Computational Visual Media
DOI 10.1007/s41095-017-0091-7 Vol. 3, No. 4, December 2017, 359–368

Research Article

Face image retrieval based on shape and texture feature fusion

Zongguang Lu[1](✉), **Jing Yang**[1], **and Qingshan Liu**[1]

Abstract Humongous amounts of data bring various challenges to face image retrieval. This paper proposes an efficient method to solve those problems. Firstly, we use accurate facial landmark locations as shape features. Secondly, we utilise shape priors to provide discriminative texture features for convolutional neural networks. These shape and texture features are fused to make the learned representation more robust. Finally, in order to increase efficiency, a coarse-to-fine search mechanism is exploited to efficiently find similar objects. Extensive experiments on the CASIA-WebFace, MSRA-CFW, and LFW datasets illustrate the superiority of our method.

Keywords face retrieval; convolutional neural networks (CNNs); coarse-to-fine

1 Introduction

One of the first visual patterns an infant learns to recognize is the face. The face provides a natural means for people to recognize each other. For this and several other reasons, face recognition and retrieval have been problems of prime interest in the fields of computer vision, biometrics, pattern recognition, and machine learning for decades. The face has been very successful used in biometrics due to its unobtrusive nature and ease of use; it is suited to both overt and covert applications. Along with advances in face analysis technology, face recognition, expression recognition, attribute analysis, and other applications have come to the fore. Also, content-based image information retrieval technology has gradually matured, and major search engines now offer a *search by image* function. Progress in face recognition and context-based information retrieval technology have made automatic similar face retrieval possible. Similar face retrieval has high application value in the fields of entertainment search, criminal surveillance, and so on. Figure 1 illustrates large-scale face retrieval in the field of prevention of terrorist crimes.

As a specific application of image retrieval, face retrieval has the same research characteristics. Unlike face recognition and face identification, the aim of face retrieval is to search for all the face images similar to an input image in a given face image database, and to sort the results by similarity. Existing face retrieval methods are usually designed to compute geometric properties and relationships between significant local features, such as the eyes, nose, and mouth [1, 2]. Bach et al. [3] manually annotated images of faces and used artificial features extracted from the annotated

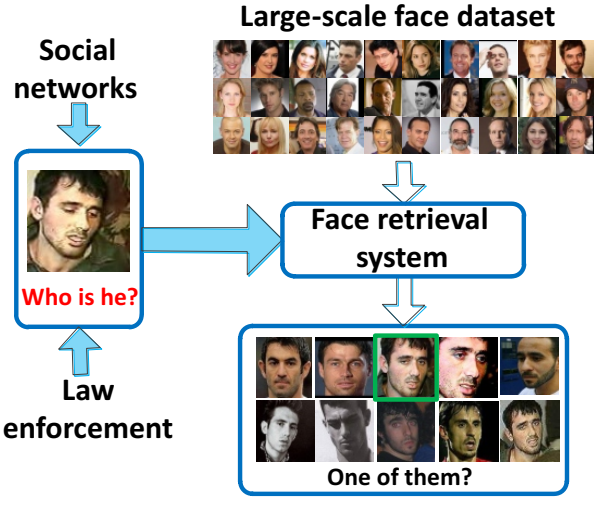


Fig. 1 Example of large-scale face retrieval problem.

1 School of Information and Control Engineering, Nanjing University of Information Science and Technology, China. E-mail: Z. Lu, zongguanglu@nuist.edu.cn (✉); J. Yang, nuist_yj@126.com; Q. Liu, qsliu@nuist.edu.cn.

Manuscript received: 2017-02-27; accepted: 2017-05-26

regions for face matching, thus providing a semi-automatic face retrieval system. Eickeler [4] applied the pseudo 2D hidden Markov model method for the first time in a face retrieval system, achieving good results. Gudivada and Raghavan [5] borrowed methods from face matching and proposed using features extracted from face matching in a face retrieval system. Wang et al. [6] proposed a multi-task learning structure using local binary patterns (LBP) [7] to solve face verification and retrieval problems.

Learning face representations via deep learning has achieved a series of breakthroughs in recent years [8–13]. The idea of mapping a pair of face images to a distance originated in Ref. [14]. They trained Siamese networks as a basis for the similarity metric, which is small for positive pairs and large for the negative pairs. This approach requires image pairs as input.

