# Masked Face Recognition Using MobileNetV2

MING LIU, Auckland University of Technology, Auckland 1010 New Zealand

WEI QI YAN, Auckland University of Technology, Auckland 1010 New Zealand

Masked face recognition has made great progress in the field of computer vision since the popularity of Covid-19 epidemic in 2020. In countries with severe outbreaks, people are asked to wear masks in public. The current face recognition methods, which take use of the whole face as input data is quite well established. However, when people are use of face masks, it will reduce the accuracy of face recognition. Therefore, we propose a mask wearing recognition method based on MobileNetV2 and solve the problem that many of models cannot be applied to portable devices or mobile terminals. The results indicate that this method has 98.30% accuracy in identifying the masked face. Simultaneously, a higher accuracy is obtained compared to VGG16. This approach has proven to be working well for the practical needs.

**Additional Keywords and Phrases:** Computer vision · Deep learning · MobileNetV2 · VGG16 · Masked face recognition

## 1 INTRODUCTION

Since the outbreak of the COVID-19 epidemic, its rapid spread has posed a serious threat to people's ordinary lives. To prevent cross-infection and the expansion of the epidemic, the World Health Organization (WHO) has proposed that wearing masks properly in public places and maintaining a safe distance is an effective way to prevent its spread [1]. However, due to lack of awareness, uncomfortable to wear masks and other reasons, it is difficult to reach the standard of people wearing masks consciously. Therefore, it is very important to detect whether masks are worn in public places and whether they are worn in a standard way.

In 2006, the theory of deep learning was proposed, the field of imaging represented by computer vision developed rapidly. Due to the increased amount of data and computing power today, it makes deep learning work well to its advantage, especially in target detection and image feature extraction [2]. Face recognition algorithms are already popularly applied to our real life [25]. However, the mainstream masked face detection algorithms before the Covid-19 epidemic required tagged samples, and the network models requires a high computer hardware configuration, with the problem that they could not be applied to portable devices or mobile [3].

The most general approach to masked face recognition is to consider face recognition as a classification task. The classification network is trained based on the large dataset. The fully connected classification layer in the last layer of the network is removed and the remaining network layer is employed as the face feature

extraction network. The output of the last layer of this network is feature data [4]. But now the sizes of deep nets are enormous. The existing experiments have shown that the accuracy of mask recognition using deep learning method RetinaFace and VGGFace2 is only 94.5% [5]. This accuracy is not optimal for practical applications. Image processing using convolution is one of the frequently used methods in the field of computer vision. For tackling less data where the face is marked, transfer learning in deep learning can handle it well which means that these models are already trained by other data [6]. MobileNetV2 from Keras is one of the representatives of transfer learning techniques [7]. The MobileNet model was firstly proposed in 2017 that is a lightweight convolutional neural network designed for embedded or mobile devices. After gradual development, optimization, and update iterations, MobileNetV2 was presented in 2019. This lightweight network transfers the mask wearing recognition problem to a classification problem by using a target detection network. It effectively reduces the number of network model parameters and greatly reduces the computing time [8]. The parameters of the model size are around 5M [9].

In this research project, we train the model using two separate datasets from Kaggle having the same faces with and without face masks. We take use of the OpenCV framework in addition to image processing techniques. We obtained a real-time detection accuracy of 98.3% in our results. This accuracy is two percent higher than the same test conducted in 2020 [10]. Simultaneously, we compared it with another deep learning model VGG16 and obtained a higher accuracy.

The rest of the paper is arranged as follows. Section 2 we introduction of the existing work and the inspiration from it, Section 3 we describe the MobileNetV2 model in detail, Section 4 we present the experiment and analyse the results, Section 5 we give our conclusion and feature works.

## 2 RELATED WORK

As an important task in the field of computer vision, face masked recognition has been modeled for a long time. The studies on mask testing have been increasing since 2016, especially after the Covid-19 epidemic. Since in the beginning for face recognition algorithms cannot collect all face image data for training and prediction, the face recognition with task is practiced on the closed set and tested on the open set [11]. This leads to learn algorithmic models that can distinguish unknown features only on a limited dataset.

