OUTLIER-ROBUST NEURAL AGGREGATION NETWORK FOR VIDEO FACE IDENTIFICATION

Stefan Hörmann, Martin Knoche, Maryam Babaee, Okan Köpüklü and Gerhard Rigoll

Technical University of Munich

ABSTRACT

Current approaches for video face recognition rely on image sets containing faces of exclusively one identity. However, as image sets are created by unsupervised methods, it is necessary to consider outlier-afflicted sets for real-life applications. In this paper, we propose an Outlier-Robust Neural Aggregation Network (ORNAN). First, we embed each image into a feature space using a Convolutional Neural Network (CNN). With the help of two cascaded attention blocks, we predict outliers within the image set. By integrating this knowledge into our aggregation network, we adaptively aggregate all feature vectors to form a single feature, mitigating the influence of outliers and noisy features. We show that our network is robust against outliers using outlier-afflicted IJB-B and IJB-C benchmarks while maintaining similar performance without outliers.

Index Terms— video face recognition, face identification, biometrics, feature aggregation

1. INTRODUCTION

Lately, video face recognition has gained more and more attention [1–11]. This is among others due to advances made in the models' identification performances. Furthermore, the omnipresence of devices being capable of capturing video causes a huge demand in utilizing the data for e.g. surveillance and access control.

State-of-the-art methods for still image face recognition embed faces into a deep feature space and assign identities to faces by measuring their feature similarity [12–15]. Compared to the identification of still images, the addition of several images to form an image set causes new challenges to emerge. Instead of only one feature vector, a set of feature vectors is obtained. This increased amount of available information requires efficient integration of that information into meaningful features. As for large-scale face recognition problems, calculating pairwise feature distances is computationally very expensive: new approaches prefer the generation of one fixed-size feature vector as a representative for the entire image set.

A simple method of merging an arbitrary amount of features is taking the average of all features. However, this implies that every feature is equally useful, which is usually not the case as image sets contain multiple (often suboptimal) face variations. The information a CNN is able to extract highly depends on the pose, size, illumination and expression of the face, and whether it is affected by motion blur. Therefore, the goal of the aggregation is to integrate all relevant information into one feature vector while at the same time omitting noisy misleading information.

In addition to the aforementioned difficulties, we want to handle image sets with more than one identity as depicted in Figure 1, the impact of which still is to be investigated. Current approaches for video face recognition rely on perfect face sets, i.e. only containing images of one identity, and are vulnerable to the addition of outliers, as we show later. However, since image sets are usually created by unsupervised methods, such as face tracking or face clustering, an outlier affliction must be considered in real-life applications.

With our ORNAN we propose an architecture capable of not only predicting outliers within a set, but also generating an aggregated feature vector with a minimized influence of the outliers.

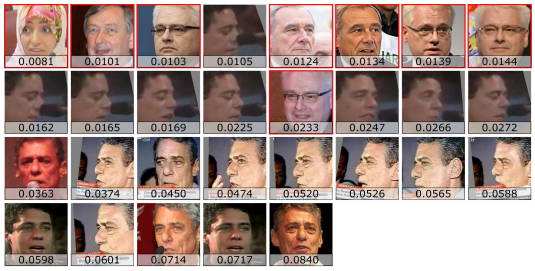

Fig. 1. An outlier-afflicted image set with their ascending relevances predicted by our ORNAN. All outliers (marked with red frame) were correctly identified by our model. Best viewed in color.

2. RELATED WORK

Recent state-of-the-art methods for (video) face recognition are dominated by CNNs, which embed a face into a deep feature space resulting in one feature vector for each image [12–15]. In order to recognize face sets, their similarity is measured by computing pairwise feature distances [12, 13], or using more sophisticated methods like a support vector machine for template adaptive matching [1] or score level fusion [16, 17]. Apart from that, simple average/max-pooling operations are used to obtain a single feature vector for the entire image set [2–5].

However, to better handle the unconstrained scenarios in video face recognition, features are aggregated adaptively based on their discriminability. These methods can be further divided into whether the aggregation takes place before, during, or after feature extraction:

Rao et al. [6] argued that feature extraction networks are not trained to reliably encode image quality, and therefore proposed to generate a single representative image by using a generative adversarial network from which the features are extracted afterwards. Lately, Xie et al. [7] showed that incorporating the aggregation within the network significantly boosts the performance. Also, Kang et al. [8] used a pairwise relational network together with LSTMs to aggregate the features within the network. As opposed to the previous methods, Yang et al. [9] and Liu et al. [10] used attention-based methods to aggregate features after being extracted. Furthermore, Xie et al. [11] introduced a new method for feature aggregation by directly estimating the feature quality.

