ARFace: Attention-Aware and Regularization for Face Recognition With Reinforcement Learning

Liping Zhang, *Member, IEEE*, Linjun Sun, *Student Member, IEEE*, Lina Yu, *Member, IEEE*, Xiaoli Dong, *Member, IEEE*, Jinchao Chen, *Member, IEEE*, Weiwei Cai, *Graduate Student Member, IEEE*, Chen Wang, and Xin Ning, *Member, IEEE*

***Abstract*—Different face regions have different contributions to recognition. Especially in the wild environment, the difference of contributions will be further amplified due to a lot of interference. Based on this, this paper proposes an attention-aware face recognition method based on a deep convolutional neural network and reinforcement learning. The proposed method composes of an Attention-Net and a Feature-net. The Attention-Net is used to select patches in the input face image according to the facial landmarks and trained with reinforcement learning to maximize the recognition accuracy. The Feature-net is used for extracting discriminative embedding features. In addition, a regularization method has also been introduced. The mask of the input layer is also applied to the intermediate feature maps, which is an approximation to train a series of models for different face patches and provide a combined model. Our method achieves satisfactory recognition performance on its application to the public prevailing face verification database.**

***Index Terms*—Attention-aware, reinforcement learning, regularization, face recognition.**

I. Introduction

FACE recognition technology has become an inseparable part of our daily life, and its applications have penetrated into various areas such as financial services, public security, government affairs, and transport as well in retail services. Through continuous efforts, the tremendous improvement in computer vision, especially in the wild of face recognition, [1]–[5] has been brought due to the development of the deep convolutional neural network. Focusing on the solutions of face recognition issue, it is often considered as a multi-classification task, using the softmax loss function and its variants, such as SphereFace [3] and ArcFace [4], which explicitly enforce the inter-class angle constraint to maximize face class separability. Moreover, approaches such as triplet loss [6], center loss [7] are adopted to realize metric learning of face similarity by using Euclidean distance instead of cosine distance.

Through the investigation of the mentioned literature, the existing studies commonly place more emphasis on designing the loss function to obtain discriminative features [3], [4], [8], [9]. These methods have greatly improved the performance of face recognition, but defects still exist in facing an unconstrained situation in the wild. When dealing with unconstrained face recognition, the performance of face recognition will be greatly affected by the factors of posture change and large scale occlusion. Many studies have confirmed that different face regions make different contributions on recognition [2], [10], [11]. Especially in the wild environment, with a lot of interferences, those differences will be further magnified. To investigate the discrepancies influence caused by different face regions, Sun *et al.* implemented DeepID [2], which extracts features from face patches corresponding to different regions, scales and channels to get a complementary and over-complete representation. Then, in DeepID2 [10] and DeepID2+ [11], the face partitioning method was adopted. The face recognition method proposed by Baidu [12] also follows this method, where the face image is divided the face image into multiple patches with overlapping parts by face landmarks. And same network is used to extract features of different face regions.

DeepID [2], [10], [11] series and the method proposed by Baidu [12] all use the face partitioning strategy. And the features of multiple face blocks are combined to obtain over-complete features. However, both of these algorithms have a large number of parameters, and training and inference are time-consuming. We believe that patches in different face region have different effects on the performance of face recognition. Some unimportant facial blocks, such as severe occlusion, may even have a negative impact on recognition. Face

Manuscript received April 11, 2021; revised June 29, 2021; accepted August 8, 2021. Date of publication August 11, 2021; date of current version February 25, 2022. This work was supported in part by the National Natural Science Foundation of China under Grant 61901436, and in part by the Key Research Program of the Chinese Academy of Sciences under Grant XDPB22. This article was recommended for publication by Associate Editor S. Zafeiriou upon evaluation of the reviewers' comments. *(Liping Zhang and Linjun Sun contributed equally to this work.) (Corresponding author: Xin Ning.)*

Liping Zhang, Linjun Sun, Lina Yu, Xiaoli Dong, and Xin Ning with the Institute of Semiconductors, Chinese Academy of Sciences, Beijing 100083, China, also with the Center of Materials Science and Optoelectronics Engineering & School of Microelectronics, University of Chinese Academy of Sciences, Beijing 100049, China, also with the Beijing Key Laboratory of Semiconductor Neural Network Intelligent Sensing and Computing Technology, the Institute of Semiconductors, Chinese Academy of Sciences, Beijing 100083, China, and also with the Cognitive Computing Technology Joint Laboratory, Wave Group, Beijing, China (e-mail: zliping@semi.ac.cn; sunlinjun@semi.ac.cn; yulina@semi.ac.cn; dongxiaoli@semi.ac.cn; ningxin@semi.ac.cn).

Jinchao Chen is with the School of Computer Science, Northwestern Polytechnical University, Xi'an 710072, Shaanxi, China (e-mail: cjc@nwpu.edu.cn).

Weiwei Cai is with the School of Logistics and Transportation, Central South University of Forestry and Technology, Changsha 410004, China (e-mail: vivitsai@ieee.org).

Chen Wang is with the School of Computer Science and Engineering, State Key Laboratory of Software Development Environment, Jiangxi Research Institute, Beihang University, Beijing 100191, China.

Digital Object Identifier 10.1109/TBIOM.2021.3104014

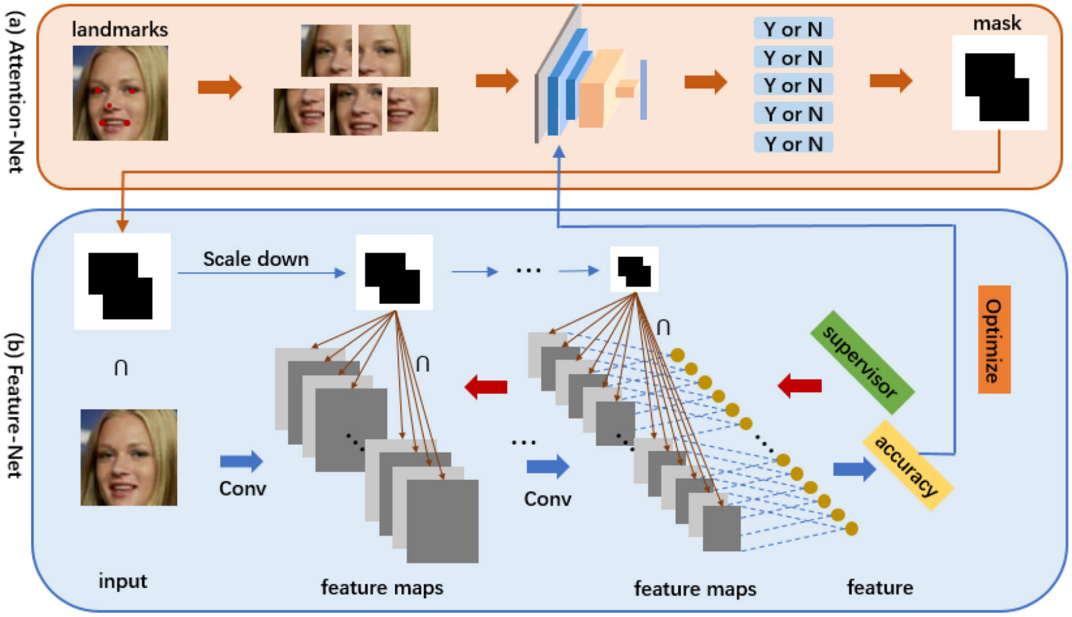


Fig. 1. The framework of ARFace. We propose ARFace for face recognition. (a) The Attention-Net classifies 5 face blocks and then outputs a face mask, which will be masked in the face image and feature maps of the Feature-Net. (b) The Feature-Net is used to extract embedding features. Accuracy is used to optimize the parameters of the Attention-Net.

partitioning strategy is also used in our previous paper [13] PickPatch. PickPatch is a professional regularization method for face recognition. PickPatch improves the robustness and recognition performance of the model by dropping some face blocks during training.

Current prevailing face recognition methods have focused on the enhancement of the discriminative power of face features. Comparatively, only a few studies have investigated regularization methods for face recognition. In fact, the generalization ability of the model is also important when dealing with a large dataset containing noise in unconstrained face recognition. One common generalization method is the model combination [17], which aims at improving their overall performance by fusing models that are complementary to each other. However this requires massive computing resources, huge training and inference time. The dropout [18] has been considered to be effective for obtaining the same performance as model combination, while utilizing fewer resources. When training networks with dropout, the activations are set to zero with some fixed probability. The network then approximates the result of exponentially-sized ensembles of sub-networks. Existing studies have shown that dropout is powerful for fully-connected layers, but has limited effect for convolution layer [19]. DropBlock [20] claims that information flow can still be transmitted, even with dropout, because of the spatial correlation of the features in the convolution layers. Inspired by DropBlock [20], we also apply the mask to intermediate convolution layers in this work, which is a regularization method professionally for face recognition.

