



EfficientMask-Net for face authentication in the era of COVID-19 pandemic

Neda Azouji¹ · Ashkan Sami¹ · Mohammad Taheri¹

Received: 4 July 2021 / Revised: 20 November 2021 / Accepted: 21 January 2022 / Published online: 21 April 2022
© The Author(s), under exclusive licence to Springer-Verlag London Ltd., part of Springer Nature 2022

Abstract

Today, we are facing the COVID-19 pandemic. Accordingly, properly wearing face masks has become vital as an effective way to prevent the rapid spread of COVID-19. This research develops an Efficient Mask-Net method for low-power devices, such as mobile and embedded models with low-memory requirements. The method identifies face mask-wearing conditions in two different schemes: I. Correctly Face Mask (CFM), Incorrectly Face Mask (IFM), and Not Face Mask (NFM) wearing; II. Uncovered Chin IFM, Uncovered Nose IFM, and Uncovered Nose and Mouth IFM. The proposed method can also be helpful to unmask the face for face authentication based on unconstrained 2D facial images in the wild. In this study, deep convolutional neural networks (CNNs) were employed as feature extractors. Then, deep features were fed to a recently proposed large margin piecewise linear (LMPL) classifier. In the experimental study, lightweight and very powerful mobile implementation of CNN models were evaluated, where the novel “EfficientNetb0” deep feature extractor with LMPL classifier outperformed well-known end-to-end CNN models, as well as conventional image classification methods. It achieved high accuracies of 99.53 and 99.64% in fulfilling the two mentioned tasks, respectively.

Keywords COVID-19 · EfficientNet · Face mask-wearing · Face authentication · Large margin classifier · Deep feature extraction

1 Introduction and motivation

It is necessary to design a model for automatic identification of face mask-wearing conditions and use it as a first step to unmask the faces for face authentication in mobile devices and security systems, such as ATMs, banks, airport security checkpoints, and facial-biometric attendance systems.

The face mask condition identification is a very challenging task because while samples from the different classes are highly similar, samples from the same class may be much

different. In other words, there are a great intra-class variation and a small inter-class variation, which make it difficult to learn discriminant features. Figure 1 depicts some samples from the three classes.

In this paper, a new method has been developed for face mask-wearing identification using well-known deep convolutional neural networks (CNNs) as feature extractors and a novel large margin piecewise linear (LMPL) [1] as a classifier.

The proposed method contains four main steps: image pre-processing, deep feature extraction, face mask-wearing classification, and face unmasking. The proposed method showed an excellent performance in a computational resource-limited environment, for both classification tasks with 99.53 and 99.64% accuracy, respectively. Moreover, unmasking the masked faces showed a promising result. It can be concluded that the proposed EfficientMask-Net method is effective in face mask-wearing identification, as well as face unmasking. Therefore, it can be used in many security systems for epidemic prevention and face authentication.

✉ Ashkan Sami
sami@shirazu.ac.ir

Neda Azouji
azouji@shirazu.ac.ir

Mohammad Taheri
motaheri@shirazu.ac.ir

¹ Department of Computer Science and Engineering and IT,
School of Electrical and Computer Engineering, Shiraz
University, Shiraz, Iran

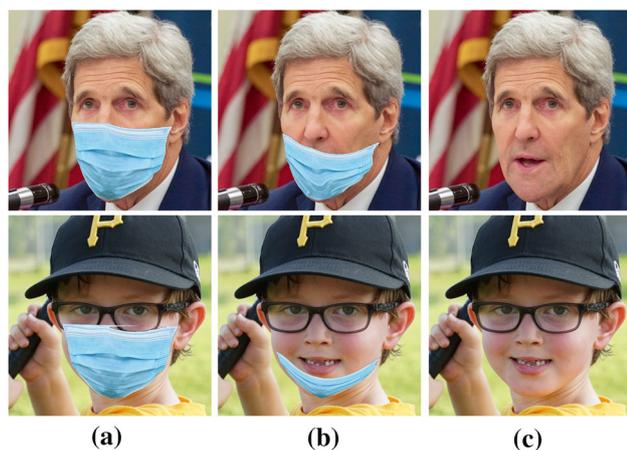


Fig. 1 Sample images of MaskedFace-Net dataset. **a** Correct Face Mask (CFM), **b** Incorrectly Face Mask (IFM), and **c** Not Face Mask (NFM) wearing show the challenge of this task: Samples of different classes are highly similar, while those of the same classes are so different

