFN_Manual* includes the local information by fusing local CNNs trained with manually selected patches. The patches are chosen intuitively around eyes, nose and mouth corners as shown in Fig. 6. Comparison between *CNN_Single* and *CFN_Manual*

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2015.2406191, IEEE Transactions on Circuits and Systems for Video Technology

TABLE I
COMPARISON OF MEAN ACCURACY AND STANDARD VARIANCE ON YOUTUBE FACES DATABASE. THE BEST PERFORMANCE IS ILLUSTRATED IN BOLD.

Methods	Acc. ± Err.(%)
MBGS L2 mean, LBP [1]	76.4 ± 1.8
MBGS+SVM [38]	78.9 ± 1.9
APEM-FUSION [5]	79.1 ± 1.5
STFRD+PMML [4]	79.5 ± 2.5
VSOF+OSS [39]	79.7 ± 1.8
DDML (LBP) [32]	81.3 ± 1.6
DDML (combined) [32]	**82.3** ± 1.5
CNN_Single	78.3 ± 1.4
CFN_Manual	79.6 ± 1.2
DMM+CNN_Average	79.5 ± 1.2
DMM+CFN (1)	80.9 ± 0.9
DMM+CFN (3)	**82.8** ± 0.9

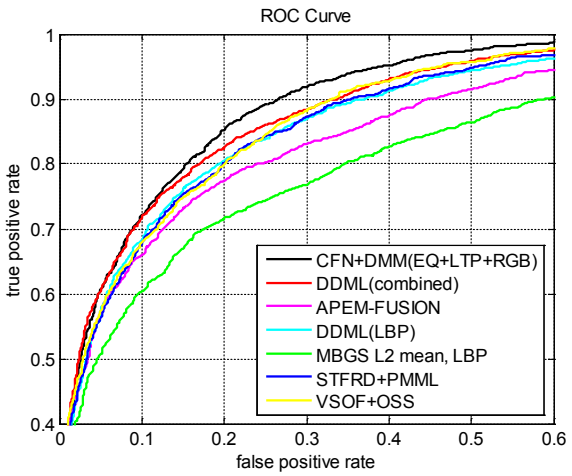


Fig. 7. Comparison of ROC curves with the state-of-the-arts on YouTube Faces Database.

indicates that the local information can bring considerate improvements (1.3% in our experiments) over holistic only approach. *DMM+CNN_Average* simply averages over pre-trained local CNNs. Different from *CFN_Manual*, local CNNs in this methods are learnt from patches acquired with the deep mixture model. As shown in the table, such simple approach can achieve almost the same performance as *CFN_Manual*. The performance is further improved by including the fusion stage into the learning process. *DMM+CFN(1)* is conducted on the images with only histogram equalization and improves *DMM+CNN_Average* by 1.4%. Fusion of more models is shown to be effective. The images used in *DMM+CFN(3)* are pre-processed with HE and LTP, respectively. Together with the original RGB images, the fusion model improves over single illumination based method *DMM+CFN(1)* by 1.9%.

Comparing with the state-of-the-art method on YTF – *DDML (combined)*, our approach improves the performance by 0.5%. *DDML (combined)* is also based on deep learning, but the networks learn a Mahalanobis distance metric from the hand-crafted features (LBP, DSIFT and SSIFT). However, our fusion network is directly learnt on the raw-pixel images.

The ROC curve is illustrated in Fig. 7. Consistent with the comparisons in Table I, our approach outperforms the existing methods reported on YTF.

Here we also list some of the latest results published after our submission. Li et al. [40] proposed the Eigen-PEP model for video face recognition, and achieved 85.04 ± 1.49 on YTF and 88.97 ± 1.32 on LFW. In [40], the performance is largely improved by including flipped frames and corrected labels, which are not used in our method. The accuracy without flipping is 82.40 ± 1.7, which is close to our results. Hu et al. [41] learnt the distance metrics form multiple features and achieved 81.28 ± 1.17 on YTF. Lu et al. [42] applied a reconstruction criterion to metric learning and achieved 81.86 ± 1.55.