To the best of our knowledge, outlier robustness was not integrated into a network and analyzed for video face recognition before.

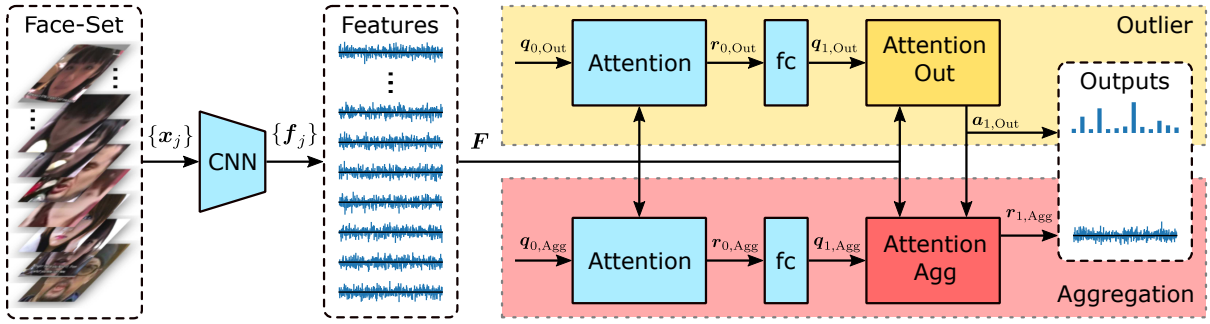

Fig. 2. Our approach for robust video face recognition: Face images are embedded into a deep feature space using a CNN. Based on the extracted features, we then use two cascaded attention blocks separated by a fully connected layer (fc) each to predict outliers within the image set and aggregate all features into one single feature vector $\boldsymbol{r}_{1,\text{Agg}}$ mitigating the outliers' influence.

3. OUTLIER-ROBUST NEURAL AGGREGATION NETWORK

3.1. Feature extraction network

In order to embed face images into a deep feature space, we adapt the Inception-ResNet-v1 architecture from Szegedy et al. [18]. In contrast to the proposed architecture, we insert a 512-dimensional fully connected layer between the dropout and logits layers. Since we train on 8631 identities, this layer acts as bottleneck and improves the network's generalization performance substantially.

3.2. Aggregation network

The goal of video face identification tasks is to predict the ground truth subject ID y_i for a given set of face images $\mathcal{X}_i$. In total, the task consists of K pairs of face data $(\mathcal{X}_i, y_i)_{i=1}^{K}$. Each set $\mathcal{X}_i = \{\boldsymbol{x}_1^i, \boldsymbol{x}_2^i, \cdots, \boldsymbol{x}_{N_i}^i\}$ contains N_i face images $\boldsymbol{x}_j^i$ for $j \in [1; N_i]$. With help of the mapping function $\text{id}(\boldsymbol{x})$, which returns the subject ID for a given a face image, the following equation must hold for all images within a set:

$$y_i = \text{id}\left(\boldsymbol{x}_j^i\right) \quad \forall\, \boldsymbol{x}_j^i \in \mathcal{X}_i \tag{1}$$

For the remaining part of this paper, we omit the index denoting the i-th pair where appropriate to improve readability.

Each face image $\boldsymbol{x}_j \in \mathcal{X}$ is embedded independently into a deep feature space by the CNN described in subsection 3.1 yielding a set of normalized feature vectors $\mathcal{F} = \{\boldsymbol{f}_1, \boldsymbol{f}_2, \cdots, \boldsymbol{f}_N\}$. In order to simplify the subsequent equations, we write $\mathcal{F}$ in matrix form $\boldsymbol{F} \in \mathbb{R}^{512\times N}$. Even though the matrix notation implies dependency between columns, we want to state that the aggregation network is invariant to the order of the features by design.