Unlike the above-mentioned methods, in this paper, we propose an attention-aware face recognition system. The idea comes from the mechanism of attention in the human visual system. Humans selectively focus on salient areas to make accurate visual judgments. Not all parts of the face image give us positive information [14], some unimportant face blocks, such as severe occlusion, may even have a negative impact on recognition. The overall algorithm framework is shown in Figure 1. The algorithm is mainly composed of two parts, namely Attention-Net and Feature-Net. Attention-Net finds the face patches that are ineffective or counter-productive, and drops them in the input image. The decision-making process is implemented by reinforcement learning and the positions of patches are decided by facial landmarks. To improve the generalization ability of the model, the dropping behavior is also applied to part of convolution layers. The Feature-net is used to extract discriminative embedding features. The input of the Feature-Net is the image masked by the output of Attention-Net and the Attention-Net is trained by the accuracy of Feature-Net. Our method achieves competitive face recognition performance on the public face verification database.

The structure of the paper is as follows: the related works on attention mechanism, reinforcement learning and regularization methods are presented in Section II. It is followed by Section III, in which a detailed description of the algorithm is introduced. The experimental results and the attention-aware image in face recognition are presented in Section IV. Conclusions from the work are given in Section V.

II. Related Work

A. Attention Mechanism

The attention mechanism is an important part of the human visual system, in which humans selectively focus on salient features to make accurate visual judgments. In the field of computer vision, attention mechanisms have been widely used in fields, including, but not limited to, image classification [21], [22], [23], [24], [25], object detection [26], visual interpretation [27] and person re-identification [28], [29]. The class activation mapping (CAM) [26] module was proposed to demonstrate that this module can locate areas of interest in the image by category supervision, visually demonstrating the effect of the attention mechanism. Attention mechanisms based on convolution neural networks (CNNs) can be divided into the channel domain and the spatial domain. The squeeze-and-excitation network (SENet) [21], as the champion of 2017 ImageNet classification task, introduced an attention mechanism in the channel dimension of the network design, enabling the model to focus on to the channel features with the most information and suppress the unimportant features. The convolution block attention module (CBAM) [30] consisting of a channel attention module and spatial attention module. In this, the attention map is multiplied by the input feature map to carry out feature adaptive learning. The combination of channel attention and spatial attention makes the model pay more attention to the target object itself.

Unlike the previous approaches, we focus on the data level rather than the network structure. In our method, some patches are selected as the active region according to the face landmarks, and the rest are set to zero. This will suppress the features which are unimportant or even have negative effect on recognition in the data source.

B. Reinforcement Learning

As a branch of machine learning, reinforcement learning emphasizes on how to act based on the environment in order to maximize the expected reward. Reinforcement learning is widely used in network structure search tasks [31], [32], [33], [34]. Neural architecture search network(NASNet) [31] proposed by Google is a typical example of using reinforcement learning to search the network structure. NASNet [31] includes a controller based on a recurrent neural network (RNN) and a real network structure generated by the combination of the output of the controller. Network architecture search using reward-oriented reinforcement learning requires less professional knowledge than designing a network structure by hand. It is not only the network structure, but also the data augmentation can get the optimal solution using the reinforcement learning search. AutoAugment [35] creates a search space for data augmentation policy and selects suitable strategies for a particular data set that can be migrated to similar data sets.

Inspired by the above works, reinforcement learning is used to solve the problem of face block selection in this work, which simulates the attention mechanism of the human visual system. We use the policy gradient, which directly outputs the behavior without analyzing the reward and punishment values to optimize the Attention-Net, and then obtain the input of the Feature-Net that is beneficial for identification.

C. Regularization

Deep convolution neural networks with a large amount of data and regularization tend to achieve strong hierarchical feature representation. A considerable number of regularization methods have been proposed through data augmentation [36], [37], [38] or adding noise during training [18], [20], [39], [42] in order to prevent over-fitting. Cutout [36] creates out-of-distribution examples by randomly dropping out a square region of input image, which makes the network more robust to noise and occlusion. However, smart augmentation [38] used the idea of generative adversarial neural networks [40]. Smart augmentation uses a deep neural network to generate new samples by inputting some images from the same class. Not at the data level, DropBlock [20] randomly drops out a square region in a feature map, while SpatialDropout [19] drops out entire channels. Similar to data augmentation, shake-shake [41] and ShakeDrop [42] introduce gradient augmentation, which involves adding noise in gradient propagation. The common idea of all the above-mentioned regularization methods is that the over-fitting information flow can be disturbed with noise.

Inspired by DropBlock [20], Our method is a face-specific regularization method by choosing patches on the basis of face landmarks during the training phase. The position of patch in the image and feature map is consistent, which adds noise not only in the data source but also training steps.

III. ARFace

In this paper, we propose ARFace, an attention-aware face recognition method that can be integrated into various networks for end-to-end training. The ARFace algorithm is mainly composed of two parts, namely the Attention-Net and the Feature-Net. The Attention-Net outputs the mask of face images, which is used to determine whether each patch will be dropped or not. Feature-net is used for extracting discriminative embedding features, with the mask is applied on input image and the intermediate feature maps. The Feature-net is trained by gradient back propagation. At the same time, the classification accuracy of the Feature-Net is used for learning parameters of the Attention-Net, which completes a self-driven optimization process. The overall algorithm framework is shown in Figure 1.

A. Attention-Net

Simulating the attention mechanism of the human visual system, we believe that not all area of the face has a positive effect on the recognition performance, especially when dealing with unconstrained face recognition. Therefore, instead of taking the whole face image as the input, we introduce an Attention-Net to determine which blocks will be used for extracting face features. The overall algorithm framework of Attention-Net is shown in Figure 2.

Firstly, for all face images, their facial landmarks are detected using the multitask cascaded CNN algorithm [44], and the detected faces are aligned with similarity transformation according to their landmarks. Consequently five landmarks for each image are obtained. We extend a certain radius around the landmarks so that five patches can be obtained.

Then a simple CNN network structure that contains only two layers of convolution is used to determine whether a face patch is dropped or not. Here, we mainly consider the shallow

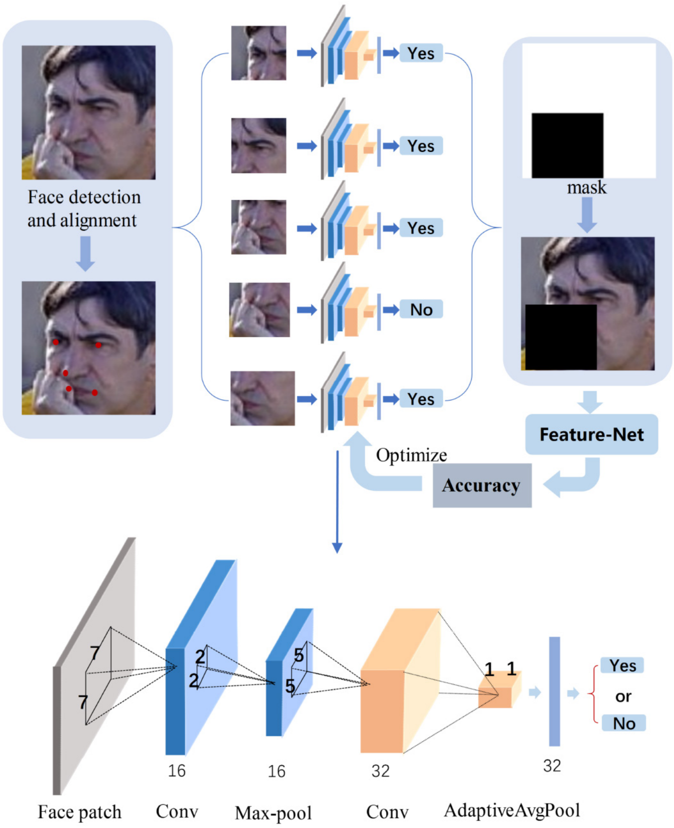


Fig. 2. The algorithm framework of Attention-Net. It takes face patchs as input, and then a five-layer CNN network structure is used to determine whether a block is picked.