2 Related work

2.1 Masked face detection

Prasad et al. [2] proposed a lightweight model called “MaskedFaceNet” for real-time mask detection using a progressive semi-supervised approach. Fafous et al. [3] presented BinaryCoP (Binary COVID-mask Predictor) to detect correct face mask-wearing and positioning. The proposed BinaryCoP was a low-power binary neural network (BNN) classifier, which performed the classification on edge devices, such as embedded FPGA accelerator. They used the MaskedFaceNet dataset with four classes, including IMFD Nose and Mouth, IMFD Nose, IMFD Chin, and CMFD, and balanced the dataset with data augmentation techniques. As a result, accuracy of up to 98% was obtained for the wearing positioning problem.

2.2 Mobile-based face mask detection

Cabani et al. [4] introduced the MaskedFace-Net dataset with 137,016 images. This large-scale dataset includes Correctly Masked Face Dataset (CMFD) and Incorrectly Masked Face Dataset (IMFD), in which masked faces are created by applying a deformable model on the Flickr-Faces-HQ3 (FFHQ) face dataset. Qin et al. [5] proposed image super-resolution and classification network (SRCNet), in which a super-resolution method was applied to improve the performance of low-quality images. They classified the face mask-wearing situations into three classes, including correct mask-wearing, incorrect mask-wearing, and no mask-wearing, and achieved an accuracy of 98.70%. The training and evaluation were performed on the public Medical Masks Dataset containing 3835 images.

2.3 Identification of face mask-wearing conditions

Dey et al. [6] proposed a deep learning and multi-stage face mask detection method called “Mobile-Net Mask.” They used two different datasets with 5200 images to detect Masked or NotMasked faces from still images and video streams. The Mobile-Net Mask reached an accuracy of 93%. Jiang et al. [7] presented a RetinaFaceMask detector based on the one-stage RetinaNet for high-accuracy face mask detection. The introduced model contained ResNet or MobileNet as a backbone, along with a feature pyramid network (FPN) and context attention modules. The authors achieved a 93.4% precision, which was higher than baseline results.

As presented in this section, although researchers have introduced several approaches to identify face mask-wearing conditions, face authentication lacks a unified system. In this study, we developed a unified, efficient method for face mask-wearing identification besides unmasking the masked faces, which can be useful in authentication systems.

3 Materials and methods

This section describes the overall process of the proposed EfficientMask-Net method. Figure 2 demonstrates the diagram of the proposed mask-wearing system.

3.1 Image preprocessing

Image preprocessing enhances the visual appearances of images and results in higher accuracy of the detection system.

3.2 Resizing face images

The input images of EfficientNet were resized to $224 \times 224 \times 3$ using bicubic interpolation.

