FACE AGGREGATION NETWORK FOR VIDEO FACE RECOGNITION

Stefan Hörmann *Zhenxiang Cao* *Martin Knoche* *Fabian Herzog* *Gerhard Rigoll*[*]

Technical University of Munich

ABSTRACT

Typical approaches for video face recognition aggregate faces in a feature space to obtain a single feature representing the entire video. Unlike most previous approaches, we aggregate the faces directly in order to additionally obtain a single representative face as an intermediate output, from which a more discriminative feature vector is extracted. To overcome the limitation of a fixed number of input images of the state of the art in face aggregation, we incorporate a permutation invariant U-Net architecture capable of processing an arbitrary number of frames, which is employed in a generative adversarial network. We demonstrate the effectiveness of our method on three popular benchmark datasets for video face recognition. Our approach outperforms the baselines on the YouTube Faces dataset, obtaining an accuracy of $96.62\,\%$. Besides, we show that our method is robust against motion blur.

Index Terms— Video Face Recognition, Face Aggregation, Generative Adversarial Network, Biometrics

1. INTRODUCTION

Compared to still image Face Recognition (FR), current FR approaches for videos still face numerous challenges as video frames are typically acquired under arguably poor conditions leading to a high variety in head poses, expressions, and motion blur. This requires the model to recognize when a frame does not contain useful information and consequently mitigate its influence. Besides, architectures need to be designed such that they are capable of handling an arbitrary number of frames.

The majority of video FR approaches [1–9] first extract feature vectors from every frame separately and aggregate them based on their relevance into a single feature vector representing the entire video. However, feature extractors are trained on identification and, therefore, do not directly embed feature quality. Thus, estimating the relevance of every feature after its extraction is limited. Moreover, since videos consist of a high number of frames, extracting features from every frame introduces a high computational cost.

As depicted in Figure 1, we propose a framework to aggregate the valuable information from an image set $\mathcal{I}$ consisting of N frames $\boldsymbol{I}_n$ into a single image $\boldsymbol{A}$. This disentanglement of aggregation and recognition not only performs the aggregation when the input quality can be extracted directly but also provides the aggregated image as an additional output. We can state the overall task as follows:

$$\underbrace{\{\boldsymbol{I}_1, \boldsymbol{I}_2, \cdots \boldsymbol{I}_N\}}_{\mathcal{I}} \mapsto \boldsymbol{A} \tag{1}$$

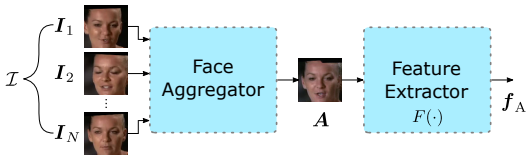

Fig. 1: Overview of our approach: N images $\boldsymbol{I}_n$ are aggregated into one image $\boldsymbol{A}$, whose feature $\boldsymbol{f}_\mathrm{A}$ represents the whole video $\mathcal{I}$.

subject to:

$$\left\|F(\mathcal{I}_k^i) - F(\mathcal{I}_l^j)\right\| \geq \left\|F(\boldsymbol{A}_k^i) - F(\boldsymbol{A}_l^j)\right\| \;\forall\, k, l, i = j \tag{2}$$

$$\left\|F(\mathcal{I}_k^i) - F(\mathcal{I}_l^j)\right\| < \left\|F(\boldsymbol{A}_k^i) - F(\boldsymbol{A}_l^j)\right\| \;\forall\, k, l, i \neq j \tag{3}$$

with the superscripts $\cdot^i$ and subscripts $\cdot_k$ denoting the k-th image of identity i and $F(\cdot)$ a feature extraction network. Hence, we not only aggregate N images into a single image $\boldsymbol{A}$, but also want to ensure an easier differentiation of identities in the feature space, i.e., more discriminative features. Through this aggregation, the influence of outliers is mitigated by only aggregating relevant information into $\boldsymbol{A}$.

The contributions of our work can be summarized as follows:

- We propose the first approach to face aggregation capable of aggregating an *arbitrary number* of faces in a *permutation invariant* manner.
- Our exhaustive evaluation shows that our method outperforms the state of the art and is robust against motion blur degradation.

2. RELATED WORK

Video FR approaches can be divided into set- and sequence-based methods. While set-based algorithms are considering an orderless set of images, sequence-based methods take into account the temporal dependency between frames.