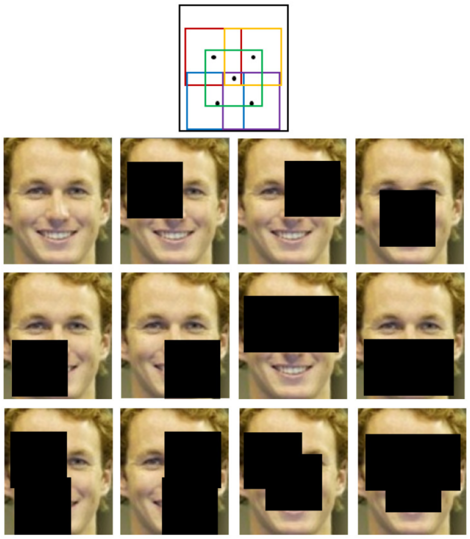


Fig. 3. The effect of part mask modes applied on the input image.

semantics of the image, that is, whether a face patch contains a large range of occlusion, pose, expression changes and other factors that are not conducive to face recognition. At the same time, the large kernel size and pooling operation are used for increasing the receptive field and convergent context. Further, we use the parametric rectified linear unit (PReLU) [43] to increase the nonlinearity of the network.

Attention-Net takes the face patchs as the input and output a face mask, M, in which dropped patches are set to zero. Following this, M is applied on the input image and the intermediate feature maps of the Feature-Net, namely

$$A = A * M, \tag{1}$$

here, A represents the input image or intermediate feature map, and $*$ represents Hadamard product.

In order to select the face blocks which are conducive to recognition, we train this simple CNN with reinforcement learning to maximize the expected accuracy with the generated mask is applied on the input image and the intermediate feature maps. The input of Attention-Net is the state s_t, and the output is the action a_t, respectively in reinforcement learning. The sequence $\{s_1, a_1, s_2, a_2, \ldots\}$ is called the trajectory τ.

The optimization solution of Attention-Net is realized by the policy gradient to maximize the expected reward on the training set. In supervised learning, we use cross-entropy to compare the differences between the two distributions. In this work, the face recognition accuracy obtained from the Feature-Net has been used as the label of a trajectory, τ, in the policy gradient optimization solution. When training the parameters, θ, of Attention-Net, we fix the parameters, w, of Feature-Net. Then, the loss function of Attention-Net can be expressed as

$$L(\theta) = -\frac{1}{N}\sum_{\tau} R(\tau) log p_{\theta}(\tau), \tag{2}$$

here, a trajectory τ is defined as the feature extraction of a batch of data. Then, $R(\tau)$, which is the accuracy of the Feature-Net on that masked batch data, can be obtained. $p_{\theta}(\tau)$ is the probability distribution of the trajectory τ.

Similarly, we use the gradient descent algorithm to solve it. Then the parameter update rule of Attention-Net can be expressed as

$$\theta \leftarrow \theta - \alpha \nabla_{\theta}(R(\tau) - b) log p_{\theta}(\tau), \tag{3}$$

where b is a reinforcement baseline, which is employed to reduce the variance of this estimate [65]. As long as the baseline b does not depend on the current action, then this is still an unbiased estimate for the gradient. In this work, we set b according the code provided by [32].

In this work, Attention-Net and Feature-net alternately update parameters in a batch. After multiple optimizations, face patches that are not important or have a negative impact on recognition will be dropped. This method can filter out interference information in the face, so as to give more weightage to the face parts which are beneficial for the recognition. What's more, it is also a regularization method. In the following section, the effect of this method on the recognition performance, when face patches are randomly dropped, has been assessed.

The number of patches used in this paper is 5, and there are 32 mask modes except that all patchs are not picked, Figure 3 shows the effect of part mask modes applied on the input image.

B. Feature-Net

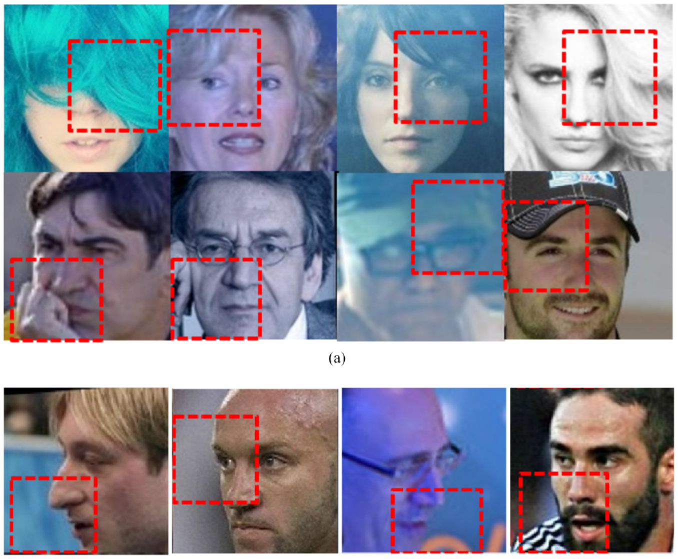
Feature-Net is used for extracting embedded features and supervision information. In this work, we choose two basic networks to verify the accuracy of the experiment, namely, MobileFaceNet [45] and ResNet [46]. In the field of face recognition, MobileFaceNet [45] is the representative of a lightweight network, with extremely high efficiency for real-time face verification on mobile devices. The main building block of MobileFaceNet is the residual bottleneck which consists of convolutions with 1 × 1 kernels and depthwise convolutions with 3 × 3 kernels. The detailed architecture used in our work is consistent with the original paper. ResNet [46] is also a classic network and widely used in face recognition tasks. In our experiment, we have used ResNet50 [46] in which the shortcut was implemented by the max-pool operation or convolutions used 1 × 1 kernels, and have explored the BN-Dropout-FC-BN structure to get the final 512-D embedding feature according to ArcFace [4].

In order to further enhance the generalization ability of the network, we also applied the mask on the intermediate feature layer. This is an extension of PickPatch [13]. Instead of the random selection of face patches, the input face image and the intermediate feature maps are masked by the output of the Attention-Net according to Eqn. (1) in our method. The mask remains consistent in the entire network and is scaled down in the intermediate feature maps. Whether an intermediate layer is masked or not is determined randomly. In particular, for every batch, we randomly selected some layers and masked selected feature maps after the last operation of the residual bottlenecks or residual blocks with a fixed probability, which was set at 0.5. Similar to the DropBlock [20] and dropout [18], the mask is not applied on feature maps during inference. This approximates the result of the exponentially-sized ensembles of sub-networks.

In our experiment, there are two main hyperparameters, namely *patch_radius* and *num_layers*. *patch_radius* is the radius of a patch which is obtained by expanding to the periphery with coordinates as the center. The size of the mask is scaled down by the size of the feature map. As determined by experimental verification, when the radius of the patch was set to 30, the proposed method was found to achieve better performance. *num_layers* is the number of masked layers in the intermediate feature maps. We will subsequently discuss their influence on the results through experiments. Feature-Net was optimized by the additive margin softmax (AMSoftmax) [47]. The feature scale, s, was set to 30 and the angular margin, m, of AMSoftmax [47] was chosen to be 0.4. Next section describes the, experimental results obtained by applying our algorithm to multiple image datasets.

IV. Experiments and Results

We have performed a series of experiments for face recognition in order to verify the effectiveness of our method. Existing face verification datasets were used for the comparison, and the sensitivity of the hyperparameter *num_layers* was investigated. All experiments were implemented using the PyTorch library [48].


(a)

(b)

Fig. 4. The output results of Attention-Net. The Red dotted box represents the discarded face block decided by Attention-Net. (a) Part of the face block is discarded due to occlusion. (b) Part of the face block is discarded due to self-occlusion (pose variation).

The implementation details of the experiment are as follows. In our experiments, we separately employ publicly available VGGFace2 [49], MS1MV2 and DeepGlint-Face [50] as our training data to verify our proposed performance of the algorithm. MS1MV2 and DeepGlint-Face contain 390k and 280k cleaned and aligned data respectively, which are non-overlapping and excluded from LFW [51]. After removing the identities appearing in LFW, the VGGFace2 training set consisted of 8k identities and had approximately 3M images. After detecting their facial landmarks, the faces images were aligned using similarity transformation on the basis of their landmarks. The size of aligned face images was 112 × 112, and each pixel was normalized by subtracting 128 and divided by 128. The networks were trained on TITANX GPUs and the batch size is set to fill entire the GPU memory. During inference, the test input face was still masked by Attention-Net, but the intermediate feature maps no longer did the dropping operation.

A. Attention-Aware Input Images

In this work, the Attention-Net determines whether a face block is used to extract face features or not. Therefore, we first verified the effectiveness of Attention-Net. Figure 4 shows the results.

As shown in Figure 4, for face images having a large-scale occlusion or a self-occlusion due to large pose variation, Attention-Net can capture the parts that have a large negative impact on the recognition performance, and these are then set to zero. This is because Attention-Net is driven by accuracy. Favorable behavior will increase the probability of being selected next time, while unfavorable behavior will be decreased. Sometimes, maybe some face blocks should not be discarded according to the principle of human eyes. However, they are a natural selection in accordance with accuracy.

