2019 IEEE 4th International Conference on Signal and Image Processing

Deep Representation Learning With Feature Augmentation for Face Recognition

Jie Sun, Shengli Lu*, Wei Pang
National ASIC System Engineering Research Center
Southeast University
Nanjing, China
e-mail: {jie.sun, lsl, pw}@seu.edu.cn

Zhilin Sun
School of Mechanical and Power Engineering
Nanjing Tech University
Nanjing, China
e-mail: tychosun@163.com

***Abstract*—Deep Convolutional Neural Networks (DCNN) significantly improve the performance of many computer vision tasks, such as classification, detection, and semantic segmentation. The ideal face features are expected to have smaller maximal intra-class distance than minimal inter-class distance under open-set protocol, but the current algorithm still has the open problem of implementing the criterion. In this paper, we present a feature augmentation network for the IARPA Janus Benchmark C (IJB-C) on a small CNN. The proposed feature enhancement method is used to approximate the identity features, and the original features are augmented by a small automatic encoder-decoder which can be quickly run in an embedded system with limited resources and obtains similar accuracy to a large backbone CNN network.**

Keywords-feature augmentation; face recognition; face verification; IJB-C

I. INTRODUCTION

The emergence of deep learning greatly advances the frontier of face recognition (FR)[1]. Owing to advanced network architectures [2] and discriminate learning approaches [3], [4], deep CNN have boosted the FR performance to an unprecedented level. Face recognition can be categorized as face identification and face verification. Existing CNN still have limited ability to handle pose variability, occlusions, large illumination variation and small faces. Modern deep learning is heavily data-driven. The generalization power of deep models is usually proportional to the training data size. First, it is in-feasible to collect a massive data set that covers all possible poses with even distribution. Second, such methods would add processing burden on the embedded system that is a very limited memory size for bigger numbers of neurons in network (usually they have 0.5-1 GB RAM integrated without possibility to extend).

In order to push the development of frontier in unconstrained face recognition, a new face dataset template-based IJB-C[5] is introduced recently, whose protocols and solutions are aligned better with the requirements of real applications. The IJB-C dataset includes real-world unconstrained faces from 3531 subjects with full pose and illumination variations which are much harder than the LFW and YTF datasets.

Usually a shallow CNN model is almost impossible to identify and verify faces on IJB-C. In this paper, the distribution of template-based facial features extracted from 4-layers CNN network and 50-layers backbone network are compared. The loss function joint KL divergence and distance metric is designed. By adding the output of the encoder after the small 4-layer network, we get enhanced feature maps. Our work does not focus on image space, nor through feature fusion, but directly on the feature space.

II. RELATED WORK

Feature extraction methods such as principal component analysis (PCA), linear discriminant analysis (LDA), which works by transforming primitive features into new feature sets built on the original features of their combination[6]. In addition to the traditional feature reduction, the existing CNN-based feature augmentation research for face recognition is mainly in following three aspects.

A. GAN-based Feature Augmentation

By performing feature augmentation with GAN, Volpi et al. [7] proposed a method for generating features from noise vectors and tag codes. The domain invariance is forced in a single feature extractor trained by GAN, and then a more complex minimax game is defined to perform feature augmentation in feature space. More specifically, the feature generator trained use condition GAN (CGAN). The Minimax game uses features instead of images, allowing the generation of features that fit the desired category.

However, training GAN is quite complicated, and to some extent an art form, because the incorrect hyper-parameter setting causes the model to crash, and there is almost no explanation for the cause of the error.

B. Mapping Features of Profile Faces to Frontal Faces

Cao et al. [8] proposed a pose-robust method called DREAM, which simulates the transformation between front-side faces in high-level deep feature space. Given an input image of arbitrary pose, [8] can actually map its feature to the frontal space through a mapping function that adds residual. This method does not consider roll and pitch angles, and transform profile face features to frontal features can yield better performance than image-level frontal.

For huge unconstrained datasets like IJB-C, only this approach is not enough.

C. Supplement a Target Pose Parameter

FATTEN [9] is an encoder decoder architecture that exploit a parametrization of pose trajectories in terms of an appearance map, which captures properties such as object color and texture. The encoder maps the feature response of

978-1-7281-3660-8/19/$31.00 ©2019 IEEE

the object image into a pair of appearance and pose parameters. The decoder then takes these parameters plus the target pose and produces the corresponding feature vector. The "add" feature space by increasing the data set of the feature response with the invisible object pose, the main method is to add additional appearance descriptors, such as color or texture.

