Robust Masked Face Recognition via Balanced Feature Matching

Yu-Chieh Huang†
Graduate Institute of Communication Engineering
National Taiwan University
Taipei, Taiwan
r09942076@ntu.edu.tw

Lin-Hsi Tsao†
Department of Electrical Engineering
National Taiwan University
Taipei, Taiwan
b06901045@ntu.edu.tw

Homer H. Chen
Department of Electrical Engineering
National Taiwan University
Taipei, Taiwan
homer@ntu.edu.tw

***Abstract*—Wearing a facial mask has become a must in our daily life due to the global COVID-19 pandemic. However, the performance of conventional face recognition systems severely degrades for faces occluded by masks. How to combat the effect of occlusion on face recognition is an important issue. However, the performance of existing methods developed for masked face recognition unpleasantly degrades when dealing with unmasked faces. To address this issue for real-world applications, where the gallery image or the probe image may be a masked or unmasked face, we propose the concept of balanced facial feature matching and, based on it, design a robust masked face recognition system. The matching is balanced because it is performed on features extracted from corresponding facial regions. The system consists of a classification network and two feature extractors. The classification network classifies an input face image into a masked face or an unmasked face. One feature extractor extracts the feature of a full face, and the other uses a guided perceptual loss to focus the feature extraction on the non-occluded part of the face. The system is tested on both synthetic and real data. The face verification accuracy is improved by 2.4% for the synthetically masked LFW dataset, 1.9% for the MFR2 dataset, and 5.4% for the RMFD dataset. The results further show that the system improves masked face recognition while preserving the performance of unmasked face recognition.**

***Keywords*—Face Recognition System, Masked Face, Mask Occlusion**

I. Introduction

Face recognition has been widely adopted for user authentication, intelligent punch card, video surveillance, etc. It makes our life easier. Take user authentication as an example, face recognition automates the authentication process without requiring the user to memorize any long, complex password. The recent advances of deep learning further enable face recognition systems, such as DeepFace [7], DeepIDs [8], FaceNet [9], CosFace [2], ArcFace [3], to achieve nearly 100% verification accuracy for clean faces without mask. However, in the mist of the worldwide COVID-19 pandemic, wearing a facial mask has become a must in our daily life. Faces covered with facial masks are no longer clean faces. Such facial masks occlude the mouth and nose regions and make mouth and nose features unavailable for face recognition. In addition, the presence of the facial mask in either the gallery image or the probe image results in an imbalanced matching, where a masked face is matched against an unmasked face. Due to

†denotes equal contribution

Fig. 1. Three cases of matching.

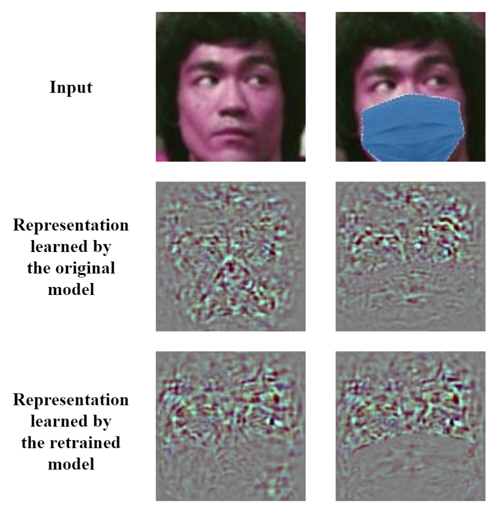

Fig. 2. Visualization of feature representations learned by the original and retrained ArcFace model. The first row shows the input face images. The second and third rows illustrate the activated regions learned by the original model and the retrained model, respectively.

these reasons, the conventional face recognition approach that was developed for clean faces cannot work well for masked faces.

Masked face recognition targets faces with facial masks. However, most existing masked face recognition methods directly retrain a model by adding synthetically masked faces [4], [5], [12] to the training data. A major drawback of this approach is that it tends to only focus on features of the upper face, making the performance of the retrained model degrades when the input is a clean face without a mask.

