Video Face Recognition: Component-wise Feature Aggregation Network (C-FAN)

Sixue Gong   Yichu Shi   Anil K. Jain

Michigan State University, East Lansing, MI

{gongsixu,shiyichu}@msu.edu, jain@cse.msu.edu

Abstract

We propose a new approach to video face recognition. Our component-wise feature aggregation network (C-FAN) accepts a set of face images of a subject as an input, and outputs a single feature vector as the face representation of the set for the recognition task. The whole network is trained in two steps: (i) train a base CNN for still image face recognition; (ii) add an aggregation module to the base network to learn the quality value for each feature component, which adaptively aggregates deep feature vectors into a single vector to represent the face in a video. C-FAN automatically learns to retain salient face features with high quality scores while suppressing features with low quality scores. The experimental results on three benchmark datasets, YouTube Faces [39], IJB-A [13], and IJB-S [12] show that the proposed C-FAN network is capable of generating a compact feature vector with 512 dimensions for a video sequence by efficiently aggregating feature vectors of all the video frames to achieve state of the art performance.

1. Introduction

Video-based face recognition has received increasing interest in recent years as a growing number of videos are continually being captured by mobile devices and CCTV systems for surveillance. Due to requirements in law enforcement and the fact that videos contain rich temporal and multi-view information, it is necessary to develop robust and accurate face recognition techniques for surveillance systems. Although the ubiquity of deep learning algorithms has advanced face recognition technology for static face images, video-based face recognition still poses a significant research challenge. Compared to static photos which are generally taken under controlled conditions (in terms of pose, illumination, expression, and occlusion) and with subject's cooperation, individual video frames have relatively low image quality because of unconstrained capture environments [12].

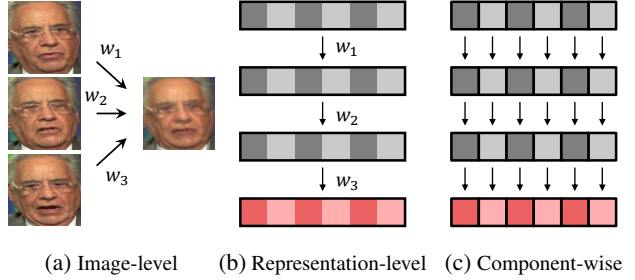(a) Image-level   (b) Representation-level   (c) Component-wise

Figure 1: Methods to aggregate identity information from a set of video frames of the same person. Figure 1a: individual frames are merged (e.g., pixel wise pooling or generative model to obtain a fused image); Figure 1b: deep feature vector for each video frame is assigned a single weight, W_i; Figure 1c: weighted average of feature components, where weights (quality value) are learned by the same network which gives the feature vector (see Figure 4)

There are two main challenges in face recognition in surveillance videos. First, subjects captured by surveillance cameras are usually moving without directly looking at the cameras, leading to large pose variations and severe motion blur. Second, the video consists of multiple frames of the same subject, providing both noisy frames with poor quality and unfavorable viewing angles, but useful temporal and multi-view information (see Figure 2d). Therefore, efficiently using a set of frames to generate a compact face representation is another challenge in video-based face recognition.

State-of-the-art face recognition algorithms for still images strive to generate robust face representations by deep neural networks trained on a large face dataset [28], [34], [18], [33]. Although the problems of image blur, pose variations, and occlusions can be partially solved by data augmentation [6] and ensemble convolutional neural networks [6], [33], [17], such strategies may not be directly applicable to video frames. Furthermore, the majority of existing deep face recognition approaches cast network predictions as point estimates, which means every face image has an explicit representation regardless of quality. However, there are many noisy video frames with low facial infor-

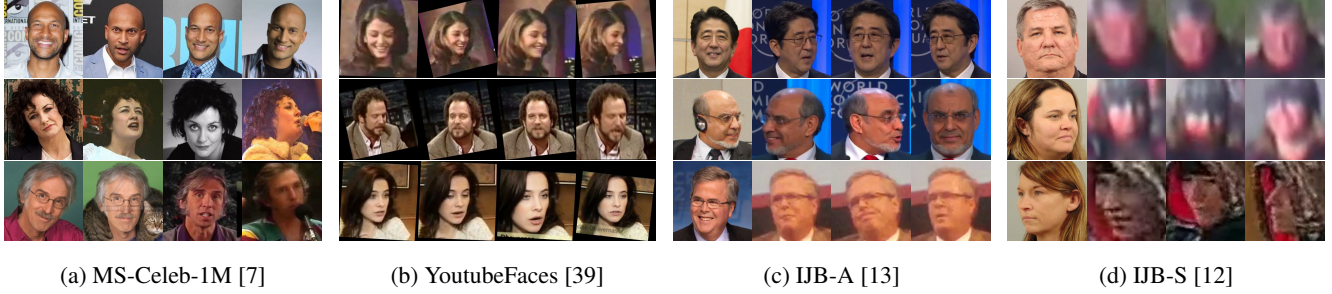| (a) MS-Celeb-1M [7] | (b) YoutubeFaces [39] | (c) IJB-A [13] | (d) IJB-S [12] |