TABLE I

Face Verification on the LFW Datasets. "Training Set" and "Num of Identities" Indicates Number of Training Set Images and the Number of People Used for All Methods Separately. "*" Indicates it Is Composed of Several Public Datasets and a Private Face Dataset, Which Is Different From the "5M" in the Second Part of the Table. "Models" Indicates the Number of Networks Used for Identification

Method	Training Set	#Image	#Identity	Models	Acc(%)
DeepFace [1]	SFC	$4.4M$	$4K$	3	97.35
FaceNet [6]	-	$200M$	$80K$	1	99.63
DeepID2+ [11]	-	$0.2M$	$12k$	25	99.47
Center Face [7]	CASIA-WebFace, CACD2000, Celebrity+	$0.7M$	$-$	1	99.28
Range Loss [54]	MS-Celeb-1M, CASIAWebFace	$5M$	$100K$	1	99.52
Marginal Loss [55]	MS-Celeb-1M	$4M$	$82K$	1	99.48
Baidu [12]	-	$1.3M$	$18k$	1	99.13
CosFace [8]	several public datasets and a private face dataset	$5M*$	$90k$	1	**99.73**
SphereFace [3]	MS-Celeb-1M	$5M$	$79k$	1	99.57
CosFace [8]	MS-Celeb-1M	$5M$	$79k$	1	99.53
ArcFace [4]	MS-Celeb-1M	$5M$	$79k$	1	99.57
AdaptiveFace [9]	MS-Celeb-1M	$5M$	$79k$	1	**99.62**
PickPatch [13]	VGGFace2	$3M$	$8k$	1	**99.65**
ARFace	VGGFace2	$3M$	$8k$	1	**99.62**

TABLE II

The Results in on BLUFR Protocol. VR(%)@FAR $= 0.1\%$ Indicates Verification Rate in Face Verification. DIR(%)@FAR $= 1\%$ Indicates Rank-1 Detection and Identification Rate in Open Set Identification

Method	Training Set	Num of Identities	VR@FAR=0.01%	DIR@FAR=1%
SphereFace [3]	$5M$	$79k$	99.12	96.72
CosFace [8]	$5M$	$79k$	99.35	97.76
ArcFace [4]	$5M$	$79k$	99.47	98.02
AdaptiveFace [9]	$5M$	$79k$	99.53	**98.19**
PickPatch [13]	$3M$	$8k$	99.72	94.64
ARFace	$3M$	$8k$	**99.83**	**97.21**

In the training phase, Attention-net output a decision on whether a face patch is picked or not by multinomial distribution sampling. In the inference phase, decision is outputted by selecting the maximum value. In the actual testing phase, when registering a face image, it is often not clipped according to experimental statistics. Because the registered face is often required to be a front face with neutral expression. In the experiment, the cropped face often has a large occlusion and pose variation. When the face block has a large occlusion, the feature extracted from this region may be from the occlusion with a large probability, which greatly affects the recognition. Similarly, when the face image has pose variation, that is, the head deflection angle is too large, the feature may be more the background information, which is also bad for recognition. Although it is true that there is a certain degree of information loss when a part of the face is cropped, most of the negative information can be filtered out and the network can pay more attention to the parts that are beneficial to recognition.

B. Test on LFW [51] Database

LFW [51] database is the most commonly used and classical database in the field of face recognition. LFW [51] database contains 13233 images from 5749 different identities, having large variations in pose, expression, and illumination. The standard unrestricted with labeled outside data [52] testing protocol is used, which reports the results on 3000 pairs of positive samples and 3000 pairs of negative samples. The similarity of all face features is given by cosine similarity. Then 10-fold cross validation method is used to calculate the recognition rate. we also test the models on the benchmark of large-scale unconstrained face recognition (BLUFR) [53] protocol which uses all the 13233 images. The BLUFR protocol involving both the verification and open-set identification scenarios is more challenging than the original LFW testing protocol. The BLUFR protocol made 10-fold experiments with 156,915 genuine matching and 46,960,863 impostor matching. The average of the verification rate(VR) and the detection and identification rate(DIR) were calculated as the final result. We report the results in Table I and Table II.

We compared the accuracy of our method with the current popular face recognition methods. Since we did not reproduce all of the above methods, the results in the first part of the Table I are from the original literature. In the second part of Table I and Table II, SphereFace [3], CosFace [8], ArcFace [4], AdaptiveFace [9] were implemented using the same ResNet50 network, which is consistent with network structure of the Feature-Net. And the results obtained are from AdaptiveFace [9]. Since the sizes of the training set are different for all results, we list the number of training set images and the number of people used for all methods. The important thing to note here is that CosFace in the first part of Table I used contains 5M images. "*" indicates it is composed of several public datasets and a private face dataset.

In ARFace, we set *num_layers* $= 9$. This is the best setting for the hyperparameters in the current experimental setup obtained after verification. As can be seen from the tables, ARFace achieved 99.62% in the case of the LFW standard test protocol, which is comparable performance with the current state-of-the-art methods [9] and PickPatch [13]. However, LFW standard protocol is relatively simple and use limited number of impostor matches, so to further test the

TABLE III
FACE VERIFICATION ON THE YTF DATASET

Method	Training Set	LFW(%)	YTF(%)
DeepFace [1]	$4.4M$	97.35	91.40
FaceNet [6]	$200M$	99.63	95.10
VGG Face [59]	$2.6M$	98.95	97.30
DeepID2+ [11]	$0.2M$	99.47	93.20
Center Face [7]	$0.7M$	99.28	94.90
Range Loss [54]	$1.5M$	99.52	93.70
Marginal Loss [55]	$4M$	99.48	95.98
SphereFace [3]	$0.5M$	99.42	95.00
CosFace [8]	$5M*$	**99.73**	**97.60**
PickPatch [13]	$3M$	99.65	97.30
ARFace	$3M$	**99.62**	**97.54**

TABLE IV
FACE VERIFICATION ON THE CALFW AND CPLFW

Method	LFW(%)	CALFW(%)	CPLFW(%)
VGGFace2 [49]	99.43	90.57	84.00
Center Face [7]	98.75	85.48	77.48
SphereFace [3]	99.27	90.30	81.40
PickPatch [13]	99.65	91.07	88.85
ARFace	**99.62**	**91.92**	**89.20**

performance of algorithm, we made testing on more difficult Blufr test protocol. Although ARFace achieved 97.21% and was slightly inferior to AdaptiveFace in BLUFR open-set tests. But our approach use fewer data resources. AdaptiveFace [9] was trained with 5M images from 79k identities, while only 3.8M images from 8,629 identities were used to train ARFace. Especially in the number of training identities, this advantage is more clearly expressed. In addition, when using the same training data set, we increase the recognition rate from 94.64% to 97.21% compared to PickPatch in BLUFR open-set tests. That is sufficient to prove the effectiveness of our method.

C. Test on the Datasets Including Large Intra-Class Variation

The second type of datasets contains large intra-class variation. The YouTube Face [56] dataset consists of 3,425 videos from 1,595 subjects, with an average of 181.3 frames per video clip. As a video-based face recognition database, the YTF [56] dataset contains a large number of low resolution and motion blur images, which greatly increases the difficulty of their verification. The videos were divided into 5,000 pairs of samples, half of which were from the same person. Similar to the LFW testing protocol, the 10-fold cross validation method was used for calculating the recognition rate. Besides YTF datasets, we also verify the performance of the algorithm on the recently introduced datasets, namely, Cross-Age LFW (CALFW) [57] and Cross-Pose LFW [58]. Based on same identities from LFW, CALFW deliberately searches and selects 3,000 positive face pairs with age gaps in order to add the aging process to the intra-class variance. Unlike CALFW, CPLFW focuses on the pose difference from the same individual in order to add the pose variation to the intra-class variance. We report the result separately in Table III and Table IV.

We compared the performance of algorithm with the current popular face recognition methods on the YTF [56], CALFW [57] and CPLFW [58]. It is important to note that all the results are from the original literature. That is, other methods that were not tested on these databases do not appear in the table, for example AdaptiveFace [9]. In order to compare the methods with greater clarity, we also list the test results on the LFW in the table. As can be seen from the table, assuming that the test results on LFW are not significantly different, ARFace obtained 97.54% face verification rate on the YTF dataset, which is outperforms other algorithms such as VGGFace [59], DeepID2+ [11], SphereFace [3], Center Face [7] and PickPatch [13]. Same as mentioned above, the dataset of CosFace used contains 5M images. It is composed of several public datasets and a private face dataset, containing about more than 90K identities. Using fewer data resources, our algorithm is observed to achieve a slightly worse result. But ARFace is far superior to other algorithms, which shows the superiority of ARFace.