Therefore, to work for both masked and unmasked faces, a face recognition system must consider three cases of matching, as shown in Fig 1. Case 1 represents the case where two unmasked faces are to be matched, Case 2 represents the case where a masked face is to be matched against an unmasked face, and Case 3 represents the case where two masked faces are to be matched. As an illustration of the three cases of feature matching, Fig 2. shows the visualization of the feature representations of a masked and an unmasked face learned by a retrained ArcFace model and the original model. The visualization is obtained via Guided-Backpropagation [11]. It can be observed that the original model focuses on the entire region of an unmasked face and the upper (or uncovered) region of a masked face. As a result, an imbalanced matching is unavoidable for Case 2. Despite the fact that the original model focuses on the upper region of a masked face, the features it extracted are not as rich as the ones extracted by the retrained model. Therefore, the original model is less powerful than the retrained model in Case 3. On the other hand, the retrained model is not as effective as the original model in Case 1 because it focuses on the upper region and misses the details in the bottom region of a face.

The primary goal of this work is to develop a robust face recognition system to handle masked faces without sacrificing its performance for unmasked faces. That is, we want our model to work for all three cases discussed above. Both synthetically masked faces and original unmasked faces are used as training data, and a guided perceptual loss function is applied to the features extracted from unmasked and masked faces to achieve a balanced feature matching. In addition, a classification network is applied to distinguish between masked and unmasked faces and makes the face recognition system as effective as possible for all three cases.

The contributions of this paper are threefold:

- The proposed concept of balanced facial feature matching leads to a robust masked face recognition system.
- The resulting face recognition system enhances the face verification accuracy for masked faces without sacrificing its performance for unmasked faces.
- Our method outperforms the state-of-the-art technique.

The rest of the paper is organized as follows. Section II reviews related work on conventional face recognition and masked face recognition. Section III describes the details of the proposed method including the training stage and the testing stage. Section IV discusses experimental results, and Section V concludes the paper.

II. RELATED WORK

A. Conventional Face Recognition for Unmasked Faces

With advances in deep learning, recent face recognition models are mostly based on Deep Convolution Neural Networks [6]. Given a face as input, such models aim to extract the essential facial feature representation with small intra-class but large inter-class distances. Different loss functions such as angular softmax loss [1], large margin cosine loss [2], and additive angular margin loss [3], etc., have been proposed to ensure that the recognition models can extract the above-mentioned facial features. Despite the outstanding recognition performance for clean faces, these models fail to recognize faces with facial masks, perhaps because conventional face recognition systems extract facial features from all parts of a face, including mouth and nose. If the mouth or nose is occluded by a facial mask, the recognition performance degrades.

B. Masked Face Recognition

To prevent the effect of facial masks on the performance of a face recognition system, methods for the recognition of masked faces were proposed. Data augmentation was applied to face recognition by synthesizing masked faces from an existing face dataset and using the resulting faces to retrain the face recognition model [4]. It was found helpful to incorporate the probability of mask-wearing into the face recognition model for masked face recognition [5]. The cropping and attention-based approach was developed to directly remove the bottom half of the face from face recognition [17]. However, these methods are not as practical as expected since they fail to preserve the original performance for unmasked faces. Therefore, a robust face recognition system that performs well for both masked and unmasked faces is needed.

III. PROPOSED METHOD

In this section, we describe the details of the proposed method, which is based on the concept of balanced feature matching.

A. Problem Definition and Annotations

To begin with, we define the notations used in this work. We denote the set of unmasked face images by $X = \{x_1, x_2, ..., x_N\}$ and the corresponding set of synthetically masked face images by $\hat{X} = \{\hat{x}_1, \hat{x}_2, ..., \hat{x}_N\}$, where N represents the number of training images. Furthermore, we denote the corresponding set of identity labels by $Y = \{y_1, y_2, ..., y_N\}$ and the number of identity classes by M.

As shown in Fig. 3, our system mainly consists of a mask classifier C, a full-face feature extractor E_F, and a partial-face feature extractor E_P. The classifier C classifies an input face image into a masked face or an unmasked face. The extractor E_F extracts the feature of a face from the full-face region. Then, the subsequent full-face ArcFace layer LR_F is designed to calculate the Additive Angular Margin Loss (ArcFace) [3] based on the extracted feature $E_F(X)$. To achieve the balanced feature matching, the extractor E_P extracts the feature from

