Mask-Compatible Face Verification System

Xinhao Liu
School of Arts and Science
New York University Shanghai
Shanghai, China
Email: xinhao.liu@nyu.edu

Wenbin Qi
School of Arts and Science
New York University Shanghai
Shanghai, China
Email: wq372@nyu.edu

***Abstract*—The "new normality" of COVID-19 makes wearing facial mask a daily routine. This causes a big problem to the widely-used facial verification systems because a number of them don't perform well when recognizing people's faces with masks on. This adds many extra work to people's daily life especially for frequently used systems like iPhone's face id, since users need to take off their masks to be identified. The difficulty of this problem mainly lies in how to recognize a person's identity only be the face that is not covered by the mask. We solve this problem mainly by introducing center loss and CBAM to traditional convolutional neural network. Though the accuracy of our model is not very high, when we apply it to to real-time videos, it has a good performance in distinguishing different identities.**

I. Introduction

Since the outbreak of COVID-19 last year, we've noticed many tiny changes to our life. One of them is that iPhone's face ID no longer works when people are wearing masks. Although there might be some tricks to let iPhone recognize you even with mask, all of them are either insecure or tedious. Recently, Apple allows users with an apple watch to unlock iPhone when they are wearing masks in the newest update of the iOS system, but it's still a very temporary solution. We've also noticed that the face access in the Academic Building has been closed since last year and it also causes lots of inconvenience.

With this motivation in mind, we want to dive deep in the field of facial recognition and see what would be the mechanism behind if we want to build such a system that can recognize people's face and distinguish people's identity even when they are wearing masks.

II. Leteriture Review

Face recognition is actually an application in computer science with a very long history. The earliest research on this field can trace back to the 1960s. The first capstone of face recognition was eigenface in the 1980s [1]. Stepping into the 21st century, SVM was introduced to facial recognition and received a relatively good performance [2]. In 2014, Heisele, Ho, Poggio first applied metric learning into the field of facial recognition, and published their DeepID2 model [3]. Then, in 2015, Google creatively transfer from classification loss into contrastive loss triplet loss with their FaceNet [4]. Moreover, in recent years, there were also some reserach exactly about what kind of effect masks would have on the facial recognition systems both before and after the outbreak of COVID-19 [5]–[7]. Based on these models and papers, we have an initial idea about the revolution on facial recognition system. We also decided to mainly use different convolutional neural networks to achieve our goal considering it is the most state-of-art achievements in the field and has a relatively higher accuracy.

III. Dataset

During our research, we found a very recent open-source face dataset called WebFace260M, which, according to the authors, is "largest public face recognition training set" [8]. This dataset contains 260 million images of human faces with 4 million identities. In other words, on average, each identity have 65 images. Currently, the authors release a subset of this dataset with 4 million images. Although this is very small compare to the original dataset, it's far from enough in terms of our computation capability. A slight drawback of this dataset is that it doesn't contain images of people wearing mask, so we have to do some modifications to it. The following are two modifications we did:

A. Crop

According to our research, many of facial recognition models and networks that are widely used now don't perform well when predicting faces with mask on. The reason is obvious and intuitive because a correctly worn mask covers around half of a human face. Thus, we would like to know specifically how will the performance of a model drop as we increase the area covered on a human's face. Therefore, we decided to crop images in the dataset into different percentages, as shown in figure 1 and 2, in order to test the performance of a model.

Fig. 1. Example of Original Images

B. Artificial Mask

To add a mask to an image of human face is actually very easy because we can use tools like Photoshop. However, it's really impossible to add mask to hundreds of thousands of images, especially when the direction, location, and angle of the faces are very different in these images. Therefore, we used

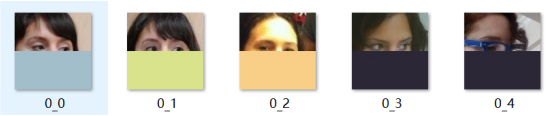

Fig. 2. Example of Images Being Cropped by 50 Percent

an program that can automatically add mask to people's face [9]. We slightly modified the program so that it will randomly add mask to an image of human face by a probability of 50 percent. Another modification to the program is that when it add masks, we let the program randomly choose a color of mask so the model trained from this dataset can be more robust to real-life application. Figure 3 shows the images of one of the identities in the dataset before and after we add mask to it. Moreover, considering the computation speed, we sliced a subset of 100 identifies and 2000 images from the dataset we downloaded. Up till now, we are ready to apply different models to this dataset.