TABLE V
FACE IDENTIFICATION AND VERIFICATION EVALUATION ON THE MEGAFACE CHALLENGE 1. ID(%) REFERS TO TOP-1 IDENTIFICATION RATE AT 1M DISTRACTORS AND VERI REFERS TO FACE VERIFICATION AT $\mathit{FAR} = 10^{-}6$ 'R' REFER TO DATA REFINEMENT BY NOISES LIST

Method	Protocol	Id(%)	Veri(%)
FaceNet [6]	Large	70.49	86.47
DeepSense V2	Large	81.29	95.99
Vocord deepVo V3	Large	91.76	94.96
CosFace [8]	Large	82.72	96.65
Softmax	Large	71.36	73.04
SphereFace-R [3]	Large	92.41	93.42
CosFace-R [8]	Large	93.94	94.11
ArcFace-R [4]	Large	94.63	94.85
AdaptiveFace-R [9]	Large	95.02	95.60
PickPatch-R [13]	Large	93.63	94.83
ARFace-R	Large	**96.40**	**96.75**

In Table IV, the test results obtained by applying several algorithms on the CPLFW and CALFW datasets are given. The test results on obtained using the LFW dataset also presented for the sake of comparison. The results listed in Table IV indicate that our method exhibits better superior performance than its competitors such as VGGFace2 [49], Center Face [7], SphereFace [3] and PickPatch [13], when applied to both the CPLFW and CALFW datasets. It is especially noteworthy that ARFace shows a significantly improved performance when applied to the CPLFW dataset, achieving an accuracy rate of 89.20%. This is because our method is naturally adaptive to the occlusion and pose variation. Compared with randomly discarding face blocks, ARFace is more targeted. It can discard the patches that have a negative impact on recognition and focus on the other parts to obtain the discriminating features.

D. Test on Challenging Large-Scale Databases

We tested the performance of algorithm on the very challenging large-scale databases, including the MegaFace [60], IARPA Janus Benchmark-B(IJB-B), IJB-C [61] and Trillion-Pairs dataset.

The MegaFace dataset is the commonly used testing benchmark for face identification and verification at the million-scale. The MegaFace dataset [60] has a collection of 1M images of 690k different individuals as distractors. The probe set consists of 100k images of 530 unique individuals from the FaceScrub dataset [62] and 1002 face images of 82 individuals from the FGNet dataset. In this paper, we use FaceScrub [62] as the probe set, which is also consistent with ArcFace [4] and AdaptiveFace [9]. Corresponding to the two test protocols, MegaFace has two tasks, identification and verification. The large test protocol is defined to have the training set containing more than 0.5M images.

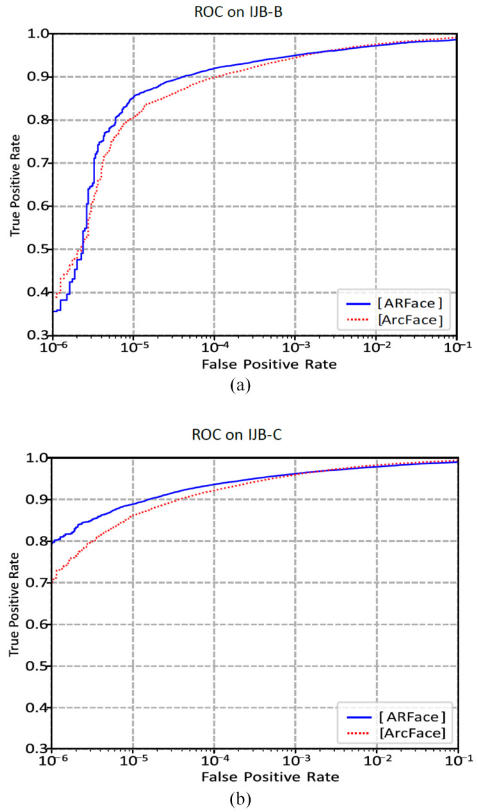
In this experiment, a large-scale dataset was used to train our algorithm, namely MS1MV2 and DeepGlint-Face provided by Trillion-Pairs [50], which consist of 5.8M images from 180k identities. According to the ArcFace [4] and AdaptiveFace [9], hence we use the noise list proposed by ArcFace [4] to clean it. The results are shown in Table V. To be fair, we focused on the results of removing the noise data. SphereFace [3], CosFace [8], ArcFace [4], AdaptiveFace [9] are implemented using the same ResNet50, which is consistent with our experimental setup. Table V shows that our method obtained 96.40% and 96.75%, which is state-of-the-art result under identification and verification scenarios separately.

The IJB-B dataset [61] is composed of face images having a huge variety of facial postures and drastic changes in illumination under completely unconstrained environment and are used for face detection and recognition. The dataset introduces the concept of template, which includes still images and video clips of the subject. Face verification is based on a template rather than a single image. The IJB-B dataset [61] contains 1, 845 subjects with 21.8k still images and 55k frames from 7, 011 videos. The IJB-C dataset [61] is an extension of IJB-B, having 3, 531 subjects with 31.3k still images and 117.5k frames from 11, 779 videos. There are 12, 115 templates and 23, 124 templates in the IJB-B and the IJB-C respectively. In Table VI, we compare the true acceptance rate (*TAR*) for a 1:1 verification protocol.

To be consistent with the baseline methods [4], we have employed the VGGFace2 dataset as the training data and the ResNet50 network was used to extract 512-D embedding feature. As can be seen from Table VI and Figure 5, ARFace shows an improves improvement from 90.8% to 92.0% at $FPR = 1 \times 10^{-4}$. Moreover, our algorithm has a higher pass rate in the case of low error rate, which can be seen more intuitively in Figure 5.

The Trillion-Pairs dataset uses 274k images from 5.7k images of the LFW [51] database as the probe set and also provides a 1.58M images from Flickr as distractors. Evaluation is conducted via the identification and verification tasks respectively. In the verification task, every pair between the probe and the distractors is used and there are 0.4 trillion pairs in total. Then the true positive rate, $TPR@FPR = 1 \times 10^{-9}$ will be used as evaluation indicator. In the identification task, the $TPR@FPR = 1 \times 10^{-3}$ is obtained on the 1.58 million-size gallery and a 270k sized query. The results obtained in the Table VII.

In this experiment, we still use ResNet50 network as the Feature-Net. The baseline is the ResNet50 network trained using VGGFace2, MS1MV2 and DeepGlint-Face separately datasets. As can be seen from the results, our method always exhibits some superior performance even on such a large test data set, which further proves the effectiveness of our method. Compared with PickPatch, ARFace showed a huge improvement from 20.17% to 49.44% to on the limited dataset. In addition, based on the test results of the above three databases, it can be seen that ARFace is significantly better than Pickpatch in large-scale databases. This is because large datasets contain more noise, and ARFace eliminates noise as much as possible, not just as a regularization method.



(a)

(b)

Fig. 5. ROC curves on the IJB-B and IJB-C dataset. (a) IJB-B. (b) IJB-C.

E. Test on Large Pose and Occlusion Datasets

In order to verify the robustness for the face pose changes, we tested ARFace on the Celebrities in Frontal-Profile in the Wild (CFP) [71]. The CFP dataset consists of images of 500 celebrities in frontal and profile views and each subject has about 10 frontal and 4 side images. The test results are shown in Table VIII.

We also carried out the recognition test for the occluded face. And the accuracy was evaluated on the Real-world Masked Face Recognition Dataset (RMFRD) [66]. RMFRD is one of the richest real-world masked face datasets. It contains 5,000 images of 525 subjects with masks, and 90,000 images without masks which represent 525 subjects. The test results are shown in Table IX.

From Table VIII we may see that ARFace can achieve superior recognition performance, which is almost the same as ArcFace, on CFP dataset. And in the test for the occluded face ARFace shows the best performance on the RMFRD compared with the common methods as shown in Table IX.

