

Feature Aggregation Network for Video Face Recognition

Zhaoxiang Liu
CloudMinds

robin.liu@cloudminds.com

Huan Hu
CloudMinds

hans.hu@cloudminds.com

Jinjiang Bai
Beihang University

baijinjiang@buaa.edu.cn

Shaohua Li
CloudMinds

shaohua.li@cloudminds.com

Shiguo Lian
CloudMinds

sg_lian@163.com

Abstract

This paper aims to learn a compact representation of a video for video face recognition task. We make the following contributions: first, we propose a meta attention-based aggregation scheme which adaptively and fine-grained weighs the feature along each feature dimension among all frames to form a compact and discriminative representation. It makes the best to exploit the valuable or discriminative part of each frame to promote the performance of face recognition, without discarding or despising low quality frames as usual methods do. Second, we build a feature aggregation network comprised of a feature embedding module and a feature aggregation module. The embedding module is a convolutional neural network used to extract a feature vector from a face image, while the aggregation module consists of cascaded two meta attention blocks which adaptively aggregate the feature vectors into a single fixed-length representation. The network can deal with arbitrary number of frames, and is insensitive to frame order. Third, we validate the performance of proposed aggregation scheme. Experiments on publicly available datasets, such as YouTube face dataset and IJB-A dataset, show the effectiveness of our method, and it achieves competitive performances on both the verification and identification protocols.

1. Introduction

Video face recognition has become more and more significant in the past few years [41, 39, 31, 18, 25, 27, 42, 28, 19, 23, 38, 12, 44, 11, 32], which plays an important role in many practical applications such as visual surveillance, access control, person identification, video search and so on. Compared to single still image-based face recognition, further useful information of a single face can be exploited in the video. However, the video faces exhibit much richer

uncontrolled variations, *e.g.*, out-of-focus blur, motion blur, occlusion, varied illuminations and a large range of pose variations, which make video face recognition a challenging task. Hence, how to design a feature model which can effectively represent the video face across different frames becomes a key issue of video face recognition.

In video face recognition task, each subject usually has a varied number of face images. A straightforward approach would be to represent a video face as a set of face descriptors extracted by a deep neural network, compare every pair of face descriptors between two face videos [31, 34], and fuse the matching results across all pairs. However, this method would be considerably memory-consuming and inefficient especially for a large-scale recognition task. Consequently, an effective aggregation scheme, requiring minimal memory storage and supporting efficient similarity computation, is desired for this task, to generate a compact representation for a face video. And what is more, the aggregated representations should be discriminative, *i.e.*, they are expected to have smaller intra-class distance than inter-class distance under a suitably chosen metric space.

So far, a variety of efforts on integrating information across different frames have been dedicated [18, 25, 27, 42, 28, 11, 32, 7, 8, 1]. Besides max pooling [8], average pooling [18, 25, 32, 8] may be the most common aggregation technique. However, it considers all frames of equal importance during feature aggregation, in which case the low quality frames with some misleading feature would degrade the performance of recognition. Considering of this problem, some other methods either just focus on high quality frames, *i.e.*, feature-rich frames, while ignoring low quality frames, such as blurred faces, occluded faces and large pose faces [27, 12] or adaptively high weigh high quality frames while framesdown weigh low quality frames [42, 44].

Despite that those aggregation strategies have been shown to be effective in the previous works, we believe that an optimal aggregation strategy should not simply and

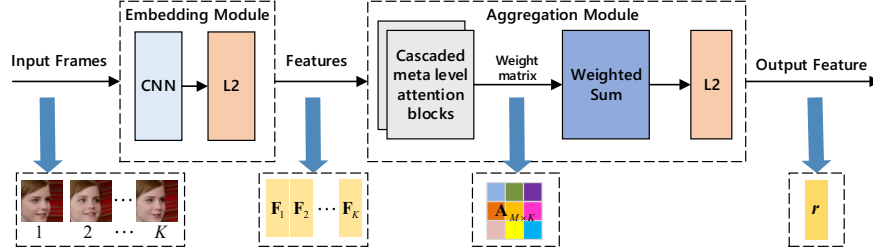


Figure 1. Network architecture. Input frames of a video are fed into feature embedding module to produce a set of normalized feature vectors. Then these features are passed through aggregation module to obtain a single fixed-size normalized feature vector for the video. The aggregation module mainly consists of cascaded two axis-level attention blocks which adaptively weighs the feature vectors along each feature axis among all frames, fusing the feature vectors organically.

crudely despise the low quality frames, because the low quality frames might even contain local discriminative features which can be complementary to high quality frames. In some sense, the low quality frames may be beneficial to video face recognition. Thus, the best aggregation result should be the composition of local discriminative features from low quality frames and other parts from high quality frames. Our intuition is simple and straightforward: an ideal algorithm should be able to emphasize the valuable part of the frame feature while suppress the worthless part of the frame feature irrespective of the face quality during aggregation, *i.e.*, it adaptively deals with each dimension of frame feature with different importance, not like NAN [42] that treats each dimension of equal importance for frame feature when aggregating. Let us imagine an extreme case: with some poor quality face images, *e.g.*, a variety of large pose faces each with different pose, it is possible to aggregate these faces into a discriminative face representation for video face recognition.

