Improving Face Recognition by Exploring Local Features with Visual Attention

Yichun Shi and Anil K. Jain
Michigan State University
East Lansing, Michigan, USA
shiyichu@msu.edu, jain@cse.msu.edu

Abstract

Over the past several years, the performance of state-of-the-art face recognition systems has been significantly improved, due in a large part to the increasing amount of available face datasets and the proliferation of deep neural networks. This rapid increase in performance has left existing popular performance evaluation protocols, such as standard LFW, nearly saturated and has motivated the emergence of new, more challenging protocols (aimed specifically towards unconstrained face recognition). In this work, we employ the use of parts-based face recognition models to further improve the performance of state-of-the-art face recognition systems as evaluated by both the LFW protocol, and the newer, more challenging protocols (BLUFR, IJB-A, and IJB-B). In particular, we employ spatial transformers to automatically localize discriminative facial parts which enables us to build an end-to-end network where global features and local features are fused together, making the final feature representation more discriminative. Experimental results, using these discriminative features, on the BLUFR, IJB-A and IJB-B protocols, show that the proposed approach is able to boost performance of state-of-the-art face recognition systems. The proposed approach is not limited to one architecture but can also be applied to other face recognition networks.

1. Introduction

Face recognition is an ongoing challenging problem in both computer vision and biometrics, due in a large part to a number of difficult issues such as pose, illumination, and expression variations, high inter-person similarity and occlusions. Thanks to the large face datasets in the public domain and rapid developments in deep convolutional neural networks, the state-of-the-art performance of unconstrained face recognition today is quite impressive and it continues to improve [20, 18, 16, 19, 17, 27, 15, 14]. On the standard LFW protocol [6] for face verification, which was regarded as a difficult task ten years ago, the performance of deep learning based methods have achieved accuracy over 99%, beating even human performance [10]. Due to this saturation of the old LFW protocol, more difficult and realistic face recognition challenges have recently been proposed, such as BLUFR [11], IJB-A [9] and IJB-B [22]. These newer protocols contain a larger number of test images, and the individual face images could be of low quality, larger pose variations and unfavorable illumination. In light of these new challenges posed by BLUFR, IJB-A, and IJB-B, we propose a new face recognition system to push state-of-the-art in face recognition performance. Our proposed method is inspired by the previous success of local parts-based face recognition systems and visual attention networks. In particular, we further improve the performance of state-of-the-art deep neural networks by fusing the semantic information from both the global features and automatically localized facial parts.

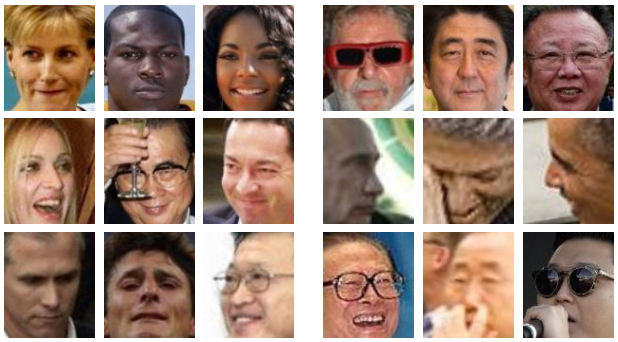

Figure 1: Example images in LFW and IJB-B after alignment using MTCNN [28]. The image in the first row are well aligned and all the facial parts are located in a consistent way. The face images in the second and third rows, although aligned, still appear in a quite different way because of large pose or occlusion.

Almost all face recognition systems include face alignment as a pre-processing step to ensure the input faces are in a similar position and orientation, reducing the intra-class variations and making the recognition task eas-

ier [20, 19, 27, 15, 14]. However, as the complexity of unconstrained face images increase, even though aligned, 2D face images can still appear very differently, as shown in Figure 1. As such, constructing global face models becomes a very difficult task. Because of this difficulty, an attractive idea is to model different facial parts individually and combine them to generate a global representation. Recognizing complex objects by their parts is a popular technique in pattern recognition. In the well-known Deformable Part Model (DPM) [2], different part filters are learned and combined with a root filter to detect complex objects in the images efficiently. Similar ideas, such as decomposing faces into different parts, have been shown to work well for face detection [12, 25, 26]. A highly successful, parts-based face recognition approach, called the DeepID series [18, 16, 19, 17], cropped a large number of different local patches either at fixed positions or around landmarks in the face image, trained a single deep convolutional network on each of these regions, and fused the representations from all the networks by training on a validation dataset. The success of works like DeepID indicate that although face is a nearly rigid object, building models for different face regions can also help improve the performance of face recognition systems.