Early set-based approaches extract features from every face separately and compute pairwise distances [1] or apply average/max-pooling to obtain a single feature vector [2]. However, these approaches disregard that not every face contains an equal amount of information. This caused new approaches to emerge, which adaptively aggregate the features based on their quality: The quality of every feature is predicted directly from the features using a cascaded attention network [3, 4], from intermediate feature maps [10], or from the input utilizing a separate network [11]. Also, the quality for every component of a feature vector is estimated by [5, 6].

Like [3, 4], Gong et al. [7, 8] predict feature quality based on the feature vector. However, they incorporate LSTMs and thus are considered sequence-based approaches. The Discriminative Aggregation Network (DAN) [12] uses 20 concatenated faces as the input for the aggregation network to obtain a single face, from which the

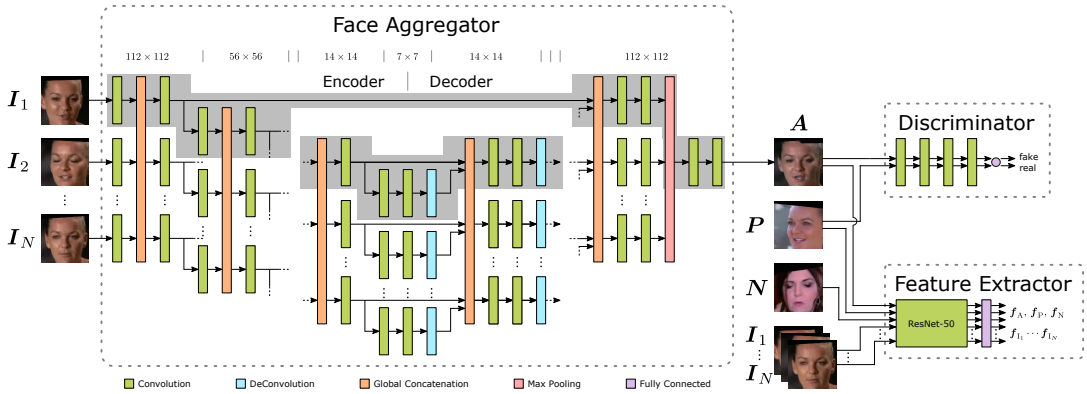

Fig. 2: Our approach for video FR: N faces $\boldsymbol{I}_n$ are aggregated by a permutation invariant U-Net yielding a single face image $\boldsymbol{A}$, whose feature $\boldsymbol{f}_{\mathrm{A}}$ is representative for all N input images. The branch for the first face $\boldsymbol{I}_1$ through the face aggregator is highlighted.

feature vector for FR is extracted. Kang et al. [9] aggregate features within the feature extractor utilizing a pairwise relational network together with LSTMs. Rao et al. [13] predict feature quality by applying attention-aware reinforcement learning in the feature and image space.

3. METHODOLOGY

3.1. Network Architecture

Figure 2 depicts our proposed framework with its three modules being described in the following subsections:

3.1.1. Face Aggregator

In order to overcome the limitation of requiring a fixed number of input faces as in [12] and aggregate the set of N input images $\mathcal{I} = \{\boldsymbol{I}_1, \boldsymbol{I}_2, \cdots \boldsymbol{I}_N\}$ into a single image $\boldsymbol{A}$, the architecture must be permutation invariant and capable of processing an arbitrary amount of input images. As opposed to typical feature aggregation approaches [3, 4, 7, 8, 10, 11], aggregating images is a more complex task. Even though all input images contain the same identity, their pairwise pixel distance is too high, which renders computing the (weighted) average of the input images impossible. Hence, the information needs to be aggregated at multiple depths throughout the network to obtain a realistic image.

Our approach incorporates the global concatenation from [14] into a modified U-Net architecture [15]. All images are processed in parallel and in the same way as weights are shared for every image. However, the global concatenation operation allows every image to include information from the remaining images. Given the intermediate feature tensor $\boldsymbol{X}_n \in \mathbb{R}^{W \times H \times C}$ of the n-th image with height H, width W, and C channels, we obtain the output of the global concatenation layer $\boldsymbol{Y}_n \in \mathbb{R}^{W \times H \times 2C}$ through concatenation along the channel dimension with the max-pooling over all images:

$$\boldsymbol{Y}_n = \boldsymbol{X}_n \oplus \max_j (\boldsymbol{X}_j) \tag{4}$$

The integration of global concatenation layers at multiple depths enables repeated back-and-forth information exchange between members of the set in a permutation invariant manner.