To this end, we propose a new attention-based aggregation network which adaptively and fine-grained weighs the feature along each feature dimension among all frames to form a compact and discriminative face representation. Different from previous methods, we neither focus only on high quality frames nor simply weigh the feature on frame-level. Instead, we design a neural network which is able to adaptively and fine-grained measure the importance of each dimension of the feature among all frames.

Our major contributions can be summarized as follows:

- We propose a novel feature aggregation scheme for video face recognition, and reveal why it could work better. It is a generalized feature aggregation scheme, and may also serve as a feature aggregation scheme for other computer vision tasks.
- Based on the proposed aggregation scheme, we construct a feature aggregation network (as shown in Figure 1) composed of two modules trained end-to-end or

one by one separately. One is the feature embedding module which is a frame-level feature extractor using deep CNN model. The other is the aggregation module which adaptively integrates the feature vectors of all the video frames together. Our feature aggregation network inherits the main advantages of the pooling techniques (*e.g.*, average pooling and max pooling), could handle arbitrary input size and produces order-invariant, fixed-size feature representation.

- We demonstrate the effectiveness of our proposed aggregation scheme in video face recognition by various comparative experiments. Trained on publicly available dataset, such as YouTube face dataset and IJB-A dataset, our method takes a lead over the baseline methods and is a competitive method compared to the state of the art methods.

2. Related works and preliminaries

Since our work is concerned with order-insensitive video or image set face recognition, any other methods exploiting the temporal information of video sequence will not be considered here.

Early traditional studies attempt to represent the face videos or image sets as manifolds [1, 14, 16, 36, 35, 37] or convex hulls [6] and compute their similarities under corresponding spaces. While those methods may work well under constrained scenarios, they are usually incapable of dealing with large face variations.

Some other methods extract the local features of frames and aggregate them across multiple frames to represent the videos [18, 17, 24]. For example, PEP-based methods [18, 17] take a part-based representation by extracting and merging LBP or SIFT descriptors, and the method in [24] applies Fisher vector encoding to represent each frame by extracting RootSIFT [3, 22] and fuses across multiple different video frames to form a video-level representation.

These years, still image-based face recognition has gained great success thank to deep learning techniques [31, 38, 34, 10, 20]. Based on this, some simple aggregation strategies are adopted in video face recognition. The methods in [31] and [34] utilize pairwise frame feature similarity computation and then fuse the matching results. Max-or average-pooling is used to aggregate the frame features in [25, 11, 7, 8]. Though DAN [27] proposes a GAN-like aggregation network which takes the video clip as input and reconstructs a single image as output to represent the video, the average pooling result of the video frames is employed to supervise the aggregation training. What is more, DAN is not suitable to tackle image set face recognition due to that a video face discriminator is used inside the GAN.

Recently, a few methods take a lead over the simple pooling techniques. The method in [12] utilizes discrete wavelet transform and entropy computation to select feature-rich frames from a video sequence and learns a joint feature from them. GhostVLAD [44] employs a modified NetVLAD [2] layer to down weigh the contribution of low quality frames. NAN [42] proposes an attention mechanism to adaptively weigh the frames, so that the contribution of low quality frames to the aggregation is down weighed. However, NAN considers each dimension of the feature vector to be of equal importance. These methods may lose some valuable information of the low quality images. This motivates us to seek a better solution in this paper.

Our work is inspired by NAN [42]. However, our aggregation scheme is a more generalized strategy, can fine-grained handle the feature vector on dimension level. Now, let us review the feature aggregation scheme of NAN [42]. Consider the video face recognition task on n pairs of video face data $(S^i, y_i)_{i=1}^n$, where S^i is a face video sequence or image set with varying image number K_i , *i.e.*, $S^i = \{x_1^i, x_2^i, \dots, x_{K_i}^i\}$ in which x_k^i , $k = 1, 2, \dots, K_i$ is the k -th frame in the video, and y_i is the corresponding subject ID of S^i . Each frame x_k^i has a corresponding normalized feature representation \mathbf{F}_k^i extracted from the feature embedding module, and the aggregated feature representation becomes

$$\mathbf{r} = \sum_{k=1}^{K_i} a_k^i \mathbf{F}_k^i, \quad (1)$$

where a_k^i is the linear weight generated from all feature vectors of a video, it can be formulated as

$$a_k^i = \frac{\exp(e_k^i)}{\sum_{j=1}^{K_i} \exp(e_j^i)}, \quad (2)$$

where e_k^i is the corresponding significance yielded via dot product with a kernel filter \mathbf{q} for each feature vector, it can be formulated as

$$e_k^i = \mathbf{q}^T \mathbf{F}_k^i. \quad (3)$$

Obviously, if $a_k^i = \frac{1}{K_i}$, Eq. (1) degrades to average pooling strategy.