Very recently, Refs. [12, 15] supervised the learning process in CNNs using challenging identification signals (with a softmax loss function), which brings richer identity-related information to deeply learned features. Subsequently, a joint identification–verification supervision signal was adopted in Refs. [10, 13], leading to more discriminative representation features. Reference [16] enhanced supervision by adding a fully connected layer and loss functions to each convolutional layer. The advantage of triplet loss has been proved in Refs. [8, 9, 17]. With deep embedding, the distance between an anchor and a positive instance is minimized, while the distance between an anchor and a negative instance is maximized until a preset margin is met. They achieved state-of-the-art performance on the LFW dataset.

We propose a method for fast large-scale face retrieval using fused shape and texture features to represent a face. Firstly, we use accurate face alignment to gain shape information, inspired by SDM [18]. Secondly, we adopt a modified GoogleNet [19] to gain texture information about the face. Thirdly, we fuse these two features to represent the face image. Furthermore, we use a coarse-to-fine structure that clusters the dataset into several dense subsets to achieve fast retrieval. We thoroughly evaluate the contributions of each part in this paper and show that it achieves excellent performance on experimental datasets.

2 Method

2.1 Overview

Figure 2 provides an overview of our shape and texture cascade face retrieval approach. Firstly we use SDM to extract face landmarks and a modified GoogleNet to extract face texture information. Secondly we fuse and balance the two features using principal component analysis (PCA). Finally, we search the face dataset using the fused features to get the result.

2.2 Shape feature representation

This section describes use of SDM in the context of face alignment. Algorithm 1 shows the main steps of the SDM evaluation procedure. SDM is based on a regressor that starts from a raw initial shape guess x_0 and progressively refines this estimate using descent directions R_k and bias terms b_k, outputting a final shape estimate x_k. The descent directions set R_k and bias terms b_k have been learned during training. The training procedure corresponds to minimizing:

$$\arg\min_{R_0,b_0} \sum_{d^i} \sum_{x_0^i} \|\Delta x_*^i - R_0\phi_0^i - b_0\|^2 \qquad (1)$$

where x_* are the manually annotated face landmarks. Minimizing this corresponds to a linear least squares problem that can be solved in closed-form.

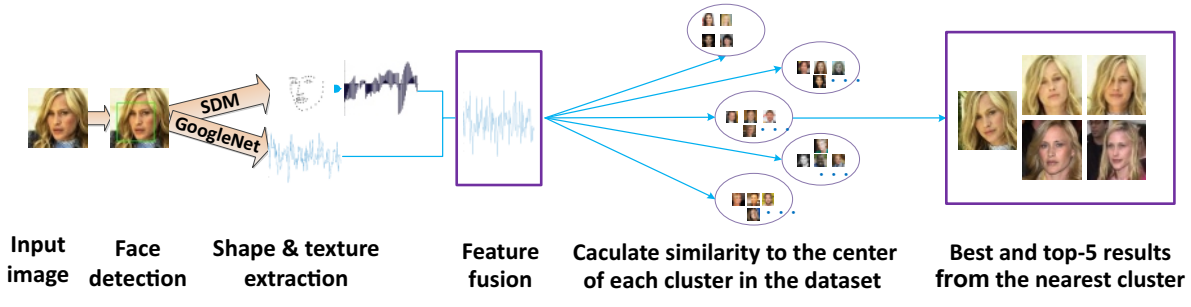


Fig. 2 Proposed large-scale face retrieval approach.

Algorithm 1 Face alignment via supervised descent method (SDM)

Input: Image I, descent directions R_k, bias terms b_k, initial guess x_0.

1: **for** $i = 1 : k$ **do**
2: $\phi = h(d(x))$;
3: $x_i = x_{i-1} + R_{i-1}\phi_{i-1} + b_{i-1}$;
4: **end for**
5: **return** final estimate x_k.