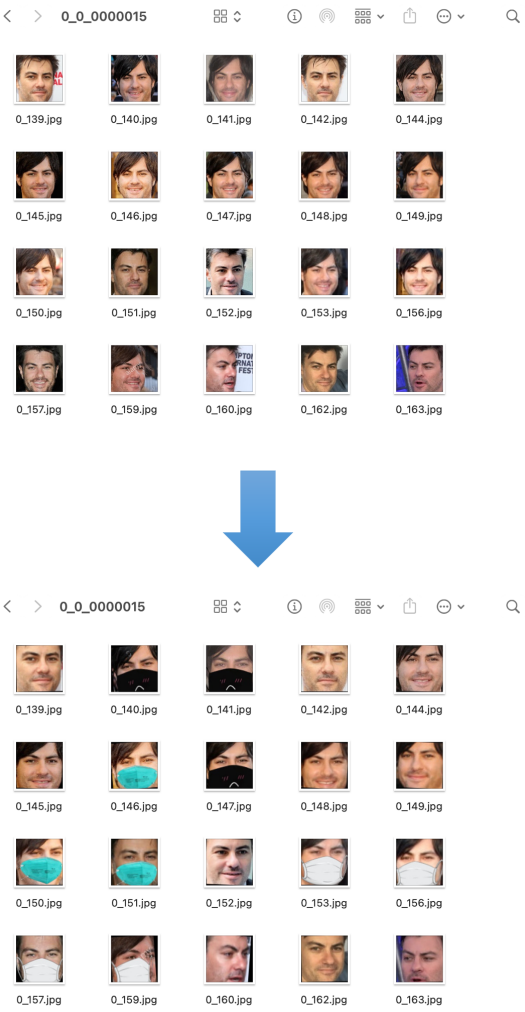

Fig. 3. Example of the images of one identity before and after we add artificial mask

IV. Modeling

A. VGGNet

Essentially, distinguishing different identity of human faces is a classification problem, so we considered some existing CNN frames and chose VGGNet [10]. We chose it because it has a relatively simple structure comparing to other CNN frames so we can have trials and errors very fast. We tried it on the dataset without mask and the result is not very good. As can be seen from figure 4, the highest accuracy reaches only about 40%. We think the reason behind the difference the theoritical high accuracy of VGGNet and the low accuracy from our test is the original purpose of VGGNet. Since it is used for image and video recognition, the classification is much wider. In other words, for example, the difference between the face of a dog and a cat is way larger than the difference between two human faces. Therefore, it's understandable that the accuracy on the unmasked dataset can't exceed 50%.

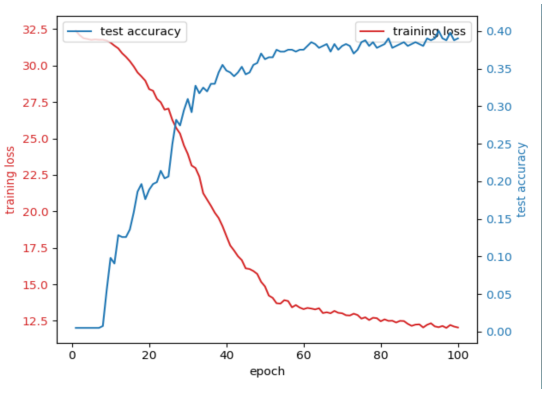

Fig. 4. Loss and accuracy of VGGNet on our dataset for 100 epochs of training

B. Siamese Network

After the failure on VGGNet, we found Siamese Network, which is another structure of network during our research. This network is used to compare if the objects in two images is similar or not. Together with contrastive loss function, it performs very well in object classification problems [11]. However, since our dataset is composed of folders of identity and each of them contains images of the same identity, if we would like to apply Siamese network to our dataset, we have to modify its structure and we think it's very difficult. Considering we only have a very limited time to work on the project, we stopped looking furthing into this network.

C. Center Loss

Although we weren't able to apply siamese network, it indeed inspired us to modify the loss functions of the CNN models. Then, we found center loss [12]. First of fall, unlike softmax loss function, center loss itself cannot be used to perform classification. Instead, it acts on the embedding data of the images (usually the last but one layer of the CNN models).The goal of center loss is to maximize the distance

between different classes and minimize the distance inside each class. Although this sound very like to the clustering problem where the goal is to minimize intra-class distance and maximize inter-class distance, the difference is vital. In clustering, the parameters of each sample is fixed and the goal is to cluster them, but for center loss, the embedding is generated by the network is can be updated through training. Surprisingly, simply by adding center loss as a supportive loss function to softmax loss, we get a nearly 98% accuracy on the dataset without mask (see figure 5).