B. Labeled Face in the Wild

LFW is a standard database collected to evaluate benchmark algorithms for face verification. It contains $13,000$ images of $5,749$ individuals downloaded from the Internet. LFW has the similar evaluation protocols as YTF: $6,000$ pre-defined image pairs are divided into 10 mutually exclusive folds and the average precision is reported.

1) Experiment Settings: In this paper, the experiments are conducted in the image-restricted scenario, i.e. only the given $6,000$ pairs are allowed for training. We follow the most strict setting, i.e. no outside training data are used, even for landmark detection. The face images are only roughly aligned with an unsupervised method – deep funnel [43]. We crop the central 144×144 region from the full-face image. DMM follows the same patch extraction strategy as that used for YTF.

Three general approaches of illumination correction are applied – Self-Quotient Image (SQI) [32], Local Ternary Pattern (LTP) [13] and Histogram Equalization (HE) . In SQI, the images are filtered with 7x7 Gaussian filter with bandwidth set as 2 and then normalized. The parameters for LTP are the same as those in YTF.

CFN Structure. The local networks are also of four layered structure – 20 convolutional kernels in $C_1^{(i)}$, 40 kernels in $C_2^{(i)}$, 100 hidden units in $L^{(i)}$ and a Softmax layer $S^{(i)}$. For LFW, we select the top-6 patches, and thus the final CFN is composed of 21 CNNs in the local layer.

2) Comparison with the State-of-the-arts: In this subsection, our approach is compared with some existing methods with the same setting, i.e. the image-restricted setting without outside training data. The only exception is NReLu [13], in which face images are well-aligned and outside data are used for unsupervised pre-training. This approach built a DBN of siamese architecture, and thus is closely related to our method.

Table II shows the results of five different settings related to the proposed network. The number after each setting indicates the number of illumination correction methods included – for the 2-correction case images are pre-processed with only SQI and LTP. *CNN_Single(2)* reports the result of training CNNs only on the full-face images. Under this

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2015.2406191, IEEE Transactions on Circuits and Systems for Video Technology

TABLE II
COMPARISON OF MEAN ACCURACY AND STANDARD VARIANCE ON LABELED FACE IN THE WILD. THE BEST PERFORMANCE IS ILLUSTRATED IN BOLD.

Methods	Acc. ± Err.(%)
NReLu [13]	80.73 ± 1.34
NReLu without Outside Data [13]	79.25 ± 1.73
Hybrid descriptor-based [44]	78.47 ± 0.51
V1/MKL [45]	79.35 ± 0.51
APEM(LBP) [5]	81.97 ± 1.90
APEM(SIFT) [5]	81.88 ± 0.94
APEM(fusion) [5]	**84.08** ± 1.2
Fisher Vector [3]	**87.47** ± 1.49
CNN_Single(2)	80.59 ± 1.54
CFN_Manual(2)	82.05 ± 1.6
DMM+CNN_Average(2)	83.93 ± 1.75
DMM+CFN (2)	85.48 ± 1.64
DMM+CFN (3)	**85.60** ± 1.67
CFN+APEM	**87.50** ± 1.57

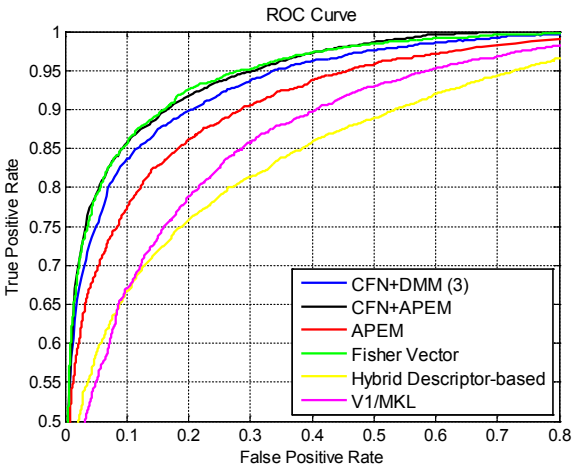


Fig. 8. Comparison of ROC curves with the state-of-the-arts on the most strict setting of Labeled Face in the Wild.