3. Method

3.1. The proposed aggregation scheme

We argue that each dimension of the feature vector shares the common weight as NAN does is not optimal. The ideal strategy should be able to adaptively weigh each dimension of feature vector separately. So we leverage a kernel matrix \mathbf{Q} to filter the feature vector \mathbf{F}_k^i via product, yielding a significance vector \mathbf{E}_k^i , which describes the importance of each dimension of \mathbf{F}_k^i . Assuming \mathbf{F}_k^i is an M -dimension vector, then, we can formulate \mathbf{Q} as

$$\mathbf{Q} = \begin{bmatrix} \mathbf{q}_1^T \\ \mathbf{q}_2^T \\ \vdots \\ \mathbf{q}_M^T \end{bmatrix}_{M \times M}, \quad (4)$$

and formulate \mathbf{E}_k^i as

$$\mathbf{E}_k^i = \begin{bmatrix} e_{1k}^i \\ e_{2k}^i \\ \vdots \\ e_{Mk}^i \end{bmatrix}_{M \times 1} = \mathbf{Q} \mathbf{F}_k^i = \begin{bmatrix} \mathbf{q}_1^T \mathbf{F}_k^i \\ \mathbf{q}_2^T \mathbf{F}_k^i \\ \vdots \\ \mathbf{q}_M^T \mathbf{F}_k^i \end{bmatrix}_{M \times 1}. \quad (5)$$

After softmax operation along each dimension, a positive

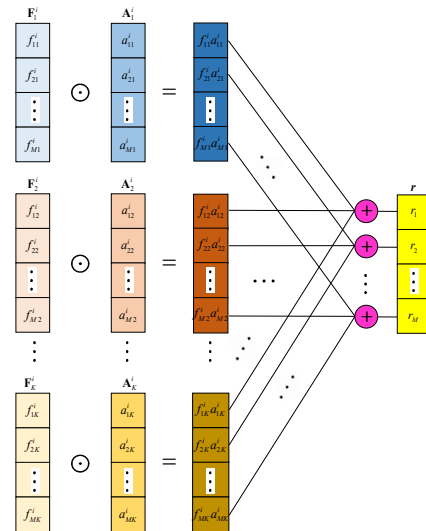


Figure 2. Element-wise weighted sum of features.

weight vector \mathbf{A}_k^i is generated as following

$$\mathbf{A}_k^i = \begin{bmatrix} a_{1k}^i \\ a_{2k}^i \\ \vdots \\ a_{Mk}^i \end{bmatrix}_{M \times 1} = \begin{bmatrix} \frac{\exp(e_{1k}^i)}{\sum_{j=1}^{K_i} \exp(e_{1j}^i)} \\ \frac{\exp(e_{2k}^i)}{\sum_{j=1}^{K_i} \exp(e_{2j}^i)} \\ \vdots \\ \frac{\exp(e_{Mk}^i)}{\sum_{j=1}^{K_i} \exp(e_{Mj}^i)} \end{bmatrix}_{M \times 1}, \quad (6)$$

where a_{mk}^i denotes the linear weight of that m -th dimension of the feature vector contributes to aggregation result, and $\sum_{k=1}^{K_i} a_{mk}^i = 1, \forall m \in \{1, 2, \dots, M\}$. So that the aggregated feature representation becomes

$$\mathbf{r} = \sum_{k=1}^{K_i} \mathbf{A}_k^i \odot \mathbf{F}_k^i, \quad (7)$$

where \odot represents element-wise product. Figure 2 shows the calculating process of \mathbf{r} . \mathbf{r} turns out to be $\tilde{\mathbf{r}}$ after $L2$ -normalization. Either cosine or $L2$ distance can be used to compute the similarity.

From above formulas and Figure 2, we can clearly see the difference between our method and NAN is that we use a kernel matrix instead of a kernel vector to adaptively weigh the feature. Therefore, we can measure the importance of feature on dimension level without constraining each dimension to share the same weight just as NAN [42] does. Compared to NAN and other pooling techniques, our method is more flexible, and can make each dimension of one feature vector adaptively contribute to the aggregation feature. In theory, it can realize optimal feature aggregation after well trained. So, our method can deal with every frame fairly regardless of face quality, and make the best to exploit its any valuable or discriminative local feature to promote the video face recognition.