Figure 2: Example images from different datasets. MS-Celeb-1M and YoutubeFaces contain only still images or video frames, respectively. IJB-A and IJB-S include both still images and videos. The first columns of IJB-A and IJB-S show still images, followed by video frames of the respective subjects in the next three columns.

mation content. For those frames, the corresponding face representation should have lower reliability. Also, in many practical surveillance applications, video clips of a range of frames for each identity can be collected. It is crucial to effectively integrate the information across different frames together from frame-level features [19], [40] or raw input frames [10], [25] by assigning each instance[1] with an adaptive instance-level weight. However, since each component of an instance encodes different subsets of facial features, noise could also be integrated when we emphasize or suppress all components simultaneously.

In this work, we propose a new approach for video face recognition by considering component-wise feature aggregation. We cast video face recognition as a template (a set of images from the same person) matching problem [2]. Figure 1 shows the proposed component-wise feature aggregation, compared to the other two instance-level aggregation methods. In component-wise aggregation, each component of the feature vectors is aggregated separately by learning the corresponding quality weight vectors as shown in Figure 1c. Based on the hypothesis that different covariates only lead to a variation in a subset of deep feature components, we train the network to predict an adaptive quality score for each deep feature component. The network can be trained on a still image dataset (e.g. MS-Celeb-1M [7], Figure 2a) with randomly generated templates. During testing, the features of each video or template are aggregated independently on each component using the respective quality scores. Experimental results on the IJB-S dataset [12] as well as other template/video matching benchmarks show that the proposed method significantly boosts the performance compared with average pooling and outperforms instance-level feature aggregation. The contributions of the paper are summarized below:

- A component-wise feature aggregation network (C-FAN) that aggregates each component of deep feature

vectors separately, leading to better representation for video face recognition.

- The quality scores predicted by C-FAN correlate with the visual image quality.

- Evaluating the proposed C-FAN on a challenging surveillance dataset IJB-S [12]. Consequently, we achieve state-of-the-art results on the five face identification protocols.

- We also attain comparable performance on the other two face recognition benchmarks, YouTube Faces [39], and IJB-A [13].

2. Related Work

2.1. Deep Face Recognition

Deep neural networks dominate the ongoing research in face recognition [34], [32], [28], [22], [18], [9], [24], [36], given their success in the ImageNet competition [14]. Taigman *et al*. [34] proposed the first face recognition application using deep neural networks. The following works have been exploring different loss functions to improve the discriminability of the embedding vector. Wen *et al*. [38] proposed center loss to reduce the intra-class variation. Other work proposed new metric learning methods for face recognition [28] [31]. More recent work has been trying to close the gap between metric learning and identification learning by learning classification on a spherical embedding space [18] [36], [37], [24].

2.2. Video Face Recognition

Previous work on video face recognition can mainly be categorized into three categories: *space*-based, *classifier*-based method and *aggregation*-based. *Space*-based methods aim to model the feature space spanned by the instances in the video, such as probabilistic distributions [29], [1], affine hulls [29], [41], [3], SPD matrices [11], n-order statistics [20] and manifolds [15], [8]. *Classifier*-based methods train a supervised classifier on each image set or

---

[1]An image or a feature vector.

[2]The same setting as the IARPA Janus Benchmark [13]

video to obtain a representation [39], [5]. Because deep convolution neural networks are shown to be able to lead to highly discriminative compact face representations from a collection of images, recent works on video face recognition mainly use an aggregation-based method to fuse a set of feature vectors or images into a single vector. Compared to previous methods as well as score-level fusion [2], aggregation methods are more efficient in terms of both computation and storage. In particular, image-aggregation methods fuse a set of images to a single image [10], [25] for feature extraction while feature-aggregation methods fuses the features extracted from different images into one vector. Although some feature-aggregation methods achieve good performance by simply using an average pooling [6], [4], a number of recent work has focused on fusing features with visual qualities, such as detection score [23], or predicted quality scores [19], [40], [26]. However, all of them only consider an instance-level aggregation while our work tries to aggregate the feature vectors in each component separately.

3. Component-wise Feature Aggregation

3.1. Motivation

Having multiple frames of a face in a video clip can be both advantageous and challenging at the same time. On the one hand, multiple frames incorporate temporal and multi-view information; on the other hand, large pose variations and motion blur of the frames will incur data noise and may impair the accuracy of video-based face recognition. Therefore, a key issue is how to aggregate the rich information in video frames to generate a face representation with stronger discriminative power than of individual frames.