The goal of our aggregation network is to merge all N features into one single feature vector $\boldsymbol{r} \in \mathbb{R}^{512\times 1}$. As the number of images (and therefore the number of features) N is varying per set $\mathcal{X}$, the network must be able to handle an arbitrary amount of features. We make use of the attention block architecture of the Neural Aggregation Network (NAN) proposed by Yang et al. [9], which consists of two basic attention blocks from Figure 3 separated by a single fully connected layer as shown in Figure 2. Each attention block aims to predict the relevance a_j of a feature $\boldsymbol{f}_j$. Next, the aggregated feature vector $\boldsymbol{r}$ is obtained by computing the weighted average of all features $\boldsymbol{F}$ weighted by their corresponding relevances $\boldsymbol{a} \in \mathbb{R}^{1\times N}$. Using the dot product this can be formulated as follows:

$$\boldsymbol{r} = \boldsymbol{F}\boldsymbol{a}^{\text{T}} \tag{2}$$

The vector of relevances $\boldsymbol{a}$ incorporates the contribution of each feature and therefore needs to contain only positive elements with $|\boldsymbol{a}|_1 = 1$. Both conditions are ensured by applying the softmax function onto the vector of unnormalized relevances $\boldsymbol{e} \in \mathbb{R}^{1\times N}$, which itself is the result of the dot product of a kernel $\boldsymbol{q} \in \mathbb{R}^{1\times 512}$ and $\boldsymbol{F}$:

$$\boldsymbol{a} = \text{softmax}\left(\boldsymbol{e}\right) \tag{3}$$

$$\boldsymbol{e} = \boldsymbol{q}\boldsymbol{F} \tag{4}$$

When setting all elements of $\boldsymbol{a}$ to $\frac{1}{N}$, Equation 2 performs a simple average pooling operation. However, with the kernel $\boldsymbol{q}$ the attention block learns to focus on highly discriminate features while mitigating low quality features. In accordance with [9], we show that this addition leads to superior performance compared to average pooling.

While the kernel of the first attention block $\boldsymbol{q}_{0,\text{Agg}}$ is trained using backpropagation and gradient descent, the kernel of the second attention block $\boldsymbol{q}_{1,\text{Agg}}$ is the output of the fully connected layer connecting both attention blocks:

$$\boldsymbol{q}_{1,\text{Agg}} = \tanh\left(\boldsymbol{W}\boldsymbol{r}_0 + \boldsymbol{b}\right) \tag{5}$$

with $\boldsymbol{W}$ and $\boldsymbol{b}$ denoting the trainable weight matrix and bias vector of the fully connected layer. In contrast to the previously used softmax function, the hyperbolic tangent does not prevent negative values in $\boldsymbol{q}$, which allows the attention model to determine the relevances based on intra-feature dependencies.

The cascaded structure of two attention blocks allows the first attention block to focus on global feature quality metrics, whereas the second attention block together with the fully connected layer elaborates identity-dependent local relevance metrics. Since feature quality may be embedded differently in the deep feature space depending on the identities, the addition of the second block improves the performance of the aggregation network substantially as shown by [9].

3.3. Increasing outlier robustness

In order to increase the aggregation network's robustness against outliers, i.e. sets of images belonging to more than one identity, we add a second task to our model, namely the detection of outliers. Therefore, we create new face data pairs $(\mathcal{X}_i', y_i)_{i=1}^{K}$ where $\mathcal{X}_i' = \mathcal{X}_i \cup \tilde{\mathcal{X}}_i$ denotes the new outlier-afflicted template unifying

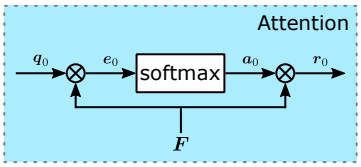

Fig. 3. Basic attention block.

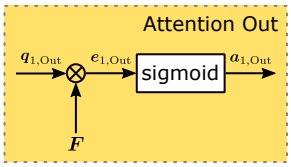

Fig. 4. Second Attention block for Outlier detection.