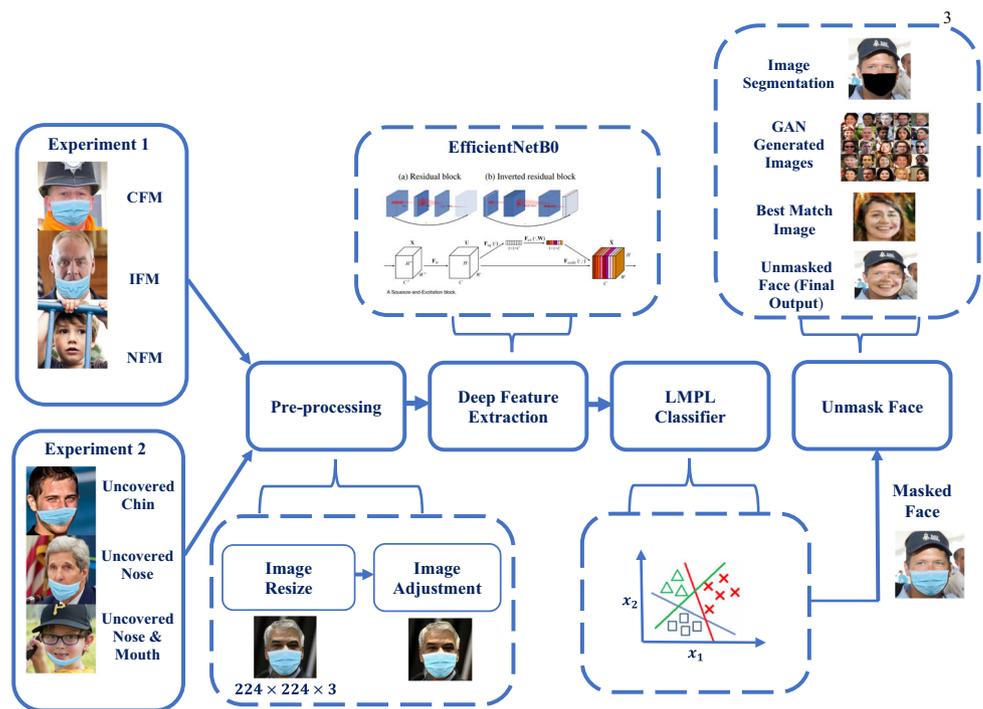
3.3 Image adjustment

Real-world images have a considerable variation in contrast and exposure. The images were adjusted by mapping input intensity to the new values to saturate 1% of the pixel values in low and high intensities. Besides, the histogram of images was calculated to determine the adjustment limit automatically.

3.4 Deep feature extraction

High-level and abstract features can be extracted by deep CNNs. This study focused on a small and efficient network in computational power. Transfer learning was used to prevent overfitting and obtain better generalization.

Fig. 2 A schematic of the proposed EfficientMask-Net



EfficientNet was introduced by Tan and Le [8] in 2019. It is one of the most efficient CNN models among well-known pre-trained networks with a small number of FLOPS. Compared to other models achieving similar ImageNet accuracy, the EfficientNet is much smaller and faster. The authors have shown that the proposed EfficientNet is five times faster for inference on mobile devices [8].

3.5 Large margin piecewise linear (LMPL) classifier

The novel large margin piecewise linear (LMPL) classifier [1] works based on a cellular structure. First of all, a grid is considered on feature space. In fact, some random hyper-planes partition feature space into subpartitions called cells. Each cell is labeled by a class label based on covered training instances. The main problem is with tuning of initial hyper-planes.

- 1) *Normal*: Ordinary samples, which are correctly classified at just one side of the hyper-plane. Their loss function is Hinge loss as defined in (1):

$$l(x)_{\text{Normal}(\tilde{y})} = \max(0, 1 - \tilde{y}(w^T \cdot x + b)) \quad (1)$$

where: $\tilde{y} = \{-1, +1\}$

where \tilde{y} is the virtual label of sample x and determines at which side of the hyper-plane, x is correctly classified.

- 2) *Negative don't care*: These samples are classified incorrectly on both sides of the hyper-plane. Their loss is defined in (2):

$$l(x)_{\text{DontCare}^-} = \max(l(x)_{\text{Normal}^{(+1)}}, l(x)_{\text{Normal}^{(-1)}}) \quad (2)$$

- 3) *Positive don't care*: This group is the opposite of *Negative don't care*, and samples are classified correctly on both sides of the hyper-plane. Their loss function is defined in (3):

$$l(x)_{\text{DontCare}^+} = \min(l(x)_{\text{Normal}^{(+1)}}, l(x)_{\text{Normal}^{(-1)}}) \quad (3)$$

The *Positive don't care* samples, which are always classified correctly, are ignored in this paper. The main reason is that the loss function in (3) is not convex. Therefore, the objective function is defined as presented in (4):

$$\min \frac{1}{2} \|w\|^2 + C_1 \sum_{x \in \text{Normal}} l(x)_{\text{Normal}(\tilde{y})} + C_2 \sum_{x \in \text{DC}^-} l(x)_{\text{DontCare}^-} \quad (4)$$

The scalar values C_1 and C_2 control the balance between the structural and empirical error. In this paper, both C_1 and C_2 were experimentally tested and set to 1000.