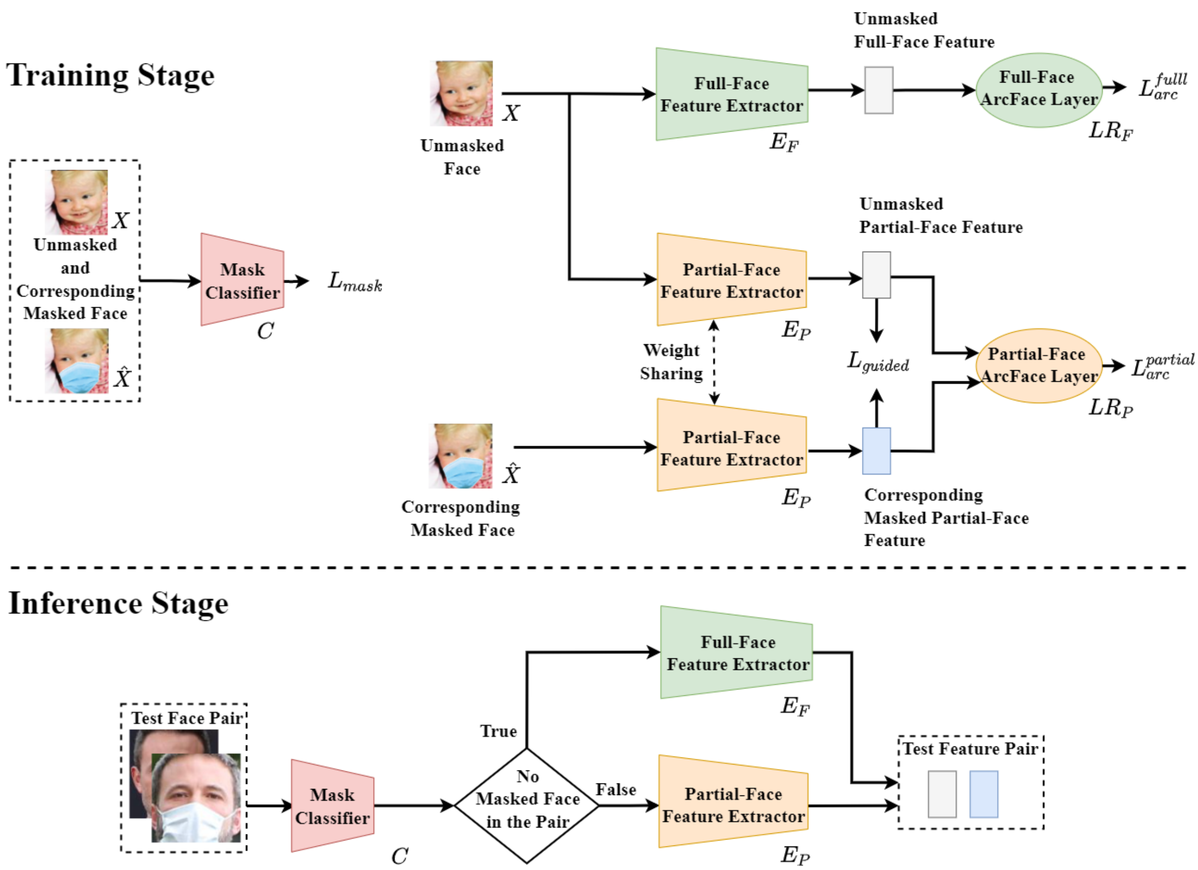

Fig. 3. Overview of the proposed method in the training and inference stages.

the upper part of a face image using a guided perceptual loss as the constraint. The partial-face ArcFace layer LR_P is utilized to calculate the ArcFace loss based on the extracted feature $E_P(\hat{X})$. Once the training process is completed, the mask classifier C, full-face feature extractor E_F, and partial-face feature extractor E_P are used for inference.

B. Learning of the Mask Classifier

We aim at training the mask classifier C to detect if the person in the input face image wears a mask or not. The output of the classifier is used to clearly distinguish the three cases of matching. The softmax loss is adopted here to calculate the *mask classification loss* L_{mask} as follows:

$$L_{mask} = -\frac{1}{B}\sum_{i=1}^{B}(log(\frac{e^{C(x_i)_1}}{e^{C(x_i)_1}+e^{C(x_i)_2}}) + log(\frac{e^{C(\hat{x}_i)_2}}{e^{C(\hat{x}_i)_2}+e^{C(\hat{x}_i)_1}})), \tag{1}$$

where B is the batch size, $C(\cdot)_1$ denotes the 1st class (i.e., unmasked face) output logit for the input face, while $C(\cdot)_2$ denotes the 2nd class (i.e., masked face) output logit.