One of the most important problems in parts-based face recognition approaches, is the localization of the target parts. In other words, although the faces are aligned, parts of a face shown in a fixed region could be quite different for different people at different poses, which reduces the discrimination ability of these parts-based models. One approach to solving this problem is to use the detected landmarks to crop rectangular patches around those respective landmarks. However, even with these landmarks, it is still difficult to decide what regions we should crop since some regions may be useful for recognition, and other may not. Given this difficulty, we turn to another technique to find and localize discriminative regions automatically that has become popular in the vision community, i.e. visual attention mechanism [1, 24, 3, 8].

By using a differentiable visual attention network, we can build an end-to-end system where the global recognition network and several parts-based networks are trained simultaneously. In this proposed end-to-end system, a fully connected layer for fusing features can be trained together with the recognition networks, which helps the subnetworks to explore more discriminative features complementary to the global representation. In addition, the visual attention network learns to localize distinct local regions automatically without any landmark supervision. Our experiments show that the proposed approach can further improve the state-of-the-art networks on challenging benchmarks such as BLUFR, IJB-A and IJB-B. More concisely, contributions of this paper can be summarized as follows:

- We designed an end-to-end face recognition system including global network, parts-based networks, attention network and a fusion layer that are trained simultaneously.
- We showed that discriminative regions can be be localized automatically without using facial landmarks by using a visual attention network.
- We showed that adding parts-based networks can further improve the performance of state-of-art deep networks on challenging protocols, including BLUFR, IJB-A and IJB-B, with little complexity increase.

2. Related Work

2.1. Parts-based Deep Face Recognition

Our proposed approach is predominantly inspired by the success of the DeepID series [18, 16, 19, 17]. In the first DeepId paper [18], ten different regions were cropped, respectively, from a face image (five large regions at fixed positions and five small regions around detected landmarks). For each region, RGB and gray-scale patches of five different scales were generated and each trained with a single convolutional neural network to output a feature vector of 160 dimensions. The features were then concatenated and the dimensionality was reduced with additional training on a validation set. In DeepID2, 400 patches at different positions, scales, color channels and horizontal flipping were cropped and used for training 200 different networks. After feature selection, 25 patches were selected to extract a $4,000$-dimensional feature vector, which was finally reduced to 180-dimensional vector with PCA. The authors showed that combining these features from different regions substantially improved the face recognition performance. In our work, unlike the DeepID methods, we combine all the elements (patch selection, sub-networks and feature fusion) together into an end-to-end system and train them simultaneously.

2.2. Visual Attention Network

Visual attention is a mechanism to automatically localize objects of interest in an image or parts of an object. Ba et al. [1] used a recurrent attention model to locate the objects in order to better perform multi-object classification. A similar scheme was used in [24] to generate captions for images. Xiao et al. [23] proposed to use visual attention proposals for fine-grained object classification by clustering the channels of a feature map into different groups and generating patches based on the activation of individual groups. In [3], a recurrent structure of a CNN and attention proposal network is proposed to zoom into small regions for fine-grained classification. The input of the attention network is the feature map of the last convolutional