The LMPL classifier optimizes each hyper-plane based on the introduced objective function with a convex optimizer.

After some iterations, the model converges to some hyper-planes in order to classify samples of different classes, and extra hyper-planes that are not useful in the classification are removed. Therefore, regarding the distribution and the complexity of the decision boundaries, the complexity of the model is tuned by removing redundant hyper-planes, and an efficient large margin approach is obtained.

3.6 Unmasking the face

Given its contactless nature, especially in the pandemic era, using faces is preferred in biometric recognition. However, these systems are designed for non-occluded faces [9], the proposed method was designed to work based on existing face authentication methods and avoid retraining them on masked face datasets. Most of the recent works have focused on the eye area exclusively [10] or retraining existing methods on the simulated masked faces [11].

1) Image segmentation

As the first step, faces were segmented into Mask and Non-Mask segments to determine missed parts of the face. Figure 3a illustrates an example of an input masked face and the resulting segmented face.

2) Generating Synthetic Faces

A generative advertising network (GAN) was trained on 15,000 real-world faces without face masks. Then, 25 synthetic faces were generated by the trained GAN to complete the masked faces, as shown in Fig. 3b.

3) Selecting the Matched generated face

The distance between a masked face and generated faces was calculated at pixel level based on the normalized root-mean-square error (NRMSE), which ranges from 0 (identical) to 1 (completely different). The synthetic face with the smallest value was selected to complete the masked face. An example is shown in Fig. 3c.

4) Face Completion

Facial parts of the mask area were extracted from the selected synthetic face to fill missed parts of the masked face. An example of the final output of the proposed method is shown in Fig. 3d.

Algorithm 1 shows the whole process of the proposed EfficientMask-Net method.

4 Experimental results

4.1 Experimental setup

All experiments were implemented using the deep learning and image processing toolboxes of MATLAB R2021a. A CPU Core i7 4.00 GHz with 24 GB RAM was applied to implement the

MaskEfficient-Net. The Adam optimizer [12] with $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 10^{-8}$ was also used. Moreover, weight decay of 10^{-4} for L2 regularization was applied to avoid overfitting.

The network was trained for five epochs with a mini-batch size of 64. The initial learning rate was set on 10^{-3} , and the learning rate drop factor was set on 0.1 for all three epochs to increase the learning speed. Besides, the training dataset shuffled every epoch.

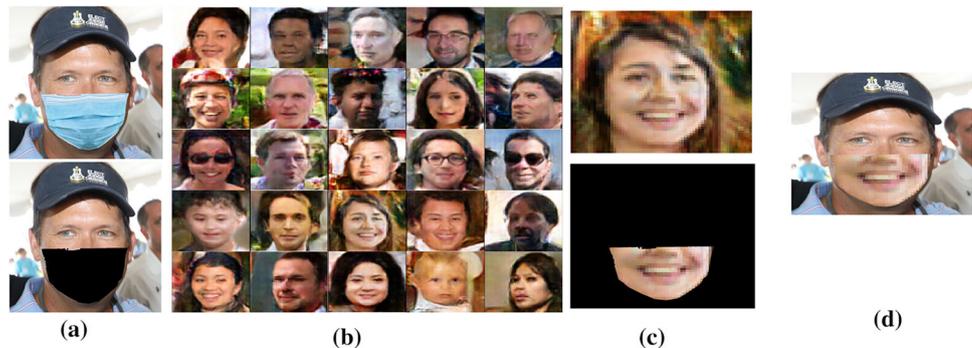
Algorithm 1. The proposed EfficientMask-Net method