Hassner *et al*. [10] found that simply eliminating poor quality frames will not necessarily help improve the face recognition accuracy. One of the possible reasons is that even poor quality frames carry some valuable information. Efforts have been made to aggregate multiple frames into one single image [10], [25]. A more straightforward solution might be extracting a feature vector for each frame and averaging the feature vectors to obtain a single video representation, which is known as average pooling. Let $T = \{I_1, I_2, \cdots, I_N\}$ be a template of face images. Assume that each subject has a noiseless embedding $\boldsymbol{\mu}$, and a feature vector of the ith image in the template T generated by face representation model is

$$\mathbf{f}_i = \boldsymbol{\mu} + \boldsymbol{\epsilon}_i, \qquad (1)$$

where $\boldsymbol{\epsilon}_i$ is the noise caused by the variations in the image. The template representation of average pooling can thus be explained as

$$\mathbf{r}^{avg} = \frac{1}{N} \sum_i (\boldsymbol{\mu} + \boldsymbol{\epsilon}_i) = \boldsymbol{\mu} + \frac{1}{N} \sum_i \boldsymbol{\epsilon}_i, \qquad (2)$$
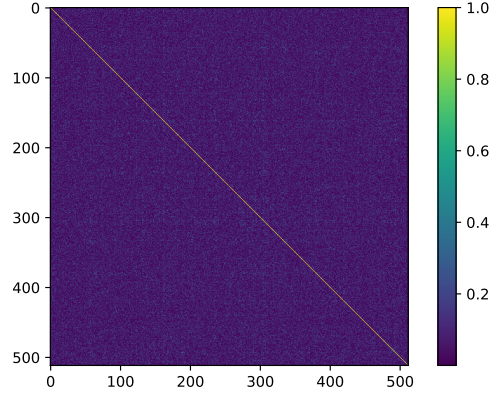

Figure 3: Heat map of the intra-class deep features extracted from the feature embedding module of C-FAN on IJB-S dataset [12]. Instead of original correlation values, we show their absolute values because we are only interested in their magnitude.

where it assumes the expectation of noise $E(\boldsymbol{\epsilon}) = \mathbf{0}$. If the number of images N is big enough, the approximation of expectation $\hat{E}(\boldsymbol{\epsilon}) = \frac{1}{N} \sum_i \boldsymbol{\epsilon}_i$ will converge to $\mathbf{0}$. However, the assumption cannot hold when N is small. Previous work [19], [40] attempts to solve this problem by learning a weight scalar for each feature vector (referred to as instance) and their template representation can formulated as:

$$\mathbf{r}^{inst} = \frac{1}{N} \sum_i w_i(\boldsymbol{\mu} + \boldsymbol{\epsilon}_i) = \boldsymbol{\mu} + \frac{1}{N} \sum_i w_i \boldsymbol{\epsilon}_i, \qquad (3)$$

where w_i is the weight for the feature vector of the ith image. Although it can reduce noise to some extent, it still assumes that individual components of a feature vector have correlated noise such that $\frac{1}{N} \sum_i w_i \epsilon_{ij} < \frac{1}{N} \sum_i \epsilon_{ij}$ for all j, where j is the index of the jth feature component. However, such assumption does not hold if each component contains different amount of identity information and their noise values are weakly correlated to each other. In Figure 3 we can observe that the intra-class correlations between individual components (off-diagonal values) of the deep feature vectors are quite small (close to 0). This inspires us to learn component-wise feature aggregation to further minimize noise on each component. The proposed fusion will generate different weights for each component of the deep feature vectors, and each component of the template representation is

$$r_j^{C-FAN} = \mu_j + \frac{1}{N} \sum_i w_{ij} \epsilon_{ij}. \qquad (4)$$

The proposed method will make better use of the components with more identity information instead of enhancing or weakening the whole feature vector if different components of a feature vector contain different identity information.
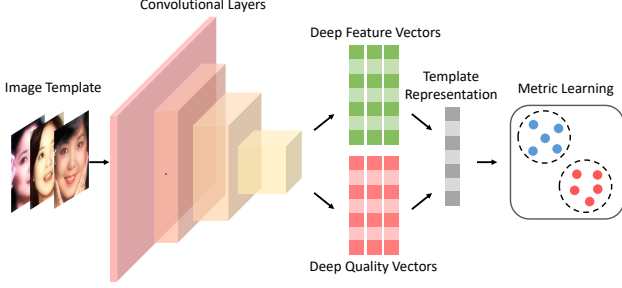
Figure 4: Framework of the proposed C-FAN.

3.2. Overall Framework

The overall framework of the proposed C-FAN is presented in Figure 4. C-FAN incorporates a base CNN model for extracting face representation and a feature aggregation module. The base model is first trained on a large scale face dataset (MS-Celeb-1M [7]). Then we fix the base model and train the aggregation module to learn the quality score for each single feature component on the same dataset using a different objective function. The aggregation module is attached to the last convolutional layer of the base model as a second branch besides the feature extraction layer. The features of a template are then pooled with the component-wise quality scores into a compact feature vector.