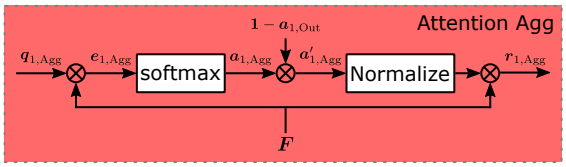

Fig. 5. Second Attention block for Aggregation.

the set $\mathcal{X}_i$ of $N_{\text{prop},i}$ *proper* images with a set $\tilde{\mathcal{X}}_i$ of $N_{\text{imp},i}$ *imposter* images. While Equation 1 still holds for all $\boldsymbol{x}_j^i \in \mathcal{X}_i$, $\tilde{\mathcal{X}}_i$ may only contain images from other identities:

$$\text{id}\left(\boldsymbol{x}_j^i\right) \neq y_i \quad \forall \boldsymbol{x}_j^i \in \tilde{\mathcal{X}}_i \tag{6}$$

Moreover, as outliers may not prevail within $\mathcal{X}_i$, the number of *imposter* images $N_{\text{imp},i}$ needs to be smaller than the number of *proper* images $N_{\text{prop},i}$. Beyond that, we do not constrain $\tilde{\mathcal{X}}_i$ any further and therefore allow more than one identity within $\tilde{\mathcal{X}}_i$.

In order to improve the robustness, we predict outliers and use the prediction to mitigate the relevance of the outliers in the aggregation network. Our outlier detection network consists of the same first attention block (Figure 3) and fully connected layer as the aggregation network. However, we change the second attention block in order to obtain a vector $\boldsymbol{a}_{1,\text{Out}} \in \mathbb{R}^{1\times N}$ predicting which image $\boldsymbol{x}_j \in \mathcal{X}$ is also member of $\tilde{\mathcal{X}}$, i.e. is *imposter*. This is achieved by applying an elementwise sigmoid function as opposed to the softmax function in Equation 3:

$$\boldsymbol{a}_{1,\text{Out}} = \text{sigmoid}\left(\boldsymbol{e}_{1,\text{Out}}\right) \tag{7}$$

By using the sigmoid function we not only enforce positive elements, but also ensure that all elements are independent among each other, which allows the simultaneous prediction of multiple outliers.

Next, we want to capitalize on the outlier predictions in the aggregation network. Since $\boldsymbol{a}_{1,\text{Out}}$ embodies the probability of the corresponding feature being *imposter* and $\boldsymbol{a}_{1,\text{Agg}}$ describes the discriminability of the features, we pool both knowledges using the Hadamard product:

$$\boldsymbol{a}'_{1,\text{Agg}} = \boldsymbol{a}_{1,\text{Agg}} \circ \left(\mathbf{1} - \boldsymbol{a}_{1,\text{Out}}\right) \tag{8}$$

In order to improve convergence of the network we normalize $\boldsymbol{a}'_{1,\text{Agg}}$ such that $\left|\boldsymbol{a}'_{1,\text{Agg}}\right|_1 = 1$. Ultimately, the final outlier-robust aggregated feature $\boldsymbol{r}_{1,\text{Agg}}$ is obtained according to Equation 2 using $\boldsymbol{a}'_{1,\text{Agg}}$.

3.4. Loss functions

For our ORNAN we use the following loss function $\mathcal{L}$:

$$\mathcal{L} = \mathcal{L}_{\text{CE,Agg}} + \lambda_1 \mathcal{L}_{\text{CE,Out}} + \lambda_2 \mathcal{L}_{\text{REG}} \tag{9}$$

with $\mathcal{L}_{\text{CE,Agg}}$ and $\mathcal{L}_{\text{CE,Out}}$ denoting the cross-entropy loss of the aggregation and outlier detection network, respectively, $\mathcal{L}_{\text{REG}}$ the regularization loss, and λ_1 and λ_2 weights for the corresponding loss term. To compute $\mathcal{L}_{\text{CE,Agg}}$, we add an additional fully connected layer followed by a softmax layer after $\boldsymbol{r}_{1,\text{Agg}}$, whereas for $\mathcal{L}_{\text{CE,Out}}$ we can directly use $\boldsymbol{a}_{1,\text{Out}}$. As $\mathcal{L}_{\text{REG}}$ we use the L^2-Norm of all trainable weights except the CNN.

4. EXPERIMENTS

4.1. Training details

We train the CNN and ORNAN separately using the VGGFace2 training dataset, which consists of over 3.1M images from 8631 identities [19]. For both trainings we align the faces using facial landmarks predict by the LGCN [20] and crop them to 160×160 pixels. With the normalized features from the CNN we achieve $99.52\,\%$ accuracy on the LFW benchmark [21]. Afterwards, we freeze the weights of the CNN and use it only as feature extractor.