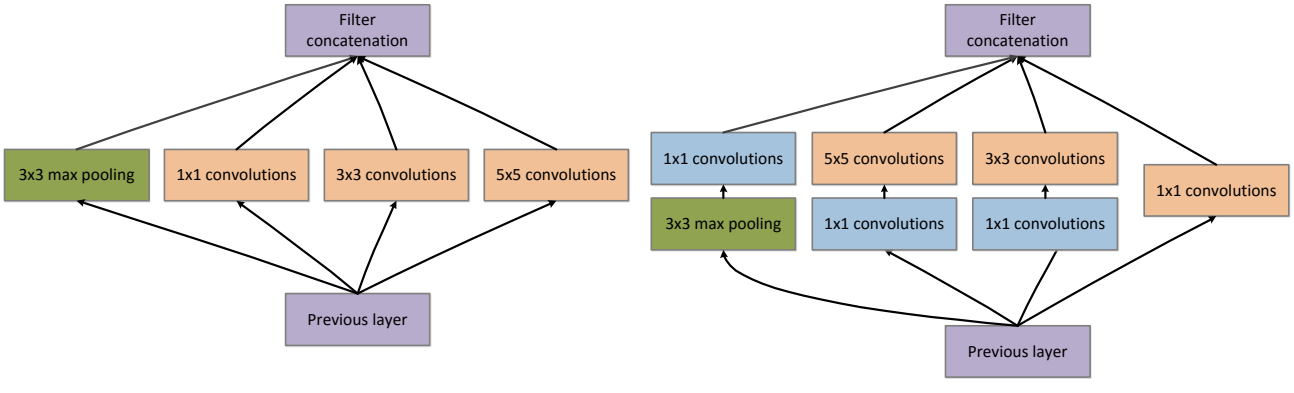
2.3 Texture feature representation

This section explains how we use CNNs modified from *GoogleNet V2* [20] to extract the texture features. Convolutional neural networks (CNNs) have played an extremely significant role in computer vision due to the revolutionary improvements they provide over the state of the art in many applications. In the field of face analysis, however, large scale public datasets are extremely scarce. Thus, here we use a face dataset containing 20,000 celebrities, each with 50–1000 images, for a total of about 2,000,000 images taken from the Internet. We combine the state of the art performance of the *GoogleNet V2* and the accurate and efficient approach of triplet loss [8] to train our face texture extraction model using the above dataset.

GoogleNet Inception V1 is the earliest version of *GoogleNet*, appearing in 2014 [19]. Generally, the most direct way to increase network performance is to increase the depth and width of the network, which means generating a massive number of parameters. However, so many parameters will not only cause overfitting but also increase the computation. Reference [19] believes that the fundamental way to solve these two drawbacks is to convert the connections, even the convolutions, to a sparse set of connections. For non-uniform sparse data, the computational efficiency of computer software and hardware is very poor, so determining an approach that not only keeps the sparsity of the network, but also permits the high computational performance associated with dense matrices, is a key issue. A large number of papers show that the computing performance can be improved by clustering the sparse matrix into dense submatrices. Inspired by those methods, the *Inception* module was designed to realize the above ideas.

Figure 3(a) shows the initial version of the *Inception* module. The different sizes of convolutions mean different sizes of receptive fields; filter concatenation fuses diverse scale features. As the network deepens, the features tend to become more abstract, and the receptive field of each feature involved is also increased. Thus, with an increasing number of layers, the proportion of 3×3 and 5×5 convolutions also increases, resulting in a huge computational load. Inspired by Ref. [21], a 1×1 convolutional kernel is applied to dimensionality reduction. The dimension-reduction form of the *Inception* module is shown in Fig. 3(b).

Although this network has been proposed, building deeper networks is becoming mainstream, but the computational efficiency reduces as the models enlarge. Hence, Szegedy et al. [20] tried to find a method to expand the network while avoiding increased computational requirements. *GoogleNet V2* was proposed in 2015, which, compared with *V1*, is an improvement in that it applies $n\times1$ rather than $n\times n$ convolutional kernels. Because of this scheme, the convolutional neural network can keep



(a) Inception module, initial form (b) Inception module with dimension reductions

Fig. 3 GoogleNet Inception V1.

a wide range of receptive fields and reduce the number of parameters needed when expanding the network, increasing the computational speed. Figure 4 illustrates the architecture of the *Inception* module of *GoogleNet V2*. Here, $n = 7$ for the 17×17 grid. In virtue of its high performance and lightweight model, we choose it as the basic network used to extract face texture features.

As an improvement, we adopt a *triplet-based loss* to learn a face embedding when we train the GoogleNet. The triplet-loss acts, in brief, such that when we compare a pair of two alike faces (a, b) and a third differing face c, the aim is to ensure that a is more similar to b than c, unlike traditional metric learning approaches.

The output $\phi(\ell_t) \in \mathbb{R}^D$ of the GoogleNet, pre-trained, is l^2-normalised and mapped to an $L \ll D$ dimensional space using an affine projection $x_t = W'\phi(l_t)/\|\phi(l_t)\|_2$, where $W' \in \mathbb{R}^{L\times D}$. There are two key differences compared to use of a linear predictor: firstly, $L \neq D$ is not equal to the number of class identities, but it is the size of the descriptor embedding; secondly, the projection W' is trained to minimise the empirical triplet loss:

$$E(W')=\sum_{(a,p,n)\in T} \max\{0, \alpha-\|x_a-x_n\|_2^2+\|x_a-x_p\|_2^2\},$$

$$x_i = W'\frac{\phi(l_i)}{\|\phi(l_i)\|_2} \tag{2}$$

where $\alpha \geqslant 0$ is a fixed scalar representing a learning *margin* and T is a set of *training triplets*. Here we do not learn the bias, unlike in the previous function. A triplet (a, p, n) is composed of an anchor face a, and furthermore a positive $p \neq a$, and negative n sample of the anchor's identity.