Because the IJB-C dataset itself does not have an appearance label, if manual marking will increase the workload and is not applicable in this task.

III. THE PROPOSED METHOD

In this work, the feature augmentation performed is different from the above methods. First, we did not use GAN (add noise) to generation features. Secondly, we analyze the difference of the feature distribution of the same template between the backbone network and the shallow CNN network, and normalize the eigenvalues of the two, then combine both eigenvalues with the Euclidean distance matrix as the auto-encoder loss function. Therefore, we generate new features for enhanced recognition accuracy of the original features by the encoder-decoder.

A. Feature Equivariance

Karel et al. [10] presented the notion of feature equivariance. Different networks have different representations for the feature extraction values of the same template. The network can be composed of different models with various layers, and the training datasets are also inconsistent, so different embedded methods perform different feature selections during the execution of the modeling algorithm.

Inspired by [9], the backbone network and the small 4-layer (convolution and activation) network in this paper are inconsistent with the features extracted by the same template, but both have similar distributions for features extracted by multiple templates of a person.

B. Image Representation With CNN

Given a template T_x , its representation $\mathbf{f}_x$ is hence extracted by CNN with the updated parameter θ, i.e.,

$$\mathbf{f}_x = \mathbf{CNN}_\theta(T_x) \tag{1}$$

Our goal is to improve more discriminative power of $\mathbf{f}_x$ at a small CNN network on a resource-constrained embedded system. According to the experiment, the existing small network can effectively verify and identify the LFW [8], but the performance of the IJB-C drops sharply. There are two reasons. First, the LFW's picture pose changes little. Second, the scale of LFW is too small. For large unconstrained data sets of IJB-C, we still hope to complete face verification and recognition on a small CNN network.

Features extracted by small and stem CNN. Our small network can be seen in Fig. 3, and the Stem CNN uses the pre-trained model SeNet (Model is available at

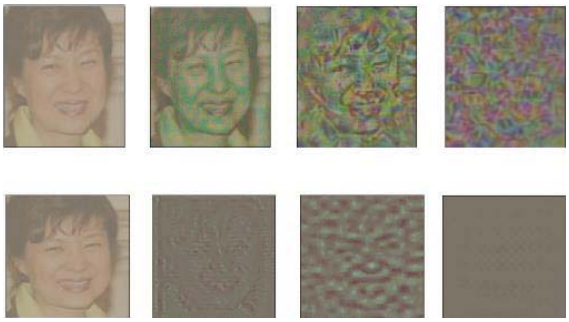
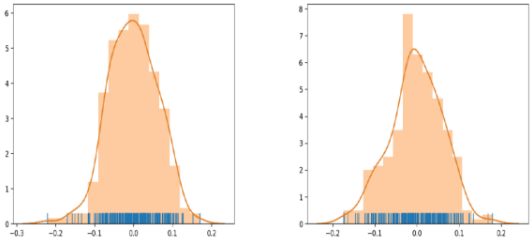


Figure 1. Visualization of features. The above four images are small CNN's, the left is input image and the right are layer3, layer4 and fully connected layer respectively. The below four are the corresponding features of the SeNet.

http://www.robots.ox.ac.uk/~vgg/data/vgg_face2/models/pytorch/senet50_256_pytorch.tar.gz).We first analyze the differences in the features of the same image extracted by the two, and Fig.1 shows their differences in the representation of the features.

Data distributions of High and Low accuracy. In the case of limited data, people make more use of data augmentation and transfer learning. Data augmentation usually use rotation or flip to make the model invariant to transformation and will not increase the real variability. Transfer learning uses filter weights from other domain-specific large datasets. We did not use transfer learning but analyzing the distribution of different CNNs on the same template. As illustrated in Fig. 2, Features for good quality task have a high L2-norm while pool quality task have low L2-norm .



(a) High accuracy distribution (b) Low accuracy distribution

Figure 2. Taking template 6 as an example, the distribution of (a) can complete the identification and verification tasks correctly, and the distribution of (b) is just the opposite

The Earth Mover's Distance (EMD) [11] is a method to evaluate dissimilarity between two multi-dimensional distributions in some feature space where a distance measure between single features. We did not use the EMD method, because it requires multiple iterations to calculate the optimal value, therefore does not suitable for resource constrained embedded systems.