After an initial 3×3 convolutional layer, the encoder block consists of a global concatenation followed by a 1×1 convolutional layer to merge the local with the global features, and a 3×3 convolutional layer. To reduce the feature map dimension, the last convolution is applied with stride 2. Four encoder blocks are stacked to downsize the initial input resolution 112×112 to feature maps of size 7×7.

The decoder block consists of a 4×4 deconvolutional layer, global concatenation, and two convolutional layers of size 1×1 and 3×3. In addition to the global concatenation in the encoder, we further concatenate the output from the encoder of the corresponding resolution to allow the model to skip the lower latent resolutions. The decoder blocks are stacked four times resulting in the initial input dimension at the output.

We conclude the face aggregator with max-pooling over all branches to obtain a single image and add two 3×3 convolutional layers for global optimizations. As activation function, we apply exponential linear unit [16] after every convolutional and deconvolutional layer.

3.1.2. Discriminator

The discriminator's task is to judge whether its input is a real face or a fake face aggregated by the face aggregator. Our discriminator comprises four convolutional layers with leaky ReLU activation function [17]. The former two layers reduce the feature map size to 28×28 using a stride of 2. After further processing of the subsequent two layers, the discriminator is concluded by a single neuron, which is connected to every pixels of the previous feature map. By employing this fully connected layer, the discriminator is capable of valuing every pixel differently. This is important as we expect the discriminator to base its decision rather on the face in the center than on the background.

3.1.3. Feature Extractor

Similar to [12], we utilize a feature extractor to guide the aggregation of the images. We use a pretrained ResNet-50 [18] with ArcFace layer according to [2] to obtain meaningful and well-generalizing features $\boldsymbol{f} \in \mathbb{R}^{512}$.

3.2. Loss Functions

To train our face aggregator, we use a weighted sum of several losses $\mathscr{L}_{\mathrm{G}}$, which are explained in the following:

$$\mathscr{L}_{\mathrm{G}} = \lambda_{\mathrm{dis}} \mathscr{L}_{\mathrm{dis}} + \lambda_{\mathrm{rec}} \mathscr{L}_{\mathrm{rec}} + \lambda_{\mathrm{adv}} \mathscr{L}^{\mathrm{G}}_{\mathrm{adv}} + \lambda_{\mathrm{tv}} \mathscr{L}_{\mathrm{tv}} \tag{5}$$

with the scalars λ_{dis}, λ_{rec}, λ_{adv}, and λ_{tv} to balance the losses.

3.2.1. Discriminative Loss $\mathscr{L}_{dis}$

To ensure more discriminative features, we employ the discriminative loss from the DAN [12] in a triplet loss manner [1]. We extract the features $\boldsymbol{f}_{\text{A}}$, $\boldsymbol{f}_{\text{P}}$, $\boldsymbol{f}_{\text{N}}$ and $\boldsymbol{f}_{\text{I}_n}$ of the aggregated image $\boldsymbol{A}$, an image of the same identity from a different video $\boldsymbol{P}$, an image from a different identity $\boldsymbol{N}$, and the input images $\boldsymbol{I}_n$, respectively.

$$\mathscr{L}_{\text{dis}} = \left(\|\boldsymbol{f}_{\text{A}} - \boldsymbol{f}_{\text{P}}\|_2^2 - \alpha\right)_+ + \left(\beta - \|\boldsymbol{f}_{\text{A}} - \boldsymbol{f}_{\text{N}}\|_2^2\right)_+ \tag{6}$$

$$\alpha = \min_n \left\|\boldsymbol{f}_{\text{I}_n} - \boldsymbol{f}_{\text{P}}\right\|_2^2 \tag{7}$$

with $\beta = 32$ denoting a constant margin and the subscript $_+$ meaning $\max(0, \cdot)$. While the first term ensures that the feature distance between the aggregated image $\boldsymbol{A}$ and a positive image $\boldsymbol{P}$ is smaller than between all input frames $\boldsymbol{I}_n$ and $\boldsymbol{P}$ (cf. Equation 2), the second term maximizes the distances between $\boldsymbol{A}$ and a negative image $\boldsymbol{N}$ (cf. Equation 3).