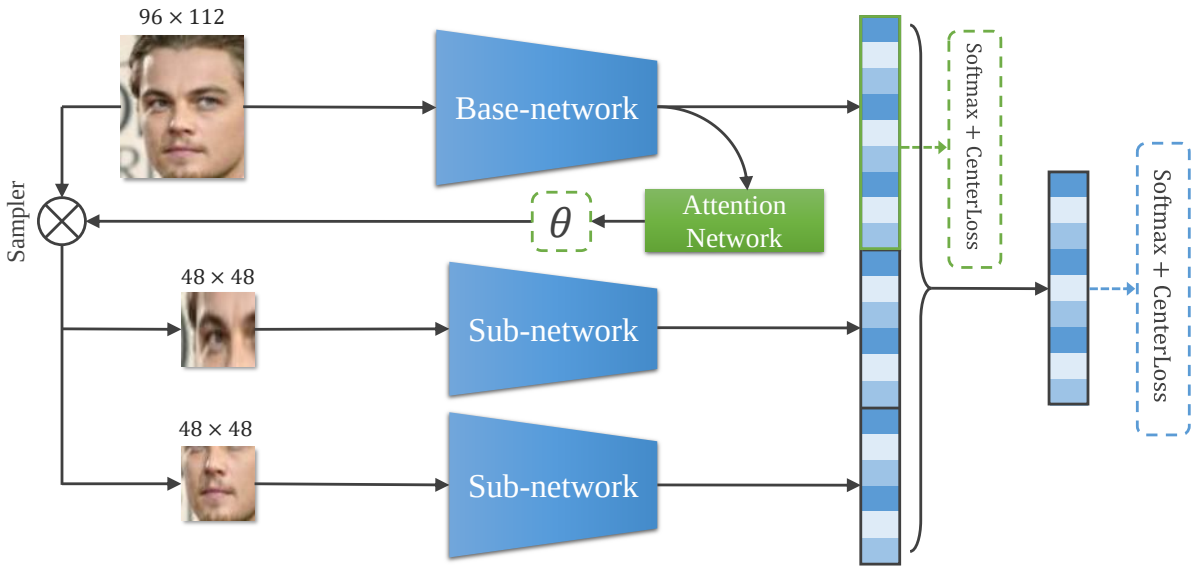

Figure 2: An example architecture of the proposed end-to-end network with $K = 2$ sub-networks. A 96×112 image is first fed into the base-network, which is a single CNN for face recognition. The feature map of the last convolutional layer of the base-network is then both used to learn a global representation with a fully connected layer, and K transformation matrices with an attention network of two-stacked fully-connected layers. The regions of interest are sampled into patches of size of 48×48. K smaller CNNs as sub-networks follow to learn local features from these automatically localized patches. All the global and local features are then concatenated and fused by another fully connected layer.

layer rather than raw images so that the computational cost can be reduced. We adopt a similar strategy in our network. Only two levels of CNNs are used in our approach but more than one patch is generated by the attention network. In addition, we use Spatial Transformers [8], which use a projective transformation matrix θ to transform the original input image, enabling us to better sample patches. By multiplying θ and the coordinates of pixels in the output image, the spatial transformer computes the corresponding coordinates of each pixel in the input image, and samples them through bi-linear interpolation. This transformer is differentiable, allowing the attention network to be learned end-to-end without labels. In [8], experiments showed that the spatial transformer network is able to automatically localize distorted digits, and street view house numbers. Subsequently, the performance of fine-grained classification is improved by generating multiple region proposals. Finally, Zhong et al. [29] showed that by training an attention network with spatial transformers, an end-to-end face recognition network which automatically learns the alignment can achieve comparable results to those with pre-aligned images.

3. Approach

In this section, we outline an end-to-end network which includes a *base-network* for learning a global representation from the whole face image, several *sub-network*s for modeling specific facial parts, an attention network for generating region proposals to feed into the sub-networks and a fusion layer to fuse the global and local features.

3.1. Overall Architecture

A graphic illustration of the overall architecture is shown in Figure 2. The input image size is 96×112. The proposed network begins with a base-network which can be any single convolutional neural network for face recognition. In particular, we employ the Face-ResNet proposed in [5] because of its good generalization ability and its state-of-the-art performance. In order to reduce the computational cost of the attention network, we adopt a similar approach as [3], where the attention network is connected to the last hidden convolutional layer rather than the input image. The attention network outputs K projective transformation matrices θ, each of which has 8 parameters. Here, K is a hyperparameter. For each of the K transformation matrices, a spatial transformer is used to sample a 48×48 patch from the region of interest via bi-linear interpolation. The sampled patch is then used by a smaller sub-network to learn local features. The global representation is of 512 dimensions, while the length of each local feature vector is 128 dimensions. All of them are concatenated together and fused by a fully connected layer to generate a 512-dimensional representation.

A softmax layer is added to both the global represen-

Type	Output Size
Batch Norm + Fully Connected	128
Batch Norm + Fully Connected	$8 \times K$

Table 1: The architecture of the attention network.

tation and the fused representation for classification in the training phase. Notice that the gradient is not propagated back through the fusion layer to the global representation. This allows the base-network to be trained independently, and it encourages the sub-networks to explore new features complementary to the global representation. Experimental result shows that such an approach enables the model to converge faster and leads to better generalizability. The softmax mainly learns to scatter the features of different classes, which is correspondent to the inter-class dissimilarity. Therefore, in order to reduce the intra-class variation, we also adopt the center loss proposed in [21] with the recommended setting of $\alpha = 0.5$ and $\lambda = 0.003$. The center loss is applied to both the global representation and fused representation.

3.2. Attention Network

Details about the attention network are shown in Table 1. Because the input to this network is the feature map of the last convolutional layer of the base-network that contains rich semantic information, the attention network is composed of only two fully-connected layers, saving a large amount of computational resources. We add a batch normalization layer [7] along with a ReLU activation layer [13] both before and after the first fully-connected layer to accelerate the training of attention network. The second fully connected layer outputs K transformation matrices. Then a spatial transformer module is used to sample the corresponding partial regions according to each of these matrices. Finally, there are several implementation subtleties to note.

