Ou, Zhonghong; Hu, Yucheng; Song, Meina; Yan, Zheng; Pan, Hui

**Redundancy Removing Aggregation Network with Distance Calibration for Video Face Recognition**

*Published in:*
IEEE Internet of Things Journal

**DOI:**
10.1109/JIOT.2020.3038776

Published: 01/05/2021

*Document Version*
Peer reviewed version

*Please cite the original version:*
Redundancy Removing Aggregation Network with Distance Calibration for Video Face Recognition

Zhonghong Ou, Member, IEEE, Yucheng Hu, Meina Song, Yan Zheng, Senior Member, IEEE, and Hui Pan, Fellow, IEEE

Abstract—Attention-based techniques have been successfully used for rating image quality, and have been widely employed for set-based face recognition. Nevertheless, for video face recognition, where the base Convolutional Neural Network (CNN) trained on large-scale data already provides discriminative features, fusing features with only predicted quality scores to generate representation is likely to cause duplicate sample dominant problem, and degrade performance correspondingly. To resolve the problem mentioned above, we propose a Redundancy Removing Aggregation Network (RRAN) for video face recognition. Compared with other quality-aware aggregation schemes, RRAN can take advantage of similarity information to tackle the noise introduced by redundant video frames. By leveraging metric learning, RRAN introduces a distance calibration scheme to align distance distributions of negative pairs of different video representations, which improves the accuracy under a uniform threshold. A series of experiments are conducted on multiple realistic datasets to evaluate the performance of RRAN, including YouTube Faces, IJB-A, and IJB-C. In comprehensive experiments we demonstrate that our method can diminish the overall influence of poor quality components with large proportion in the video and further improve the overall recognition performance with individual difference. Specifically, RRAN achieves a 96.84% accuracy on YouTube Face, outperforming all existing aggregation schemes.

Index Terms—Convolutional Neural Networks, Feature Aggregation, Metric Learning, Video-based Face Recognition.

I. INTRODUCTION

In recent years, an ever-increasing number of Internet-of-Things (IoT) devices have been deployed in various scenarios. Amongst them, as ubiquitous devices, video cameras have become an indispensable part of IoT scenarios and provide an effective approach for data acquisition [1]. Embedded in various IoT devices, video-based face recognition plays a fundamental role in various practical scenarios, including video surveillance [2], personal identification recognition [3], and access control [4], just to name a few. Although face recognition for static face images has been largely resolved with the advancement of deep learning-based techniques, video-based face recognition still poses a big challenge because of the unconstrained scenarios, and poor imaging quality, etc. A video consists of a series of frames of the same subject, how to match them with a still image or another video is a challenging metric learning problem.

Track merge is the most common and effective approach at present, which refers to integrating information across different frames. It achieves performance improvement with a simple design and minimal memory storage. A number of schemes have been proposed in this direction [5]–[11]. Besides common max pooling [5] and average pooling [5], [6], certain attention-based aggregation schemes have been proposed [7]–[11]. They mainly focus on finding discriminative frames and complementary information of low quality frames to make face video representation have bigger inter-class distance than intra-class distance. Nevertheless, they grade features or each component (dimension) of a feature separately, and then normalize the scores as weight to sum the original features. This approach ignores a fundamental fact, i.e., video as an ordered set has temporal context information and redundant frames. As shown in Fig. 1 (a), the weight distribution of three types of faces with different poses and expressions changes because of the different length of corresponding video clips. Regardless of the quality of redundant frames, it affects performance of the feature aggregation method, which is referred to as duplicate sample dominant problem.

Face recognition includes two types of protocols, i.e., identity verification and subject identification. Identity verification determines whether the person is the one it claims, while subject identification determines who the person is. For identity verification, it calculates the distance of a pair of face descriptions (usually in Euclidean distance or cosine distance) after track merge, and then compare the distance with the threshold value to determine whether they are the same person. For subject identification with open-set, every probe video needs to calculate the minimum distance from the gallery dataset, and then compare with the threshold value to determine who the person is.

Regardless of the task, the threshold has been designed to be uniform for all subjects in the past. In practical application scenarios, nevertheless, it can be observed that different individuals usually demonstrate different degrees of intra-variations, which are relevant to age, gender, and race, etc. For example, children are usually more challenging to identify than adults, and certain specific races have higher probability of becoming false positive under a uniform threshold. As an example illustrated in Fig. 1 (b), we select four subjects (individuals) from the MS-Celeb-1M dataset [12] randomly and visualize their corresponding face features with dimensionality reduction.
Experimental results on the YouTube Faces dataset (YTF) [13] is designed for recognition problem with identity specificity. Without additional training and supervision information, which during testing, distance calibration is based on statistical data compared with other existing studies. As a pluggable module acting on face set is the primary difference of our scheme and model complexity. The redundancy removing mechanism in the case of high redundancy with same storage overhead gation schemes, RRAN has good accuracy improvement even carefully chosen metric space. Unlike existing feature aggregation schemes, RRAN aggregates and calibrate the distance between other representations. By measure the statistic specificity of video face representation based on the distance distribution with other faces embedding, measure the statistic specificity of video face representation and calibrate the distance between other representations. By combining discrimination and redundancy, RRAN aggregates useful information of an input video into one vector under a uniformly threshold. A video with 51 frames is split into three subsequences of frames according to face poses. The summation of the weight percentage (red numbers) is given above each subsequence. On the other hand, we select a representative frame (gray block) from each subsequence and show them in the same way. Redundancy in video may easily distract the weight in aggregation. (b) Four individual’s intra-distances. Different color stands for different individual.