What is more, our method is a more generalized feature aggregation scheme. Obviously, if $a_{1k}^i = a_{2k}^i = \dots = a_{Mk}^i$, Eq. (7) degrades to NAN, and if $a_{mk}^i = \frac{1}{K_i}$, Eq. (7) degrades to average pooling. And max pooling can also be regarded as a special case of our method.

3.2. The proposed feature aggregation network

Based the on proposed aggregation scheme, we construct a feature aggregation network comprised of two modules. As shown in Figure 1, the network can be fed with a set of face images of a subject and produces a single feature vector as its representation for the recognition task. It is built upon a modern deep CNN model for frame feature embedding, and adaptively aggregates all frames in the video into a compact vector representation.

The image embedding module of our network adopts the backbone network of Arc-Face [10] which greatly ad-

vances the image-based face recognition recently. The embedding module mainly consists of a ResNet50 which has an improved residual unit: BN-Conv-BN-PReLU-Conv-BN structure, while using BN-Dropout-FC-BN after the last convolutional layer. The embedding module produces 512-dimension image features which are first normalized to be unit vectors then fed into the aggregation module.

In order to obtain a better aggregation representation, a cascaded two attention blocks with nonlinear transfer is designed inside aggregation module as shown in Figure 3. Each attention block consists of a kernel filter and a nonlinear transfer. The kernel filter is implemented with a FC layer, while nonlinear transfer with a \tanh activation layer. Then \mathbf{E}_k^i becomes

$$\mathbf{E}_k^i = \tanh(\mathbf{Q}_2 \bar{\mathbf{E}}_k^i + \mathbf{b}_2), \quad (8)$$

where $\bar{\mathbf{E}}_k^i$ is the output of the first block, it can be formulated as

$$\bar{\mathbf{E}}_k^i = \tanh(\mathbf{Q}_1 \mathbf{F}_k^i + \mathbf{b}_1). \quad (9)$$

Therefore, besides kernel matrices \mathbf{Q}_1 and \mathbf{Q}_2 , biases \mathbf{b}_1 and \mathbf{b}_2 are also trainable parameters of aggregation module. We have to point out that our cascaded attention blocks are totally different from NAN [42]'s in that our attention block uses an importance matrix while NAN uses an importance vector to weigh the feature vectors. In comparison, our method is more fine-grained than NAN to aggregate feature vectors. Furthermore, NAN aggregates the feature vectors twice, where the second attention block takes the aggregation result of the first attention block as input. However, our method only makes aggregation once.

In addition, our network has several other favorable properties. First, it is able to tackle arbitrary number of images for one subject. Second, the aggregation result \mathbf{r} which is of the same size as a single feature \mathbf{F}_k^i is invariant to the image order, keeps unchanged when the image sequence are reshuffled or even reversed, *i.e.*, our network is insensitive to the temporal information of the video or image set. Third, it is adaptive to the input faces and whose all parameters are trainable through supervised learning with standard backpropagation and gradient descent.

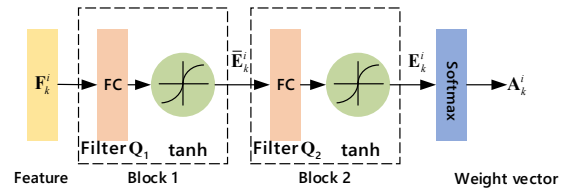


Figure 3. Cascaded two attention blocks.

3.3. Network training

To make the training faster and more stable, we divide it into three stages(as shown in Fig.4). Firstly, we train the embedding module for single image face recognition task. In this stage, the cleaned MS-Celeb-1M dataset [10, 13] is used. Secondly, we train the whole network end-to-end for set-based face recognition task, and the VGGFace2 dataset [5] is used in this stage. In order to boost the capability of handling images of varying quality that typically occur in the wild, the VGGFace2 dataset is augmented in the form of image degradation, such as blurring or compression. Finally, we finetune the whole network end-to-end on the training set of the benchmark dataset.

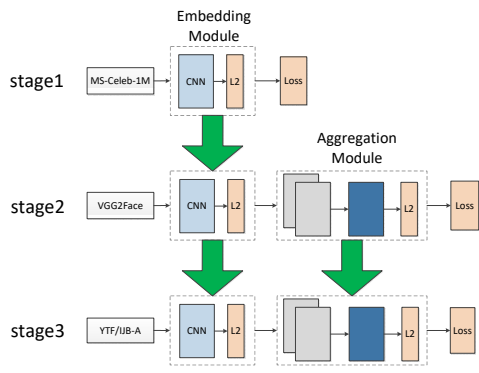


Figure 4. Network Training. In stage 1, only embedding module is trained; then the trained embedding module is copied to stage 2 for end-to-end training; finally, the whole network is copied to stage 3 for end-to-end finetuning.