Our ORNAN is trained for 10 epochs (about 20 hours on a Nvidia 1080 Ti GPU) using random sets of images containing up to 40 *proper* and 5 *imposter* images. We use ADAM-optimizer [22] with an initial learning rate of 0.05. To balance the losses in Equation 9, we set $\lambda_1 = 0.5$ and $\lambda_2 = 5 \cdot 10^{-5}$. All weights are initialized with zeros in order to start with average pooling. Data augmentation is applied separately to each image using horizontal flipping and motion blur. Moreover, we statistically augment the extracted features by adding gaussian noise $\mathcal{N}(0, 0.05)$ to $50\,\%$.

4.2. Benchmark details

We evaluated our approach following the 1:N mixed media identification protocol of the IJB-B and IJB-C datasets [23, 24], in which sets $\mathcal{X}$ contain still images and/or video frames. This protocol allows open-set and closed-set performance evaluation by splitting the data into two disjoint gallery sets $\mathcal{G}_1$ and $\mathcal{G}_2$, and a probe set $\mathcal{P}$ containing sets from all subjects. In order to evaluate the robustness of our approach against outliers we create a new benchmark $\beta\left(r_{\text{max}}, N_{\text{imp,max}}\right)$ which selects $N_{\text{imp},i}$ *imposter* images for a *proper* set $\mathcal{X}_i$ according to the following formula:

$$N_{\text{imp},i} = \max\left(\max\left(r_i, r_{\text{max}}\right) \cdot N_{\text{prop},i}, N_{\text{imp,max}}\right) \tag{10}$$

with $r_i = \frac{N_{\text{imp},i}}{N_{\text{prop},i}}$, and r_{max} and $N_{\text{imp,max}}$ being the maximum value for the ratio r_i and $N_{\text{imp},i}$, respectively. Both parameters not only ensure that the *imposter* samples are not prevailing in $\mathcal{X}'$, but also allow adjustments to the difficulty of the benchmark. In order to make the benchmark deterministic we consecutively add *imposter* images from $\mathcal{G}_1$ to $\mathcal{P}$, i.e. to $\mathcal{X}_1 \in \mathcal{P}$ we add the first $N_{\text{imp},1}$ images in $\mathcal{G}_1$, yielding the outlier-afflicted probe set $\mathcal{P}_1$. We increase r_i in steps of 0.1 from 0 to r_{max} after each $\mathcal{X}_i$ (and reset r_i after reaching r_{max}) to vary the outlier-affliction of $\mathcal{X}'_i$. The gallery sets $\mathcal{G}_1$ and $\mathcal{G}_2$ are untouched. By matching $\mathcal{P}_1$ with $\mathcal{G}_1$ (and $\mathcal{P}_2$ with $\mathcal{G}_2$) we further increase the difficulty as all *imposter* images in $\mathcal{X}'_i \in \mathcal{P}_1$ are also member of $\mathcal{G}_1$ i.e. are known beforehand.

4.3. Baseline methods

In order to evaluate the face identification performance and robustness of *ORNAN*, we compare it with the aforementioned average pooling *AvgPool* as aggregation method. Moreover, we train the

Table 1. Face identification performance of outlier-afflicted sets: the average TPIR is reported for Rank-1 and FPIR = 0.01 together with the mAP for IJB-B and IJB-C datasets in percent.

	IJB-B						IJB-C					
	$\beta\,(0.4, 5)$		$\beta\,(0.6, 10)$				$\beta\,(0.4, 5)$		$\beta\,(0.6, 10)$			
Method	Rank-1	FPIR = 0.01	Rank-1	FPIR = 0.01	mAP		Rank-1	FPIR = 0.01	Rank-1	FPIR = 0.01	mAP	
L_2-Min	37.0	0.0	28.3	0.0	73.6		33.4	0.0	25.4	0.0	74.3	
L_2-Mean	84.1	36.9	81.3	29.9	93.1		82.1	32.4	79.9	26.7	94.5	
AvgPool	85.2	60.1	83.6	**57.2**	–		84	58.6	82.9	**56.4**	–	
NAN [9]	77.4	3.4	70.2	1.1	32.2		77.7	1.4	70.9	0.6	31.1	
ONAN	87.0	56.0	84.9	37.1	58.8		87.0	59.9	85.3	44.0	59.9	
ORNAN	87.2	57.3	85.3	39.9	67.1	**94.5**	87.3	62.4	85.8	47.5	68.4	**95.6**
ORNAN^2	**87.4**	**61.2**	**85.8**	42.6	63.1	91.7	**87.5**	**63.1**	**86.3**	51.8	64.4	92.5

NAN with (*ONAN*) and without outliers (*NAN*). All methods yield a single aggregated feature vector $\boldsymbol{r}_{1,\text{Agg}}$, which we normalize, and use to compute the L_2 feature distances between probe and gallery images sets. We also capitalize on the outlier predictions of the ORNAN by removing all features where $\boldsymbol{a}_{1,\text{Out}} > 0.32$ and aggregate the features a second time (*ORNAN*^{2}).