3.2.2. Reconstruction Loss $\mathscr{L}_{rec}$

Besides $\mathscr{L}_{\text{dis}}$, we further guide the face aggregator by ensuring that $\boldsymbol{f}_{\text{A}}$ represents the corresponding identity. We compare two different forms of reconstruction loss:

The reconstruction loss proposed in [12] aims to reduce the intra-class variance by minimizing the distance between the feature of the reconstructed image $\boldsymbol{f}_{\text{A}}$ and the average feature of all input frames:

$$\mathscr{L}_{\text{rec}}^{\text{avg}} = \left\| \boldsymbol{f}_{\text{A}} - \frac{1}{N} \sum_{i=1}^{N} \boldsymbol{f}_{\text{I}_n} \right\|_2^2 \tag{8}$$

However, the downside of $\mathscr{L}_{\text{rec}}^{\text{avg}}$ is that $\boldsymbol{f}_{\text{A}}$ is trained to match only the video's local features. In order to overcome this limitation and consider global features covering the entire dataset, we propose to incorporate a variant of the center loss [19]:

$$\mathscr{L}_{\text{rec}}^{\text{cen}} = \left\| \boldsymbol{f}_{\text{A}}^j - \boldsymbol{c}^j \right\|_2^2 \tag{9}$$

with the center $\boldsymbol{c}^j$ of the j-th identity, which is updated as follows:

$$\boldsymbol{c}^j = (1 - r)\boldsymbol{c}^j - \frac{r}{N+1} \left[\boldsymbol{f}_{\text{P}} + \sum_{i=1}^{N} \boldsymbol{f}_{\text{I}_n} \right] \tag{10}$$

with $r = 0.5$ denoting the update rate. Compared to $\mathscr{L}_{\text{rec}}^{\text{avg}}$, which is constrained on the current video, the $\mathscr{L}_{\text{rec}}^{\text{cen}}$ considers all samples of the same subject averting the effect of outliers in the current video and producing more robust results.

3.2.3. Adversarial Loss $\mathscr{L}_{adv}$

To ensure that the aggregated image $\boldsymbol{A}$ looks realistic, we utilize the face aggregator as a generator in a Generative Adversarial Network (GAN) structure to employ adversarial loss. While the generator is trying to synthesize a realistic looking image to deceive the discriminator, the objective of the discriminator $D(\cdot)$ is to distinguish the aggregated image $\boldsymbol{A}$ from the real data - in our case, the image from another video $\boldsymbol{P}$:

$$\mathscr{L}_{\text{adv}}^{\text{G}} = -\log(D(\boldsymbol{A})) \tag{11}$$

$$\mathscr{L}_{\text{adv}}^{\text{D}} = -\log(1 - D(\boldsymbol{A})) - \log(D(\boldsymbol{P})) \tag{12}$$

3.2.4. Total Variation Loss $\mathscr{L}_{tv}$

To cope with artifacts being created by the deconvolution layers in the decoder of the face aggregator, we also incorporate the total variation loss [20]:

$$\mathscr{L}_{\text{tv}} = \sum_{x,y=1}^{H,W-1} (\boldsymbol{A}_{x,y} - \boldsymbol{A}_{x,y+1})^2 + \sum_{x,y=1}^{H-1,W} (\boldsymbol{A}_{x,y} - \boldsymbol{A}_{x+1,y})^2 \tag{13}$$

with $\boldsymbol{A}_{x,y}$ denoting the pixel in x-th row and y-th column, and H and W being the height and width of the image, respectively.

4. EXPERIMENTS

4.1. Training Details

We first train the feature extractor on the refined MS-Celeb-1M dataset [21] containing 5.8 M images of over 85 k identities. As preprocessing, we align the faces utilizing the landmarks obtained by MTCNN [22]. Our feature extractor achieves a face verification accuracy of $99.63\,\%$ on the LFW benchmark [23].