C. Learning of the Full-Face Feature Extractor

The full-face feature extractor E_F is designed to handle Case 1 matching. For an unmasked face image x_i, the extractor E_F extracts the feature of the image from the entire face. We apply the Additive Angular Margin Loss of ArcFace [3] to enable the extractor E_F to learn identification information for different people. The formula of the *full-face ArcFace loss* L_{arc}^{full} is

$$L_{arc}^{full} = -\frac{1}{B}\sum_{i=1}^{B}log(\frac{e^{s(cos(\theta_{y_i}+m,i))}}{e^{s(cos(\theta_{y_i}+m,i))}+\sum_{j=1,j\neq y_i}^{M}e^{s(cos(\theta_j,i))}}), \tag{2}$$

subject to

$$cos(\theta_j+m,i) = \|W_j^F\|\|E_F(x_i)\|cos(\theta_j+m), \tag{3}$$

where B is the batch size, W_j^F denotes the j-th column of the weight vector of the full-face ArcFace layer LR_F, $E_F(x_i)$ represents the extracted full-face feature of the i-th unmasked input sample x_i, y_i denotes the identity label of x_i, θ_j denotes the angle between W_j^F and $E_F(x_i)$, s is a hyper-parameter to multiply all logits by a constant scaling value, and m denotes a margin penalty used for extra intra-class minimization and inter-class maximization.

D. Learning of the Partial-Face Feature Extractor based on Balanced Feature Matching

The partial-face feature extractor E_P is designed to handle Case 2 and Case 3 matching. To achieve the balanced feature matching that constrains the extractor E_P to extract similar features between the original unmasked faces and the synthetically masked faces, the following *guided perceptual loss* L_{guided} is applied:

$$L_{guided} = \frac{1}{B}\sum_{i=1}^{B} \| \frac{E_P(x_i)}{\|E_P(x_i)\|} - \frac{E_P(\hat{x}_i)}{\|E_P(\hat{x}_i)\|} \|_2^2, \tag{4}$$

where B is the batch size, $E_P(x_i)$ denotes the feature of the i-th input face x_i extracted by the extractor E_P, and $E_P(\hat{x}_i)$ represents the feature of the input image $\hat{x}_i$ extracted by the extractor E_P.

We now discuss how the features of the partial-face contain identification information. Given the features of the unmasked and masked face extracted from E_p, we utilize the ArcFace and define the *partial-face ArcFace loss* $L_{arc}^{partial}$ by

$$\begin{aligned} &L_{arc}^{partial} = \\ &-\frac{1}{B}\sum_{i=1}^{B}(log(\frac{e^{s(cos(\theta_{y_i}+m,i))}}{e^{s(cos(\theta_{y_i}+m,i))} + \sum_{j=1,j\neq y_i}^{M} e^{s(cos(\theta_j,i))}}) \\ &+log(\frac{e^{s(cos(\hat{\theta}_{y_i}+m,i))}}{e^{s(cos(\hat{\theta}_{y_i}+m,i))} + \sum_{j=1,j\neq y_i}^{M} e^{s(cos(\hat{\theta}_j,i))}})), \end{aligned} \tag{5}$$

subject to

$$\begin{aligned} cos(\theta_j + m, i) &= \|W_j^P\| \|E_P(x_i)\| cos(\theta_j + m), \\ cos(\hat{\theta}_j + m, i) &= \|W_j^P\| \|E_P(\hat{x}_i)\| cos(\hat{\theta}_j + m), \end{aligned} \tag{6}$$

where B is the batch size, W_j^P denotes the j-th column of the weight vector of the partial-face ArcFace layer LR_P, $E_P(x_i)$ represents the extracted partial-face feature of the i-th input sample x_i while $E_P(\hat{x}_i)$ is the extracted partial-face feature of the input sample $\hat{x}_i$, y_i denotes the identity label of both x_i and $\hat{x}_i$, θ_j denotes the angle between W_j^P and $E_P(x_i)$ while $\hat{\theta}_j$ is the angle between W_j^P and $E_F(\hat{x}_i)$, s is a hyper-parameter to multiply all logits by a constant scaling value, and m denotes a margin penalty used for extra intra-class minimization and inter-class maximization.