TABLE VI
1:1 VERIFICATION TAR AT DIFFERENT FAR ON THE IJB-B DATASET AND IJB-C DATASET

Method	TAR on IJB-B dataset			TAR on IJB-C dataset		
	FAR=0.0001	FAR=0.001	FAR=0.01	FAR=0.0001	FAR=0.001	FAR=0.01
ResNet50 [49]	0.784	0.878	0.938	0.825	0.900	0.950
SENet50 [49]	0.800	0.888	0.949	0.840	0.910	0.960
MN-v [63]	0.818	0.902	0.955	0.852	0.920	0.965
MN-vc [63]	0.831	0.909	0.958	0.862	0.927	0.968
ResNet50+DCN(Kpts) [64]	0.850	0.927	0.970	0.867	0.940	0.979
ResNet50+DCN(Divs) [64]	0.841	0.930	0.972	0.880	0.944	0.981
SENet50+DCN(Kpts) [64]	0.846	0.935	0.974	0.874	0.944	0.981
SENet50+DCN(Divs) [64]	0.849	0.937	0.975	0.885	0.947	**0.983**
ArcFace [4]	0.898	0.944	**0.975**	0.921	0.959	0.982
PickPatch [13]	0.908	0.934	0.967	0.921	0.947	0.973
ARFace	**0.920**	**0.950**	0.972	**0.936**	**0.961**	0.978

TABLE VII
THE PERFORMANCE(%) ON THE TRILLION-PAIRS TEST

Method	Id@FPR=1e-3	Ver@FPR=1e-9
ResNet50+VGGFace2	15.06	14.71
PickPatch+VGGFace2	20.17	18.20
ARFace+VGGFace2	**49.44**	**47.85**
ResNet50+MS1MV2,DeepGlint-Face	69.29	69.51
PickPatch+MS1MV2,DeepGlint-Face	69.59	69.74
ARFace+MS1MV2,DeepGlint-Face	**70.32**	**69.83**

TABLE VIII
FACE RECOGNITION EVALUATION ON CFP

Method	Accuracy
Softmax	94.39%
ArcFace	**95.56%**
SphereFace	94.38%
CosFace	95.44%
ARFace	**95.52%**

TABLE IX
FACE RECOGNITION EVALUATION ON RMFRD DATASET

Method	Accuracy
Luttrell et al. [70]	85.7%
Hariri et al. [69]	84.6%
Almabdy et al. [68]	87.0%
Walid Hariri [67]	91.3%
ARFace	**91.8**%

F. Ablation Study

In order to prove the effectiveness of ARFace, we have conducted a large number of ablation experiments to analyze the effects of the attention mechanism and regularization of the middle layer on the recognition performance. In order to eliminate the long tail effect, we used the subset of VGGFace2 [49] as training set, which is consist of 50 images of each identity. Our baseline is the accuracy of the ResNet50 trained using the AMsoftmax loss function. The results are shown in Table X. In this part, we choose LFW as the test set. All the models are evaluated using two different protocols as mentioned above.

As is reported in Table X, the results obtained by employing Attention-Net to the subsets of the VGGFace2 dataset, i.e., the model of *num_layers* = 0 surpasses the baselines by a significant margin in the case of the LFW dataset, and this further demonstrates the effectiveness of Attention-Net. In this case, we only apply dropping operation for the input layer. Moreover, compared to the random selection of face blocks, Attention-Net also shows certain advantages and improvements in the performance of the algorithm in the case of the VGGFace2 subset used for training ResNet, from 98.90% to 99.13%. Attention-Net driven by accuracy is more targeted than random selection, which is just a regularization method. More weightage can be given to the parts of the face that are more useful for recognition when we employing Attention-Net.

Besides Attention-Net, we have discussed the results obtained from models with different *num_layers*. ARFace exhibits a further performance improvement as compared to models using *num_layers* = 0, and this demonstrates the superiority of employing dropping operation in the intermediate feature maps. Experimental results show that MobileFaceNet network using *num_layers* = 3 and ResNet50 network using *num_layers* = 9 achieve the best recognition performance, and thus this setting will be applied in the following experiment.

The ResNet50 network is easy to get caught up in overfitting. In order to make a fair comparison with other methods and to highlight the real performance of our algorithm, we show the results of ResNet50 trained with the whole VGGFace2 dataset in Table X. ARFace is seen to outperform in terms of the baseline values by an obvious margin. We further discuss effect of setting different values for the hyperparameter *num_layers* on recognition performance, as is shown in Figure 6.

As can be seen in Figure 6, MobileFaceNet with *num_layers* = 3 and ResNet50 with *num_layers* = 9 are the optimal model choices for carrying out the validation task. Moreover, for the open-set identification scenarios proposed in the BLUFR protocol, it is a more challenging task. As the *num_layers* increases, the overall identification rate increases and tends to reach a saturation in the later period in ResNet50. However the accuracy decreases significantly when the final convergence occurs in MobileFaceNet. We speculate that this is that it is because the model itself a lightweight model. Even if the degree of regularization increases, it is still limited by the capacity of the model itself. This shows that the regularization method has a greater important in relatively difficult tasks. However the expressiveness of the model is also limited by its size. Therefore, considering that the training and inference time of the model will be longer as the *num_layers* is increased, the most appropriate parameter setting would be *num_layers* = 3 in the MobileFaceNet network and *num_layers* = 9 in the ResNet50 network.

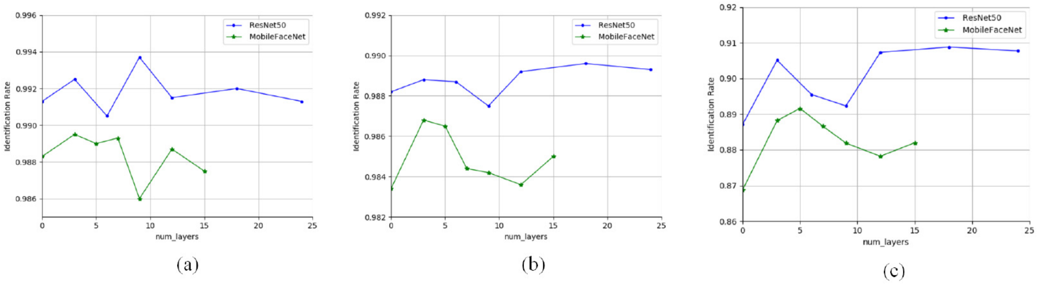


Fig. 6. The effect of hyperparameter on recognition performance. (a) Fold-10 cross validation of LFW. (b) LFW BLUFR VR@FAR = 0.1%. (c) LFW BLUFR DIR@FAR = 1%.

TABLE X
COMPARISON OF ABLATION RESULTS(%)

Architecture	Selecting schema	number of drop layers	LFW 6000pairs/%	BLUFR VR@FAR=0.1%	BLUFR DIR@FAR=1%
MobileFaceNet (VGGFace2 subset)	–	–	98.58	98.20	85.86
	Random	$num\_layers = 0$	98.77	98.52	87.39
	Attention-net	$num\_layers = 0$	98.83	98.34	87.03
	Attention-net	$num\_layers = 3$	**98.95**	**98.68**	88.82
	Attention-net	$num\_layers = 5$	98.90	98.65	**89.16**
ResNet50 (VGGFace2 subset)	–	–	98.97	97.95	80.18
	Random	$num\_layers = 0$	98.90	98.53	84.85
	Attention-net	$num\_layers = 0$	99.13	98.82	88.71
	Attention-net	$num\_layers = 9$	**99.37**	98.75	89.23
	Attention-net	$num\_layers = 12$	99.15	**98.92**	**90.73**
ResNet50 (VGGFace2)	–	–	99.52	99.54	92.04
	Random	$num\_layers = 0$	99.50	99.74	94.88
	Attention-net	$num\_layers = 0$	99.55	99.82	**97.42**
	Attention-net	$num\_layers = 9$	**99.62**	**99.83**	97.21
	Attention-net	$num\_layers = 12$	99.60	99.83	97.01

TABLE XI
COMPARISON WITH OTHER REGULARIZATION(%)

Architecture	regularization method	LFW 6000pairs/%	BLUFR VR@FAR=0.1%	BLUFR DIR@FAR=1%
MobileFaceNet (VGGFace2 subset)	–	98.58	98.20	85.86
	Cutout	98.92	98.78	89.10
	DropBlock	98.73	97.94	83.71
	PickPatch	**99.03**	98.60	88.31
	ARFace	98.95	**98.68**	**88.82**
ResNet50 (VGGFace2 subset)	–	98.97	97.95	80.18
	Cutout	98.97	98.65	85.61
	DropBlock	98.30	95.10	72.58
	PickPatch	99.10	98.46	85.99
	ARFace	**99.37**	**98.75**	**89.23**

G. Comparison With Other Regularization Methods

A number of regularization methods have been proposed to prevent overfitting. In our algorithm, we have applied mask in the intermediate feature layer, which is also a regularization method. In this section, results obtained by applying the MobileFaceNet and ResNet50 networks trained using ARFace were compared with those obtained by employing the cutout [36] and DropBlock [20] to discuss the influence of the regularization method on the performance of face recognition. The other hyperparameter settings were consistent with previous experiments. Data augmentation was achieved using the Cutout [36] by randomly dropping out a square region of input images. According to origin work, we select a cutout size of 16 × 16 pixels. In contrast to the cutout, DropBlock randomly dropped out a square region in some feature maps. We carried out experiments by applying DropBlock to both Groups 3 and 4 from MobileFaceNet and ResNet50, based on the definition of DropBlock [20]. We set *block_size* = 7 and *keep_prob* was gradually decreased from 1 to 0.75 in a linear fashion. Table XI presents the test results.