However, when we were so exited to try this model on the dataset with masks, we were disappointed to see the accuracy drops sharply to 50% even after 200 epochs of training (see figure 6). We also notice that the accuracy of this method decreases linearly with respect to the percentage covered of the images. This means that with center loss alone, the network still performs badly when it is used to process images with masks. Our first trial is to add weight on the upper part of the image. We simply multiple a scalar to the top 40% of the entries in the tensor generated by the image. We were not sure if it is mathematically rigours, but the result turned out to have very little improvement. The accuracy only increase by 0.5 to 1 percent.

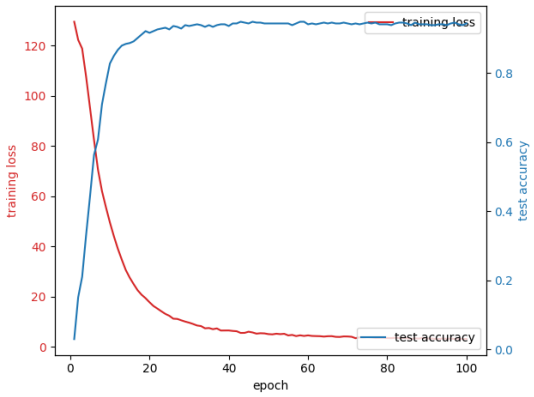

Fig. 5. Loss and accuracy of VGGNet with center loss on the dataset without mask for 100 epochs of training

D. CBAM

Although simply adding weight doesn't work efficiently, we were very sure if we add attention mechanism like that in the RNNs to our model, it's performance should increased a lot. After all, thinking intuitively, when we want to recognize a face with mask, we would focus less on the mask itself and pay more attention to the upper part of the face. We can also convince ourselves to add attention on the upper part of images when observing the decrease of accuracy in Figure.7. Different from adding weight, since the attention module can also be trained, it can adjust automatically what place need to pay more attention to. Then, we noticed the Convolutional Block Attention Module (CBAM) [13]. According to Woo et al., the module has one channel attention module and spatial attention module. Each of them focus differently on

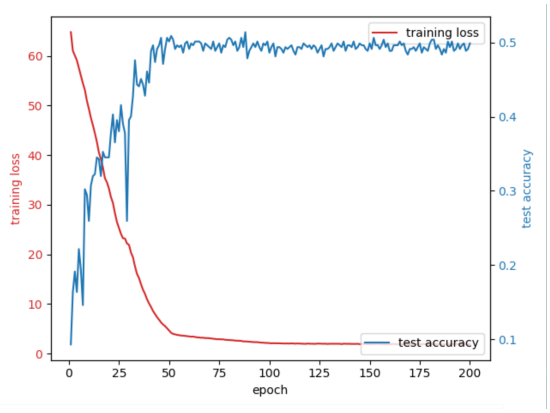

Fig. 6. Loss and accuracy of VGGNet with center loss on the dataset with mask for 200 epochs of training

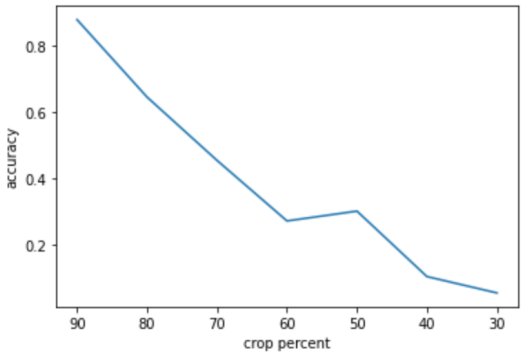

Fig. 7. Decrease of accuracy of VGGNet with center loss on the cropped dataset

the different dimensions of the tensor generated by an image. Since in the paper, the authors build the module on the basis of Resnet [14], we transferred our model to Resnet as well. Although the authors use traditional softmax loss in the model, we still made some modifications to add centor loss to it in order to let the model have a better performance on human face. After doing this, we have a slightly bigger improvement comparing to the previous models (see figure 8). This is where we stopped because of the time limit. Our final model is a composition of Resnet, CBAM, and center loss. We think it both accommodates the tiny difference between human faces and the existence of masks. Although the final accuracy, 60%, doesn't sound like a very good result, we'll see in the next section that it works relatively well in real-time implementation.