First, because we are using a projective transformation, the sampled region is not restricted to be a rectangular shape. This means that the original image could be warped. However, Zhong et al. [29] showed that a better performance can be achieved with a projective transformation than a similarity transformation for face alignment. One plausible explanation for this is that neural networks do not perceive images in the same way as human do. As such, networks are able to learn better features from warped images.

Second, we multiply the learning rate of the attention network by 0.0001. Without performing this scaling, the output transformation deviates too much before the network is able to learn a set of reasonable parameters.

Third, the weights of the last fully connected layer are initialized as zero, while its biases are initialized as the flat-

Type	Output Size	Filter Size/Stride
Convolution	$48 \times 48 \times 32$	$3 \times 3/1$
Convolution	$48 \times 48 \times 64$	$3 \times 3/1$
Max Pooling	$24 \times 24 \times 64$	$2 \times 2/2$
Convolution	$24 \times 24 \times 64$	$3 \times 3/1$
Convolution	$24 \times 24 \times 128$	$3 \times 3/1$
Max Pooling	$12 \times 12 \times 128$	$2 \times 2/2$
Convolution	$12 \times 12 \times 96$	$3 \times 3/1$
Convolution	$12 \times 12 \times 192$	$3 \times 3/1$
Max Pooling	$6 \times 6 \times 192$	$2 \times 2/2$
Convolution	$6 \times 6 \times 128$	$3 \times 3/1$
Convolution	$6 \times 6 \times 256$	$3 \times 3/1$
Fully Connected	128	

Table 2: The architecture of the sub-networks.

ten vector of the initial K transformation matrices. In experiments, we use manual initialization for these matrices if K is small and random initialization if K is large.

3.3. Sub-Network for Modeling Facial Parts

Since the information in a local region is relatively small, it would be unnecessarily complex to use a network with as many parameters as the base-network to learn representations from these patches. As such, we use a simple architecture for all the sub-networks, as shown in Table 2. It is very similar to the network used in [27] except that it uses fewer layers. We add a fully connected layer at the end of the sub-network to learn a compressed local feature vector. Finally, we add a batch normalization along with a ReLU layer after every convolution and fully connected layer. Because the sub-networks take a smaller input and have much less parameters compared with base-network, they only add little extra run-time to the whole model, as shown in 4.1.

3.4. Promoting Sub-networks for Feature Exploration

Although theoretically the larger the number of sub-networks, the more complementary local features can be learned to improve the robustness of the fused representation, we find that the improvement of the performance after adding a large number of sub-networks is usually negligent. An explanation for this is found by the magnitude of the weights in the fusion layer for each dimension in the concatenated feature. Figure 3 shows that many local features have very small weights in the fusion layer. This indicates that there are some sub-networks which contribute little to the final fused representation. Additionally, this could diminish the loss propagated back to the base-networks and prevents the sub-networks from learning efficiently. As such, some sub-networks become "dead" during training. Therefore, inspired by [4], we add a promotion loss to explicitly promote the weights in the fusion layer for those local features. Notice that in [4], the promoted parameters are those related to a certain output class, however, in our case they are those related to a certain input dimension. In particular, let's denote an input feature vector as $\mathbf{x} = [\mathbf{x}^g, \mathbf{x}^l]$

where $\mathbf{x}^g$ is the global feature vector and $\mathbf{x}^l$ is the vector of all local features concatenated into one column. The fused representation y is obtained with a fully connected layer $y = W\mathbf{x} + \mathbf{b}$. Corresponding to x^g and x^l, W can be viewed as the concatenation of two matrices W^g and W^l, where

$$y = W^g\mathbf{x}^g + W^l\mathbf{x}^l + \mathbf{b} \tag{1}$$

The goal of the proposed promotion loss L_p is to encourage the local weights to be similar to the global weights:

$$L_p = \frac{1}{D_l}\sum_{i=0}^{D_l} \|\|W_i^l\|^2 - \alpha\|^2, \tag{2}$$

where

$$\alpha = \frac{1}{D_g}\sum_{i=0}^{D_g} \|W_i^g\|^2 \tag{3}$$

and D_l, D_g refer to the number of dimensions in the local and global feature vectors, respectively. W_i^l refers to the i_{th} column of W^l, similar for W_i^g. The promotion loss is added as a regularization loss with coefficient λ. As shown in 3, after adding promotion loss, the distribution of the weights in the fusion layer become much more uniform, thus avoiding the problem of "dead" sub-network and encouraging the sub-networks to find more discriminative features.