With the development of face masked recognition algorithms, a series of loss algorithms for faces have emerged. From the initial Softmax-Loss [12], Triplet-Loss, Center-Loss to the subsequent A-Softmax [13], L-Softmax, Arcface-Loss [14], and then to the AdaCos [15] proposed in CVPR2019. AdaCos does not require hyperparameters compared to the loss function and takes use of adaptive scaling parameters to automatically enhance training supervision during the training process, showing the advantages of improving the speed of network convergence and making the network more stable, which can significantly improve the accuracy of face recognition.

Video-based passenger masked face recognition has been accomplished and applied to railroad transportation systems. Krishan et al. proposed a face mask detection and normative wear recognition method based on YOLOv3 and YCrCb. YOLOv3 was applied to detect whether the mask is worn or not, the elliptical skin color model of YCrCb is applied to detect skin color in the mask region, and then to determine whether the mask is worn. The mAP for masked face detection is 89.07% in the experiment, and the recognition rate of mask regulation wearing reached 82.48% [16]. This value is proved to be unsatisfactory in our subsequent experiments.

After YOLOv4 was released in 2020. Sharma et al. proposed a mask wearing detection method based on fusion of high and low frequency components of images with YOLOv4 net [17]. The experiments were conducted by web crawlers to build the dataset and manual data annotation, trained by Darknet framework to conduct object detection, the model detection accuracy reached 98.5% after model training, with an average detection speed of 35.2 ms. Compared to YOLOv3, YOLOv4 offers a significant improvement in accuracy. Since YOLOv4 is use of a mish activation function that is smoother than the Leaky ReLU activation function in YOLOv3, the gradient descent is much effective.

Since face recognition models are easily exposed to sunlight in open-air environments, changes in sunlight and facial expressions can have an impact on algorithm performance. Lahasan et al. [18] conducted a work to address these challenges. The evaluation is classified into three parts: Occlusion feature extraction, occlusion recognition, and occlusion recovery. The mask is employed as an example of the object of facial expression recognition system, categorized it into holistic and part-based approaches. Experimental results show that the local matching method has better performance compared to the reconstruction method in partial-based mechanisms. They demonstrated that the combination of local matching methods and optimization based on metaheuristic techniques can increase the stability of marked face recognition. However, it requires a large enough number of facial images for training and are not available for models such as e-passports, which are trained for a single sample.

## 3 METHOD

### 3.1 MobileNetV2

MobileNetV2 is a lightweight convolutional neural network designed for embedded or mobile devices. The structure of the network has two types of stride blocks, which have three layers in both blocks.

The network mainly takes use of deeply separable convolution. In the first layer, a number of channels expanded by 1×1 convolution with ReLU. This allows for feature extraction in higher dimensions. The 3×3 depth-wise convolutional contains in second layer. In the third layer, the feature dimensionality reduced by 1×1 convolutional. The activation function is not applied in the final dimensionality reduction layer because using the activation function for low-dimensional features would lose some of the extracted image space features. Deeply separable convolution can greatly reduce the number of model parameters and the amount of computation, which is able to improve the computational speed of the network, the training process can also make full use of device resources, the model can also be built in embedded devices and mobiles to achieve the effect of real-time recognition. The first layer is stride 1 block. The second layer is depth-separable convolution with residual module as shown in Table 1.

Table 1: Deeply separable convolution with residual structure

| input | operator | output |
|---|---|---|
| $h\times w\times c$ | 1×1 conv2d, ReLU | $h\times w\times tc$ |
| $h\times w\times tc$ | 3×3 dw, ReLU | $(h/s)\times(w/s)\times tc$ |
| $(h/s)\times(w/s)\times tc$ | 1×1 conv2d, ReLU | $(h/s)\times(w/s)\times c_1$ |

From Table 1, where *h*, *w*, *c* represent the height of the image, width of the image and the number of channels, respectively. Since the MobileNetV2 network has stride 2 layers, through using convolutional filtering with a step size of 2, it will cause a large loss of information. In this paper, we consider using the attention optimization module according to Squeeze-and-Excitation Networks as shown in Table 2 [19].