Besides methods providing an aggregated feature vector, we measure the similarity of two image sets by computing the pairwise L_2 feature distances. By taking the min- and mean-operation on all distances we obtain a single representative distance. These methods are denoted by *L_2-Min* and *L_2-Mean*, respectively.

In contrast to [9], we decided to exclude max pooling and *L_2-Max* as both methods yielded poor results.

4.4. Face identification results

Following the protocol, we computed the average True Positive Identification Rate (TPIR) of both galleries at Rank-1 and 0.01 False Positive Identification Rate (FPIR) without any *imposter* images. The results are reported in Table 2. Computing $\boldsymbol{a}'_{1,\text{Agg}}$ takes ≈ 17 min for IJB-C with the ORNAN using only $\approx 1\,\%$ of the processing time. All variants of the NAN provide a significant advantage over naive average pooling or computing pairwise feature distances. Moreover, training with *imposter* images only slightly decreases the accuracies. While the TPIR of the ORNAN can not directly compete with other state-of-the-art methods, we do not consider it a disadvantage as the outlier detection module can be integrated into any method.

Table 2. Face identification without outliers: the average TPIR at Rank-1 and FPIR = 0.01 of both galleries are reported in percent.

	IJB-B		IJB-C	
Method	Rank-1	FPIR = 0.01	Rank-1	FPIR = 0.01
GOTS [23, 24]	42	–	38	–
VGGFace [3]	78	–	79	–
VGGFace2 [19]	90.2	74.3	91.4	76.3
FPN [25]	91.1	–	–	–
Light CNN-29 [26]	91.9	–	–	–
PRN [8]	93.5	81.4	–	–
L_2-Min	85.3	57.4	85.4	56.1
L_2-Mean	85.4	43.8	83.3	38.5
AvgPool	85.4	61.7	84.5	59.9
NAN [9]	88.2	68.8	88.7	67.9
ONAN	88.2	67.7	88.4	66.8
ORNAN	88.1	68.0	88.3	66.7
ORNAN^2	88.1	68.0	88.2	66.7

4.5. Outlier robustness

In order to evaluate the prediction of outliers we compute the mean Average Precision (mAP), which maximizes when the queried outliers are at the beginning of a prioritized list. To generate the list of predicted outliers we sort $\boldsymbol{a}_{1,\text{Agg}}$ and $\boldsymbol{a}_{1,\text{Out}}$ by ascending and descending values, respectively. Moreover, we compute the pairwise intra-set distances of the features of each set and use the min- and mean-operation to get a prioritized list. As reported by the mAP values in Table 1, the ORNAN is able to reliably predict the outliers.

When comparing Table 2 with Table 1 we can clearly see the drop of TPIR in all baseline methods due to the addition of outliers. However, the ORNAN outperforms the baseline methods for outlier-afflicted image sets, which is also depicted in Figure 6. The accuracy can be further improved by removing the predicted outliers from the set and aggregate the features a second time (ORNAN^2). Using this method, our network is able to remove 114k (of 154k) *imposter* and 9k (of 915k) *proper* images for $\beta\,(0.6, 10)$ on IJB-C. On the contrary, for low FPIR the average pooling may still be a viable option.

Furthermore, we found that it is best to disentangle the outlier prediction and aggregation task and not to enforce high mAP for $\boldsymbol{a}_{1,\text{Agg}}$ as otherwise the network loses its capability of reliably estimating the feature quality.

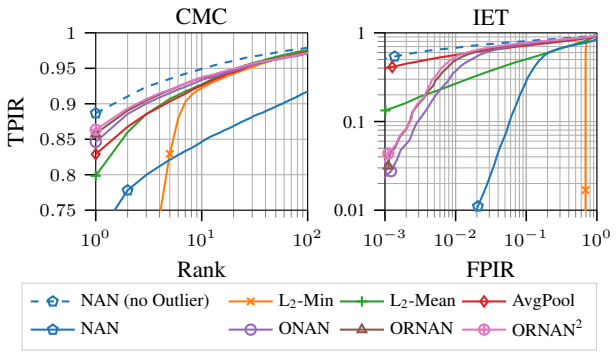

Fig. 6. Average Cumulative Match Characteristic (CMC) and Identification Error Trade-off (IET) for the $\beta\,(0.6, 10)$ benchmark on the IJB-C dataset. Both y-axes refer to the TPIR. Best viewed in color.