We preprocess the VoxCeleb2 dataset [24] by extracting 5 frames per utterance, which yields 5.3 M images of 6 k identities divided into 1.1 M videos. We train the face aggregator with $\mathscr{L}_{\text{G}}$ ($\lambda_{\text{dis}} = \lambda_{\text{adv}} = 1$, $\lambda_{\text{rec}} = 0.5$, and $\lambda_{\text{tv}} = 10^{-4}$) using $\mathscr{L}_{\text{rec}}^{\text{cen}}$ as reconstruction loss, and train the discriminator with $\mathscr{L}_{\text{adv}}^{\text{D}}$ in an alternating manner with Adam optimizer [25]. We train for 6 epoch with a batch size of 6 and an initial learning rate of $5 \cdot 10^{-5}$, which is decreased to $1.25 \cdot 10^{-5}$ after 3 epochs. The input images are aligned as during the pretraining of the feature extractor and are augmented with left-right flipping, random contrast, brightness and saturation, and motion blur with a filter kernel size of up to 9 pixels with a probability of $50\,\%$. Even though our architecture supports an arbitrary number of input images N, we train with constant $N = 10$ to obtain comparable results with DAN [12]. Nevertheless, we demonstrate in the evaluation that our approach is also capable of aggregating $N \neq 10$ images. Note that compared with [12], we do not need an additional pretraining step on MSE loss to initialize the face aggregator and facilitate the convergence as our approach also converges with randomly initialized weights.

4.2. Benchmark Details

We evaluated our proposed framework on face verification using the popular YouTubeFaces (YTF) dataset [26], which comprises 3425 videos of 1595 identities with 48 to 6070 frames per video. We follow the benchmark protocol of DAN [12] by resampling the video to $N \cdot n_{\text{avg}}$ frames. Then, we aggregate every N consecutive frames separately into, in total, n_{avg} images, whose average feature vector is used to represent the entire video. As a distance measure, we employ euclidean distance after L^2 normalizing the features. Besides the related work, we compare our approach with the average of all $N \cdot n_{\text{avg}}$ features *avg* and the DAN trained with our feature extractor and on VoxCeleb2, denoted by *DAN**, for a fair comparison.

We further analyze the face identification performance following the 1:N mixed media protocol of the IJB-B and IJB-C datasets [27, 28], in which sets contain a mix of still images and video frames. IJB-B contains ≈ 67 k pieces of media of 1845 subjects, whereas IJB-C consists of ≈ 138 k pieces of media of 3531 subjects. Both datasets are split into two disjoint galleries allowing open- and closed-set identification. As a baseline, we compute the average of all features and compare it with the aggregation of all video frames into a single representative image followed by averaging the resulting features.

Table 1: Effect of different loss functions on the verification accuracy on the YTF dataset. The last column denotes results averaged over all benchmark parameters. Results marked with ⋄ were obtained by duplicating the input frames due to the limitation of the architecture.

Method	Losses					$N=$	Accuracy [%]						
							1	5	10				
	$\mathscr{L}_{\text{adv}}$	$\mathscr{L}_{\text{dis}}$	$\mathscr{L}_{\text{rec}}^{\text{avg}}$	$\mathscr{L}_{\text{rec}}^{\text{cen}}$	$\mathscr{L}_{\text{tv}}$	$n_{\text{avg}}=$	2	2	2	4	6	8	Avg
avg							95.80	96.40	96.32	96.34	96.34	96.52	96.29
DAN*	√	√	√				95.44 ⋄	96.10 ⋄	96.24	96.48	96.20	96.56	96.17
		√	√		√		95.58	95.90	96.26	96.42	96.34	**96.62**	96.19
		√		√	√		95.52	96.08	96.24	96.44	96.28	96.42	96.16
		√			√		95.44	95.72	96.40	96.42	96.42	96.36	96.13
ours	√	√	√		√		95.58	96.44	96.36	96.44	96.28	96.44	96.26
	√	√		√	√		**95.90**	**96.52**	**96.56**	**96.60**	96.48	**96.62**	**96.45**
	√	√			√		95.58	96.26	96.40	96.58	**96.62**	96.52	96.33
	√	√		√			95.58	96.14	96.42	96.40	96.60	96.58	96.29

4.3. Comparison with State of the Art

The verification accuracy on the YTF dataset is illustrated in Table 2. It is evident that our approach outperforms most but not all state-of-the-art methods, which is mainly due to a more powerful architecture for the feature extractor as in [2, 7, 11] or higher input resolution [3, 10]. Moreover, we did not further finetune the model on the YTF dataset as [12, 13]. Our model surpasses the baselines *avg* and *DAN**, and thus can not only be considered the best performing face aggregation network but also provides the aggregated image as an additional output compared to feature aggregation methods.