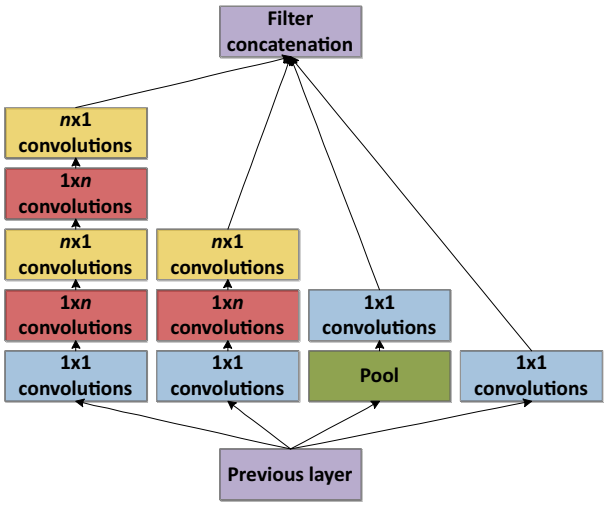


Fig. 4 Inception module after factorization of $n\times n$ convolutions.

We obtain our texture feature representation by training using a face dataset that contains 2,000,000 images; the model size is 58.7 MB.

2.4 Fast face retrieval via coarse-to-fine procedure

This section explains we achieve fast face retrieval for large-scale databases, using two main steps. The first fuses face shape and texture features. The above two features are 132 and 256 dimensional vectors respectively. We apply PCA to reduce the combined features to a final fused feature vector of 128 dimensions. All face data is used in this operation. The second step clusters the combined feature vectors for each dataset into several dense subclusters. We determine the number of clusters according to the number of images in each dataset. Our experiments show that about 100,000 images per cluster give the best balance between speed and precision of retrieval. Therefore, we choose 5 and 2 clusters respectively for the *CASIA-WebFace* [22](abbreviated as CASIA in the following) and *MRSA-CFW* [23] (abbreviated as CFW) datasets.

3 Results and discussion

3.1 Experimental data

As Table 1 shows, we have performed experiments on three datasets. As most identities contain only one image in LFW [24], we conduct face verification on this dataset to demonstrate the excellent selectivity of our face feature representation. The other two datasets are used for face retrieval. Figure 5 shows some examples of face images in these three face datasets. All face images from CASIA are cropped to a uniform size but we use the original images from CFW. Thus, CASIA only contains face images while CFW includes many busts and full-body pictures.

Table 1 Datasets used in experiments

Dataset	Identity	Image
LFW	5,749	13,233
MSRA-CFW	1,583	202,792
CASIA-WebFace	10,575	494,414

(a) CASIA (b) MRSA-CFW (c) LFW

Fig. 5 Example of face images from the three face datasets.

3.2 Evaluation

We now explain how we carried out the experiments. Because both CASIA and CFW were collected for training face recognition tasks, and do not give a standard test set for face retrieval, we therefore manually selected a test sample for each identity in both datasets. Extensive experiments on the LFW dataset were used to evaluate the performance of the features extracted by our method.

As there is no benchmark for face image retrieval using CASIA and CFW, in the following evaluations, we selected 10,575 representative face images using each identity in CASIA as its test set, and used the same method to set up a test set for CFW with 1583 representative face images. Following standard image retrieval experimental practice, we use top-1 and top-5 retrieval precisions as our performance metric. Top-1 and top-5 precisions are calculated using:

$$\frac{\sum_{i=1}^{n} C(X_i, Y_i)}{n} \tag{3}$$

where n represents the number of representative face images in the test set, and $C(X_i, Y_i)$ compares the ground truth X_i and the retrieval result Y_i. In top-1 retrieval mode, Y_i contains just the most similar retrieval result, and if $X_i = Y_i$, $C(X_i, Y_i) = 1$, otherwise $C(X_i, Y_i) = 0$. In top-5 retrieval mode, Y_i contains the five most similar retrieval results, and as long as one of the five results is equal to the ground truth, $C(X_i, Y_i) = 1$, otherwise $C(X_i, Y_i) = 0$.