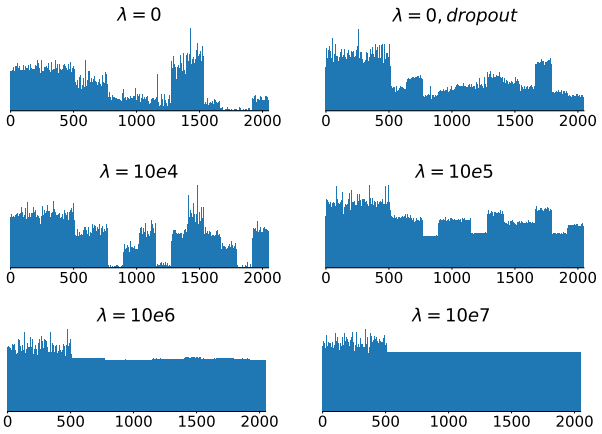

Figure 3: Magnitude of the weights of the fusion layer over different input dimensions when using different λ for the promotion loss. Without promotion loss, many dimensions have little weight, resulting in "dead" sub-networks. Dropout helps to promote the weights, but diminishes the performance.

4. Experiments

4.1. Implementation Details

. We conduct all of our experiments using Tensorflow 1.2. First, we implement the Face-ResNet in [5] as a baseline. We follow the same settings for the learning rate and center loss. All the images are first aligned using landmarks detected using MTCNN [28] and trained on the CASIA-Webface dataset [27]. The resulting network achieves a verification accuracy of 98.77% on the standard LFW protocol. This result is quite comparable to the performance originally reported in [5], however, we do note a slight drop in performance (from 99.00% to 98.77%). The most plausible explanation is that we are using a different library for implementation. All the following experiments are compared to this baseline result.

For the sub-networks, we adopt two schemes to initialize the transformation matrices θ:

- **Model A**: a small network with $K = 3$ rectangular regions initialized in the upper, middle and bottom face, respectively.
- **Model B**: a relatively larger network with K randomly initialized rectangular regions, whose widths and heights are between 30% and 60% of the original image.

The reason that we manually initialize Model A is that when K is rather small, the randomly initialized regions are not guaranteed to be distributed well. For example, they may have a large amount of overlap and only cover a small part of the entire face image. This would result in leaving behind crucial information useful for recognition. Therefore we manually choose three rectangular regions that cover different parts of the face for Model A.

We follow the same training settings as [5] with a batch size of 256 and $28,000$ training steps. The promotion loss weight is set to $\lambda = 10^5$ based on the results of a grid search. We use two Nvidia Gefore GTX 1080 Ti GPUs to train Model A and four for Model B. As for time complexity, there is only a slight increase in run-time: for base-network, Model A and Model B, it takes 0.003s, 0.003s and 0.004s per image to extract features with one GPU, respectively.

In order to evaluate the proposed method and implementation, we first study the effectiveness of the proposed modules using LFW dataset with both standard and BLUFR protocol [11]. Then we evaluate the proposed model on more challenging IJB-A [9] and IJB-B [22] benchmarks. Because the purpose of this paper is to present a system to improve any face recognition network instead of achieving the best result on these specific protocols, and since most results on the benchmarks are based on different architectures and training datasets, we believe it is not fair to compare the absolute performances. Thus, we only compare the relative performance of the proposed system with the original base-network.

4.2. Evaluation of Proposed Modules on LFW

In the proposed network, we use an attention network to localize K discriminative regions rather than cropping a

Figure 4: Example pairs that are misclassified by base-network but are classified correctly on LFW dataset. Pairs in the green box are genuine pairs and pairs in the red box are impostor pairs. We use the average threshold of BLUFR face verification for VR@FAR$= 0.1\%$ on 10 splits.

fixed patch, train a fusion layer to compress the concatenated feature and add promotion loss encouraging the sub-networks to explore more discriminative features. Here we evaluate the effectiveness of these modules by comparing the results with and without these modules on two protocols on LFW dataset: standard and BLUFR [11]. The standard verification protocol of the original LFW dataset contains only $6,000$ pairs of faces in all, which is insufficient to evaluate deep learning methods, evidenced by the fact that results are almost saturated on this protocol. Because of this, Liao et al. [11] made use of the whole LFW dataset to build the BLUFR protocol. In this protocol, a 10-fold cross-validation test is defined for both *face verification* and *open-set face identification*. For *face verification*, a verification rate (VR) is reported for each split with strict false alarm rate (FAR$= 0.1\%$) by comparing around $156,915$ genuine pairs and $46,960,863$ imposter pairs[1], which is more close to real-world scenario than the accuracy metric in the standard LFW protocol. For *open-set identification*, an identification rate (DIR) at Rank-1 corresponding to FAR$= 1\%$ is computed. We first test the performance of Model B withoug certain modules to ensure their effectiveness. Then we train the proposed Model A, Model B with all modules and compare them with base-network.