scenario, the fusion network only has two sub-CNNs on the full-face images after SQI and LTP respectively. The accuracy outperforms that of *NReLu* without unsupervised pre-training, and is comparable to their best performance with unsupervised pre-training based on outside unlabeled data. *DMM+CNN_Average(2)* simply averages over the confidence scores returned by pre-trained sub-CNNs. Performance with such a setting is even comparable with *APEM (fusion)* – only 0.1% difference. Further improvement is achieved by holistic back-propagation over the whole network, as shown by *DMM+CFN(2)*. The increase on mean accuracy is 1.55%, and can be up to 2.6% for some folds. The best results are achieved by fusion with all three illumination correction methods as shown for *DMM+CFN(3)*.

APEM [5] is also based on selection of patches, and our method surpasses *APEM* with a single feature, either SIFT or LBP, by around 3.6%. The advance over APEM with feature fusion is 1.52%. There is a gap of 1.9% between fisher vector [3] and our method alone. However, by simply averaging with the results of APEM – *CFN+APEM*, we achieve the accuracy of *Fisher Vector*. The improvement by simply averaging with APEM demonstrates the features learnt in our fusion network is different from the hand-crafted features. Note that both *APEM* and *Fisher Vector* are built on images of large size (100×100 in APEM and 160×125 in FV), while our fusion network is only trained on images of small size 32×32.

The ROC curve in Fig. 8 illustrates the average performance over 10 folds. It is clear that our method outperforms APEM significantly and achieves a comparable performance with *Fisher Vector*.

3) Fusion Result Analysis: We conduct two sets of experiments to analyze the effect of several factors on fusion. The first set fuses the local patches with the full-face images. The second set studies the fusion among only the local patches. For each experiment set, we include three groups tested on the images after SQI, images after LTP and images after both SQI and LTP (*Combined* in Table III), respectively. We also examine the influence of local patches in fusion by varying the number of patches included. These patches are added in the descending order with regard to their confidence scores defined by Eqn. 13.

Referring to the results in Table III, sub-CNNs trained with full-face images have a considerate influence in fusion. Fusion with full-face images outperforms fusion with only local patches by approximately 1.1%. Note that the local patches also demonstrate great influence. Generally, more local patches lead to higher accuracy in both experiment sets. As more patches are included, the performance gradually saturates. Fusing different pre-processing methods also contributes to the final fusion performance, and the increase on accuracy is around 1%.

C. Computation Analysis

The proposed framework can be divided into two parts – DMM and CFN. Both networks are implemented based on Theano[1] and Pylearn2[2]. All experiments are conducted on a single-core computer with GeForce GTX TITAN Black GPU. For both data sets, we extract 44 local patches from each face image, and random sample 60,000 patches for YTF and 45,000 patches for LFW as the inputs for DMM, respectively. In YTF, the training set of CFN includes 4,500 video pairs. Within each video pair, 20 frame pairs are randomly chosen. Accordingly, DMM takes 45s per iteration in training and CFN takes 33s per iteration for each sub-net. In LFW, the training set includes 5,400 image pairs for CFN. We also include random shifting, scaling and rotation to increase the diversity and scale of the training samples. As a result, the network is trained with 21,600 image pairs in total. Accordingly, DMM takes 36s per iteration in training and CFN takes 9s per iteration for each sub-nets. For faster computation, we can fix the convolution

[1] http://deeplearning.net/software/theano/
[2] http://deeplearning.net/software/pylearn2/

TABLE III
FUSION RESULTS. IN EACH EXPERIMENT SET, RESULTS ARE REPORTED BY VARYING THE NUMBER OF LOCAL PATCHES INCLUDED. 0 MEANS ONLY THE FULL-FACE IMAGES ARE USED FOR TRAINING.