3.2.1 Face retrieval evaluation

As Table 1 shows, CASIA contains 494,414 face images with 10,575 identities while CFW contains 202,792 face images with 1583 identities. Here we conduct two kinds of experiments. The first strategy performs face retrieval by directly calculating the Euclidean distance between the test image and all images in the test database (the *linear scan approach*). Sorting the distances gives the top-1 and top-5 retrieval results. We also use a coarse-to-fine strategy (the *coarse-to-fine approach*). Firstly, we adopt k-means to cluster the database image features into k dense subsets ($k = 5$ and $k = 2$ respectively for CASIA and CFW). Secondly, we find the nearest subset to the test image. Finally, we search this closest subset for the final top-1 and top-5 results.

Our retrieval results are shown in Table 2. For the CASIA dataset we find that our features give excellent performance, achieving 96.62% and 99.34% precisions in top-1 and top-5 modes respectively using linear scan to find the top-k face images. However, the linear scan method is time consuming. The average search time per probe face is nearly 3 s, which is unacceptable. Therefore, we use a coarse-to-fine structure to speed up the retrieval. It takes about 0.3 s to produce retrieval results per probe image. The retrieval speed increased by 8–9 times, at a cost of precision decrease by approximately 2%.

We also achieve outstanding performance on CFW, the retrieval precisions in top-1 and top-5

Table 2 Face retrieval results for CASIA and CFW; retrieval time is the average search time per probe face

	CASIA				CFW			
Retrieval method	Linear scan		Coarse-to-fine		Linear scan		Coarse-to-fine	
Retrieval mode	Top-1	Top-5	Top-1	Top-5	Top-1	Top-5	Top-1	Top-5
Retrieval time	2.87 s	2.78 s	0.33 s	0.35 s	0.49 s	0.52 s	0.50 s	0.52 s
Precision	96.62%	99.34%	94.02%	97.20%	98.61%	99.30%	97.56%	98.61%

modes using linear scan being 98.61% and 99.30% respectively. As the dataset is much smaller than CASIA, the retrieval time is only about 0.5 s. When we applied the coarse-to-fine procedure to the retrieval, the results were quite different from those expected. In top-1 mode, the time cost of each retrieval did not reduce, but increased. This experiment illustrates that if the dataset is not large, the coarse-to-fine operation does not reduce the retrieval time, but increases the complexity of the search.

In order to prove that the fusing features gives better retrieval results, we performed comparative experiments on both CASIA and CFW with fused features, and only texture feature. Table 3 shows the retrieval results, which confirm our expectations. For CASIA, using only texture features, top-1 and top-5 retrieval accuracies decreased by 8% and 5%. The reduction for CFW is more severe, top-1 and top-5 retrieval accuracies being reduced by 17% and 11% respectively. The differences between the two databases led to these quite different accuracy reductions: all face images of CASIA are cropped to uniform size but CFW still contains the original images. As expected, the facial shape information indeed contributes to the good performance.

We demonstrate some results using real examples. Figures 6 and 7 show top-10 results for CASIA and CFW retrieved by the coarse-to-fine method. All retrieval experiments were carried out on a desktop computer with an Intel i7-2600 CPU and 24 GB RAM.

3.2.2 Face verification evaluation

We conducted a face verification evaluation using the LFW dataset, which is the standard test set for face verification in an unconstrained environment.

Table 3 Face retrieval results for different kinds of features

	CASIA				CFW			
Feature	Shape + texture		Texture only		Shape + texture		Texture only	
Retrieval mode	Top-1	Top-5	Top-1	Top-5	Top-1	Top-5	Top-1	Top-5
Precision	96.62%	99.34%	88.33%	94.34%	98.61%	99.30%	81.46%	87.91%