4. Experiments

4.1. Datasets and protocols

We conduct experiments on two widely used datasets including the YouTube Face dataset (YTF) [40], IJB-A dataset [15]. In this section, we will first introduce our implementation details, and then report the performance of our method on above two datasets.

4.2. Training details

Embedding module training: As aforementioned, the cleaned MS-Celeb-1M dataset [10, 13] which contains about 3.8M images of 85k unique identities is used to train our feature embedding network for the single image face recognition task. MTCNN [43] is employed to detect 5 facial landmarks in the face images. The faces are aligned to 112×112 by using similarity transformation according to the landmarks detected, and then fed into embedding network for training. The Additive Angular Margin Loss [10], which is a kind of modified softmax loss is used to supervise the training. After training, the classification loss layer

is removed from the trained network. The rest network is fixed and used to extract a single fixed-size representation for the face image.

End-to-End training: We use the VGGFace2 dataset [5] to train the whole network end-to-end for the set-based face recognition task. The VGGFace2 Dataset [5] consists of about 3 million images, covering 8631 identities, and there are on average 360 face images for each identity. To perform set-based face recognition training, the image sets are built by repeatedly sampling a fixed number of images which belong to the same identity. All the images sampled are aligned using the same way as in the embedding module training. After alignment, the data augmentation is performed by image degradation. Following the same strategy as in GhostVLAD [44], four methods: isotropic blur, motion blur, decreased resolution and JPEG compression are adopted to degrade the face image for training. The Additive Angular Margin Loss [10] is also adopted to supervise the end-to-end training. In order to speed up the training, we initialize all the parameters of the aggregation module to be zero. That means the aggregation module begins with average pooling to search the optimal parameters.

Finetuning: All the video face dataset are also aligned by using MTCNN [43] algorithm and similarity transformation. Then the whole network is finetuned on the training set of each video face dataset using the Additive Angular Margin Loss [10].

4.3. Baseline methods

Since average pooling is a widely used aggregation method in many previous works [18, 25, 32, 8], we choose average pooling as one of our baselines. For fairness, the average pooling method shares common embedding module with our method after the whole network is finetuned on each benchmark dataset. We also choose NAN [42] as our

Method	Accuracy(%)	AUC
EigenPEP[18]	84.80 \pm 1.40	92.60
DeepFace-single[34]	91.40 \pm 1.1	96.30
DeepID2+[33]	93.20 \pm 0.20	92.30
FaceNet[31]	95.12 \pm 0.39	92.30
Wen et al.[38]	94.90	92.30
TBE-CNN[11]	94.96 \pm 0.31	92.30
NAN[42]	95.72 \pm 0.64	98.80
ADRL[29]	96.52 \pm 0.54	-
Deep FR[25]	97.30	92.30
AvgPool	95.70 \pm 0.61	98.69
NAN*	95.93 \pm 0.62	98.92
Ours	96.21 \pm 0.63	99.1

Table 1. Performace evaluation on YTF benchmark. (NAN* represents the NAN [42] method we reproduce with our embedding module.)

another baseline. We reproduce the NAN which consists of cascaded two attention blocks as [42] describes. The reproduced NAN is trained in the same way as our method. The two baselines as well as our method produce 512-d feature representation for each video and compute the similarity in $O(1)$ time. Besides the above two baselines, we also compare with some other state-of-the-art methods.

4.4. Results on YouTube Face Dataset

We first evaluate our method on the YouTube Face Dataset [40] which contains 3425 videos of 2595 different subjects. The lengths of videos vary from 48 to 6070 frames and the average length is 181.3 frames per video. The dataset is splitted into 10 folds, and each fold consists of 250 positive (intra-subject) pairs and 250 negative (inter-subject) pairs. We follow the standard verification protocol to test our method.

Table 1 shows the results of our method, the baseline and some other state of the art methods. We can see that our method outperforms the two baselines, reducing the error of the best-performing baseline: NAN* by 6.88%. This can be regarded as a proof of the effectiveness of our method. Our method also performs better than all the other state-of-the-art methods (including the original NAN [42]) except the deep FR methods and ADRL method [29]. The reason is that, the deep FR method benefits a lot from front face selection and triplet loss embedding with carefully selected triplets, and ADRL method [29] benefits from exploiting the temporal information from the video sequence. Compared to deep FR method, our aggregation method is more straightforward and elegant without hand-crafted rules. And compared to ADRL method [29], our method is order-invariant, can be used in more potential scenarios. It is noteworthy that our reproduced NAN also performs better than original NAN [42]. That is because both the embedding module and aggregation module of the reproduced NAN is trained end-to-end instead of separately, and compared to separate training, more training data is used during the end-to-end training stage.