E. Loss Function

Integrating all the equations described above, the total loss function used to optimize the proposed system takes the following form:

$$L_{total} = L_{mask} + L_{guided} + L_{arc}^{full} + L_{arc}^{partial} \tag{7}$$

F. Inference Stage

The inference stage for face verification is shown in Fig. 3 as well. The mask classifier C identifies whether the test face pair contains at least a masked face image. If the face pair contains no masked face, the full-face feature extractor E_F is used to extract the test feature pair from the face pair. On the other hand, if one of the faces in the face pair is predicted as a masked face, the partial face feature extractor E_P is used to extract the feature pair.

IV. EXPERIMENTS

This section describes the experimental setup, including preprocessing techniques, training and testing datasets, baseline models, and implementation details. Both qualitative and quantitative experimental results are shown.

A. Experimental Setup

1) Preprocessing techniques: We use MaskTheFace [4] and MTCNN [13] to prepare the datasets required for training and testing in our face recognition task. The MaskTheFace tool is used to convert existing face datasets into masked face datasets. In order to increase the diversity of facial masks, we randomly synthesize masked faces with different types of masks, including cloth, surgical, and N95. The MTCNN tool is used for face detection and face alignment, and the aligned faces are cropped out for face recognition. We normalize all the face images to $112 \times 112 \times 3$ pixels and feed them to the proposed system and the baseline models. To make sure that preprocessed datasets are reliable, we also manually clean up the datasets by deleting wrongly detected faces due to failures of MTCNN.

2) Dastasets: The CASIA-WebFace dataset [14] and the corresponding Masked CASIA-WebFace generated by the MaskTheFace tool are used as training datasets. After preprocessing and cleaning, both datasets consist of 404,280 face images of 10,575 identities. The testing datasets include the Labeled Faces in the Wild (LFW) [10], the synthetically masked LFW (Masked LFW), the real-world masked face verification dataset (RMFD) [12], and the masked faces in real world face recognition (MFR2) [4]. The details of these four testing datasets are described in the following:

- LFW: This is a well-known public dataset containing 13,233 face images of 5,749 identities and 6,000 face pairs for face verification. All face pairs belong to Case 1.
- Masked LFW: This is a synthetically masked face dataset generated from LFW by the MaskTheFace tool. After manual cleaning, the dataset contains 13,134 face images of 5,749 identities and 5,910 face pairs. All face pairs belong to Case 3.
- RMFD: This is a real-world masked face dataset consisting of face images crawled from the website. After manual cleaning, the dataset contains 3,588 face images of 426 identities and 4,586 face pairs. All face pairs belong to Case 2.
- MFR2: This is a small real-world masked face dataset with face images collected from the Internet. After manual cleaning, the dataset contains 259 face images of 53 identities and 788 face pairs. The face pairs include all three cases of feature matching discussed in Sec. I.