In Table 3, *Base-net* indicates the baseline single CNN network, which is used as the base-network in our model. *Attention Net* indicates whether an attention network is used to automatically localize the regions for sub-networks or crop the fixed regions that are randomly initialized. *Fusion Layer* indicates whether to add a fully connected fusion layer or directly use the concatenated layer as the representation. *Promotion Loss* means whether we add promotion loss as regularization to the fusion layer. The accuracy is tested on the standard protocol, while *Verification Rate* (VR) and *Detect and Identification Rate* (DIR) are tested on BLUFR protocol. Although all the results are similar on standard LFW protocol, distinct differences can be observed on BLUFR results. This is because standard protocol only contains $6,000$ pairs which is not adequate to precisely reflect the performance of a highly sophisticated model. Based on the results on BLUFR, we can see that Model B consistently outperforms base-network even without certain modules. And also every module is making a contribution and is essential to guarantee the final performance of the whole model. After using all modules, the proposed Model A and Model B surpasses the baseline by four percent in terms of DIR@FAR$= 1\%$ at rank-1. This demonstrates that the proposed idea of an auto-aligned parts-based model does improve the performance of a single neural network. And with more sub-networks added, Model B (12 sub-networks) consistently outperforms Model A (3 sub-networks).

Type	AN	FL	PL	Accuracy	VR @FAR$= 0.1\%$	DIR Rank-1 @FAR$= 1\%$
Base-net				98.77%	94.96%	72.96%
Model B	N	Y	Y	98.67%	95.54%	74.33%
Model B	Y	N	Y	98.78%	95.63%	76.37%
Model B	Y	Y	N	98.75%	95.83%	75.75%
Model A	Y	Y	Y	98.85%	95.90%	77.51%
Model B	Y	Y	Y	$\mathbf{98.98\%}$	$\mathbf{96.44\%}$	$\mathbf{77.96\%}$

Table 3: Evaluation results of the proposed model with/without certain modules on standard LFW and BLUFR protocols. "AN" means "Attention Network"; "FL" means "Fusion Layer"; "PL" refers to "Promotion Loss". "Y" indicates the module is used while "N" indicates that module is not used. Accuracy is tested on the standard LFW verification protocol. Verification Rate (VR) and Detection and Identification Rate (DIR) are tested on the BLUFR protocol.

To further evaluate the attention networks, we visualize the localized patches in the Model A. Some examples are shown in Figure 5. Notice the different distribution of facial parts, even after alignment, due to the challenging pose of the input image. The attention network can still accurately find the target facial parts. In the localized patches in the column, all the facial parts are distributed in a similar way. These accurately localized patches make it an easier task for the sub-networks to learn robust features from certain facial parts. The attention network also allows adjusting which part to localize so that the sub-networks can find more discriminative features. Notice that the attention network is trained without the landmark labels and as such, the computation is almost free.

[1] the numbers are averaged over ten splits.

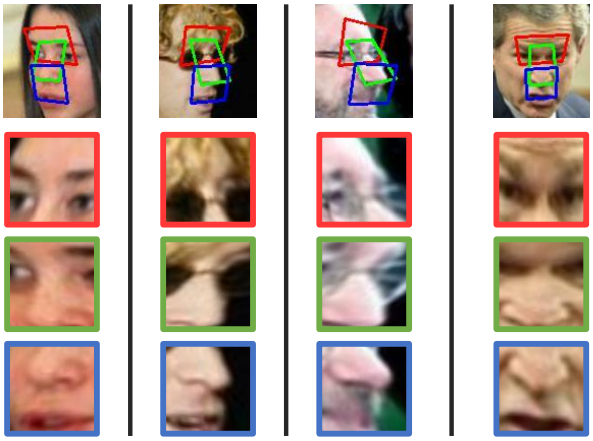

Figure 5: Examples of the localized regions in Model A. The attention network localizes the eyes, nose and mouth accurately by learning without landmark labels. These accurately localized patches make it an easier task for the sub-networks to learn robust features from certain facial parts.