Patch #	Full-face Included			Without Full-face		
	SQI	LTP	Combined	SQI	LTP	Combined
0	80.11 ± 1.73	81.07 ± 1.01	82.45 ± 1.40	-	-	-
1	81.67 ± 1.24	83.14 ± 1.61	84.48 ± 1.42	78.18 ± 1.54	77.92 ± 2.48	80.10 ± 2.10
2	83.25 ± 1.75	83.55 ± 1.49	85.18 ± 1.90	82.37 ± 2.27	81.95 ± 2011	84.35 ± 2.26
3	83.24 ± 1.72	83.67 ± 1.65	84.92 ± 1.72	82.33 ± 1.67	82.20 ± 2.02	83.98 ± 1.73
4	83.09 ± 1.94	83.7 ± 1.76	85.15 ± 1.46	82.27 ± 1.92	82.60 ± 2.33	84.18 ± 2.68
5	**83.34** ± 1.89	83.74 ± 1.76	85.24 ± 1.46	**82.38** ± 1.93	**83.10** ± 2.35	**84.50** ± 2.40
6	83.21 ± 1.95	**83.74** ± 1.69	**85.48** ± 1.64	82.10 ± 2.30	82.43 ± 2.43	84.20 ± 2.12

layers in the sub-nets of CFN, and only fine-tune the later fully-connected layers as many previous papers did. The corresponding results are only slightly degraded. The reported results are derived by setting the maximal training iteration number as 160 for DMM and 120 for CFN, respectively.

VII. CONCLUSIONS

In this paper, we proposed a part-based learning scheme for face verification in the wild by introducing Convolutional Fusion Network. We fuse multiple sub-CNNs pre-trained on the local patches to take into account both local and holistic information. A deep mixture model is also proposed to further address the mis-alignment brought by pose variation. DMM captures the spatial-appearance distribution over faces to acquire the correspondences of the local patches. Without relying on the hand-crafted features, the proposed framework automatically learns an effective representation of face images to build an end-to-end system. We achieve the state-of-the-art performance with automatic feature learning in the two benchmark datasets in the wild.

REFERENCES

[1] L. Wolf, T. Hassner, and I. Maoz, "Face recognition in unconstrained videos with matched background similarity," in *Computer Vision and Pattern Recognition*. IEEE, 2011.

[2] G. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," Tech. Rep., 2007.

[3] K. Simonyan, O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Fisher Vector Faces in the Wild," in *British Machine Vision Conference*, 2013.

[4] 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 *Computer Vision and Pattern Recognition*, 2013.

[5] H. Li, G. Hua, Z. Lin, J. Brandt, and J. Yang, "Probabilistic elastic matching for pose variant face verification," in *Computer Vision and Pattern Recognition*, 2013.

[6] T. Ojala, M. Pietikäinen, and D. Harwood, "A comparative study of texture measures with classification based on featured distributions," *Pattern recognition*, vol. 29, no. 1, pp. 51–59, 1996.

[7] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[8] C. Liu and H. Wechsler, "Gabor feature based classification using the enhanced fisher linear discriminant model for face recognition," *IEEE Transactions on Image Processing*, vol. 11, no. 4, pp. 467–476, 2002.

[9] A. Krizhevsky, I. Sutskever, and G. Hinton, "Imagenet classification with deep convolutional neural networks," in *Neural Information Processing Systems*, 2012.

[10] C. Farabet, C. Couprie, L. Najman, and Y. LeCun, "Learning hierarchical features for scene labeling," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2013.

[11] G. B. Huang, H. Lee, and E. Learned-Miller, "Learning hierarchical representations for face verification with convolutional deep belief networks," in *Computer Vision and Pattern Recognition*, 2012.

[12] Y. Sun, X. Wang, and X. Tang, "Deep convolutional network cascade for facial point detection," in *Computer Vision and Pattern Recognition*, 2013.

[13] V. Nair and G. Hinton, "Rectified linear units improve restricted boltzmann machines," in *International Conference on Machine Learning*, 2010.

[14] Y. Sun, X. Wang, and X. Tang, "Deep learning face representation from predicting 10,000 classes," in *Computer Vision and Pattern Recognition*. IEEE, 2014.

[15] Y. Sun, Y. Chen, X. Wang, and X. Tang, "Deep learning face representation by joint identification-verification," in *Advances in Neural Information Processing Systems*, 2014.

[16] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "Deepface: Closing the gap to human-level performance in face verification," in *Computer Vision and Pattern Recognition*, 2014.