3.3. Feature Aggregation Module

Let $T = \{I_1, I_2, \cdots, I_N\}$ be a template of face images. Notice that N here is not a fixed number but changes with template size. Let $\mathcal{H}(\cdot)$ denote deep convolutional layers before the feature extraction layer, and $\mathcal{F}(\cdot)$ denote the feature extraction layer. The feature map of ith image in the template is generated by: $\mathcal{H}(I_i) = \mathbf{m}_i$, whose feature vector is then extracted by: $\mathcal{F}(\mathbf{m}_i) = \mathbf{f}_i$, where $\mathbf{f}_i$ is a D-dimensional feature vector. The corresponding quality vector is obtained by $\mathcal{Q}(\mathbf{m}_i) = \mathbf{q}_i$, where $\mathcal{Q}(\cdot)$ is the feature aggregation module, and $\mathbf{q}_i$ has the same dimension D as the feature vector. The quality vectors are then normalized with a softmax operator. Formally, given a set of quality vectors $\{\mathbf{q}_1, \mathbf{q}_2, \ldots, \mathbf{q}_N\}$, the jth component of the ith vector is normalized by:

$$w_{ij} = \frac{exp(q_{ij})}{\sum_{k=1}^{N} exp(q_{kj})} \quad (5)$$

The final face representation of a template is obtained by pooling the feature vectors using the normalized quality vectors

$$\mathbf{r} = \sum_{i=1}^{N} \mathbf{f}_i \odot \mathbf{w}_i, \quad (6)$$

where $\odot$ represents the operation of element-wise multiplication. The aggregated feature vector of a template $\mathbf{r}$ has the same dimensionality as a single face image frame extracted by the embedding module.

3.4. Training C-FAN

Given a pre-trained CNN for face feature extraction, C-FAN adds an aggregation module to the original network to predict the quality for each component. Unlike previous work [40] which takes the feature vector as input, we use the last feature map as the input of the aggregation layer since it contains more information. The feature aggregation module includes only one batch normalization layer and one fully connected layer. The batch normalization layer is added here to alleviate the difficulty of hyper-parameter tuning by normalizing the input of the aggregation module. During training, we fix the base CNN model and only update the aggregation layer. That is, the new model only learns how to aggregate the original features without changing them. To train the aggregation layer, we use a template triplet loss, where each triplet consists of three random templates. The templates are randomly constructed online for each mini-batch. The template features are obtained by Equation (6). Online hard triplet mining is adopted and the loss function is given by:

$$\mathcal{L}_{triplet} = \sum_{i=1}^{M} [\alpha + d(\mathbf{r}^a, \mathbf{r}^+) - d(\mathbf{r}^a, \mathbf{r}^-)]_+, \quad (7)$$

where M is the number of triplets, $\mathbf{r}^a$, $\mathbf{r}^+$ and $\mathbf{r}^-$ are the fused features of anchor template, positive template and negative template, respectively. $[x]_+ = \max\{0, x\}$, α is a margin parameter and $d(\cdot)$ is the squared euclidean distance.

4. Experiments

4.1. Datasets and Protocols

We evaluated C-FAN on three datasets including two video face datasets, the IARPA Janus Benchmark – Surveillance (IJB-S) [12], the YouTube Face datast (YTF) [39]; and a template-based face dataset, the IARPA Janus Benchmark A (IJB-A) [13].

IJB-S: The IJB-S dataset [12] is a new dataset released by IARPA for evaluating face recognition systems on surveillance videos. The dataset includes 350 surveillance videos spanning 30 hours in total, 5, 656 enrollment images, and 202 enrollment videos. This dataset provides a number of challenging face videos that are captured in real-world environments with a variety of subject activities and interactions. There are five 1:N face identification evaluations (supporting both open- and closed-set evaluation) involving surveillance video, including (i) surveillance-to-still [3],

---
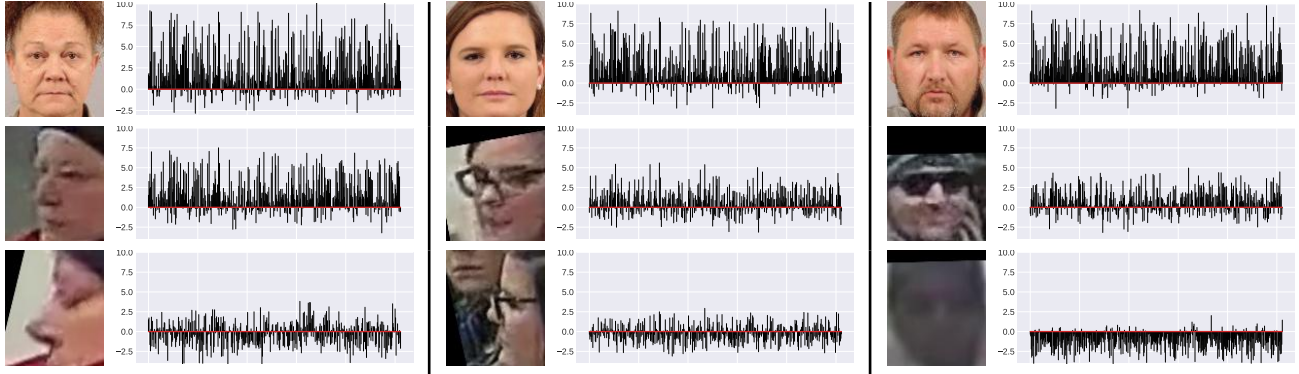[3]"Still" refers to single frontal still images.