Table XI show that our method consistently outperforms DropBlock, cutout and PickPatch, which is clearly evident from the results obtained from the ResNet50 network. DropBlock [20] shows competitive performance on ImageNet classification because it drops more semantic information and then learns more discriminative regions. However, for face recognition, it is easy to discard important facial features, and our method alleviates this problem. For every batch, the masked layers are randomly selected,and the information can be complementary in next batch.

V. Conclusion

In this work, ARFace, an attention-aware face recognition method has been introduced. It consists of two components, namely Attention-Net and Feature-Net. By simulating the attention mechanism of the human visual system, the Attention-Net outputs the mask of a face image and is optimized by reinforcement learning. Feature-Net aims to obtain the embedding features from the image and provide supervision signals for the Attention-Net. In addition to the above system, we have also proposed an effective regularization method, in which the mask is scaled down by the size of the intermediate feature maps and employed in the feature maps. Extensive experiment was conducted to compare performance of the proposed method with baseline. The results thus obtained reveal that ARFace achieves competitive face recognition performance.

This work has demonstrated the superiority of our method as well as the feasibility of using the attention mechanism in face recognition. However, we only used five face landmarks as the center of dropping in our experiments. In future, we could even subdivide the face area with ten or more points, which would lead to higher accuracy results. Our work only demonstrates the feasibility of this approach, and Attention-Net based on reinforcement learning can help improve the performance of face recognition to some extent.

References

[1] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "DeepFace: Closing the gap to human-level performance in face verification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Columbus, OH, USA, 2014, pp. 1701–1708.

[2] Y. Sun, X. Wang, and X. Tang, "Deep learning face representation from predicting 10,000 classes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Columbus, OH, USA, 2014, pp. 1891–1898, doi: 10.1109/CVPR.2014.244.

[3] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song, "SphereFace: Deep hypersphere embedding for face recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, 2017, pp. 6738–6746, doi: 10.1109/CVPR.2017.713.

[4] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "ArcFace: Additive angular margin loss for deep face recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Long Beach, CA, USA, 2019, pp. 4685–4694, doi: 10.1109/CVPR.2019.00482.

[5] X. Cheng, J. Lu, B. Yuan, and J. Zhou, "Face segmentor-enhanced deep feature learning for face recognition," *IEEE Trans. Biometr., Behav., Identity Sci.*, vol. 1, no. 4, pp. 223–237, Oct. 2019.

[6] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, 2015, pp. 815–823, doi: 10.1109/CVPR.2015.7298682.

[7] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, "A discriminative feature learning approach for deep face recognition," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2016, pp. 499–515.

[8] H. Wang *et al.*, "CosFace: Large margin cosine loss for deep face recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, 2018, pp. 5265–5274.

[9] H. Liu, X. Zhu, Z. Lei, and S. Z. Li, "AdaptiveFace: Adaptive margin and sampling for face recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Long Beach, CA, USA, 2019, pp. 11939–11948, doi: 10.1109/CVPR.2019.01222.

[10] Y. Sun, Y. Chen, X. Wang, and X. Tang, "Deep learning face representation by joint identification-verification," in *Proc. 27th Adv. Neural Inf. Process. Syst.*, 2014, pp. 1988–1996.

[11] Y. Sun, X. Wang, and X. Tang, "Deeply learned face representations are sparse, selective, and robust," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, 2015, pp. 2892–2900, doi: 10.1109/CVPR.2015.7298907.

[12] J. Liu, Y. Deng, T. Bai, Z. Wei, and C. Huang, "Targeting ultimate accuracy: Face recognition via deep embedding," 2015. [Online]. Available: arxiv.abs/1506.07310.

[13] L. Sun, W. He, X. Ning, W. Li, and Y. Shi, "Pickpatch: A regularization method for deep face recognition," *J. Pharmaceutical Health Care Sci.*, vol. 1487, no. 1, 2020, Art. no. 012024.

[14] X. Min, G. Zhai, and K. Gu, "Visual attention on human face," in *Proc. Vis. Commun. Image Process. (VCIP)*, Singapore, 2015, pp. 1–4.

[15] R. Bellman, "A Markovian decision process," *J. Math. Mech.*, vol. 6, no. 5, pp. 679–684, 1957.

[16] M. L. Littman, "Reinforcement learning improves behaviour from evaluative feedback," *Nature*, vol. 521, no. 7553, pp. 445–451, May 2015, doi: 10.1038/nature14540.

[17] Z. H. Zhou, *Ensemble Methods: Foundations and Algorithms*. London, U.K.: Chapman and Hall, 2012.

[18] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.

[19] J. Tompson, R. Goroshin, A. Jain, Y. LeCun, and C. Bregler, "Efficient object localization using convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Boston, MA, USA, 2015, pp. 648–656, doi: 10.1109/CVPR.2015.7298664.

[20] G. Ghiasi, T.-Y. Lin, and Q. V. Le, "DropBlock: A regularization method for convolutional networks," in *Proc. 31st Adv. Neural Inf. Process. Syst.*, 2018, pp. 10727–10737.

[21] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, 2018, pp. 7132–7141, doi: 10.1109/CVPR.2018.00745.

[22] Y. Cao, J. Xu, S. Lin, F. Wei, and H. Hu, "GCNet: Non-local networks meet squeeze-excitation networks and beyond," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshop (ICCVW)*, Seoul, South Korea, 2019, pp. 1971–1980, doi: 10.1109/ICCVW.2019.00246.

[23] D. Linsley, D. Shiebler, S. Eberhardt, and T. Serre, "Learning what and where to attend," Jun. 2019. [Online]. Available: arXiv:1805.08819.

[24] F. Wang *et al.*, "Residual attention network for image classification," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, 2017, pp. 6450–6458, doi: 10.1109/CVPR.2017.683.

[25] W. Zheng, M. Yue, S. Zhao, and S. Liu, "Attention-based spatial–temporal multi-scale network for face anti-spoofing," *IEEE Trans. Biometr., Behav., Identity Sci.*, vol. 3, no. 3, pp. 296–307, Jul. 2021.

[26] B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba, "Learning deep features for discriminative localization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, 2016, pp. 2921–2929, doi: 10.1109/CVPR.2016.319.

[27] H. Fukui, T. Hirakawa, T. Yamashita, and H. Fujiyoshi, "Attention branch network: Learning of attention mechanism for visual explanation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Long Beach, CA, USA, 2019, pp. 10697–10706, doi: 10.1109/CVPR.2019.01096.

[28] W. Li, X. Zhu, and S. Gong, "Harmonious attention network for person re-identification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, 2018, pp. 2285–2294, doi: 10.1109/CVPR.2018.00243.

[29] W. Yang, H. Huang, Z. Zhang, X. Chen, K. Huang, and S. Zhang, "Towards rich feature discovery with class activation maps augmentation for person re-identification," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Long Beach, CA, USA, 2019, pp. 1389–1398, doi: 10.1109/CVPR.2019.00148.

[30] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "CBAM: Convolutional block attention module," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 3–19.

[31] B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le, "Learning transferable architectures for scalable image recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Salt Lake City, UT, USA, 2018, pp. 8697–8710, doi: 10.1109/CVPR.2018.00907.

[32] H. Pham, M. Y. Guan, B. Zoph, Q. V. Le, and J. Dean, "Efficient neural architecture search via parameter sharing," Feb. 2018. [Online]. Available: arXiv:1802.03268.

[33] M. Tan *et al.*, "MnasNet: Platform-aware neural architecture search for mobile," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Long Beach, CA, USA, 2019, pp. 2815–2823, doi: 10.1109/CVPR.2019.00293.

[34] M. Tan and Q. V. Le, "EfficientNet: Rethinking model scaling for convolutional neural networks," Nov. 2019. [Online]. Available: arXiv:1905.11946.

[35] E. D. Cubuk, B. Zoph, D. Mane, V. Vasudevan, and Q. V. Le, "AutoAugment: Learning augmentation strategies from data," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Long Beach, CA, USA, 2019, pp. 113–123, doi: 10.1109/CVPR.2019.00020.

[36] T. DeVries and G. W. Taylor, "Improved regularization of convolutional neural networks with cutout," Nov. 2017. [Online]. Available: arXiv:1708.04552.

[37] H. Zhang, C. Moustapha, D. N. Yann, and L. David, "MIXUP: Beyond empirical risk minimization," in *Proc. 6th Int. Conf. Learn. Represent. (ICLR)*, Vancouver, BC, Canada, 2018, p. 9.