The results on the IJB-B and IJB-C datasets are depicted in Table 3. We outperform all baselines by a substantial margin in terms of True Positive Identification Rate (TPIR) at Rank-1 and 0.01 False Positive Identification Rate (FPIR). Furthermore, due to the face aggregation we reduce the computational cost on feature extraction by $77.1\,\%$ (IJB-B) and $80.9\,\%$ (IJB-C).

Table 2: Comparisons of the verification accuracy with the state of the art on the YTF dataset.

Method	Accuracy [%]	Method	Accuracy [%]
DAN [12]	95.01	MARN [7]	96.44
FaceNet [1]	95.12	C-FAN [6]	96.50
NAN [3]	95.72	ADRL [13]	96.52
QAN [10]	96.17	REAN [8]	96.60
FAN [5]	96.21	ArcFace [2]	98.02
PRN [9]	96.3	DDL [11]	**98.18**
avg	96.52	DAN*	96.56
		ours	96.62

Table 3: The average TPIR [%] at Rank-1 and FPIR $= 0.01$ of both galleries on the IJB-B and IJB-C datasets.

	IJB-B		IJB-C	
Method	Rank-1	FPIR $= 0.01$	Rank-1	FPIR $= 0.01$
avg	89.54	75.81	89.18	72.06
DAN*	89.69	74.43	89.71	74.96
ours	**90.44**	**75.93**	**90.67**	**77.44**

4.4. Robustness Analysis

Since motion blur is frequently occurring in video-related tasks, we analyze the robustness of our approach by synthetically applying motion blur to 9 out of $N = 10$ images. As depicted in Table 4, our approach clearly outperforms *avg* and *DAN**. This demonstrates that the face aggregator can correctly identify the untouched frame and mainly uses it for the aggregation.

Table 4: Verification accuracy [%] for $N = 10$ and $n_{\text{avg}} = 4$ on the YTF dataset for different motion blur filter sizes applied to 9 images.

Filter Size	7	9	11	13	15	17
avg	96.28	95.90	95.40	94.78	93.76	92.66
DAN*	96.48	95.68	94.48	93.92	91.28	89.44
ours	**96.50**	**96.48**	**96.40**	**96.18**	**95.82**	**95.92**

4.5. Ablation Study

Table 1 analyzes the influence of different losses and different benchmark parameters on the verification accuracy. In accordance with the findings from Rao et al. [12], we see that utilizing the adversarial Loss $\mathscr{L}_{\text{adv}}$ is beneficial. Moreover, using the global center loss $\mathscr{L}_{\text{rec}}^{\text{cen}}$ increases the performance.

Regarding the benchmark parameters, we see that our approach is flexible concerning the number of input frames N as it outperforms both baselines for $N \in \{1, 5\}$. For $N = 10$ and $n_{\text{avg}} = 2$, the baseline *avg* averages 20 features, whereas our approach creates 2 images, from which the features are averaged. Hence, when comparing our method's accuracy for this case $96.56\,\%$ with the accuracy of *avg* when averaging two images $95.80\,\%$ ($N = 1$ and $n_{\text{avg}} = 2$), we can clearly see that our approach achieves a precise fusion of all relevant information into solely 2 images.

5. CONCLUSION

In this paper, we present a novel approach for video FR based on prior face aggregation. Compared to previous methods, we lift the limitation of aggregating only a fixed number of faces by incorporating a permutation invariant U-Net. Our analysis shows that we outperform state of the art on multiple established video FR benchmarks. Moreover, by synthetically applying motion blur, we show that our approach yields satisfying results despite the lack of details.

6. REFERENCES

[1] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A Unified Embedding for Face Recognition and Clustering," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 815–823.

[2] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "ArcFace: Additive Angular Margin Loss for Deep Face Recognition," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 4690–4699.

[3] J. Yang, P. Ren, D. Zhang, D. Chen, F. Wen, H. Li, and G. Hua, "Neural Aggregation Network for Video Face Recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 5216–5225.

[4] S. Hörmann, M. Knoche, M. Babaee, O. Köpüklü, and G. Rigoll, "Outlier-Robust Neural Aggregation Network for Video Face Identification," in *IEEE International Conference on Image Processing (ICIP)*, 2019, pp. 1675–1679.

[5] Z. Liu, H. Hu, J. Bai, S. Li, and S. Lian, "Feature Aggregation Network for Video Face Recognition," in *IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, 2019, pp. 990–998.