4.5. Results on IJB-A Dataset

The IJB-A Dataset [15] contains 5712 images and 2085 videos, covering 500 subjects in total. The average numbers of images and videos per subject are 11.4 images and 4.2 videos. This dataset is more challenging than YouTube Face dataset [40] due to it covers large range of pose variations and different kinds of image conditions.

We follow the standard benchmark procedure for IJB-A to evaluate our method on both the ‘compare’ protocol for 1 : 1 face verification and the ‘search’ protocol for 1 : N face identification. The true accept rates (TAR) vs. false positive rates (FAR) are reported for verification, while the true positive identification rates (TPIR) vs. false positive

identification rates (FPIR) and the Rank-N accuracies are reported for identification. Table 2 and Table 3 show the evaluation results of different methods for verification task and identification task respectively.

From above two tables, we can see that our method outperforms the two baselines by appreciable margins in both verification task and identification task, especially reducing the error of best-performing baseline by 7.12%, 11.97% and 17.83% at FAR=0.001, FAR=0.01 and FAR=0.1 respectively in verification task. These solidly prove the effectiveness of our method. Compared to all the state-of-the-art methods except for Ranjan *et al.* [26], our method performs a little better at FAR=0.001 and FAR=0.01, performs on par with them at FAR=0.01 where TAR values have almost saturated to a 99% mark, and beats all of them except on Rank-10 metric where our method is on par with them and the TPIR values have saturated to a 99.4% mark. Though Ranjan *et al.* [26] performed better than our method in three of all the eight metrics, they fused two deeper networks for feature embedding. One is ResNet101, the other is Inception ResNet-v2 that has 244 convolution layers, both of which are much deeper than our backbone ResNet50. What is more, they used more diverse datasets than us to train the feature embedding module. Not only the still image dataset but also other additional video dataset which is beneficial to video face recognition were utilized to train the networks. Besides, the reproduced NAN also outperforms the original NAN [42] as on YTF benchmark. It is noteworthy that the gap between our method and the original NAN [42] on IJB-A dataset is larger compared to the results on YTF dataset. This is because the face variations in IJB-A dataset is much larger than in YTF dataset, our method can extract more beneficial information for video face recognition.

5. Conclusion

We introduced a new feature aggregation network for video face recognition. Our network can adaptively and fine-grained weigh the input frames along each dimension of the feature vector and fuse them organically into a compact representation which is invariant to the frame order. Our aggregation scheme can make the best to exploit any valuable part of the features regardless of the frame quality to promote the performance of video face recognition. Experiments on YTF and IJB-A benchmarks show our method is a competitive aggregation method.

References

- [1] O. Arandjelovic, G. Shakhnarovich, J. Fisher, R. Cipolla, and T. Darrell. Face recognition with image sets using manifold density divergence. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 581–588. IEEE, 2005. 1, 2

Method	1:1 verification TAR(%)		
	FAR=0.001	FAR=0.01	FAR=0.1
DREAM[4]	86.8 ± 1.5	94.4 ± 0.9	-
Triplet Embedding[30]	81.3 ± 2	91 ± 1	96.4 ± 0.5
Template Adaptation[9]	83.6 ± 2.7	93.9 ± 1.3	97.9 ± 0.4
NAN[42]	88.1 ± 1.1	94.1 ± 0.8	97.8 ± 0.3
QAN[21]	89.31 ± 3.92	94.20 ± 1.53	98.02 ± 0.55
VGGFace2[5]	92.1 ± 1.4	96.8 ± 0.6	99.0 ± 0.2
GhostVLAD[44]	93.5 ± 1.5	97.2 ± 0.3	99.0 ± 0.2
Ranjan <i>et al.</i> [26]	95.2	96.9	98.4
AvgPool	88.82 ± 1.22	96.18 ± 0.92	98.16 ± 0.40
NAN*	93.12 ± 1.16	96.91 ± 0.83	98.71 ± 0.599
Ours	93.61 ± 1.51	97.28 ± 0.28	98.94 ± 0.31

Table 2. Performance evaluation for verification on IJB-A benchmark. The true accept rates (TAR) vs. false positive rates (FAR) are reported. (NAN* represents the NAN [42] method we reproduce with our embedding module.)