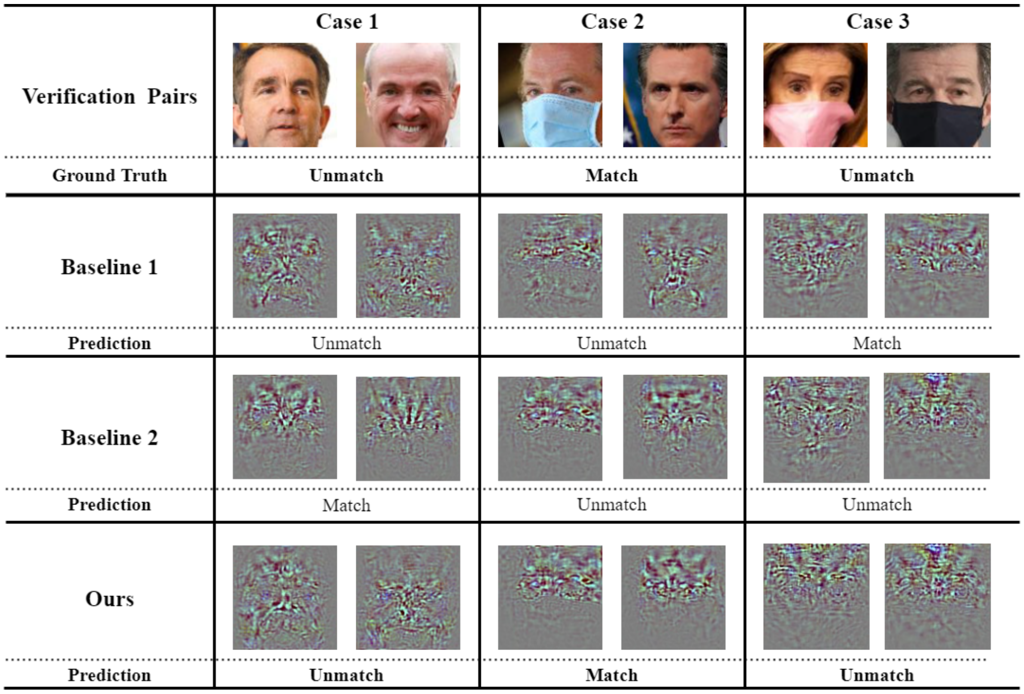

Fig. 4. Visualization of the features learned by the two baseline models and the proposed method. The first row shows three pairs of face images. The second to fourth row compare the feature representations learned by the Baseline 1 model, the Baseline 2 model, and our proposed method.

Mask Classifier

Layer	Chan./Stri.	Outp.Size
conv4×4	16/4	28×28×16
batchnorm2d	-/-	28×28×16
relu	-/-	28×28×16
flatten	-/-	12544
linear	-/-	512
batchnorm1d	-/-	512
linear	-/-	2

Full-Face ArcFace Layer

Layer	Intp.Size	Outp.Size
linear	512	M

Partial-Face ArcFace Layer

Layer	Intp.Size	Outp.Size
linear	512	M

TABLE I
THE STRUCTURE DETAILS OF THE MASK CLASSIFIER, THE FULL-FACE ARCFACE LAYER, AND THE PARTIAL-FACE ARCFACE LAYER.

Method	LFW	Masked LFW	RMFD	MFR2
Baseline 1	**99.3%**	96.1%	80.2%	94.5%
Baseline 2	98.8%	98.4%	85.4%	96.0%
Ours	99.3%	**98.5%**	**85.6%**	**96.4%**

TABLE II
THE VERIFICATION ACCURACY COMPARISONS BETWEEN BASELINES AND THE PROPOSED METHOD.

Method	LFW	Masked LFW	RMFD	MFR2
Ours	99.9%	99.6%	97.6%	98.5%

TABLE III
THE MASK CLASSIFICATION ACCURACY OF THE PROPOSED METHOD.

3) Baseline models: We compare our proposed method with two baseline models. Baseline 1 is the state-of-the-art ArcFace method with ResNet-50 [15] feature extractor trained on the CASIA-WebFace dataset. Baseline 2 has the same structure as Baseline 1 but is trained on the original CASIA-WebFace and the synthetically masked CASIA-WebFace dataset.

4) Implementation details: All the experiments are implemented with Pytorch. We train the models on four TITAN Xp COLLECTORS EDITION GPUs. The batch size is set to 128 and the SGD optimizer [16] is used for optimization with a total of 100 epochs. We set the momentum to 0.9 and the weight decay to 0.0005. The initial learning rate is set to 0.1 and divided by 10 at the 30th, 60th, and 90th epochs. We use ResNet-50 for the proposed full-face feature extractor E_F and partial-face feature extractor E_P. The architectures of the mask classifier C, the full-face ArcFace layer LR_F, and the partial-face ArcFace layer LR_P are detailed in Table I.

B. *Quantitative Performance Analysis*

A face verification task is performed for quantitative performance and computational cost evaluations. In this task, we take a pair of face images as input and determine whether the pair of images are of the same identity by calculating the $L2$ distance between the facial features of the pair of images. The best threshold is obtained by 10-fold cross-validation for each testing dataset. Table II summarizes the verification accuracy of the proposed method and the baselines. We can see that our method outperforms Baseline 1 and Baseline 2 for the Masked LFW, RMFD, and MFR2 datasets, which contain masked faces. We can also see that, when tested on LFW, our method performs at the same level as Baseline 1. This is due to the high accuracy of the mask classification network, see Table III. Also, the additional computational cost of the mask classification network is only 70.7M FLOPs, whereas the computational cost of feature extraction is 6.3G FLOPs regardless of which of the four feature extractors (partial-face feature extractor, full-face feature extractor, Baseline 1, and Baseline 2) is used.