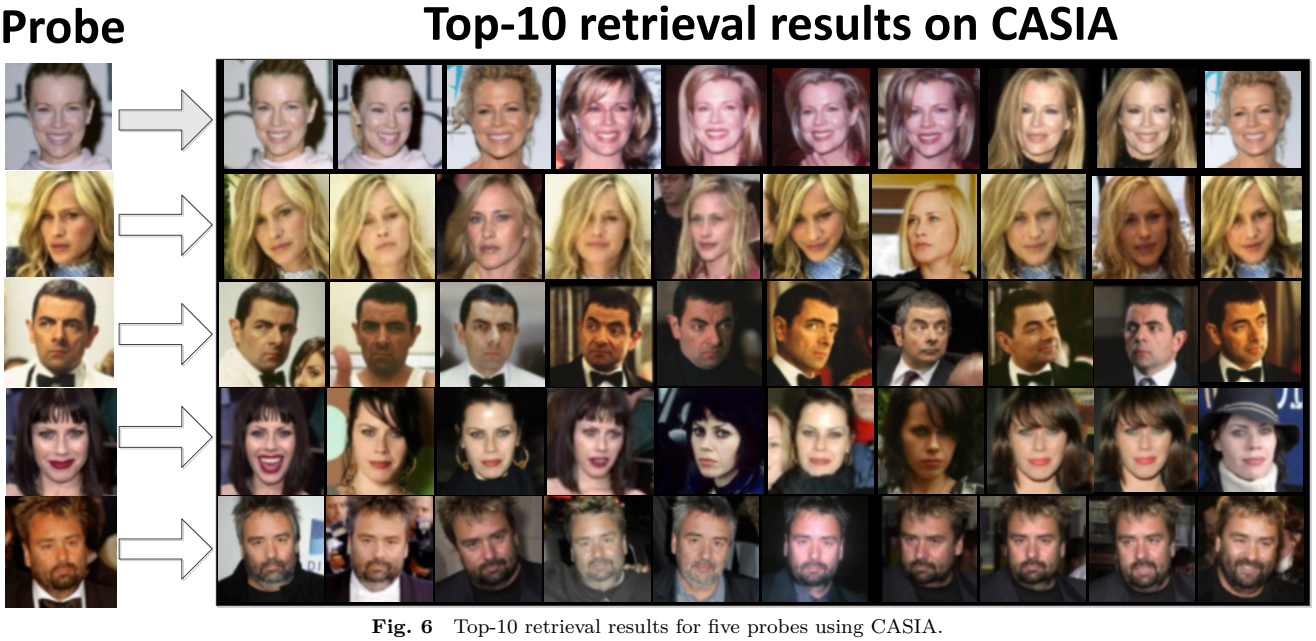


Fig. 6 Top-10 retrieval results for five probes using CASIA.

Fig. 7 Top-10 retrieval results for five probes using CFW.

We report mean face verification accuracy and the receiver operating characteristic (ROC) curve on the 6000 given face pairs in LFW. We rely on a huge outside dataset for training our face representation model, like all recent high-performance face representation methods [12, 15, 25–34]. We compared our method with these methods which all used unrestricted, labeled outside data for training. Furthermore, we used SVM to learn a threshold to verify whether two faces have the same identity or not. In this way, we achieved 97.68% face verification accuracy. We also only used texture features to conduct a face verification evaluation, and achieved 96.70% face verification accuracy, once again proving the advantages of our fused features. The comparison of accuracy and ROC curves to previous state-of-the-art methods using LFW are shown in Table 4 and Fig. 8, respectively. We achieve outstanding results that demonstrate the excellence of our face representation model.

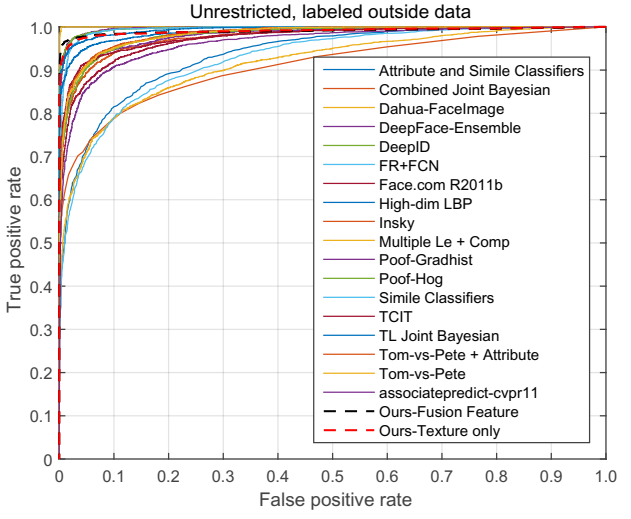


Fig. 8 ROC comparison with previous best methods using LFW.

4 Conclusions

We have designed a face image retrieval method with a novel fused face shape and texture feature representation that exploits specific facial attributes to achieve both scalability and outstanding retrieval performance, as shown by experiments with CASIA and CFW datasets. Extensive experiments on the LFW dataset demonstrate the excellence of our face representation model. In our retrieval experiments, the scale of the test database is still small. In future we plan to set up a larger face retrieval test set with millions of face images and perform experiments on it. We will improve our method and apply it in a system for similar face retrieval.