Method	1:N identification TPIR(%)				
	FPIR=0.01	FPIR=0.1	Rank-1	Rank-5	Rank-10
DREAM[4]	-	-	94.6 ± 1.1	96.8 ± 1.0	-
Triplet Embedding[30]	75.3 ± 3	86.3 ± 1.4	93.2 ± 1	-	97.7 ± 0.5
Template Adaptation[9]	77.4 ± 4.9	88.2 ± 1.6	92.8 ± 1.0	97.7 ± 0.4	98.6 ± 0.3
NAN[42]	81.7 ± 4.1	91.9 ± 0.9	95.8 ± 0.5	98.0 ± 0.5	98.6 ± 0.3
VGGFace2[5]	88.3 ± 3.8	94.6 ± 0.4	98.2 ± 0.4	99.3 ± 0.2	99.4 ± 0.1
GhostVLAD[44]	88.4 ± 5.9	95.1 ± 0.5	97.7 ± 0.4	99.1 ± 0.3	99.4 ± 0.2
Ranjan <i>et al.</i> [26]	92.0	96.2	97.5	98.6	98.9
AvgPool	86.43 ± 4.81	94.05 ± 1.02	95.69 ± 0.62	98.52 ± 0.45	99.04 ± 0.33
NAN*	87.92 ± 5.44	94.83 ± 1.01	97.23 ± 0.57	99.05 ± 0.58	99.24 ± 0.44
Ours	88.51 ± 5.86	95.18 ± 1.02	97.92 ± 0.32	99.23 ± 0.36	99.39 ± 0.25

Table 3. Performance evaluation for identification on IJB-A benchmark. The true positive identification rate (TPIR) vs. false positive identification rate (FPIR) and the Rank-N accuracies are presented. (NAN* represents the NAN [42] method we reproduce with our embedding module.)

- [2] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic. Netvlad: Cnn architecture for weakly supervised place recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5297–5307, 2016. 3
- [3] R. Arandjelovic and A. Zisserman. Three things everyone should know to improve object retrieval. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2911–2918. IEEE, 2012. 2
- [4] K. Cao, Y. Rong, C. Li, X. Tang, and C. Change Loy. Pose-robust face recognition via deep residual equivariant mapping. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5187–5196, 2018. 7
- [5] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, and A. Zisserman. Vggface2: A dataset for recognising faces across pose and age. In *2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018)*, pages 67–74. IEEE, 2018. 5, 7
- [6] H. Cevikalp and B. Triggs. Face recognition based on image sets. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 2567–2573. IEEE, 2010. 2
- [7] J.-C. Chen, R. Ranjan, A. Kumar, C.-H. Chen, V. M. Patel, and R. Chellappa. An end-to-end system for unconstrained face verification with deep convolutional neural networks. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 118–126, 2015. 1, 3
- [8] A. R. Chowdhury, T.-Y. Lin, S. Maji, and E. Learned-Miller. One-to-many face recognition with bilinear cnns. In *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1–9. IEEE, 2016. 1, 3, 5
- [9] N. Crosswhite, J. Byrne, C. Stauffer, O. Parkhi, Q. Cao, and A. Zisserman. Template adaptation for face verification and identification. *Image and Vision Computing*, 79:35–48, 2018. 7
- [10] J. Deng, J. Guo, and S. Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. *arXiv preprint arXiv:1801.07698*, 2018. 3, 4, 5
- [11] C. Ding and D. Tao. Trunk-branch ensemble convolutional neural networks for video-based face recognition. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):1002–1014, 2018. 1, 3, 5
- [12] G. Goswami, M. Vatsa, and R. Singh. Face verification via learned representation on feature-rich video frames. *IEEE Transactions on Information Forensics and Security*, 12(7):1686–1698, 2017. 1, 3