Table 2: MobileNetV2 architecture

| Input | Operator | t | c | n | s |
|---|---|---|---|---|---|
| $224^2 \times 3$ | conv2d | - | 32 | 1 | 2 |
| $112^2 \times 32$ | bottleneck | 1 | 16 | 1 | 1 |
| $112^2 \times 16$ | bottleneck | 6 | 24 | 2 | 2 |
| $56^2 \times 24$ | bottleneck | 6 | 32 | 3 | 2 |
| $28^2 \times 32$ | bottleneck | 6 | 64 | 4 | 2 |
| $14^2 \times 64$ | bottleneck | 6 | 96 | 3 | 1 |
| $14^2 \times 96$ | bottleneck | 6 | 160 | 3 | 2 |
| $7^2 \times 160$ | bottleneck | 6 | 320 | 1 | 1 |
| $7^2 \times 320$ | conv2 1×1 | - | 1280 | 1 | 1 |
| $7^2 \times 1280$ | avgpool 7×7 | - | - | 1 | - |
| $1 \times 1 \times 1280$ | conv2 1×1 | - | k | - | |

Pertaining to global average pooling in MobileNetV2 net, the use of an average pooling layer degrades network performance. In a 7×7 feature map, the perceptual domain of the center point and the edge points are the same, the center point includes the complete image while the edge points have only part of the image, so each point has a different weight. However, the average pooling layer represents all pixels with the same weight, it leads to a decrease in network performance. In this paper, we take advantage of 7×7 size convolution kernels for grouped convolution instead of global average pooling in MobileNetV2 network, which allows the network to learn the weights by itself instead of treating the weights of each point as the same, it makes the network has more generalization ability.

The input image size of the MobileNetV2 lightweight model for face recognition is 224×224, the model consists of four parts. The first part outputs thirty-two 112×112 features maps through 3×3 ordinary convolutions with a step size of 2, padding of 1 and takes a grouped convolution. The second part is composed of six different mobile modules and finally outputs 160 7×7 feature maps. In the third part, the feature dimension is expanded by 1×1 ordinary convolution, the final face feature map is obtained by 1280 1×1 convolutions. In the last part, we implement the classification layer through full connectivity. The network structure as shown in Table 2 [20]. Where c represents the number of channels, *n* represents number of repetitions of the residual structure, s represents the step size for the first layer of the inverted residual architecture for n repetitions.

## 4 EXPERIMENT AND RESULTS

### 4.1 Dataset

We are use of the data from the face mask dataset uploaded to GitHub [21]. The dataset contains 69 files, each has 950 images of the same face with and without the mask. We select one of these files separately for training and testing. The class imbalance is avoided because the proportion of photos with and without masks is the same. We set the size of all images to 224×224. The experimental environment is MS Windows 10 operating system. We make use of an NVIDIA GeForce RTX 3080 to train the model based on Anaconda using Jupyter for simulation.

### 4.2 Experimental procedure

Regarding the model to perform well on the test set, we aim to achieve generalization. In this experiment, we take use of Stochastic Gradient Descent (SGD) algorithm for model training. In order to speed up the convergence and reduce the oscillation in the process of model convergence, the momentum factor is added to the experimental training process in this experiment, the model weight update strategy is shown as follows.

$$\nabla \mathrm{w} = \frac{1}{m} \sum_{j=1}^{m} \frac{\partial L(y^j, f(x^j; w))}{\partial w} . \tag{1}$$

The parameter is updated as

$$\nu = \beta\nu - \alpha\nabla \mathrm{w}. \tag{2}$$

where $\beta$ represents the momentum factor which was set to 0.9 in the experiment, $\alpha$ is learning rate and the initial value is set to 0. 01. The learning rate is set to 0. 001, 0. 000 1, and 0. 000 01 for epochs of 40, 50, and 60, respectively.