Figure 5: Component-wise quality scores given by C-FAN for images of three subjects in the IJB-S dataset. Unnormalized quality scores are shown here for fair comparison. In the plots, quality values are shown for each of the 512 components. Each column shows a high-quality photo, a medium-quality frame and a low-quality frame from the same person, respectively.

(ii) surveillance-to-booking [4], (iii) Multi-view surveillance-to-booking, (iv) surveillance-to-surveillance, and (v) UAV [5] surveillance-to-booking. In our experiments, we follow all five identification protocols to evaluate our method. Because of the poor quality of the images in IJB-S, we are only able to detect around 9M out of 16M faces. In the original protocol of IJB-S, the recognition performance is normalized by the detection rate [12]. *However, since our work focuses on recognition and not detection, we do not follow the original metric. Instead, we report the standard Identification Rate (IR) and TPIR@FPIR*[6]. Failure-to-enroll images are ignored during feature aggregation. For templates that do not contain any detected faces, we set its representation as a zero-vector.

YTF: The YouTube Face dataset is a video face dataset released in 2011 [39]. It contains $3,425$ videos of $1,595$ different subjects. The number of frames in the YTF face videos ranges from 48 to $6,070$, and the average number of frames is 181. Compared with IJB-S, the media of YTF is more photojournalistic [12]. In experiments, we follow the 1:1 face verification protocol with the given $5,000$ video pairs.

IJB-A: IJB-A[13] is a template-based unconstrained face recognition benchmark. Although its images present similar challenges as IJB-S, the templates in the IJB-A include images from mixed media sources, with average image quality being better than IJB-S. The benchmark provides template-based 1:1 verification and 1:N identification protocols. IJB-A contains 500 subjects with a total of $25,813$ images, and has been widely used by a number of both still image and video-based face recognition algorithms.

4.2. Implementation Details

Pre-processing: We employ a facial landmark detection algorithm MTCNN [42] to detect and align all faces in training and testing. Each face is cropped from the detected face region and resized to 112×96 pixels using a similarity transformation based on the detected five facial landmarks.

Training: We first train a base CNN model on a cleaned version[7] of MS-Celeb-1M dataset [7] to learn face recognition of still images, for which we adopts the Face-ResNet (DeepVisage) architecture [9]. The component-wise quality module is then trained on the same training dataset using the template triplet loss. The parameters of the aggregation layer are optimized using stochastic gradient descent with a momentum of 0.9 and a weight decay of 0.01. Gaussian blur and motion blur are randomly applied to augment the training data. We set $\alpha = 1.0$. Each mini-batch samples 240 images of 80 templates from 20 random subjects, 3 images per template. The module is trained for $4,000$ steps in all. All these hyper-parameters are chosen by testing on a validation set, which is a small subset of IJB-S. All the experiments are conducted on a Nvidia Geforce GTX 1080 Ti GPU. The average feature extraction speed on this GPU is 1ms per image. After training on MS-Celeb-1M, the feature network and aggregation module are applied to the testing datasets without fine-tuning.

4.3. Baseline

We compare the proposed component-wise pooling with the other two pooling types, average pooling and instance-level pooling, in particular we compare between three methods:

- **Average** directly applies the average pooling to the base CNN features to generate the template representation.

---

[4]The "booking" reference template comprises the full set of images captured of a single subject at enrollment time.

[5]UAV is a small fixed-wing unmanned aerial vehicle that was flown to collect images and videos.

[6]True positive identification and false positive identification rate.