References

[1] Chan, C. H.; Tahir, M. A.; Kittler, J.; Pietikainen,

Table 4 Accuracy comparison with previous best methods using LFW

Method	Accuracy (%)
Multiple Le + Comp [25]	84.45 ± 0.46
Simile Classifiers [26]	84.72 ± 0.41
Attribute and Simile Classifiers [26]	85.54 ± 0.35
Associate-Predict [27]	90.57 ± 0.56
Face.com R2011b [28]	91.30 ± 0.30
Combined Joint Bayesian [29]	92.42 ± 1.08
Poof-Hog [30]	92.80 ± 0.47
Tom-vs-Pete [31]	93.10 ± 1.35
Poof-Gradhist [30]	93.13 ± 0.40
Tom-vs-Pete + Attribute [31]	93.30 ± 1.28
TCIT [35]	93.33 ± 1.24
High-Dim LBP [32]	95.17 ± 1.13
Insky.so [36]	95.51 ± 0.13
TL Joint Bayesian [33]	96.33 ± 1.08
FR+FCN [34]	96.45 ± 0.25
DeepFace-Ensemble [12]	97.35 ± 0.25
DeepID [15]	97.45 ± 0.26
Ours-Texture only	96.70 ± 0.45
Ours-Fusion Feature	97.68 ± 0.54
Dahua-FaceImage [37]	99.78 ± 0.07

M. Multiscale local phase quantization for robust component-based face recognition using kernel fusion of multiple descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 35, No. 5, 1164–1177, 2013.

[2] Wu, Z.; Ke, Q.; Sun, J.; Shum, H.-Y. Scalable face image retrieval with identity-based quantization and multireference reranking. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 33, No. 10, 1991–2001, 2011.

[3] Bach, J. R.; Paul, S.; Jain, R. A visual information management system for the interactive retrieval of faces. *IEEE Transactions on Knowledge and Data Engineering* Vol. 5, No. 4, 619–628, 1993.

[4] Eickeler, S. Face database retrieval using pseudo 2D hidden Markov models. In: Proceedings of the 5th IEEE International Conference on Automatic Face Gesture Recognition, 58–63, 2002.

[5] Gudivada, V. N.; Raghavan, V. V. Modeling and retrieving images by content. *Information Processing & Management* Vol. 33, No. 4, 427–452, 1997.

[6] Wang, X.; Zhang, C.; Zhang, Z. Boosted multi-task learning for face verification with applications to web image and video search. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 142–149, 2009.

[7] Ojala, T.; Pietikainen, M.; Maenpaa, T. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 24, No. 7, 971–987, 2002.

[8] Parkhi, O. M.; Vedaldi, A.; Zisserman, A. Deep face recognition. In: Proceedings of the British Machine Vision Conference, 2015. Available at http://www.bmva.org/bmvc/2015/papers/paper041/abstract041.pdf.

[9] Schroff, F.; Kalenichenko, D.; Philbin, J. FaceNET: A unified embedding for face recognition and clustering. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 815–823, 2015.

[10] Sun, Y.; Chen, Y.; Wang, X.; Tang, X. Deep learning face representation by joint identification-verification. In: Proceedings of the Advances in Neural Information Processing Systems 27, 1988–1996, 2014.

[11] Sun, Y.; Wang, X.; Tang, X. Hybrid deep learning for face verification. *IEEE Transactions on Pattern Analysis and Machine Intelligence* Vol. 38, No. 10, 1997–2009, 2016.

[12] Taigman, Y.; Yang, M.; Ranzato, M.; Wolf, L. DeepFace: Closing the gap to human-level performance in face verification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1701–1708, 2014.

[13] Wen, Y.; Li, Z.; Qiao, Y. Latent factor guided convolutional neural networks for age-invariant face recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 4893–4901, 2016.

[14] Chopra, S.; Hadsell, R.; Lecun, Y. Learning a similarity metric discriminatively, with application to face verification. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Vol. 1, 539–546, 2005.

[15] Sun, Y.; Wang, X.; Tang, X. Deep learning face representation from predicting 10,000 classes. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1891–1898, 2014.

[16] Sun, Y.; Wang, X.; Tang, X. Deeply learned face representations are sparse, selective, and robust. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2892–2900, 2015.

[17] Liu, J.; Deng, Y.; Bai, T.; Wei, Z.; Huang, C. Targeting ultimate accuracy: Face recognition via deep embedding. *arXiv preprint* arXiv:1506.07310, 2015.

[18] Xiong, X.; la Torre, F. D. Supervised descent method and its applications to face alignment. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 532–539, 2013.