### 4.3 Evaluation Indicators

In this paper, we introduce accuracy, recall, and precision to evaluate the performance of image classification task models. The accuracy rate is the proportion of correct predictions over all samples using the formula as shown in (3). TP in the formula denotes the sample predicted to be positive among all positive samples; TN shows the sample predicted to be negative among all negative samples; FP indicates the sample predicted to be positive among all negative samples; FN reflects all positive samples predicted to be negative samples. Although accuracy can be used to determine overall correctness, it is not a good indicator of results in the case of an unbalanced sample. The high accuracy obtained can be invalidated by sample imbalance.

$$\mathrm{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \tag{3}$$

Recall is the proportion of the number of correctly predicted positive samples to the actual number of positive samples, its equation is shown in (4).

$$\mathrm{Recall} = \frac{TP}{TP+FN} \tag{4}$$

The precision is the probability of predicting the actual positive sample in a positive sample, its equation is shown in (5), where a higher recall means a higher probability that an actual useless user will be predicted

$$\mathrm{Precision} = \frac{TP}{TP+FP} \tag{5}$$

### 4.4 Training Result Analysis

The model converged with 40 iterations on the pre-trained model, with training accuracy rate of 98.7% as shown in Fig.1.

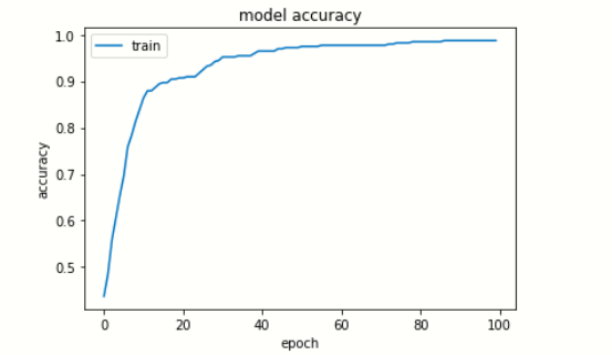


Fig. 1. Accuracy of MobileNetV2 based on lightweight face recognition algorithm

Combined with the evaluation metrics, the accuracy, recall and average accuracy of each classification of the two datasets were calculated, the evaluation results are shown in Table 3.

Table. 3: Experimental results in the training data

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| with_mask | 0.97 | 0.99 | 0.98 | 138 |
| without_mask | 0.99 | 0.97 | 0.98 | 138 |
| accuracy | | | 0.98 | 276 |
| macro avg | 0.98 | 0.98 | 0.98 | 276 |
| weighted avg | 0.98 | 0.98 | 0.98 | 276 |

From the evaluation results, MobileNetV2 neural network model has good effect on the recognition of whether the mask is worn or not and the average accuracy, recall and precision rate of each category are above 97%. To further verify the effectiveness of the algorithm, we compare the algorithm in this paper with other deep learning algorithm VGG16 using the same dataset. The result as shown in Table 4. In our experiments, we made use of batch size variations of 2, 4, 8, 16 and 32. A batch size of 8 means that the data set is divided into eight batches for neural network training.

Table 4: Different model using batch size variations

| Pre-trained Model | 2 batch | | 4 batch | | 8 batch | | 16 batch | | 32 batch | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Train Acc. | Val Acc. | Train Acc. | Val Acc. | Train Acc. | Val Acc. | Train Acc. | Val Acc. | Train Acc. | Val Acc. |
| VGG16 | 0.9276 | 0.9571 | 0.9529 | 0.9706 | 0.9841 | 0.9844 | 0.9751 | 0.9844 | 0.9857 | 0.9844 |
| MobileNetV2 | 0.9558 | 1.0000 | 1.0000 | 1.0000 | 1.0000 | 1.0000 | 1.0000 | 1.0000 | 1.0000 | 1.0000 |

The result is shown In Table 4, the more batches get the better training and validation accuracy. Even on the batch four, MobileNetV2 already got 100% accuracy of training and validation. Compared to VGG16, the maximum accuracy rate of 98.57% was only obtained in 32 batches, we chose to use 32 batches as in the number of epoch experiment as shown in Table 5.