[7]https://github.com/AlfredXiangWu/face_verification_experiment.
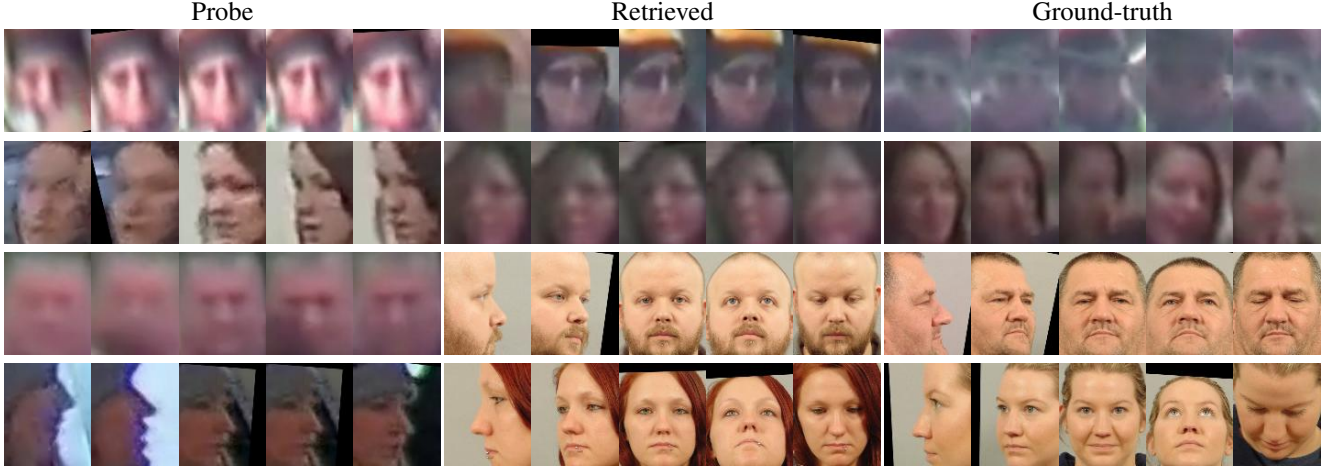
|Probe|Retrieved|Ground-truth|

Figure 6: Examples of failed templates on the IJB-S dataset. The first two rows show the failure cases from "surveillance-to-surveillance" protocol. The last two show the failure cases from "surveillance-to-booking" protocol. Because of space limit, we only show 5 randomly selected images for each template.

- **Instance** is trained similarly as C-FAN, but it only outputs one quality scalar for each image and aggregates the base CNN feature at the instance-level.
- **C-FAN** generates a quality vector for each image and does component-wise aggregation.

After the aggregation, cosine similarity is used to compare the template representations.

4.4. Qualitative Analysis on IJB-S

To explore the relationship between visual quality and the quality scores generated by C-FAN, we visualize the component-wise quality scores of images from IJB-S dataset. Figure 5 shows the images of three subjects with the corresponding 512-dimensional quality vectors. We can observe that images with high visual quality have higher quality scores than those with low visual quality. The images with motion blur, occlusion, and deformity tend to be assigned with negative quality values. We also observe that each component of a deep feature vector presents different response to the noise in images.

Figure 6 shows some failure examples of face identification on IJB-S dataset. For the "Surveillance-to-Surveillance" protocol, both probe templates and gallery templates are frames from surveillance videos, making it very difficult to discriminate identities from the images with such low quality and significant variations. However, we observe that the proposed C-FAN tends to retrieve templates with good visual quality. The ground-truth images are more blurry than the retrieved ones in the top two rows of Figure 6. Even though probe frames have low visual quality in the "Surveillance-to-booking" protocol, we can still observe some facial patterns shared by the probe and the retrieved templates, e.g., the same hair style, the blurry cap and the hair bang.

4.5. Quantitative Analysis on IJB-S

Table 1 reports identification results on IJB-S dataset. We compare C-FAN with average pooling and instance-level aggregation. Both instance-level aggregation and C-FAN outperforms average pooling on almost all protocols of IJB-S, showing that the features aggregated by quality weights are more discriminative than by simply computing the mean representations. Furthermore, compared with instance-level aggregation, C-FAN achieves higher identification rates on most of the protocols, showing the advantage of the aggregating different features components separately. With the expectation of the Multi-view protocol, the performance of our model on IJB-S dataset is significantly lower than on other datasets (See Section 4.6), showing that this dataset is indeed much more challenging. Notice that the open-set TPIR of "UAV Surveillance-to-booking" are all 0%. This is because 29 out of 79 probe templates are empty (no faces detected) and thus it is impossible to control the FPIR under 36.71%.

4.6. Performance Comparison on YTF and IJB-A

Table 2 and Table 3 report the results of C-FAN, the corresponding baseline methods, and other state-of-the-art on YTF and IJB-A, respectively. C-FAN achieves the best performance on YTF dataset. In particular, C-FAN outperforms the state-of-the-art algorithm NAN [40] by 0.78% and QAN by 0.33%. On IJB-A dataset, C-FAN achieves the best performance on 1:1 verification and open-set identification tasks, compared to previous methods. It can be seen that C-FAN outperforms NAN by 3.49% at 0.1% FAR, and by 5.18% at 1% FPIR, respectively. Notice that the gaps between C-FAN and two baseline pooling methods are relatively small. This is because the images in YTF and IJB-A datasets are not typical video frames from real-world

Table 1: Performance comparisons on IJB-S dataset.