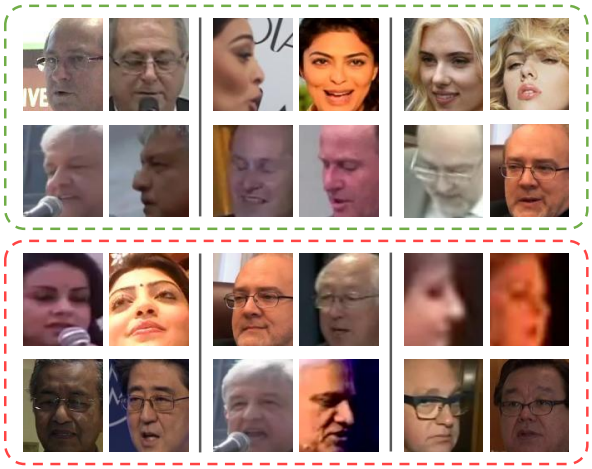

Figure 6: Example pairs that are misclassified by base-network but are classified correctly by Model B on IJB-B dataset. Pairs in the green box are genuine pairs and pairs in the red box are impostor pairs. We use the threshold of IJB-B 1:1 Baseline Verification for TAR@FAR$= 0.1\%$.

4.3. Evaluation on IJB-A and IJB-B Benchmarks

Recently, the IARPA Janus Benchmarks, including IJB-A and IJB-B, were released to push forward the frontiers of unconstrained face recognition systems. In IJB-A, a manually labeled dataset containing images both from photos and video frames is used to build a protocol for *face identification* (1:N Search) and *face verification* (1:1 Comparison). In comparison to LFW, the $5,712$ images and $2,085$ videos in the IJB-A benchmark have a wider geographic variation, larger pose variation and images of low resolution or heavy occlusion, making it a much harder benchmark than both standard LFW and BLUFR benchmarks. Again, a 10-fold cross-validation test is designed for both identification and verification in IJB-A. True Accept Rate (TAR) at False Accept Rate (FAR) is used to evaluate verification performance. For closed-set identification, Cumulative Match Characteristic (CMC) measures the fraction of genuine gallery templates that are retrieved within a certain rank. And False Negative Identification Rate (FNIR) at False Positive Identification Rate (FPIR) is reported to evaluate the performance in terms of open-set identification.

IJB-B is an extension of IJB-A benchmark. It consists of $21,798$ still images and $55,026$ frames from $7,011$ videos from $1,845$ subjects. There is no cross-validation in IJB-B. In particular, we use the 1:1 Baseline Verification protocol and 1:N Mixed Media Identification protocol for IJB-B.

From the results in Table 4 and Table 5, we can see that the proposed models do improve the performance of the base-net on both the IJB-A and IJB-B benchmarks. This shows the effectiveness of the proposed idea which fuses features from local regions together with a global feature representation, although the base-network is already quite sophisticated. Second, Model B outperforms Model A in most protocols, which indicates that more local regions and sub-networks could help achieve even larger performance gains.

5. Conclusion

In this paper, we have proposed a scheme for incorporating parts-based models into state-of-the-art CNNs for face recognition. A set of sub-networks are added to learn features from certain facial parts. An spatial transformer-based attention network learns to automatically localize the discriminative regions. We have further added a fusion layer to combine the global and local features, which, with the proposed promotion loss, encourages the sub-networks to find more discriminative features. The proposed approach can be applied to any single CNN to build an end-to-end system. Experiments on the most novel and challenging benchmarks show that the proposed strategy can help improve the performance of a single CNN without significant increase in run-time. Evidence suggests that in the future, we can further improve the performance with even more sub-networks.

References

[1] J. Ba, V. Mnih, and K. Kavukcuoglu. Multiple object recognition with visual attention. *arXiv:1412.7755*, 2014.

[2] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *IEEE Trans. on PAMI*, 32(9), 2010.

	TAR@FAR (Verification)		CMC (Closed-set Identification)		FNIR (Open-set Identification)	
Type	0.001	0.01	Rank-1	Rank-5	0.01	0.1
Base-net	0.542 ± 0.0917	0.7883 ± 0.0917	0.882 ± 0.0190	0.954 ± 0.0079	0.426 ± 0.0170	0.355 ± 0.0140
Model A	0.583 ± 0.0832	0.8075 ± 0.0264	0.889 ± 0.0068	0.957 ± 0.0068	0.418 ± 0.0147	$\mathbf{0.353 \pm 0.0137}$
Model B	$\mathbf{0.602 \pm 0.0692}$	$\mathbf{0.8231 \pm 0.0219}$	$\mathbf{0.898 \pm 0.0092}$	$\mathbf{0.960 \pm 0.0061}$	$\mathbf{0.411 \pm 0.0164}$	0.353 ± 0.0142

Table 4: Evaluation results on IJB-A 1:1 Comparison and 1:N Search protocols.