Table 5: Different model using epochs variations

| Pre-trained Model | 10 Epoch | | 20 Epoch | | 30 Epoch | | 40 Epoch | | 50 Epoch | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Train Acc. | Val Acc. | Train Acc. | Val Acc. | Train Acc. | Val Acc. | Train Acc. | Val Acc. | Train Acc. | Val Acc. |
| VGG16 | 0.1293 | 0.2188 | 0.5707 | 0.4688 | 0.8628 | 0.8906 | 0.9569 | 0.9219 | 0.9857 | 0.9844 |
| MobileNetV2 | 0.9723 | 0.9688 | 1.0000 | 1.0000 | 1.0000 | 1.0000 | 1.0000 | 1.0000 | 1.0000 | 1.0000 |

An epoch is a node, the dataset is trained based on the neural network until it is reset to the beginning of the round. In Table 5, MobileNetV2 achieved 100% training and validation accuracy on epoch 20. Meanwhile VGG16 only gets 57.07% and 46.88% training and validation accuracy, respectively. VGG16 still does not reach 100% accuracy on epoch 50. As a result, the pre-trained MobileNetV2 model has better accuracy than VGG16 which can obtain best model.

## 5 CONCLUSION AND FUTURE WORK

In this paper, we proposed the MobileNetV2 lightweight based on marked face recognition algorithm which is offered for public place detection in the current epidemic environment. In our experiments, we propose to use transfer learning to extract facial features from the data and perform classification. According to the experiments, we find that MobileNetV2 model has better accuracy by comparing with VGG16. The deep learning method in this paper generates higher efficiency. The current version is applied to desktop applications using real-time video detection [22,23,24].

## REFERENCES


[1] Asadi S., Cappa C., Wexler. A. Aerosol emission and superemission during human speech increase with voice loudness. pp. 1--10. Scientific Reports 9.1 (2019)

[2] Kaiming, H., Shao, Q.R., Xiang, Y.Z. Deep residual learning for image recognition. In: Computer Vision and Pattern Recognition pp. 102-108 (2015)

[3] Jamadar, S., Joshi, P., Surve, M., Vharkate, M. ZIB automatic attendance system using face recognition technique. In: J. Recent Technol. Eng, vol. 9, no. 1, pp. 2134--2138. Sciences Publication (2020)

[4] Adjabi, I., Benzaoui, A., Ouahabi, A. Past, present and future of face recognition: A review, vol.9, no.8, pp. 1-53. Electronics (2020)

[5] Aswal, V., Charniya, N., Shaikh, S., Tupe, O. Single camera masked face. In: IEEE International Seminar on Research of Information Technology and Intelligent Systems, pp. 57--60. (2020)

[6] Arymurthy, A., Gultom, Y., Masikome, J. Batik classification using deep convolutional network transfer. In: Komput, J. (eds.) vol. 11, no. 2, pp. 59 Journal Ilmu Komputer Dan Informasi (2018)

[7] Goh, Y.H.., Lee, Y.B., Lum, K.Y. American sign language recognition based on MobileNetV2. In: Adv.Sci.Technol, vol. 5, no. 6, pp. 481-488. ASTES (2020)

[8] Andrew, G., Marco, A., Weijun, W., Weyand, T. MobileNets: Efficient convolutional neural networks for mobile vision applications, NASA, vol.12, pp 233-240 (2017)

[9] Elmahmudi, A., Ugail, H. Deep face recognition using imperfect facial data, Futur.Gener.Comput.Syst., vol. 99, pp. 213-225, Springer (2019)

[10] Hu, L., Ge, Q. Automatic facial expression recognition based on MobileNetV2 in Real-time, Phys, J, vol. 15449, no. 2, IOP science (2020)

[11] Cai, Q., Peng, C., Shi, X. Lightweight face recognition algorithm based on MobileNetV2, vol 17231, no.1, pp 144-147(2020)