[17] J. Wright and G. Hua, "Implicit elastic matching with random projections for pose-variant face recognition," in *Computer Vision and Pattern Recognition*. IEEE, 2009.

[18] X. Tan and B. Triggs, "Enhanced local texture feature sets for face recognition under difficult lighting conditions," *IEEE Transactions on Image Processing*, vol. 19, no. 6, pp. 1635–1650, 2010.

[19] T. Ahonen, A. Hadid, and M. Pietikainen, "Face description with local binary patterns: Application to face recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 28, no. 12, pp. 2037–2041, 2006.

[20] S. Hussain, T. Napoleon, and F. Jurie, "Face Recognition using Local Quantized Patterns," in *British Machine Vision Conference*, 2012.

[21] M. A. Turk and A. P. Pentland, "Face recognition using eigenfaces," in *Computer Vision and Pattern Recognition*, 1991.

[22] K. Sikka, T. Wu, J. Susskind, and M. Bartlett, "Exploring bag of words architectures in the facial expression domain," in *European Conference on Computer Vision Workshops*, 2012.

[23] T.-K. Kim, H. Kim, W. Hwang, S. Kee, and J. Kittler, "Independent component analysis in a facial local residue space." in *Computer Vision and Pattern Recognition*, 2003.

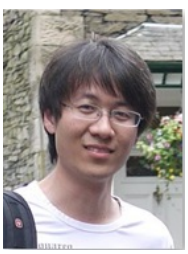
[24] T.-K. Kim, H. Kim, W. Hwang, and J. Kittler, "Component-based LDA face description for image retrieval and mpeg-7 standardisation," *Image and Vision Computing*, vol. 23, no. 7, pp. 631–642, 2005.

[25] X. Zhao, T.-K. Kim, and W. Luo, "Unified face analysis by iterative multi-output random forests," in *Computer Vision and Pattern Recognition*, 2013.

[26] P. Luo, X. Wang, and X. Tang, "Hierarchical face parsing via deep learning," in *Computer Vision and Pattern Recognition*, 2012.

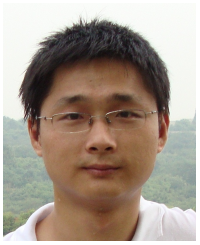
[27] P. Zhu, L. Zhang, Q. Hu, and S. Shiu, "Multi-scale patch based collaborative representation for face recognition with margin distribution optimization," in *European Conference on Computer Vision*, 2012.

[28] G. Hinton and R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, 2006.
[29] W. Ouyang and X. Wang, "Joint deep learning for pedestrian detection," in *IEEE International Conference on Computer Vision*. IEEE, 2013.
[30] S. Chopra, R. Hadsell, and Y. Lecun, "Learning a similarity metric discriminatively, with application to face verification," in *Computer Vision and Pattern Recognition*, 2005.
[31] Q. Liao, J. Z. Leibo, Y. Mroueh, and T. Poggio, "Can a biologically-plausible hierarchy effectively replace face detection, alignment, and recognition pipelines?" *arXiv preprint arXiv:1311.4082*, 2013.
[32] J. Hu, J. Lu, and Y.-P. Tan, "Discriminative deep metric learning for face verification in the wild," in *Computer Vision and Pattern Recognition*, 2013.
[33] G. Hua and A. Akbarzadeh, "A robust elastic and partial matching metric for face recognition," in *International Conference on Computer Vision*, 2009.
[34] J. Weston, S. Mukherjee, O. Chapelle, M. Pontil, T. Poggio, and V. Vapnik, "Feature selection for svms," in *Neural Information Processing Systems*, 2000.
[35] Y. Zhai, M. Tan, I. W. Tsang, and Y. Ong, "Discovering support and affiliated features from very high dimensions," in *International Conference on Machine Learning*, 2012.
[36] D. Erhan, Y. Bengio, A. Courville, P.-A. Manzagol, P. Vincent, and S. Bengio, "Why does unsupervised pre-training help deep learning?" *Journal of Machine Learning Research*, vol. 11, pp. 625–660, 2010.
[37] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," *arXiv preprint arXiv:1207.0580*, 2012.
[38] L. Wolf and N. Levy, "The svm-minus similarity score for video face recognition," in *Computer Vision and Pattern Recognition*. IEEE, 2013.
[39] H. Mendez-Vazquez, Y. Martinez-Diaz, and Z. Chai, "Volume structured ordinal features with background similarity measure for video face recognition," in *International Conference on Biometrics*. IEEE, 2013.
[40] H. Li, G. Hua, X. Shen, Z. Lin, and J. Brandt, "Eigen-pep for video face recognition," *Asian Conference on Computer Vision*, 2014.
[41] J. Hu, J. Lu, J. Yuan, and Y.-P. Tan, "Large margin multi-metric learning for face and kinship verification in the wild," in *Asian Conference on Computer Vision*, 2014.
[42] J. Lu, G. Wang, W. Deng, and K. Jia, "Reconstruction-based metric learning for unconstrained face verification," *Information Forensics and Security, IEEE Transactions on*, 2015.
[43] G. Huang, M. Mattar, H. Lee, and E. Learned-Miller, "Learning to align from scratch," in *Neural Information Processing Systems*, 2012.
[44] L. Wolf, T. Hassner, and Y. Taigman, "Descriptor based methods in the wild," in *European Conference on Computer Vision workshop*, 2008.
[45] N. Pinto, J. DiCarlo, and D. Cox, "How far can you get with a modern face recognition test set using only simple features?" in *Computer Vision and Pattern Recognition*, 2009.