C. Feature Augmentation Architecture

Fig. 3 is our proposed architecture. Inputs during training and inference are based on templates rather than traditional still images or video frames. A template refers to a collection

of all media (images and/or video frames) of an interested face captured under various environments.

The small network has four layers, and each layer includes convolution and activation to form a residual network. The number and size of convolution kernels for each layer are indicated on Fig.3.

Traditional encoder compress the input and then produce the code, the decoder then reconstructs the input only using this code, its mainly for dimension reduction. We want to augment the input 256-dimensional features, so the encoder output is still 256 dimensions. Original $\phi(x)$ representation is first normalized, then used as the input to the *encoder, and will form a corresponding matrix with* $\phi(y)$ according to the number of epochs. We set the matrix as hollow so that it can effectively measure the distance between two feature distributions.

Both Stem CNN and small CNN are pre-trained, and their output features are based on template representations, each of which is a 256-dimensional features. By feeding the features of the probe template to the auto-encoder for training, the output of the encoder is the augmenting features.

D. Evaluate Feature Distribution

For discrete probability distributions P and Q defined on the same probability space, the Kullback-Leibler (KL) divergence between P and Q is defined as in

$$D_{\mathrm{KL}}(P \parallel Q) = -\sum_{x \in \mathcal{X}} P(x) \log\left(\frac{Q(x)}{P(x)}\right) \qquad (2)$$

Where x is a template element of template set $\mathcal{X}$. If P represents the distribution of a small network, then Q represents the distribution of the backbone network and vice versa. KL Divergence has its origins in information theory, we can calculate exactly how much information is lost when we approximate one distribution with another.

In general, we cannot use KL Divergence to measure the distance between two distributions. The reason is that the KL divergence is not symmetrical. In this work, we use KL Divergence as an objective function to find the optimal value for approximating the better distribution $\phi(y)$.

E. Pairwise Distance Matrix

We define a hollow matrix for measuring distances as follows:

$$D_1 = \left(x_{ij}\right) \qquad (3)$$

where $1 \le i, j \le N$, each of x_{ij} is the difference of the corresponding template features. The entries on the main diagonal are all zero, i.e. $x_{ij} = 0$ for all $1 \le i \le N$, all the off-diagonal entries are positive i.e. $x_{ij} > 0$ if $\neq j$. In this paper, N is equal to feature dimension 256.We define the matrix as a symmetric matrix $\left(x_{ij} = x_{ji}\right)$, $x_i = \phi(x)$, $x_j = \phi(y)$, then the elements of D_1 is defined as follows:

$$x_{ij} = d_{ij}^2 = \left\| x_i - x_j \right\|_2^2 \qquad (4)$$

Where $\|\cdot\|_2$ denotes the 2-norm.

F. Cost Function and Optimization

To train the feature augmentation network, we propose two objectives. The first one is that the feature distribution of the trained template should be as close as possible to the distance of the stem network where the feature distribution extracted by the same template. The second one is that the learned features should be consistent with that extracted from the original template.

- **Cost Function**. The KL divergence is the expectation of the log difference between the probability of data in the original distribution with the approximating distribution. We rewrite Eq. (2) in terms of expectation:

  $$D_{KL}(\phi(x) \parallel \phi(y)) = E[log\phi(x) - log\phi(y))] \qquad (5)$$

  The network loss function in terms of a pair of training features $\phi(x)$ and $\phi(y)$ is defined as follows:

  $$\mathcal{L}(\phi(x), \phi(y)) = \left\| f_x - f_y \right\|^2 + \left\| \tilde{f}_x - \tilde{f}_y \right\|^2 \qquad (6)$$

  Where f_x and f_y denote the features extracted by corresponding pretrained classification network small network and stem network, respectively. $\tilde{f}_x = E[\log\phi(x)]$, $\tilde{f}_y = E[\log\phi(y)]$. All features are L_2-normalized before the distance computation.
- **Optimization Strategy**. This loss function consists of two components: a fidelity preserving constraint $\left\| f_x - f_y \right\|^2$ and similarity constraint $\left\| \tilde{f}_x - \tilde{f}_y \right\|^2$. The fidelity preserving constraint compute the distance of the newly learned features and original features from the pre-trained small network.The similarity constraint $\left\| \tilde{f}_x - \tilde{f}_y \right\|^2$ ensures that the network outputs similar features for template in feature space.