Fig. 1. Some factors affecting the performance of video face recognition. (a) Weight distribution of quality-aware pooling [8] on a video sequence and three representative frames respectively. A video with 51 frames is split into three subsequences of frames according to face poses. The summation of the weight percentage (red numbers) is given above each subsequence. On the other hand, we select a representative frame (gray block) from each subsequence and show them in the same way. Redundancy in video may easily distract the weight in aggregation. (b) Four individual’s intra-distances. Different color stands for different individual.

From the figure, we can clearly see that different individual has different intra-distances, which lead to different false acceptance rate for each individual. Thus, it is more logical to set different thresholds for different pairs. From the perspective of practical scenarios, it is equivalent to adding different bias for each distance under a uniform threshold.

In this paper, we propose a Redundancy Removing Aggregation Network (RRAN) with post-processing of distance calibration for video-based face recognition. Like most settings, video face recognition can be regarded as a template matching problem, wherein template is a upper-level concept of video, composed of a set of face images from the same subject. The whole pipeline of RRAN consists of three steps: (i) acquire features of each frame from a base Convolutional Neural Network (CNN) for static image face recognition; (ii) acquire weight scores of features from an aggregation scheme, and turn the weighted sum of the features into a single vector to represent the face in a video; (iii) based on the distance distribution with other faces embedding, measure the statistic specificity of video face representation and calibrate the distance between other representations. By combining discrimination and redundancy, RRAN aggregates useful information of an input video into one vector under a carefully chosen metric space. Unlike existing feature aggregation schemes, RRAN has good accuracy improvement even in the case of high redundancy with same storage overhead and model complexity. The redundancy removing mechanism acting on face set is the primary difference of our scheme compared with other existing studies. As a pluggable module during testing, distance calibration is based on statistical data without additional training and supervision information, which is designed for recognition problem with identity specificity. Experimental results on the YouTube Faces dataset (YTF) [13] and other template/video matching benchmarks demonstrate that RRAN boosts performance significantly compared with other feature aggregation methods.

The primary contributions are summarized as follows: (1) We propose a Redundancy Removing Aggregation Network (RRAN) to utilize discrimination and redundancy effectively, and provides an anatomy of the contributing factors. It is noteworthy that RRAN is a generic feature aggregation scheme applicable for other computer vision tasks. (2) As an independent and aggregate component, we propose an identity sensitive distance calibration scheme. It can further improve recognition performance without introducing additional architectural unit. (3) We conduct a comprehensive set of experiments to verify the effectiveness of RRAN. Experimental results demonstrate that RRAN achieves the best performance on face recognition benchmark YouTube Faces [13].

The rest of the paper is structured as follows. We introduce the existing video face recognition schemes in Section II, then describe the proposed RRAN video face recognition architecture in Section III. We compare the RRAN to other state-of-the-art schemes and present experimental results and analysis in Section IV and draw conclusions in Section V.

II. RELATED WORK

A. Video Face Recognition

Existing video face recognition schemes can be roughly categorized into two classes: space-based and aggregation-based. Aggregation-based schemes can be further divided into image-aggregation-based and feature-aggregation-based schemes.

Space-based Schemes: For space-based schemes, each video is usually modeled as a set of images that obeys specific constraints. After learning a lower dimensional subspace, the similarity or distance between videos is computed by the properties of image sets. In previous studies, a number of models have been proposed for manifold modeling, including affine hull [14], Symmetric Positive Definite (SPD) manifolds [15], [16], Grassmann manifolds [17], and n-order statistics [18], [19]. In these between-distribution similarities or distance measures, video frames are considered equally important. Thus, these models are easy to lose the discriminative information when the number of low quality image frames increases.

Image-aggregation-based Schemes: Deep CNNs have proven to be capable of producing compact and highly discriminative face representations from a single image. Thus, fusing a set of images to a single image is promising. By combining metric learning and adversarial learning, Rao et al. [20] proposed a Generative Adversarial Networks (GAN)-like aggregation network. It reconstructs a single image as output by average pooling the video frames to supervise the aggregation training. Similar practices include [21], [22], which aggregate each frame adaptively to a few representations through attention mechanism and improved loss function. In order to solve the out-of-focus blur and motion blur in video face, some schemes [23]–[25] try to employ super-resolution technique to tackle the problem. For example, Ustinova et al. [25] proposed a neural network architecture to restore...
Fig. 2. Overall architecture of the proposed RRAN. For each frame in the video, the base CNN model generates a feature vector as face descriptor. The architectural unit of feature aggregation is attached to the last convolutional layer of the base CNN model, which learns the quality score and predicts the weight for each single feature component. After pooling features into a compact representation, distance calibration post-treatment calibrates the distance between video representations.

the central frame of each input sequence. In addition to aggregating a video into a single 2D image, some schemes also leverage 3D modeling using a large range of pose variations. Mokhayeri et al. [26] proposed a domain-specific face synthesizing technique to reconstruct the gallery dataset and project it under different illumination. For the schemes mentioned above, the fundamental challenge is to generate discriminative images from video clips. Generative network with strong generalization ability is necessary, but it introduces additional computation and causes instability.