- [13] Y. Guo, L. Zhang, Y. Hu, X. He, and J. Gao. Ms-celeb-1m: A dataset and benchmark for large-scale face recognition. In *European Conference on Computer Vision*, pages 87–102. Springer, 2016. 5
- [14] T.-K. Kim, O. Arandjelović, and R. Cipolla. Boosted manifold principal angles for image set-based recognition. *Pattern Recognition*, 40(9):2475–2484, 2007. 2
- [15] 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: larpa janus benchmark a. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1931–1939, 2015. 5, 6
- [16] K.-C. Lee, J. Ho, M.-H. Yang, and D. Kriegman. Video-based face recognition using probabilistic appearance manifolds. In *Computer vision and pattern recognition, 2003. proceedings. 2003 IEEE computer society conference on*, volume 1, pages I–I. IEEE, 2003. 2
- [17] H. Li, G. Hua, Z. Lin, J. Brandt, and J. Yang. Probabilistic elastic matching for pose variant face verification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3499–3506, 2013. 2
- [18] H. Li, G. Hua, X. Shen, Z. Lin, and J. Brandt. Eigen-pep for video face recognition. In *2014 Asian Conference on Computer Vision (ACCV)*, pages 17–33, 2014. 1, 2, 5
- [19] L. Liu, L. Zhang, H. Liu, and S. Yan. Toward large-population face identification in unconstrained videos. *IEEE Transactions on Circuits and Systems for Video Technology*, 24(11):1874–1884, 2014. 1
- [20] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song. Sphereface: Deep hypersphere embedding for face recognition. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, volume 1, page 1, 2017. 3
- [21] Y. Liu, J. Yan, and W. Ouyang. Quality aware network for set to set recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5790–5799, 2017. 7
- [22] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004. 2
- [23] O. M. Parkhi, K. Simonyan, A. Vedaldi, and A. Zisserman. A compact and discriminative face track descriptor. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1693–1700, 2014. 1
- [24] O. M. Parkhi, K. Simonyan, A. Vedaldi, and A. Zisserman. A compact and discriminative face track descriptor. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1693–1700, 2014. 2
- [25] O. M. Parkhi, A. Vedaldi, A. Zisserman, et al. Deep face recognition. In *BMVC*, volume 1, page 6, 2015. 1, 3, 5
- [26] R. Ranjan, A. Bansal, J. Zheng, H. Xu, J. Gleason, B. Lu, A. Nanduri, J.-C. Chen, C. D. Castillo, and R. Chellappa. A fast and accurate system for face detection, identification, and verification. *IEEE Transactions on Biometrics, Behavior, and Identity Science*, 1(2):82–96, 2019. 6, 7
- [27] Y. Rao, J. Lin, J. Lu, and J. Zhou. Learning discriminative aggregation network for video-based face recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3781–3790, 2017. 1, 3
- [28] Y. Rao, J. Lu, and J. Zhou. Attention-aware deep reinforcement learning for video face recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3931–3940, 2017. 1
- [29] Y. Rao, J. Lu, and J. Zhou. Attention-aware deep reinforcement learning for video face recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3931–3940, 2017. 5, 6
- [30] S. Sankaranarayanan, A. Alavi, C. D. Castillo, and R. Chellappa. Triplet probabilistic embedding for face verification and clustering. In *2016 IEEE 8th international conference on biometrics theory, applications and systems (BTAS)*, pages 1–8. IEEE, 2016. 7
- [31] F. Schroff, D. Kalenichenko, and J. Philbin. Facenet: A unified embedding for face recognition and clustering. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 815–823, June 2015. 1, 3, 5
- [32] K. Sohn, S. Liu, G. Zhong, X. Yu, M.-H. Yang, and M. Chandraker. Unsupervised domain adaptation for face recognition in unlabeled videos. In *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, pages 3210–3218, 2017. 1, 5
- [33] Y. Sun, X. Wang, and X. Tang. Deeply learned face representations are sparse, selective, and robust. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2892–2900, 2015. 5
- [34] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1701–1708, 2014. 1, 3, 5
- [35] P. Turaga, A. Veeraraghavan, A. Srivastava, and R. Chellappa. Statistical computations on grassmann and stiefel manifolds for image and video-based recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(11):2273–2286, 2011. 2
- [36] R. Wang, S. Shan, X. Chen, and W. Gao. Manifold-manifold distance with application to face recognition based on image set. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008. 2
- [37] W. Wang, R. Wang, Z. Huang, S. Shan, and X. Chen. Discriminant analysis on riemannian manifold of gaussian distributions for face recognition with image sets. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2048–2057, 2015. 2
- [38] Y. Wen, K. Zhang, Z. Li, and Y. Qiao. A discriminative feature learning approach for deep face recognition. In *European Conference on Computer Vision*, pages 499–515. Springer, 2016. 1, 3, 5
- [39] L. Wolf, T. Hassner, and I. Maoz. Face recognition in unconstrained videos with matched background similarity. In *2011 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 529–534, June 2011. 1
- [40] L. Wolf, T. Hassner, and I. Maoz. Face recognition in unconstrained videos with matched background similarity. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 529–534. IEEE, 2011. 5, 6

- [41] L. Wolf and N. Levy. The svm-minus similarity score for video face recognition. In *2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3523–3530, 2013. [1](#)
- [42] J. Yang, P. Ren, D. Zhang, D. Chen, F. Wen, H. Li, and G. Hua. Neural aggregation network for video face recognition. In *CVPR*, volume 4, page 7, 2017. [1](#), [2](#), [3](#), [4](#), [5](#), [6](#), [7](#)
- [43] 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):1499–1503, 2016. [5](#)
- [44] Y. Zhong, R. Arandjelović, and A. Zisserman. Ghostvlad for set-based face recognition. *arXiv preprint arXiv:1810.09951*, 2018. [1](#), [3](#), [5](#), [7](#)