Chao Xiong received his MSc degree in Communication and Signal Processing from Imperial College London, UK, in 2011. Currently, he is a PhD candidate at the Department of Electrical and Electronic Engineering, Imperial College London, UK. His research interests include computer vision and pattern recognition.

Luoqi Liu is currently working toward the Ph.D. degree with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore. His research interests include computer vision, multimedia and machine learning.



Xiaowei Zhao received the Ph.D. degree in computer science from the Institute of Computing Technology (ICT), Chinese Academy of Sciences (CAS), Beijing, China, in 2013. He is a Post-doctoral researcher in Imperial College London from June 2013. His research interests include computer vision, pattern recognition. He especially focuses on face detection and face alignment, *etc*.

Shuicheng Yan is currently an Associate Professor in the Department of Electrical and Computer Engineering at National University of Singapore, and the founding lead of the Learning and Vision Research Group (http://www.lv-nus.org). Dr. Yan's research areas include computer vision, multimedia and machine learning, and he has authored/co-authored over 300 technical papers over a wide range of research topics, with Google Scholar citation over 12,000 times and H-index-47. He is an associate editor of IEEE Transactions on Circuits and Systems for Video Technology (IEEE TCSVT) and ACM Transactions on Intelligent Systems and Technology (ACM TIST), and has been serving as the guest editor of the special issues for TMM and CVIU. He received the Best Paper Awards from ACM MM12 (demo), PCM'11, ACM MM10, ICME10 and ICIMCS'09, the winner prizes of the classification task in PASCAL VOC 2010-2012, the winner prize of the segmentation task in PASCAL VOC 2012, the honorable mention prize of the detection task in PASCAL VOC'10, 2010 TCSVT Best Associate Editor (BAE) Award, 2010 Young Faculty Research Award, 2011 Singapore Young Scientist Award, 2012 NUS Young Researcher Award, and the co-author of the best student paper awards of PREMIA'09, PREMIA'11 and PREMIA'12.

Tae-Kyun Kim is a Lecturer in computer vision and learning at the Imperial College London, UK, since 2010. He obtained his PhD from the Univ. of Cambridge in 2008 and had been a Junior Research Fellow of Sidney Sussex College in Cambridge during 2007-2010. His research interests span various topics including: object recognition and tracking, face recognition and surveillance, action/gesture recognition, semantic image segmentation and reconstruction, and man-machine interface. He has co-authored over 40 academic papers in top-tier conferences and journals in the field, 6 MPEG7 standard documents and 17 international patents. His co-authored algorithm is an international standard of MPEG-7 ISO/IEC for face image retrieval.