IV. EXPERIMENTS AND RESULTS

We evaluate the verification and identification protocol on the IJB-C .In our experiments, we utilize the ground-truth bounding box to crop face image from the original one and resize to 96×112 for each image or frame.

A. Datasets

The IJB-C (IARPA Janus Benchmark-C)[5] dataset contains 3,531 subjects, including a total of 31,334 (21,294 faces and 10,040 non-face) still images, with an average of ~6 images per subject and 11,779 full-motion videos. There are 117,542 frames, each of which has an average of about 33 frames, and each subject has about 3 videos.

The subjects in IJB-C are not duplicated with the subjects contained in the VGG-Face and CASIA-Webface data sets.

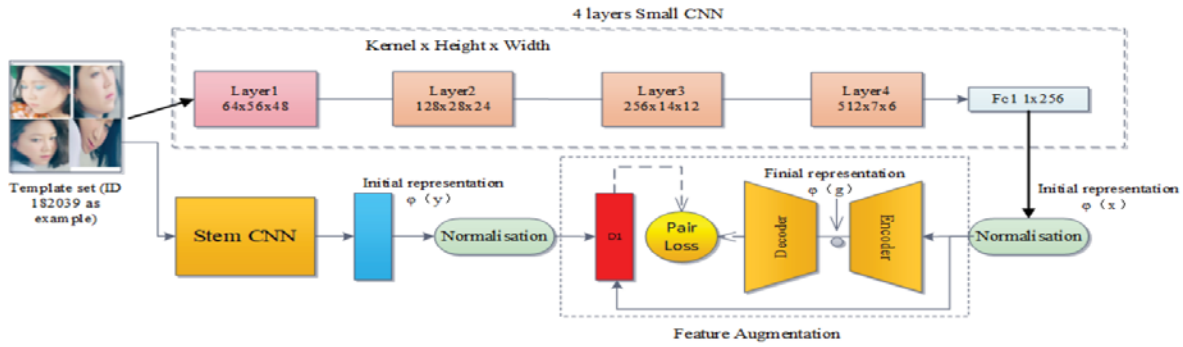


Figure 3. Feature augmentation network architecture. The output dimension of the fully connected layer of the small network and the stem network is 256. The feature maps of the former and the latter are normalized to form a hollow metric matrix as a loss function for the auto-encoder to augment original feature maps.

TABLE I. PERFORMANCE COMPARISON WITH THE STATE-OF-THE-ART CNN-BASED FACE VERIFICATION AND RECOGNITION METHODS ON IJB-C (MARKED BY "C") OR IJB-A(MARKED BY "A") . THE RESULTS REPORTED WITH THE FEATURE AUGMENTATION ARE MARKED BY "+FA ",TOKEN "-" MEANS NO DATA

Methods	1 : 1 Verification TAR			1 : N Recognition TPIR				
	FAR=0.001	FAR=0.01	FAR=0.1	FPIR=0.01	FPIR=0.1	Rank 1	Rank 5	Rank 10
FF-GAN A	0.66	0.85	-	-	-	0.90	0.95	-
DR-GAN A	0.69	0.83	-	-	-	0.90	0.95	-
VGG-Face[12] C	-	0.80	-	0.46	0.67	0.91	-	0.98
L2-Softmax[15] A	0.93	0.96	0.98	0.90	0.95	0.97	-	0.99
SeNet50 C	0.90	0.95	0.98	0.58	0.73	0.89	0.93	0.95
TDFF[16] A	0.91	0.96	0.98	0.87	0.94	0.96	0.98	0.99
Ours C	0.24	0.44	0.72	0.11	0.17	0.26	0.35	0.39
Ours +FA C	0.69	0.76	0.88	0.57	0.66	0.68	0.78	0.86

The stem network we choose to train on VGG-Face[12]. Our own small network trained on CASIA-Webface[13], so we can objectively evaluate the generalization ability of our feature augmentation network.

B. Implementation Details

This article uses PyTorch to complete all the experiments. The small network is pre-trained by CASIA-Webface[13], and all images are pre-processed to a size of 96×112 by using MTCNN[14]. Since the image required by the stem network is 224×224, we need to adjust the image of 96×112 to 224×224, normalized by subtracting the mean [91.4953, 103.8827, 131.0912] from each image. In addition, the stem network is trained by caffe and we need to adjust our image from RGB format to BGR format.