[6] S. Gong, Y. Shi, N. Kalka, and A. Jain, "Video Face Recognition: Component-wise Feature Aggregation Network (C-FAN)," in *International Conference on Biometrics (ICB)*. IEEE, 2019, pp. 1–8.

[7] S. Gong, Y. Shi, and A. Jain, "Low Quality Video Face Recognition: Multi-Mode Aggregation Recurrent Network (MARN)," in *IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, 2019, pp. 1027–1035.

[8] S. Gong, Y. Shi, and A. Jain, "Recurrent Embedding Aggregation Network for Video Face Recognition," *arXiv preprint arXiv:1904.12019*, 2019.

[9] B. Kang, Y. Kim, and D. Kim, "Pairwise Relational Networks for Face Recognition," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.

[10] Y. Liu, J. Yan, and W. Ouyang, "Quality Aware Network for Set to Set Recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 4694–4703.

[11] M. Zhang, G. Song, H. Zhou, and Y. Liu, "Discriminability Distillation in Group Representation Learning," in *European Conference on Computer Vision (ECCV)*. Springer, 2020, pp. 1–19.

[12] Y. Rao, J. Lu, and J. Zhou, "Learning Discriminative Aggregation Network for Video-Based Face Recognition and Person Re-identification," *International Journal of Computer Vision*, vol. 127, no. 6-7, pp. 701–718, 2019.

[13] Y. Rao, J. Lu, and J. Zhou, "Attention-aware Deep Reinforcement Learning for Video Face Recognition," in *IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 3951–3960.

[14] M. Aittala and F. Durand, "Burst Image Deblurring Using Permutation Invariant Convolutional Neural Networks," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 731–747.

[15] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," in *International Conference on Medical Image Computing and Computer-Assisted Interventio (MICCAI)*. Springer, 2015, pp. 234–241.

[16] D.-A. Clevert, T. Unterthiner, and S. Hochreiter, "Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs)," *arXiv preprint arXiv:1511.07289*, 2015.

[17] A. L. Maas, A. Y. Hannun, and A. Y. Ng, "Rectifier Nonlinearities Improve Neural Network Acoustic Models," in *ICML Workshop on Deep Learning for Audio, Speech and Language Processing*, 2013, vol. 30, p. 3.

[18] K. He, X. Zhang, S. Ren, and J. Sun, "Identity Mappings in Deep Residual Networks," in *European Conference on Computer Vision (ECCV)*. Springer, 2016, pp. 630–645.

[19] Y. Wen, K. Zhang, Z. Li, and Y-Qiao, "A Discriminative Feature Learning Approach for Deep Face Recognition," in *European Conference on Computer Vision (ECCV)*. Springer, 2016, pp. 499–515.

[20] A. Mahendran and A. Vedaldi, "Understanding Deep Image Representations by Inverting Them," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 5188–5196.

[21] Y. Guo, L. Zhang, Y. Hu, X. He, and J. Gao, "MS-Celeb-1M: A Dataset and Benchmark for Large-Scale Face Recognition," in *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer, 2016, pp. 87–102.

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

[23] E. G. Huang, G. B. Learned-Miller, "Labeled Faces in the Wild: Updates and New Reporting Procedures," Tech. Rep. UM-CS-2014-003, University of Massachusetts, Amherst, May 2014.

[24] J. S. Chung, A. Nagrani, and A. Zisserman, "VoxCeleb2: Deep Speaker Recognition," *arXiv preprint arXiv:1806.05622*, 2018.

[25] D. P. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," in *International Conference in Learning Representations (ICLR)*, 2015.

[26] L. Wolf, T. Hassner, and I. Maoz, "Face Recognition in Unconstrained Videos with Matched Background Similarity," in *CVPR 2011*, 2011, pp. 529–534.

[27] C. Whitelam, E. Taborsky, A. Blanton, B. Maze, J. Adams, T. Miller, N. Kalka, A. K. Jain, J. A. Duncan, K. Allen, J. Cheney, and P. Grother, "IARPA Janus Benchmark-B Face Dataset," in *IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2017, pp. 592–600.

[28] B. Maze, J. Adams, J. A. Duncan, N. Kalka, T. Miller, C. Otto, A. K. Jain, W. T. Niggel, J. Anderson, J. Cheney, and P. Grother, "IARPA Janus Benchmark - C: Face Dataset and Protocol," in *International Conference on Biometrics (ICB)*, 2018, pp. 158–165.