C. Qualitative Performance Analysis

The Guided-Backpropagation [11] technique is used to visualize the feature representations learned by the proposed method. Fig. 4 shows the visualizations together with the verification results of three sample pairs in the MFR2 dataset. Noted that the three pairs belong to three different matching cases. We can see that the Baseline 2 model fails on the face pair of Case 1 since it ignores the bottom region of the face. As a result, faces with similar facial features in the upper region are identified as similar faces in disregard of the differences in the bottom region. In contrast, through mask classification, the proposed method is able to extract the features of the whole face in this case and generates the correct match. For the face pair of Case 2 in the middle column, the proposed method outputs a correct match prediction because it performs a balanced feature matching between masked and unmasked faces, whereas Baseline 2 fails due to the bias of features in the nose region. For the face pair of Case 3 in the right column, the proposed method is able to ignore the region covered by the facial mask, better than what the Baseline 1 model is able to do.

V. CONCLUSION

Masked face recognition is a real-world problem of critical importance, considering the current COVID-19 pandemic. To combat the effect of facial mask occlusion, we propose the concept of balanced feature matching and design a robust masked face recognition system based on this idea. The proposed mask classification network of the system classifies an input image pair into one of three cases for matching. This helps the two feature extractors to extract the facial features effectively for all three cases. Moreover, the guided perceptual loss constrains the partial-face feature extractor to focus on the upper region of a face so as to achieve balanced feature matching and reduce the impact of mask occlusion on face recognition. As a result, the proposed system is able to deal with masked faces without sacrificing its performance for unmasked faces. The effectiveness of our method is verified by testing its face verification accuracy on public datasets and comparing it with the state-of-the-art techniques. The visualizations provide additional insight into the advantage of the proposed method. In conclusion, the proposed method enables a face recognition system to work robustly even if people have to wear facial masks to combat COVID-19.

ACKNOWLEDGMENT

This research was supported in part by the Ministry of Science and Technology of Taiwan (MOST 109-3111-8-002-002 and MOST 110-2221-E-002-108-MY3), National Taiwan University (NTU 109-3111-8-002-002), and Cathay United Bank.

REFERENCES

[1] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song, “Sphereface: Deep hypersphere embedding for face recognition,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2017.

[2] H. Wang et al., “Cosface: Large margin cosine loss for deep face recognition,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2018.

[3] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, “Arcface: Additive angular margin loss for deep face recognition,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019.

[4] A. Anwar and A. Raychowdhury, “Masked face recognition for secure authentication,” arXiv preprint arXiv:2008.11104 (2020).

[5] D. Montero, M. Nieto, P. Leskovsky, and N. Aginako, “Boosting Masked Face Recognition with Multi-Task ArcFace,” arXiv preprint arXiv:2104.09874 (2021).

[6] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” Communications of the ACM 60 (2012): 84 - 90.

[7] Y. Taigman, M. Yang, M. Ranzato and L. Wolf, “DeepFace: Closing the Gap to Human-Level Performance in Face Verification,” 2014 IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 1701-1708, doi: 10.1109/CVPR.2014.220.

[8] W. Ouyang et al., “Deepid-net: Deformable deep convolutional neural networks for object detection,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2015.

[9] F. Schroff, D. Kalenichenko, and J. Philbin, “Facenet: A unified embedding for face recognition and clustering,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2015.

[10] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, “Labeled faces in the wild: A database for studying face recognition in unconstrained environments,” Univ. Massachusetts, Amherst, Amherst, MA, USA, Tech. Rep. 07-49, 2007

[11] J. T. Springenberg, A. Dosovitskiy, T. Brox, and M. Riedmiller, “Striving for simplicity: The all convolutional net,” arXiv preprint arXiv:1412.6806 (2014).

[12] Z. Wang, et al., “Masked face recognition dataset and application,” arXiv preprint arXiv:2003.09093 (2020).

[13] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, “Joint face detection and alignment using multitask cascaded convolutional networks,” IEEE Signal Processing Letters 23.10 (2016): 1499-1503.

[14] D. Yi, Z. Lei, S. Liao, and S. Z. Li, “Learning face representation from scratch,” arXiv preprint arXiv:1411.7923 (2014).

[15] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016.

[16] H. Robbins, and S. Monro, “A Stochastic Approximation Method,” Annals of Mathematical Statistics 22 (2007): 400-407.

[17] Y. Li, K. Guo, Y. Lu, and L. Liu, “Cropping and attention based approach for masked face recognition,” Applied Intelligence, pp. 1–14, 2021.