**Feature-aggregation-based Schemes:** Compared to image-aggregation-based schemes, feature-aggregation-based schemes are more efficient in terms of storage, and more effective in terms of precision. GhostVLAD [27] introduces the clustering of face descriptors in aggregation to improve the weight of high quality images. NAN [8] adopts an attention scheme to weigh the frames adaptively. Nevertheless, NAN considers that each component of the feature vector has equal importance. Similar practices include [9], [28]. C-FAN [10] leverages a kernel matrix to weigh the features, and makes better use of the components with more identity information rather than enhancing or weakening the whole feature vector. Some other studies [11], [29] leverage temporal-based models, e.g., Long Short-Term Memory (LSTM), to improve the attention module. By doing so, it can take advantage of context information to mitigate the noise introduced by redundant video frames.

In this work, we utilize redundant information in a novel way. Independent of any temporal-based modules, RRAN introduces a similarity constraint to calculate the weight for aggregation without any additional parameter. Namely, RRAN can make use of general still face images as the training set.

**B. Deep Metric Learning**

Many metric learning algorithms have been successfully applied to improve recognition performance [30]–[33]. Deep metric learning [30] aims to produce discriminative descriptors through combination of deep learning and metric learning. For example, Guillaumin et al. [31] employed a nearest neighbour approach to compute the probability for two images. Schroff et al. [32] presented a triplet loss function to directly optimize the feature embedding in metric learning. Some other loss functions [33] are devoted to decreasing intra-class distance further and increasing inter-class distance with additive margin.

Complementary to these schemes, we propose a metric space alignment method to calibrate the distance of features for comparison under a uniform threshold. The solution to recognition problem is usually to separate positive pairs from negative pairs of all subjects by a threshold value. The distance calibration scheme we propose is the first one to estimate and eliminate the discrepancy in the distance distribution of negative pairs of different subjects, which directly affects or improves the classification accuracy under the optimal threshold.

### III. Methodology

**A. Overall Architecture of RRAN**

The overall architecture of the proposed RRAN is presented in Fig. 2. RRAN incorporates a base CNN model to extract face features, feature aggregation to aggregate deep feature vectors, and distance calibration post-treatment with no training. The training of the whole model is divided into two stages (in Algorithm 1): (i) train the base CNN model on a large scale face dataset (MS-Celeb-1M [12]) using classification loss; (ii) remove the classification layer, fix the base CNN model, and train the architectural unit of feature aggregation on the same dataset. For the distance calibration post-treatment, we need to use a dataset to estimate the statistical distribution and calibrate the distance, which is called calibration set. The calibration set can be an unlabeled face set, or a face dataset without identity intersection with benchmarks. The distance calibration post-treatment tries to align boundaries of positive pairs and negative pairs of every template representation. By
Algorithm 1 Pseudocode of RRAN in a PyTorch-like style.

```python
# train the base CNN model with K1 epochs
for x, y in loader_stage1:
    # x: data, y: labels
    # N*D, N: batch size, D: feature dimension
    features = BACKBONE(x)
    # implement of ArcFace
    outputs = HEAD(features, y)
    loss = nn.CrossEntropyLoss(outputs, y)
    # SGD update: the base CNN model
    loss.backward()
    optimizer_stage1.step()

# train the aggregation module with K2 epochs
for t1, t2 in loader_stage2:
    # t1 and t2 include different templates
    # of same subject subset of size M
    f1 = MODEL(t1)  # M*D
    f2 = MODEL(t2)  # M*D
    loss = NPairLoss(f1, f2, MARGIN)
    # SGD update: the aggregation module
    loss.backward()
    optimizer_stage2.step()
```

doing that, it can improve the accuracy of recognition under the uniform threshold.