1. **Initialization:** Experiment 1
2. Experiment 2
3. **Mask-wearing condition identification:**
4. **Input:** Face images
5. **Image preprocessing:**
6. Image resizing to $224 \times 224 \times 3$ pixels
7. Image adjustment
8. **Feature extraction:**
9. Fine-tuning the pre-trained EfficientNet-B0 by transfer learning
10. Deep feature extraction by EfficientNet-B0
11. **Classification:**
12. **Training:**
13. Train the LMPL classifier on the training set x
14. **Testing:**
15. Classifying the test image x' by LMPL model
16. **Output:**
17. y' : Predicted class label of the test image
18. **Face Unmasking:**
19. **Input:** Masked face
20. **Image segmentation:**
21. Image segmentation to Mask and Non-Mask segments
22. **Generating synthetic faces:**
23. 25 faces generated by a trained GAN
24. **Selecting the best matched generated image:**
25. Calculating NRMSE between the masked face and 25 generated faces
26. Selecting the synthetic face with the smallest value
27. **Face completion:**
28. Extracting facial parts of the mask area from the selected synthetic face
29. Filling the missed parts of the masked face
30. **Output:** Unmasked face

In this study, two experiments were carried out for two different classification schemes:

- I. *Experiment 1:* Correctly Face Mask (CFM), Incorrectly Face Mask (IFM), and Not Face Mask (NFM) wearing
- II. *Experiment 2:* Uncovered Chin IFM, Uncovered Nose IFM, and Uncovered Nose and Mouth IFM

Fig. 3 The steps of unmasking the faces. **a** A masked face and the resulting segmented face. **b** 25 generated images by GAN. **c** The selected synthetic face and extracted facial parts of the mask area. **d** The unmasked face



B. MaskedFace Dataset

4.2 MaskedFace dataset

In this study, we combined the novel MaskedFace-Net¹ and the well-known Flickr-Face-HQ² (FFHQ) datasets. FFHQ is an open-access high-quality dataset with PNG images of 1024×1024 resolution. The original FFHQ was used as the Not mask-wearing dataset. The details of class samples and the related experiments are listed in Table 1. Finally, 14,783 and 4992 face images were used in Experiments 1 and 2, respectively. The complete dataset for each experiment can be found in the Zenodo repository (<https://zenodo.org/record/4892677>).

4.3 Experimental results and analysis

1) Performance Analysis

Several lightweight deep networks were compared as an end-to-end network and a feature extractor with the novel LMPL classifier (called CNN⁺) in terms of different metrics, as shown in Tables 2 and 3 for both experiments. In both experiments, EfficientNetB0 achieved the best results in both schemes as an end-to-end network and a feature extractor with the LMPL classifier (EfficientNetB0⁺).

The novel LMPL was also compared with well-known classifiers. According to the results, LMPL outperformed all other classifiers in terms of performance metrics. As illustrated in Tables 4 and 5, the LMPL achieved the best classification accuracy in both experiments.

2) Statistical Analysis

¹ see “MaskedFace-Net dataset” <https://github.com/cabani/MaskedFace-Net>

² see “dataset of face images Flickr-Faces-HQ (FFHQ)” <https://github.com/NVlabs/ffhq-dataset>

Table 1 Details of face image dataset

Experiment	Types	No. of x-ray images	Source database
Experiment 1	Correctly Face Mask (CFM)-wearing	4792	MaskedFace Net, Correctly Maskedface Dataset (CMD)
	Incorrectly Face Mask (IFM)-wearing	4991	MaskedFace Net, Incorrectly Maskedface Dataset (IMD)
	Not Face Mask (NFM)-wearing	5000	Flicker-Face-HQ (FFHQ)
Experiment 2	Uncovered Chin, IFM	1815	MaskedFace Net, Incorrectly Maskedface Dataset (IMD)
	Uncovered Nose, IFM	1608	MaskedFace Net, Incorrectly Maskedface Dataset (IMD)
	Uncovered Nose and Mouth, IFM	1569	MaskedFace Net, Incorrectly Maskedface Dataset (IMD)
Unmasking face	Not face mask (NFM)-wearing	15,000	Flicker-Face-HQ (FFHQ)

Friedman test is a popular statistical analysis for simple, nonparametric, and safe comparison of at least three-related samples. It has no assumption about primary data distribution. This test ranks methods for each metric independently. Indeed, R_j is the average rank of the j th method based on different metrics. Note that in the case of tie, i.e., identical performance, the same ranks are assigned.