After training the 4 layers small network, the performance is first verified on the LFW dataset. When the accuracy mAP is lower than 98%, the network weights are re-initialized and the network parameters are re-adjusted , including the learning rate. epoches, momentum, etc., the small model can be used for the feature extractor of IJB-C until mAP is greater than 98%.

The encoder only has one linear layer with Relu as the activation function, and also decoder has only one linear layer which uses Sigmoid to activate the output. Our encoder does not require additional annotation information, but FATTE N[9] requires an additional pose label.

C. Comparison with State-of-the-Art

We compare the proposed method with existing ones on the IJB-C in the aspects of both verification and recognition accuracy. As shown in Table I, the proposed method outperform the original method without FA. By jointing distance metric and KL divergence, the final representation further boosts the performance and achieve 30%, 41% and 38% improvements in mAP ,Rank-1,and Rank-5, respectively.

Since there are fewer papers reported on IJB-C, we also show in Table I the results with the state-of-the-art models on IJB-A. Our shallow CNN model also runs smoothly on the XILINX ZYNQ 7000 development board. As far as we know, this is the first attempt of the small CNN model on the IJB-C.

V. CONCLUSIONS

This paper is dedicated to achieve the verification and identification protocol for huge IJB-C dataset on a small shallow CNN network. Different from the previous work, we don't use the GAN to generate features. The template-specific features extracted by the stem network and small network are regularized to form a distance metric matrix. At the same time, the KL divergence is used to make the generated features

approximate the original distribution. Finally, the encoder is added to the fully connect layer of any existing CNN models, and the output features of the encoder are used to identify and verify the face. Experiments on IJB-C show that the method is efficient and consumes less resources, and is suitable for embedded systems with limited resources. In the future, we plan to combine the automatic encoder-decoder with the existing small CNN network for end-to-end training to further enhance the feature augmentation.

REFERENCES

[1] Y. Sun, Y. Chen, X. Wang, and X. Tang, "Deep learning face representation by joint identification-verification," neural information processing systems, pp. 1988–1996, 2014.

[2] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation etworks," computer vision and pattern recognition, pp. 7132–7141, 2018

[3] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song, "Sphereface: Deep hypersphere embedding for face recognition," computer vision and pattern recognition, pp. 6738–6746, 2017.

[4] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," computer vision and pattern recognition, pp. 815–823, 2015.

[5] B. Maze, J. C. Adams, J. A. Duncan, N. D. Kalka, T. Miller, C. Otto,A. K. Jain, W. T. Niggel, J. Anderson, J. Cheney, et al., "Iarpa janus benchmark - c: Face dataset and protocol," pp. 158–165, 2018.

[6] A. Jovic, K. Brkic, and N. Bogunovic, "A review of feature selection methods with applications," pp. 1200–1205, 2015.

[7] R. Volpi, P. Morerio, S. Savarese, and V. Murino, "Adversarial feature augmentation for unsupervised domain adaptation," computer vision and pattern recognition, pp. 5495–5504, 2018.

[8] K. Cao, Y. Rong, C. Li, X. Tang, and C. C. Loy, "Pose-robust face recognition via deep residual equivariant mapping," computer vision and pattern recognition, pp. 5187–5196, 2018.

[9] B. Liu, X. Wang, M. Dixit, R. Kwitt, and N. Vasconcelos, "Feature space transfer for data augmentation," computer vision and pattern recognition, pp. 9090–9098, 2018.

[10] K. Lenc and A. Vedaldi, "Understanding image representations by measuring their equivariance and equivalence," in Computer Vision and Pattern Recognition, pp. 1–21, 2015.

[11] Y. Rubner, C. Tomasi, and L. J. Guibas, "The earth mover"s distance as a metric for image retrieval," International Journal of Computer Vision, vol. 40, no. 2, pp. 99–121, 2000.

[12] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, and A. Zisserman, "Vggface2: A dataset for recognising faces across pose and age," ieee international conference on automatic face gesture recognition, pp. 67–74, 2018.

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

[14] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," IEEE Signal Processing Letters, vol. 23, no. 10, pp. 1499–1503, 2016.

[15] R. Ranjan, C. D. Castillo, and R. Chellappa, "L2-constrained softmax loss for discriminative face verification.," arXiv preprint arXiv: 1703.09507, 2017.

[16] L. Xiong, J. Karlekar, J. Zhao, J. Feng, S. Pranata, and S. Shen, "A good practice towards top performance of face recognition: Transferred deep feature fusion.," arXiv: Computer Vision and Pattern Recognition, 2017.