B. Feature Aggregation

We introduce feature aggregation in RRAN to adaptively aggregate deep feature vectors into a single vector to represent the face in a video or template. The core of feature aggregation is to adaptively learn the weight to each feature component considering discrimination and redundancy. Let \( V = \{ I_1, I_2, ..., I_N \} \) denote a template of face images where \( N \) is the template size, not a fixed number. Inspired by CFAN [10], we use the last feature map instead of feature vector as the input of the feature aggregation because it provides more information. The base CNN model includes deep convolutional layers \( H(\cdot) \) and feature extraction layer which \( L_2 \)-normalizes the output \( F(\cdot) \). The last feature map of the \( i \)-th image in the template is obtained as follows:

\[
H(I_i) = h_i. \tag{1}
\]

The face feature vector is then extracted by the following equation:

\[
F(h_i) = f_i. \tag{2}
\]

The corresponding quality vector is obtained by:

\[
Q(h_i) = q_i. \tag{3}
\]

Wherein \( Q(\cdot) \) is the quality extraction layer in feature aggregation as a second branch besides the feature extraction layer. Note that \( f_i \) and \( q_i \) have the same dimension \( D \). In order to keep the scale of feature vector at a constant level after aggregation, softmax operator is used for normalization. Formally, given a set of quality vectors \( \{q_1, q_2, ..., q_N\} \), the \( j \)-th component of the \( i \)-th quality vector is normalized by:

\[
\tilde{q}_{ij} = \frac{\exp(q_{ij})}{\sum_{k=1}^{N} \exp(q_{kj})}. \tag{4}
\]

On the other hand, the similarity matrix is computed as:

\[
S = FF^T = \begin{bmatrix}
    f_1^T f_1 & \cdots & f_1^T f_N \\
    \vdots & \ddots & \vdots \\
    f_N^T f_1 & \cdots & f_N^T f_N
\end{bmatrix}. \tag{5}
\]

Based on the similarity matrix, we define the difference coefficient of the \( i \)-th frame as follows:

\[
\tilde{s}_i = \frac{\exp(f_i^T f_i)}{\sum_{k=1}^{N} \exp(f_i^T f_k)}. \tag{6}
\]

Given a video, the more video frames with similar pose and illumination to a certain frame, the smaller the difference coefficient of this frame, greater on the contrary. \( \beta \) is the scaling factor. When image quality and redundancy are taken into consideration, a positive weight vector of the \( i \)-th frame is generated as follows:

\[
w_{ij} = \frac{\tilde{q}_{ij} \cdot \tilde{s}_i}{\sum_{k=1}^{N} \tilde{q}_{kj} \cdot \tilde{s}_k}. \tag{7}
\]

Wherein \( j \) denotes the linear weight of the \( j \)-th component of the feature vector contributing to the aggregation result, and \( \sum_{k=1}^{N} w_{kj} = 1, \forall j \in \{1, 2, ..., D\} \). The aggregated template representation is obtained by pooling the feature vectors using the weight vectors as follows:

\[
r = \sum_{i=1}^{N} f_i \odot w_i. \tag{8}
\]

Wherein \( \odot \) represents the operation of element-wise multiplication. \( r \) turns into \( \hat{r} \) for subsequent steps after \( L_2 \)-normalization.

C. Distance Calibration Post-treatment

Distance calibration post-treatment tries to utilize prepared dataset to estimate and reduce the bias in the distance distribution of negative pairs of different subjects and make distance distributions consistent. We use \( V_1 \) and \( V_2 \) to denote two videos of face images to measure similarity. After the processing of the base CNN model and the feature aggregation, two representations are generated, i.e., \( r_1 \) and \( r_2 \). \( L_2 \)-distance is used to compute the similarity. Given a calibrated set \( G \) and the video face \( V \), we define the bias function \( \sigma(\cdot) \) as follows:

\[
Pr[d(g, V) - \sigma(V) < 0, g \in G] = 10^{-4}. \tag{9}
\]

Wherein \( d(\cdot) \) is the squared Euclidean distance between representations of the two face sets. \( 10^{-4} \) denotes a fixed False
Acceptance Rate (FAR) which can be adjusted according to different protocols. We use the bias function to estimate the distance tendency with negative samples and calibrate distance as a reference. Distance calibration can make the positive and negative pairs of different subjects divided more accurately under a uniform threshold. Given the $10^{-4}$, the one-side calibrated distance is listed as follows:

$$d(V_1, V_2) = d(V_1, V_2) - \lambda \sigma(V_1).$$

(10)

Wherein $\lambda$ is defined as the coefficient of bias. If we have the gallery and probe calibration set, we can align the metric space with both of them:

$$d(V_1, V_2) = d(V_1, V_2) - \lambda_1 \sigma(V_1) - \lambda_2 \sigma(V_2).$$

(11)

If the calibration sets are the same, then $\lambda_1 = \lambda_2$.

D. Training RRAN

Given a pre-trained base CNN model, RRAN introduces feature aggregation to the original network to learn the weight for each deep feature component. The feature aggregation module includes only one fully connected layer and one scaling factor. During training, we only update the parameters of aggregation module, which means RRAN will not change this original feature, just learn the way to merge them. Unlike previous studies [11], [20], which take the video frames as the training set, we simulate the redundant situation by sampling static face images repeatedly since it contains cheaper and richer identity information.