[38] J. Lemley, S. Bazrafkan, and P. Corcoran, "Smart augmentation learning an optimal data augmentation strategy," *IEEE Access*, vol. 5, pp. 5858–5869, 2017, doi: 10.1109/ACCESS.2017.2696121.

[39] G. Huang, Y. Sun, Z. Liu, D. Sedra, and K. Weinberger, "Deep networks with stochastic depth," in *Proc. ECCV*, 2016, pp. 646–661.

[40] I. J. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014, pp. 2672–2680.

[41] X. Gastaldi, "Shake-shake regularization," May 2017. [Online]. Available: arxiv:1705.07485.

[42] Y. Yamada, M. Iwamura, T. Akiba, and K. Kise, "Shakedrop regularization for deep residual learning," *IEEE Access*, vol. 7, pp. 186126–186136, 2019, doi: 10.1109/ACCESS.2019.2960566.

[43] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Santiago, Chile, 2015, pp. 1026–1034, doi: 10.1109/ICCV.2015.123.

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

[45] S. Chen, Y. Liu, X. Gao, and Z. Han, "Mobilefacenets: Efficient CNNs for accurate real-time face verification on mobile devices," in *Proc. Chin. Conf. Biometric Recognit.*, 2018, pp. 428–438.

[46] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, 2016, pp. 770–778.

[47] F. Wang, W. Liu, H. Liu, and J. Cheng, "Additive Margin Softmax for Face Verification," *IEEE Signal Process. Lett.*, vol. 25, no. 7, pp. 926–930, Jul. 2018.

[48] A. Paszke *et al.*, "PyTorch: An imperative style, high-performance deep learning library," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 8026–8037.

[49] Q. Cao, L. Shen, W. Xie, O. M. Parkhi, and A. Zisserman, "VGGFace2: A dataset for recognising faces across pose and age," in *Proc. 13th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG)*, Xi'an, China, 2018, pp. 67–74, doi: 10.1109/FG.2018.00020.

[50] *Trillionpairs*. [Online]. Available: http://trillionpairs.deepglint.com/overview

[51] G. B. Huang, M. Mattar, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database forstudying face recognition in unconstrained environments," in *Proc. Workshop Faces Real Life Images Detect. Alignment Recognit.*, Oct. 2008, p. 9.

[52] G. B. Huang and E. Learned-Miller, "Labeled faces in the wild: Updates and new reporting procedures," Dept. Comput. Sci., Univ. Massachusetts Amherst, Amherst, MA, USA, Rep. 14-003, 2014.

[53] S. Liao, Z. Lei, D. Yi, and S. Z Li, "A benchmark study of large-scale unconstrained face recognition," in *Proc. IEEE Int. Joint Conf. Biometr.*, 2014, pp. 1–8.

[54] X. Zhang, Z. Fang, Y. Wen, Z. Li, and Y. Qiao, "Range loss for deep face recognition with long-tailed training data," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, 2017, pp. 5419–5428.

[55] J. Deng, Y. Zhou, and S. Zafeiriou, "Marginal loss for deep face recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Honolulu, HI, USA, 2017, pp. 2006–2014.

[56] L. Wolf, T. Hassner, and I. Maoz, "Face recognition in unconstrained videos with matched background similarity," in *Proc. CVPR*, 2011, pp. 529–534, doi: 10.1109/CVPR.2011.5995566.

[57] T. Zheng, W. Deng, and J. Hu. (2017). *Cross-Age LFW: A Database for Studying Cross-Age Face Recognition in Unconstrained Environments*. [Online]. Available: http://arxiv.org/abs/1708.08197.

[58] T. Zheng and W. Deng, "Cross-pose LFW: A database for studying cross-pose face recognition in unconstrained environments," Beijing Univ. Posts Telecommun., Beijing, China, Rep. 18-01, Feb. 2018.

[59] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," in *Proc. Brit. Mach. Vis. Conf.*, 2015, pp. 41.1–41.12, doi: 10.5244/C.29.41.

[60] I. Kemelmacher-Shlizerman, S. M. Seitz, D. Miller, and E. Brossard, "The MegaFace benchmark: 1 million faces for recognition at scale," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, 2016, pp. 4873–4882.

[61] C. Whitelam *et al.*, "IARPA janus benchmark-B face dataset," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Honolulu, HI, USA, 2017, pp. 592–600.

[62] H. Ng and S. Winkler, "A data-driven approach to cleaning large face datasets," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Paris, France, 2014, pp. 343–347.

[63] W. Xie and A. Zisserman, "Multicolumn networks for face recognition," in *Proc. BMVC*, 2018, p. 111.

[64] W. Xie, L. Shen, and A. Zisserman, "Comparator networks," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2018, pp. 782–797.

[65] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," *Mach. Learn.*, vol. 8, nos. 3–4, pp. 229–256, 1992.

[66] Z. Wang *et al.*, "Masked face recognition dataset and application," 2020. [Online]. Available: arXiv:2003.09093.

[67] H. Walid, "Efficient masked face recognition method during the COVID-19 pandemic," 2021. [Online]. Available: arXiv:2105.03026.

[68] S. Almabdy and L. Elrefaei, "Deep convolutional neural network-based approaches for face recognition," *Appl. Sci.*, vol. 9, no. 20, p. 4397, 2019.

[69] W. Hariri, H. Tabia, N. Farah, A. Benouareth, and D. Declercq, "3D face recognition using covariance based descriptors," *Pattern Recognit. Lett.*, vol. 78, pp. 1–7, Jul. 2016.

[70] J. Luttrell *et al.*, "A deep transfer learning approach to fine-tuning facial recognition models," in *Proc. 13th IEEE Conf. Ind. Electron. Appl. (ICIEA)*, 2018, pp. 2671–2676.

[71] S. Sengupta, J.-C. Chen, C. Castillo, V. M. Patel, R. Chellappa, and D. W. Jacobs, "Frontal to profile face verification in the wild," in *Proc. WACV*, 2016, p. 1–9.

Liping Zhang (Member, IEEE) received the Ph.D. degree from the Institute of Semiconductors, Chinese Academy of Sciences in 2018. She is currently an Assistant Research Fellow of the Laboratory of High-Speed Circuit and Artificial Neural Networks, Institute of Semiconductors, Chinese Academy of Sciences. Her research interests include biometrics and pattern analysis.

Linjun Sun (Student Member, IEEE) received the B.E. degree from the School of Physics, Huazhong University of Science and Technology, China, in 2013. He is currently pursuing the Ph.D. degree with the Institute of Semiconductors, Chinese Academy of Sciences. His main research interests are computer vision and deep learning.

Lina Yu (Member, IEEE) received the Ph.D. degree from the College of Information and Electrical Engineering, China Agricultural University in June 2016. From July 2016 to July 2018, she was a Postdoctoral Research Fellow of the Laboratory of High-speed Circuit and Artificial Neural Networks, Institute of Semiconductors, Chinese Academy of Sciences, where she is currently an Assistant Research Fellow. Her researches focus on machine learning, deep modeling, and intelligent system.

Xiaoli Dong (Member, IEEE) received the Ph.D. degree from the Institute of Semiconductors, Chinese Academy of Sciences in 2018. She is currently an Assistant Research Fellow of the Laboratory of High-Speed Circuit and Artificial Neural networks, Institute of Semiconductors, Chinese Academy of Sciences. Her research interests include image processing and pattern recognition.

Jinchao Chen (Member, IEEE) received the Ph.D. degree in computer science from Northwestern Polytechnical University, Xi'an, China, in 2016, where he is an Assistant Professor with the School of Computer Science. He has published more than 30 papers in journals and refereed conferences. He focuses on the multiprocessor scheduling, embedded and real-time systems, simulation and verification, decision-making, and intelligent control of unmanned aerial vehicles. He is a member of CCF.

Weiwei Cai (Graduate Student Member, IEEE) is currently pursuing the master's degree with the Central South University of Forestry and Technology, Changsha, China. Prior to that, he worked in IT industry for more than ten years in the roles of an Enterprise Architect and a Program Manager. His research interests include machine learning, deep learning, and computer vision.

Chen Wang received the B.E. degree from the School of Computer Science, Northwestern Polytechnical University in 2013. He is currently pursuing the Ph.D. degree with the School of Computer Science, Beihang University. His research interests include computer vision, stereo matching, 3-D reconstruction, and camera localization.

Xin Ning (Member, IEEE) received the B.S. degree in software engineering in 2012, and the Ph.D. degree in electronic circuit and system from the University of Chinese Academy of Sciences in 2017. He is currently an Associate Professor with the Laboratory of Artificial Neural Networks and High Speed Circuits, Institute of Semiconductors, Chinese Academy of Sciences. He has published by first or corresponding author more than 40 papers in journals and refereed conferences. His current research interests include pattern recognition, computer vision, and image processing.