5. CONCLUSION

Using the example of NAN [9] we showed that state-of-the-art methods are vulnerable to outlier affliction even when trained with outliers. With our approach the drop in accuracy can be mitigated substantially, and outliers can be predicted and removed reliably to further increase the performance. As future work, we plan to demonstrate the effectiveness of our outlier detection module by integrating it into other approaches.

6. REFERENCES

[1] N. Crosswhite, J. Byrne, C. Stauffer, O. Parkhi, Q. Cao, and A. Zisserman, "Template Adaptation for Face Verification and Identification," in *12th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition*, 2017, pp. 1–8.

[2] 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 *IEEE International Conference on Computer Vision workshops*, 2015, pp. 360–368.

[3] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep Face Recognition," in *Proceedings of the British Machine Vision Conference*, 2015, pp. 41.1–41.12.

[4] A. R. Chowdhury, T.-Y. Lin, S. Maji, and E. Learned-Miller, "One-to-many face recognition with bilinear CNNs," in *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2016, pp. 1–9.

[5] C. Ding and D. Tao, "Trunk-Branch Ensemble Convolutional Neural Networks for Video-Based Face Recognition," 2018, vol. 40, pp. 1002–1014.

[6] Y. Rao, J. Lin, J. Lu, and J. Zhou, "Learning Discriminative Aggregation Network for Video-Based Face Recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 3801–3810.

[7] W. Xie, L. Shen, and A. Zisserman, "Comparator Networks," in *The European Conference on Computer Vision (ECCV)*, 2018.

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

[9] 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*, 2017, pp. 5216–5225.

[10] X. Liu, B.V.K Vijaya Kumar, C. Yang, Q. Tang, and J. You, "Dependency-aware Attention Control for Unconstrained Face Recognition with Image Sets," in *The European Conference on Computer Vision (ECCV)*, 2018.

[11] W. Xie and A. Zisserman, "Multicolumn Networks for Face Recognition," in *BMVC*, 2018.

[12] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *IEEE Conference onComputer Vision and Pattern Recognition*, 2015, pp. 815–823.

[13] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "DeepFace: Closing the Gap to Human-Level Performance in Face Verification," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1701–1708.

[14] J. Deng, J. Guo, X. Niannan, and S. Zafeiriou, "ArcFace: Additive Angular Margin Loss for Deep Face Recognition," 2018.

[15] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song, "SphereFace: Deep Hypersphere Embedding for Face Recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 6738–6746.

[16] L. Leng, J. Zhang, G. Chen, M. Khan, and P. Bai, "Two Dimensional PalmPhasor Enhanced by Multi-orientation Score Level Fusion," in *Secure and Trust Computing, Data Management and Applications*, 2011, pp. 122–129.

[17] L. Leng and J. Zhang, "PalmHash Code vs. PalmPhasor Code," 2013, vol. 108, pp. 1 – 12.

[18] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, Inception-ResNet and the Impact of Residual Connections on Learning," in *ICLR Workshop*, 2016.

[19] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, and A. Zisserman, "VGGFace2: A dataset for recognising faces across pose and age," in *International Conference on Automatic Face and Gesture Recognition*, 2018.

[20] D. Merget, M. Rock, and G. Rigoll, "Robust Facial Landmark Detection via a Fully-Convolutional Local-Global Context Network," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 781–790.

[21] 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.

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

[23] 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*, 2017, pp. 592–600.

[24] 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.

[25] F.-J. Chang, A. T. Tran, T. Hassner, I. Masi, R. Nevatia, and G. Medioni, "FacePoseNet: Making a Case for Landmark-Free Face Alignment," in *IEEE International Conference on Computer Vision Workshops*, 2017, pp. 1599–1608.

[26] X. Wu, R. He, Z. Sun, and T. Tan, "A Light CNN for Deep Face Representation With Noisy Labels," 2018, vol. 13, pp. 2884–2896.