To train the feature aggregation module, we leverage the N-pair loss [34]. It generalizes triplet loss by allowing joint comparison among multiple negative examples, and improves the utilization of samples in each batch as an efficient batch construction scheme. There are $2M$ templates of $M$ subjects that are constructed online in each mini-batch randomly. The template representations are obtained by the procedure described in Section III-B. Hard negative class mining is adopted, and the loss function is given by the following equation:

$$\mathcal{L}_{N-pair}(\{\tilde{r}, \tilde{r}_+\}^M_{i=1}) = \frac{1}{M} \sum_{i=1}^M \sum_{j \neq i} \left[ \alpha + d(\tilde{r}_i, \tilde{r}_i^+) - d(\tilde{r}_i, \tilde{r}_j) \right]_+$$

(12)

Wherein $M$ is the number of subjects, $\alpha$ is a margin parameter, $\tilde{r}_i^+$ is a positive template feature to $\tilde{r}_i$, and $\{\tilde{r}_i\}_{i=1}^{M-1}$ are negative. $[x]_+ = max\{0, x\}$.

IV. EXPERIMENTS

In this section, we present experimental evaluation of RRAN. We first describe the datasets and protocols in Section IV-A, we then present implementation details in Section IV-B and baselines in Section IV-C. We describe the concrete experimental results in the subsequent sections.

A. Benchmarks

We evaluate RRAN on three datasets, including one video face dataset, i.e., the YouTube Faces dataset [13], and two template-based face datasets, i.e., the IARPA Janus Benchmark A (IJB-A) [35] and the IARPA Janus Benchmark C (IJB-C) [36].

YTF: The YouTube Faces (YTF) dataset is a video face dataset released in 2011 [37]. It contains 3425 videos of 1595 different subjects, and the number of frames ranges from 48 to 6070. Compared with other video benchmarks, the media of YTF is more photojournalistic [37]. We follow the standard verification protocol with the given 5000 video pairs to test our method.

IJB-A: IJB-A [35] contains 25813 face images of 500 different subjects as a template-based benchmark. Each template includes a set of static images or video frames. Compared with YouTube Faces, the faces in IJB-A have larger variations and present a more unconstrained scenario. We follow the standard benchmark procedure for IJB-A, and evaluate on “1:1 face verification” and “1:N face identification”. Verification performance is assessed by measuring the trade-off between the True Accept Rates (TAR) vs. FAR. For 1:N identification, the performance is measured using the True Positive Identification Rate (TPIR) vs. False Positive Identification Rate (FPIR) and Rank-N.

IJB-C: IJB-C [36] is an extension of IJB-A with 140740 face images of 3531 different subjects. The verification protocol of IJB-C includes more impostor pairs. Thus, we can compute TAR at a lower FAR.

B. Implementation Details

Pre-processing: A facial landmark detector MTCNN [38] is employed to detect five facial landmarks in the face images for training and testing. The faces are aligned to 112×112 pixels by using a similarity transformation according to the detected landmarks. If detection fails, we use the default landmarks.

Training: A base CNN model is firstly trained on a cleaned version1 of MS-Celeb-1M dataset [12]. It contains approximately 3.8 M images of 85 K unique identities to learn face recognition for static face images and produce 512-d feature. We adopt the widely used ArcFace configuration [33] including IR-50 CNN architectures (an improved version of the vanilla ResNet-50 [39]) and Additive Angular Margin Loss. Then we train the feature aggregation module using the N-pair loss with the margin parameter $\alpha = 3.0$ on the same training dataset. We use the stochastic gradient descent optimizer with learning rate 0.001 and set the scaling factor $\beta = 0.5$. Each mini-batch samples 800 images (after augment) of 80 templates from 40 random subjects. The 10 images in each template are resampled from 5 images randomly. Data augmentation is randomly performed by image degradation including Gaussian blur and motion blur. The aggregation module is trained for 8000 steps in all. In order to make the hyper-parameters setting in the network better match the redundancy in real scenes, we sample 500 subjects from

1https://github.com/ZhaoJ9014/face.evoLVe.PyTorch
the IJB-C dataset randomly as the validation set to adjust the hyper-parameters. For the stability of training, all the parameters of the feature aggregation are initialized to zero, which means that RRAN begins with the same quality score to search for optimal parameters. Unlike previous work [7], after training on MS-Celeb-1M, RRAN is applied to the testing datasets without other datasets for fine-tuning. Two Nvidia Geforce GTX 1080 Ti GPU are used for acceleration.

Distance calibration: Out of a trade-off between computation and performance, we sample 30000 face images of 3000 subjects from MS-Celeb-1M [12] as a calibration set. The trained network removes the aggregation module and produces a feature for each frame. The features are stored to calculate the distance from each video representation and then sorted for distance calibration. The requirement in the protocol or the FAR corresponding to the threshold determines the value of $10^{-5}$. We set $\lambda_1 = \lambda_2 = 0.1$.

C. Baselines

We compare the proposed RRAN with three other schemes:

AvgPool: Average pooling scheme is widely used as a common baseline in previous studies [5], [6]. To make a fair comparison, the average pooling scheme shares common feature extraction module (the base CNN model) with our scheme.