Test Name	Method	Closed-set (%)			Open-set (%)	
		Rank-1	Rank-5	Rank-10	1 % FPIR	10 % FPIR
Surveillance-to-still	Average	47.93	58.86	63.20	15.22	21.77
	Instance	49.57	59.58	64.07	**16.48**	23.14
	C-FAN	**50.82**	**61.16**	**64.95**	16.44	**24.19**
Surveillance-to-booking	Average	49.31	59.92	64.44	26.27	27.05
	Instance	51.14	61.43	65.37	26.64	28.16
	C-FAN	**53.04**	**62.67**	**66.35**	**27.40**	**29.70**
Multi-view Surveillance-to-booking	Average	92.57	97.52	99.01	58.42	77.72
	Instance	94.55	99.01	99.50	61.39	83.17
	C-FAN	**96.04**	**99.50**	**99.50**	**70.79**	**85.15**
Surveillance-to-Surveillance	Average	9.38	17.44	**22.14**	0.03	0.54
	Instance	8.90	16.61	21.38	0.06	0.54
	C-FAN	**10.05**	**17.55**	21.06	**0.11**	**0.68**
UAV Surveillance-to-booking	Average	1.27	8.86	13.92	0.00	0.00
	Instance	5.06	11.39	15.19	0.00	0.00
	C-FAN	**7.59**	**12.66**	**20.25**	0.00	0.00

Table 2: Verification performance on YouTube Face dataset, compared with baseline methods and other state-of-the-art methods.

Method	Accuracy (%)	Method	Accuracy (%)
EigenPEP [16]	84.8 ± 1.4	DeepFace [34]	91.4 ± 1.1
DeepID2+ [33]	93.2 ± 0.2	Wen *et al.* [38]	94.9
FaceNet [28]	95.52 ± 0.06	DAN [25]	94.28 ± 0.69
NAN [40]	95.72 ± 0.64	QAN [19]	96.17 ± 0.09
Average	96.36 ± 1.01	*Instance*	96.42 ± 0.95
		C-FAN	**96.50 ± 0.90**

surveillance scenarios and environments. Most images in these datasets contain rich information for face recognition. Thus, a smaller improvement is expected for quality-aware feature aggregation.

5. Conclusion

In this paper, we propose a component-wise feature aggregation network (C-FAN) for video face recognition. It adaptively predicts quality values for each component of a deep feature vector and aggregates the most discriminative components to generate a single feature vector for a set of face images. We empirically demonstrate that the quality scores predicted by C-FAN fit the visual quality of images and are also beneficial in template representation by retaining discriminative components of feature vectors. Our future work will explore an aggregation network that combines different levels of fusion.

References

[1] O. Arandjelovic, G. Shakhnarovich, J. Fisher, R. Cipolla, and T. Darrell. Face recognition with image sets using manifold density diver-gence. In *CVPR*, 2005.

[2] L. Best-Rowden, H. Han, C. Otto, B. F. Klare, and A. K. Jain. Unconstrained face recognition: Identifying a person of interest from a media collection. *IEEE Trans. on Information Forensics and Security*, 2014.

[3] H. Cevikalp and B. Triggs. Face recognition based on image sets. In *CVPR*, 2010.

[4] J.-C. Chen, R. Ranjan, S. Sankaranarayanan, A. Kumar, C.-H. Chen, V. M. Patel, C. D. Castillo, and R. Chellappa. Unconstrained still/video-based face verification with deep convolutional neural networks. *IJCV*, 2018.

[5] N. Crosswhite, J. Byrne, C. Stauffer, O. Parkhi, Q. Cao, and A. Zisserman. Template adaptation for face verification and identification. In *IEEE FG*, 2017.

[6] C. Ding and D. Tao. Trunk-branch ensemble convolutional neural networks for video-based face recognition. *IEEE Trans. on PAMI*, 2018.

[7] Y. Guo, L. Zhang, Y. Hu, X. He, and J. Gao. Ms-celeb-1m: A dataset and benchmark for large scale face recognition. In *ECCV*, 2016.

[8] M. T. Harandi, C. Sanderson, S. Shirazi, and B. C. Lovell. Graph embedding discriminant analysis on grassmannian manifolds for improved image set matching. In *CVPR*, 2011.

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

[10] T. Hassner, I. Masi, J. Kim, J. Choi, S. Harel, P. Natarajan, and G. Medioni. Pooling faces: template based face recognition with pooled face images. In *CVPR Workshops*, 2016.

[11] Z. Huang, R. Wang, S. Shan, X. Li, and X. Chen. Log-euclidean metric learning on symmetric positive definite manifold with application to image set classification. In *ICML*, 2015.

[12] N. D. Kalka, B. Maze, J. A. Duncan, K. J. OConnor, S. Elliott, K. Hebert, J. Bryan, and A. K. Jain. IJB-S : IARPA Janus Surveillance Video Benchmark . In *BTAS*, 2018.

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

[14] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

Table 3: Comparisons of verification and identification on IJB-A dataset.