V. Real-time Implementation

After training the model, we decided to test its performance in a real life scenario. Specifically, we would like to capture human faces using a live camera or a recorded video in every frame, segment the human face and predict the identity of the face being captured. Fortunately, we found Voila-Jones algorithm [15] performing on Haar features to detect the position of a human face in an image, which helps the segmen-

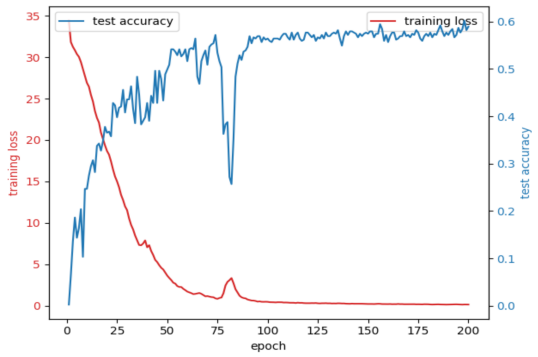

Fig. 8. Loss and accuracy of ResNet with CBAM and center loss on the dataset with mask for 200 epochs of training

tation. According to our experiment, this algorithm succeeds in detecting the position of human faces with mask on under a proper illumination and a simple background, although it fails sometimes. After the segmentation, we pass the section of the human face in the image into our CBAM model and predict the identity. In a dataset containing 100 identities and 2000 images, including images of authors, CBAM model successfully predicted our identities for approximately half of all frames in a recorded video (see this link). In a dataset containing only 20 identities and 400 images, the model successfully predicted our identities for approximately 87.5% of all frames in a recorded video. In both datasets, the model was able to correctly label an unknown identity.

VI. Discussion and Improvements

To recognise the identity of a face image with mask on is not a easy task to perform because it requires a correct prediction without any information below the nose area. However, we reach an accuracy of 60% on a dataset of 1000 identities, which is an acceptable result. There are also lots of spaces for future improvements, for example, a fine tuning on the hyper-parameters and to improve the performance of the model when treating unknown identities by setting a better threshold or trying other metrics. We can also try to extract more features of a human face from sections that are not covered by masks, for example, the overall shape of the face and the ears.

Acknowledgment

We would like to acknowledge Prof. Li Guo and our teaching assistant Yunjie Song for delivering us a fantastic Machine Learning course this semester. Without the knowledge we learned in the course, it would be impossible for us to finish this project. We would also like to acknowledge Prof. Li Guo for the assistance to get access to the WebFace260M dataset, and the access to NYU Shanghai's High Performance Computer, as well as the suggestions and advice during we carry out our project. Last but not Least, we would like to thank Daniel Zhou Liu for letting us use a video of him to test our model.

References

[1] L. Sirovich and M. Kirby, "Low-dimensional procedure for the characterization of human faces," *Josa a*, vol. 4, no. 3, pp. 519–524, 1987.

[2] G. Guo, S. Z. Li, and K. Chan, "Face recognition by support vector machines," in *Proceedings fourth IEEE international conference on automatic face and gesture recognition (cat. no. PR00580)*. IEEE, 2000, pp. 196–201.

[3] Y. Sun, X. Wang, and X. Tang, "Deep learning face representation by joint identification-verification," *arXiv preprint arXiv:1406.4773*, 2014.

[4] 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, pp. 815–823.

[5] N. Kose and J.-L. Dugelay, "Countermeasure for the protection of face recognition systems against mask attacks," in *2013 10th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*. IEEE, 2013, pp. 1–6.

[6] N. Damer, J. H. Grebe, C. Chen, F. Boutros, F. Kirchbuchner, and A. Kuijper, "The effect of wearing a mask on face recognition performance: an exploratory study," in *2020 International Conference of the Biometrics Special Interest Group (BIOSIG)*. IEEE, 2020, pp. 1–6.

[7] M. Jiang, X. Fan, and H. Yan, "Retinamask: A face mask detector," *arXiv preprint arXiv:2005.03950*, 2020.

[8] Z. Zhu, G. Huang, J. Deng, Y. Ye, J. Huang, X. Chen, J. Zhu, T. Yang, J. Lu, D. Du, and J. Zhou, "Webface260m: A benchmark unveiling the power of million-scale deep face recognition," 2021.

[9] Z. Wang, G. Wang, B. Huang, Z. Xiong, Q. Hong, H. Wu, P. Yi, K. Jiang, N. Wang, Y. Pei, H. Chen, Y. Miao, Z. Huang, and J. Liang, "Masked face recognition dataset and application," 2020.

[10] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2015.

[11] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Icml*, 2010.

[12] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, "A discriminative feature learning approach for deep face recognition," in *European conference on computer vision*. Springer, 2016, pp. 499–515.

[13] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "Cbam: Convolutional block attention module," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 3–19.

[14] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," 2015.

[15] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, vol. 1, 2001, pp. I–I.