QualityPool: In order to verify the effect of the redundancy removing on feature aggregation, component-wise quality aggregation module without similarity calculation is trained similarly as C-FAN [10]. After feature aggregation, distance calibration is not applied.

RRAN− (RRAN without distance calibration): To observe the performance contribution made by the post-treatment module, we remove the distance calibration scheme from RRAN.

For all baselines, $L_2$-normalization is applied before and after aggregation. Besides the three baselines mentioned above, we also compare with some other state-of-the-art methods.

<table>
<thead>
<tr>
<th>Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>EigenPEP [6]</td>
</tr>
<tr>
<td>DeepID2+ [40]</td>
</tr>
<tr>
<td>FaceNet [32]</td>
</tr>
<tr>
<td>NAN [8]</td>
</tr>
<tr>
<td>DeepFace [41]</td>
</tr>
<tr>
<td>AvgPool</td>
</tr>
<tr>
<td>QualityPool</td>
</tr>
<tr>
<td>RRAN</td>
</tr>
<tr>
<td>RRAN</td>
</tr>
</tbody>
</table>

TABLE I: Verification Performance on YouTube Faces Benchmark

<table>
<thead>
<tr>
<th>Method</th>
<th>Accuracy(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>EigenPEP [6]</td>
<td>84.8 ± 1.4</td>
</tr>
<tr>
<td>DeepID2+ [40]</td>
<td>93.20 ± 0.20</td>
</tr>
<tr>
<td>FaceNet [32]</td>
<td>95.52 ± 0.06</td>
</tr>
<tr>
<td>NAN [8]</td>
<td>95.72 ± 0.64</td>
</tr>
<tr>
<td>DeepFace [41]</td>
<td>91.40 ± 1.10</td>
</tr>
<tr>
<td>AvgPool</td>
<td>96.28 ± 0.12</td>
</tr>
<tr>
<td>QualityPool</td>
<td>96.59 ± 0.24</td>
</tr>
<tr>
<td>RRAN</td>
<td>96.70 ± 0.24</td>
</tr>
</tbody>
</table>

D. Results on YouTube Faces Dataset

Table I demonstrates the comparison of RRAN with the baselines and some other existing methods. Although the accuracy on YouTube Faces benchmark is nearly saturated, our method still improves the performance of video representation. Specifically, the proposed RRAN reduces the error of the best performing baseline (RRAN−) by 4.24%, and performs better than all the other state-of-the-art algorithms. It is noteworthy that RRAN outperforms the state-of-the-art method C-FAN [10] by 9.71% and REAN [11] by 7.06%.

As a video face benchmark, the images are from consecutive frames containing faces in public videos, which have a large number of redundant frames. To evaluate the difference of the feature aggregation by RRAN, we visualize the weight distribution on two example video subsequences from YouTube Faces. As shown in Fig. 3, we sample 16 frames randomly from two original video on YouTube Faces dataset respectively to form two example video clips. Then two models, i.e., QualityPool and RRAN− (or RRAN), are used to compute the weight for the frames in the clips. The component-wise weight vector of each face frame is averaged into one scalar for better visualization. From the figure, we can see that QualityPool can evaluate the quality of the given image effectively, but introduces redundancy information. Correspondingly, it is vulnerable to redundant low-quality images. Namely, when the

\[ \frac{(1-y^2)}{(1-x^2)} \times 100\% (x \text{ and } y \text{ are the corresponding precision}) \]

2\text{The calculation of reduced error rate: } \frac{(1-x^2)}{(1-y^2)} \times 100\% (x \text{ and } y \text{ are the corresponding precision}).
correspondingly, they contain less redundant information than the YouTube Faces dataset. For this reason, the proposed RRAN− does not demonstrate obvious advantages over QualityPool. Nevertheless, it is noteworthy that there is a larger gap between RRAN and other aggregation schemes on IJB-A and IJB-C dataset because face variations, including pose, occlusion, and other factors, in IJB-A and IJB-C dataset are much larger than in the YouTube Faces dataset. Thus, the differences between facial features are likely to be more pronounced. As such, we can see that RRAN achieves a more significant improvement on the IJB-A and IJB-C benchmarks.

**E. Results on IJB-A and IJB-C Datasets**

Table II and Table III illustrate the results on two more challenging benchmarks, i.e., IJB-A and IJB-C. TAR vs. FAR are reported for verification, while the TPIR vs. FPIR, and the Rank-N accuracies are reported for identification.

From Table II, we can see that RRAN outperforms all previous superior video-based face recognition solutions on verification and open-set identification protocols on IJB-A dataset. It can be seen that RRAN reduces the error of the best-performing method [7] by 4.43% at 1% FPIR, and by 4.69% at 0.1% FAR, respectively. For IJB-C, we compute TAR at a lower FAR according to general practice [45]–[49]. The results in Table III show that RRAN can improve performance in most cases. In particular, at FAR=0.01%, RRAN outperforms PFE [49] by 0.94%. Based on the experimental results mentioned above, it can be seen that distance calibration scheme can improve the recognition performance robustly.