[12] Agrawal, A., Choudhary, A. Gopalakrishnan, K., Khaitan, K.: Deep convolutional neural networks with transfer learning for computer vision-based data-driven pavement distress detection, Constr,Buid.Matter., vol. 157, no. September, pp. 322-330, Springer (2017)

[13] Li, M., Liu, W., Wen, Y., Yu, Z. Sphere face: Deep hypersphere embedding for face recognition, In: IEEE Comuter Vision and Pattern Recognition, pp. 6738-6746. (2017)

[14] Deng, J., Guo, J., Xue, N., Yang, J. ArcFace additive angular margin loss for deep face recognition, In: IEEE International Conference on

Computer Vision, Visual Communications and Image Processing, vol. 124, no.1, pp 133-137 (2016)

[15] Zhang, T. Adaptive forward-backward greedy algorithm for learning sparse representations, In: IEEE Transactions on Information Theory, pp. 4689-4708. (2011)

[16] Ahuja, U., Kumar, K., Kumar, M., Sachdeva, M., Singh, S. Face mask detection using YOLO3 and faster R-CNN models: COVID-19 environment, pp. 19753-19768, Springer (2021)

[17] Akhil, K., Kalia, A., Kaushal, M., Sharma, A. A hybrid tiny YOLO v4-SPP module based improved face mask detection vision system. Journal of Ambient Intelligence Humanized Computing, vol. 14752, no.5, pp 142-145 (2021)

[18] Lahasan, B., Lutfi, S.L., Segundo, R. A survey on techniques to handle face recognition challenges: Occlusion, single sample per subject and expression. Artificial Intelligence Review, pp. 949-979, Springer (2019)

[19] Albanie, S., Hu, J., Shen, L., Sun, G., Wu, E. Squeeze-and-Excitation Networks. IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 42, no. 8, pp. 2011-2023, IEEE (2019)

[20] Chen, L., Howard, A., Sandler, M., Zhmoginov, A., Zhu, M. MobileNetv2: Inverted residuals and linear bottlenecks, In: IEEE Conference on Computer Vision and Pattern Recognition. Vol.142, no.7, pp 87-95 (2020)

[21] Cabani, A., Benhabiles,H., Hammoudi, K., Melkemi, M. Masked face-net-A dataset of correctly/incorrectly masked face images in the context of COVID-19. Smart Health, vol. 19, pp 31-40 (2020)

[22] Wei Qi Yan. Computational Methods for Deep Learning Theoretic, Practice and Applications. 2021, Springer.

[23] Wei Qi Yan. Introduction to Intelligent Surveillance: Surveillance Data Capture, Transmission, and Analytics, 2019 Springer.

[24] Xinyi Gao, Minh Nguyen, Wei Qi Yan. Face image inpainting based on generative adversarial network. International Conference on Image and Vision Computing New Zealand, 2021.

[25] Wei Cui, Wei Qi Yan. A scheme for face recognition in complex environments. International Journal of Digital Crime and Forensics (IJDCF) 8 (1), 26-36, 2016.

**Attachment:**

CCS Concepts: General and reference -> Document types -> General literature

XML Code:

```
\begin{CCSXML}
<ccs2012>
  <concept>
      <concept_id>10002944.10011122.10002946</concept_id>
      <concept_desc>General and reference~Reference works</concept_desc>
      <concept_significance>500</concept_significance>
      </concept>
 </ccs2012>
\end{CCSXML}

\ccsdesc[500]{General and reference~Reference works}
```

**Please fill in the all authors' background:**

| **Position can be chosen from:**<br>**Prof. / Assoc. Prof. / Asst. Prof. / Lecture / Dr. / Ph. D Candidate / Postgraduate, etc.** | | | | |
|---|---|---|---|---|
| Full Name | Email Address | Position | Research Interests | Personal Website (if any) |
| Ming Liu | Tsw8517@autuni.ac.nz | Postgraduate | Deep learning | |
| Wei Qi Yan | wyan@aut.ac.nz | A/Professor | Deep learning | |
| | | | | |
| | | | | |

*This form helps us to understand your paper better; the form itself will not be published.