[19] Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich,

A. Going deeper with convolutions. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1–9, 2015.

[20] Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the inception architecture for computer vision. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2818–2826, 2016.

[21] Lin, M.; Chen, Q.; Yan, S. Network in network. *arXiv preprint* arXiv:1312.4400, 2013.

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

[23] MSRA-CFW: Data set of celebrity faces on the web. Available at https://www.microsoft.com/en-us/research/project/msra-cfw-data-set-of-celebrity-faces-on-the-web/.

[24] Huang, G. B.; Mattar, M.; Berg, T.; Learned-Miller, E. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. In: Proceedings of the Workshop on Faces in "Real-Life" Images: Detection, Alignment, and Recognition, 2008.

[25] Cao, Z.; Yin, Q.; Tang, X.; Sun, J. Face recognition with learning-based descriptor. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2707–2714, 2010.

[26] Kumar, N.; Berg, A. C.; Belhumeur, P. N.; Nayar, S. K. Attribute and simile classifiers for face verification. In: Proceedings of the IEEE 12th International Conference on Computer Vision, 365–372, 2009.

[27] Yin, Q.; Tang, X.; Sun, J. An associate-predict model for face recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 497–504, 2011.

[28] Taigman, Y.; Wolf, L. Leveraging billions of faces to overcome performance barriers in unconstrained face recognition. *arXiv preprint* arXiv:1108.1122, 2011.

[29] Chen, D.; Cao, X.; Wang, L.; Wen, F.; Sun, J. Bayesian face revisited: A joint formulation. In: *Computer Vision–ECCV 2012*. Fitzgibbon, A.; Lazebnik, S.; Perona, P.; Sato, Y.; Schmid, C. Eds. Springer Berlin Heidelberg, 566–579, 2012.

[30] Berg, T.; Belhumeur, P. N. POOF: Part-based one-vs.-one features for fine-grained categorization, face verification, and attribute estimation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 955–962, 2013.

[31] Berg, T.; Belhumeur, P. N. Tom-vs-Pete classifiers and identity-preserving alignment for face verification. In: Proceedings of the British Machine Vision Conference, 2012. Available at http://www.bmva.org/bmvc/2012/BMVC/paper129/paper129.pdf.

[32] Chen, D.; Cao, X.; Wen, F.; Sun, J. Blessing of dimensionality: High-dimensional feature and its efficient compression for face verification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 3025–3032, 2013.

[33] Cao, X.; Wipf, D.; Wen, F.; Duan, G.; Sun, J. A practical transfer learning algorithm for face verification. In: Proceedings of the IEEE International Conference on Computer Vision, 3208–3215, 2013.

[34] Zhu, Z.; Luo, P.; Wang, X.; Tang, X. Recover canonical-view faces in the wild with deep neural networks. *arXiv preprint* arXiv:1404.3543, 2014.

[35] TCIT. Available at http://www.tcit-us.com/.

[36] INSKY. Available at http://www.insky.so/.

[37] DaHua-FaceImage. Available at http://www.dahuatech.com/.

Zongguang Lu received his B.E. degree in information engineering (system engineering) from Nanjing University of Information Science and Technology, Nanjing, China, in 2015. Since 2015, he has been a master student in the School of Information and Control Engineering at Nanjing University of Information Science and Technology, Nanjing, China. His research interests include pattern recognition, face analysis, and computer vision.

Jing Yang has been a master student in the School of Information and Control Engineering at Nanjing University of Information Science and Technology since September 2014. She received her bachelor degree in system engineering from Nanjing University of Information Science and Technology in June 2014. Her research interests include machine learning and computer vision.

Qingshan Liu is a professor in the School of Information and Control Engineering at Nanjing University of Information Science and Technology, Nanjing, China. He received his Ph.D. degree from the National Laboratory of Pattern Recognition, Chinese Academy of Sciences, Beijing, China, in 2003, and his M.S. degree from the Department of Auto Control at Southeast University, Nanjing, China, in 2000. He was an assistant research professor in the Department of Computer Science, Computational Biomedicine Imaging and Modeling Center, Rutgers, the State University of New Jersey, from

2010 to 2011. Before that, he was an associate professor in the National Laboratory of Pattern Recognition, Chinese Academy of Sciences, and an associate researcher in the Multi-media Laboratory, Chinese University of Hong Kong, in 2004–2005. He was a recipient of the President's Scholarship of the Chinese Academy of Sciences in 2003. His current research interests are image and vision analysis, including face image analysis, graph and hypergraph-based image and video understanding, medical image analysis, and event-based video analysis.