**TABLE II: Performance Evaluation for Face Verification and Identification on IJB-A Benchmark**

<table>
<thead>
<tr>
<th>Method</th>
<th>Verification TAR (%)</th>
<th>Identification (%)</th>
<th>Open-set Identification (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.1% FAR</td>
<td>1% FAR</td>
<td>Rank-1</td>
</tr>
<tr>
<td>Masi et al. [42]</td>
<td>72.5</td>
<td>88.6</td>
<td>90.6</td>
</tr>
<tr>
<td>Triplet Embedding [43]</td>
<td>81.3 ± 2.0</td>
<td>90.0 ± 1.0</td>
<td>93.2 ± 1.0</td>
</tr>
<tr>
<td>Shi et al. [44]</td>
<td>60.2 ± 6.9</td>
<td>82.3 ± 2.2</td>
<td>95.8 ± 0.5</td>
</tr>
<tr>
<td>NAM [8]</td>
<td>88.1 ± 1.1</td>
<td>94.1 ± 0.8</td>
<td>94.57 ± 0.82</td>
</tr>
<tr>
<td>C-FAN [10]</td>
<td>91.59 ± 0.99</td>
<td>93.97 ± 0.78</td>
<td>97.97 ± 0.32</td>
</tr>
<tr>
<td>C-FAN [10]</td>
<td>89.31 ± 3.92</td>
<td>94.20 ± 1.53</td>
<td>97.77 ± 0.4</td>
</tr>
<tr>
<td>GhostVLAD [27]</td>
<td>93.5 ± 1.5</td>
<td>97.2 ± 0.3</td>
<td>97.92 ± 0.32</td>
</tr>
<tr>
<td>Liu et al. [7]</td>
<td>93.61 ± 1.51</td>
<td>97.28 ± 0.28</td>
<td>97.92 ± 0.32</td>
</tr>
<tr>
<td>AvgPool</td>
<td>93.14 ± 0.12</td>
<td>95.99 ± 0.07</td>
<td>96.44 ± 0.12</td>
</tr>
<tr>
<td>QualityPool</td>
<td>93.78 ± 0.34</td>
<td>97.01 ± 0.11</td>
<td>97.62 ± 0.20</td>
</tr>
<tr>
<td>RRAN−</td>
<td>93.82 ± 0.34</td>
<td>97.20 ± 0.13</td>
<td>97.63 ± 0.20</td>
</tr>
<tr>
<td>RRAN</td>
<td>93.91 ± 0.56</td>
<td>98.01 ± 0.17</td>
<td>97.92 ± 0.33</td>
</tr>
</tbody>
</table>

Because the templates of IJB-A and IJB-C are not exactly continuous video frames, but a mixture of images and discontinuous video frames captured by cameras of different specifications. Correspondingly, they contain less redundant information than the YouTube Faces dataset. For this reason,

**TABLE III: Results of Our Method and State-of-the-art Methods on IJB-C Benchmark**

<table>
<thead>
<tr>
<th>Method</th>
<th>Verification TAR (%)</th>
<th>0.001% FAR</th>
<th>0.01% FAR</th>
<th>0.1% FAR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yin et al. [45]</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>69.3</td>
</tr>
<tr>
<td>Cao et al. [46]</td>
<td>74.7</td>
<td>84.0</td>
<td>91.0</td>
<td>91.0</td>
</tr>
<tr>
<td>Multicolumn [47]</td>
<td>77.1</td>
<td>86.2</td>
<td>92.7</td>
<td>92.7</td>
</tr>
<tr>
<td>DCN [48]</td>
<td>88.5</td>
<td>93.25</td>
<td>95.49</td>
<td>95.49</td>
</tr>
<tr>
<td>PFE [49]</td>
<td>89.64</td>
<td>93.25</td>
<td>95.49</td>
<td>95.49</td>
</tr>
<tr>
<td>AvgPool</td>
<td>85.03</td>
<td>91.96</td>
<td>95.42</td>
<td>95.42</td>
</tr>
<tr>
<td>QualityPool</td>
<td>86.68</td>
<td>93.23</td>
<td>96.07</td>
<td>96.07</td>
</tr>
<tr>
<td>RRAN−</td>
<td>87.10</td>
<td>93.45</td>
<td>96.24</td>
<td>96.24</td>
</tr>
<tr>
<td>RRAN</td>
<td>89.53</td>
<td>94.19</td>
<td>96.52</td>
<td>96.52</td>
</tr>
</tbody>
</table>

**Fig. 4. Distance distribution before and after distance calibration.**

(a) Distance distribution and fitted gaussian curve between the negative pairs formed by two different templates of IJB-C before distance calibration. The vertical reference lines represent the threshold values at FAR=10−3, and the bias is calculated using the calibrated set. (b) Distance distribution after distance calibration. Best viewed in color.