Method	Verification TAR (%)		Close-set Identification (%)		Open-set Identification (%)	
	0.1% FAR	1% FAR	Rank-1	Rank-5	1% FPIR	10% FPIR
LSFS [35]	51.4 ± 6.0	73.3 ± 3.4	82.0 ± 2.4	92.9 ± 1.3	38.3 ± 6.3	61.3 ± 3.2
Pose-aware Models [21]	65.2 ± 3.7	82.6 ± 1.8	84.0 ± 1.2	92.5 ± 0.8	-	-
Masi *et al.* [22]	72.5	88.6	90.6	96.2	-	-
Triplet Embedding [27]	81.3 ± 2	90.0 ± 1.0	93.2 ± 1.0	-	75.3 ± 3.0	86.3 ± 1.4
Shi *et al.* [30]	60.2 ± 6.9	82.3 ± 2.2	89.8 ± 0.92	96.0 ± 0.6	58.9 ± 1.6	64.7 ± 1.4
QAN [19]	89.3 ± 3.9	94.2 ± 1.5	-	-	-	-
NAN [40]	88.1 ± 1.1	94.1 ± 0.8	95.8 ± 0.5	98.0 ± 0.5	81.7 ± 4.1	91.7 ± 0.9
Pooling Faces [10]	63.1	81.9	84.6	93.3	-	-
Average	91.14 ± 1.13	$\mathbf{93.99 \pm 0.79}$	94.44 ± 0.88	96.22 ± 0.53	86.23 ± 4.30	92.36 ± 1.00
Instance	91.52 ± 0.73	93.78 ± 0.77	94.54 ± 0.87	96.24 ± 0.57	86.58 ± 4.76	92.45 ± 0.99
C-FAN	$\mathbf{91.59 \pm 0.99}$	93.97 ± 0.78	$\mathbf{94.57 \pm 0.82}$	$\mathbf{96.27 \pm 0.53}$	$\mathbf{86.88 \pm 4.70}$	$\mathbf{92.85 \pm 1.02}$

[15] K.-C. Lee, J. Ho, M.-H. Yang, and D. Kriegman. Video-based face recognition using probabilistic appearance manifolds. In *CVPR*, 2003.

[16] H. Li, G. Hua, X. Shen, Z. Lin, and J. Brandt. Eigen-pep for video face recognition. In *ACCV*, 2014.

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

[18] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song. Sphereface: Deep hypersphere embedding for face recognition. In *CVPR*, 2017.

[19] Y. Liu, J. Yan, and W. Ouyang. Quality aware network for set to set recognition. In *CVPR*, 2017.

[20] J. Lu, G. Wang, and P. Moulin. Image set classification using holistic multiple order statistics features and localized multi-kernel metric learning. In *ICCV*, 2013.

[21] I. Masi, S. Rawls, G. Medioni, and P. Natarajan. Pose-aware face recognition in the wild. In *CVPR*, 2016.

[22] I. Masi, A. T. Trn, T. Hassner, J. T. Leksut, and G. Medioni. Do we really need to collect millions of faces for effective face recognition? In *ECCV*, 2016.

[23] R. Ranjan, A. Bansal, H. Xu, S. Sankaranarayanan, J.-C. Chen, C. D. Castillo, and R. Chellappa. Crystal loss and quality pooling for unconstrained face verification and recognition. *arXiv:1804.01159*, 2018.

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

[25] Y. Rao, J. Lin, J. Lu, and J. Zhou. Learning discriminative aggregation network for video-based face recognition. In *CVPR*, 2017.

[26] N. Sankaran, S. Tulyakov, S. Setlur, and V. Govindaraju. Metadata-based feature aggregation network for face recognition. In *ICB*, 2018.

[27] S. Sankaranarayanan, A. Alavi, C. D. Castillo, and R. Chellappa. Triplet probabilistic embedding for face verification and clustering. In *BTAS*, 2016.

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

[29] G. Shakhnarovich, J. W. Fisher, and T. Darrell. Face recognition from long-term observations. In *ECCV*, 2002.

[30] Y. Shi and A. K. Jain. Improving face recognition by exploring local features with visual attention. In *ICB*, 2018.

[31] K. Sohn. Improved deep metric learning with multi-class n-pair loss objective. In *NIPS*, 2016.

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

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

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

[35] D. Wang, C. Otto, and A. K. Jain. Face search at scale. *IEEE Trans. on PAMI*, 2017.

[36] F. Wang, W. Liu, H. Liu, and J. Cheng. Additive margin softmax for face verification. *arXiv:1801.05599*, 2018.

[37] H. Wang, Y. Wang, Z. Zhou, X. Ji, Z. Li, D. Gong, J. Zhou, and W. Liu. Cosface: Large margin cosine loss for deep face recognition. *arXiv:1801.09414*, 2018.

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

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

[40] J. Yang, P. Ren, D. Zhang, D. Chen, F. Wen, H. Li, and G. Hua. Neural aggregation network for video face recognition. In *CVPR*, 2017.

[41] M. Yang, P. Zhu, L. Van Gool, and L. Zhang. Face recognition based on regularized nearest points between image sets. In *IEEE FG*, 2013.

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