	TAR@FAR (Verification)		CMC (Closed-set Identification)		FNIR (Open-set Identification)	
Type	0.001	0.01	Rank-1	Rank-5	0.01	0.1
Base-net	0.631	0.851	0.749	0.861	0.149	0.032
Model A	0.652	0.861	0.768	**0.875**	0.139	**0.031**
Model B	**0.659**	**0.865**	**0.769**	0.874	**0.135**	0.032

Table 5: Evaluation results on IJB-B 1:1 Baseline Verification and 1:N Mixed Media Identification protocols.

[3] J. Fu, H. Zheng, and T. Mei. Look closer to see better: Recurrent attention convolutional neural network for fine-grained image recognition. In *CVPR*, 2017.

[4] Y. Guo and L. Zhang. One-shot face recognition by promoting underrepresented classes. *arXiv:1707.05574*, 2017.

[5] A. Hasnat, J. Bohné, S. Gentric, and L. Chen. Deepvisage: Making face recognition simple yet with powerful generalization skills. *arXiv:1703.08388*, 2017.

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

[7] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015.

[8] M. Jaderberg, K. Simonyan, A. Zisserman, et al. Spatial transformer networks. In *NIPS*, 2015.

[9] B. F. Klare, B. Klein, E. Taborsky, A. Blanton, J. Cheney, K. Allen, P. Grother, A. Mah, M. Burge, and A. K. Jain. Pushing the frontiers of unconstrained face detection and recognition: IARPA Janus benchmark A. In *CVPR*, 2015.

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

[11] S. Liao, Z. Lei, D. Yi, and S. Z. Li. A benchmark study of large-scale unconstrained face recognition. In *IJCB*, 2014.

[12] M. Mathias, R. Benenson, M. Pedersoli, and L. Van Gool. Face detection without bells and whistles. In *ECCV*, 2014.

[13] V. Nair and G. E. Hinton. Rectified linear units improve restricted boltzmann machines. In *ICML*, 2010.

[14] O. M. Parkhi, A. Vedaldi, A. Zisserman, et al. Deep face recognition. In *BMVC*, 2015.

[15] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *CVPR*, 2015.

[16] Y. Sun, Y. Chen, X. Wang, and X. Tang. Deep learning face representation by joint identification-verification. In *NIPS*, 2014.

[17] Y. Sun, D. Liang, X. Wang, and X. Tang. Deepid3: Face recognition with very deep neural networks. *arXiv:1502.00873*, 2015.

[18] Y. Sun, X. Wang, and X. Tang. Deep learning face representation from predicting 10,000 classes. In *CVPR*, 2014.

[19] Y. Sun, X. Wang, and X. Tang. Deeply learned face representations are sparse, selective, and robust. In *CVPR*, 2015.

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

[21] Y. Wen, K. Zhang, Z. Li, and Y. Qiao. A discriminative feature learning approach for deep face recognition. In *ECCV*, 2016.

[22] C. Whitelam, E. Taborsky, A. Blanton, B. Maze, J. Adams, T. Miller, N. Kalka, A. K. Jain, J. A. Duncan, K. Allen, J. Cheney, and P. Grother. Iarpa janus benchmark-b face dataset. In *CVPR Workshop on Biometrics*, 2017.

[23] T. Xiao, Y. Xu, K. Yang, J. Zhang, Y. Peng, and Z. Zhang. The application of two-level attention models in deep convolutional neural network for fine-grained image classification. In *CVPR*, 2015.

[24] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio. Show, attend and tell: Neural image caption generation with visual attention. In *ICML*, 2015.

[25] J. Yan, X. Zhang, Z. Lei, and S. Z. Li. Face detection by structural models. *Image and Vision Computing*, 32(10), 2014.

[26] S. Yang, P. Luo, C.-C. Loy, and X. Tang. From facial parts responses to face detection: A deep learning approach. In *ICCV*, 2015.

[27] D. Yi, Z. Lei, S. Liao, and S. Z. Li. Learning face representation from scratch. *arXiv:1411.7923*, 2014.

[28] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE Signal Processing Letters*, 23(10), 2016.

[29] Y. Zhong, J. Chen, and B. Huang. Toward end-to-end face recognition through alignment learning. *IEEE Signal Processing Letters*, 24(8), 2017.