**F. Qualitative Analysis**

**Why does distance calibration improve performance?**

We select 23124 templates from the benchmark IJB-C [36],
and calculate the distance distribution between the negative pairs formed by two representative templates ($V_1$ and $V_2$) and the remaining templates. The results are presented in Fig. 4 (a). According to the experimental setup of distance calibration, we acquire $\sigma(V_1)$ and $\sigma(V_2)$ at $10^{-1} = 10^{-3}$. They have the same polarity relationship as the thresholds at FAR=$10^{-3}$. Considering $V_1$ and $V_2$ before distance calibration, respectively, there is a large bias between the two threshold values at FAR=$10^{-3}$. When the difference of distance distribution between different subject and probe faces is determined, if a uniform threshold is used to make a trade-off with the performance of $V_1$ and $V_2$, the bias directly affects the overall accuracy. After distance calibration, the bias between these thresholds decreases from 0.1 to 0.03, and the distance distributions are more consistent. The accuracy loss under the unified threshold is controlled to a certain extent. The results are illustrated in Fig. 4 (b). In other words, the distance calibration can align the metric space of different representations effectively, and improve the performance of the protocol under the uniform threshold.

V. Conclusion

We introduced a Redundancy Removing Aggregation Network (RRAN) for video face recognition in this paper. RRAN can weigh the input frames with redundant information adaptively in a fine-grained manner. It fuses the video frames into a compact embedding vector organically, which is invariant to the order of input frames. Moreover, we employed a distance calibration scheme to make the distances between positive and negative pairs more consistent for the representation. By doing this, it improves performance in comparison with uniform threshold. Extensive experiments demonstrate the effectiveness of RRAN which achieve competitive performance on three widely used datasets. In the future, we will explore how to address dependency of calibration set in distance calibration post-treatment, and learn RRAN in an end-to-end manner.

Acknowledgment

This work was supported by the National Key R&D Program of China (Grant No. 2020AAA01007500) and National Natural Science Foundation of China (Grant No. 62076035).
REFERENCES


ZHONGHONG OU received the Ph.D degree from the University of Oulu, Finland, in 2010. He has been an associate professor in the School of Computer Science, Beijing University of Posts and Telecommunications, China, since December 2015. From 2010 to 2015, he was a post-doc researcher with Aalto University, Finland. He has a wide spectrum of research interests, including big data analytics, image processing, and deep learning. He is a member of the IEEE.

YUCHENG HU was born in Jiangxi, China, in 1995. He received the B.S. degree from Qingdao University, China, in 2017. He is currently pursuing the M.S. degree in School of computer science, Beijing University of Posts and Telecommunications. His research interests include deep learning, computer vision, especially face recognition and metric learning.

MEINA SONG received the Ph.D. degree in electronic engineering from BUPT in 2004. She is currently a professor with the School of Computer Science, BUPT. She has published hundreds of academic or technical papers on journals and conferences and served as the director of Engineering Research Center on Information Network for Education Ministry. Her research interests include mobile Internet, cloud computing, big data and artificial intelligence.

YAN ZHENG received the BEng degree in electrical engineering and the MEng degree in computer science and engineering from the Xi’an Jiaotong University, Xi’an, China in 1994 and 1997, respectively, the second MEng degree in information security from the National University of Singapore, Singapore in 2000, and the licentiate of science and the doctor of science in technology in electrical engineering from Helsinki University of Technology, Helsinki, Finland. She is currently a professor at the Xidian University, Xi’an, China and a visiting professor at the Aalto University, Espoo, Finland. Her research interests are in trust, security, privacy, and security-related data analytics. Prof. Yan serves as a general or program chair for 30+ international conferences and workshops. She is a steering committee co-chair of IEEE Blockchain international conference. She is also an associate editor of many reputable journals, e.g., IEEE Internet of Things Journal, Information Sciences, Information Fusion, JNCA, etc.

HUI PAN received his PhD from the Computer Laboratory at University of Cambridge, and both his Bachelor and MPhil degrees from the University of Hong Kong. He is the Nokia Chair Professor in Data Science and Professor of Computer Science at the University of Helsinki. He is also the director of the HKUST-DT System and Media Lab at the Hong Kong University of Science and Technology. He has published more than 300 research papers and with over 18,000 citations. He has 30 granted and filed European and US patents. He has founded and chaired many IEEE/ACM conferences/workshops, and has served as track chair, senior program committee member, organising committee member, and program committee member of numerous top conferences including ACM WWW, ACM SIGCOMM, ACM Mobisys, ACM MobiCom, ACM CoNext, IEEE Infocom, IEEE ICNP, IEEE ICDCS, IJCAI, AAAI, and ICWSM. He is an Associate Editor for IEEE Transactions on Mobile Computing (since 2014) and the Springer journal of Computational Social Networks. He has also served as Associate Editor for IEEE Transactions on Cloud Computing (2014 - 2018) and guest editor for various journals including IEEE Journal on Selected Areas in Communications (JSAC), IEEE Transactions on Secure and Dependable Computing, IEEE Communications Magazine, and ACM Transactions on Multimedia Computing, Communications, and Applications. He is an ACM Distinguished Scientist, an IEEE Fellow, and a member of Academia Europaea.