As can be seen in Tables 2, 3, 4, 5, the novel LMPL improved the performance metrics significantly and obtained the best average ranks in all cases. These tables reveal the

Table 2 Comparison of deep CNNs as an end-2-end network and as a feature extractor, along with the proposed LMPL classifier (CNN⁺) in experiment 1: correctly Face Mask (CFM), Incorrectly Face Mask (IFM), and Not Face Mask (NFM) wearing

Method	Performance Metrics (%)					
	Sensitivity (Recall)	Specificity (TNR)	Precision (PPV)	F1-score	Accuracy	Average rank
EfficientNet	97.24 ± 0.22(1)	97.62 ± 0.11(1)	97.25 ± 0.22(1)	97.24 ± 0.22(1)	97.24 ± 0.22(1)	1.0
EfficientNet ⁺	99.54 ± 0.16(1)	99.77 ± 0.08(1)	99.54 ± 0.16(1)	99.54 ± 0.16(1)	99.53 ± 0.16(1)	1.0
MobileNetV2	97.23 ± 0.16(2)	97.61 ± 0.08(2)	97.23 ± 0.16(2)	97.23 ± 0.16(2)	97.22 ± 0.16(2)	2.0
MobileNetV2 ⁺	98.50 ± 0.11(2)	98.75 ± 0.06 (2)	98.51 ± 0.11(2)	98.50 ± 0.11(2)	98.50 ± 0.11(2)	2.0
NasNetMobile	96.93 ± 0.14(4)	97.46 ± 0.07(4)	96.95 ± 0.13(4)	96.94 ± 0.14(4)	96.93 ± 0.14(4)	4.0
NasNetMobile ⁺	98.45 ± 0.15(4)	98.72 ± 0.08(4)	98.46 ± 0.14 (4)	98.45 ± 0.15(4)	98.45 ± 0.15(4)	4.0
ShuffleNet	97.10 ± 0.11(3)	97.54 ± 0.05(3)	97.10 ± 0.10(3)	97.10 ± 0.10(3)	97.09 ± 0.11 (3)	3.0
ShuffleNet ⁺	98.47 ± 0.20(3)	98.73 ± 0.10(3)	98.48 ± 0.20(3)	98.47 ± 0.20(3)	98.47 ± 0.20(3)	3.0
SqueezeNet	96.92 ± 0.12(5)	97.46 ± 0.06(4)	96.93 ± 0.11(5)	96.92 ± 0.12(5)	96.92 ± 0.12(5)	4.8
SqueezeNet ⁺	98.21 ± 0.27(5)	98.60 ± 0.14(5)	98.22 ± 0.26(5)	98.21 ± 0.27(5)	98.21 ± 0.27(5)	5.0
Avg. on CNN	97.08 ± 0.16	97.54 ± 0.08	97.09 ± 0.15	97.09 ± 0.15	96.04 ± 0.22	
Avg on CNN⁺	98.63 ± 0.52	98.91 ± 0.48	98.64 ± 0.52	98.63 ± 0.52	98.63 ± 0.52	

*Bold numbers indicate the best performance

Table 3 Comparison of deep CNNs as an end-2-end network and as a feature extractor, along with the proposed LMPL classifier (CNN⁺) in experiment 2: uncovered chin IFM, uncovered nose IFM, and uncovered nose and mouth IFM

Method	Performance metrics (%)					
	Sensitivity (Recall)	Specificity (TNR)	Precision (PPV)	F1-score	Accuracy	Average rank
EfficientNet	96.47 ± 0.19(1)	96.74 ± 0.09(1)	96.48 ± 0.18(1)	96.47 ± 0.18(1)	96.48 ± 0.18(1)	1.0
EfficientNet ⁺	99.64 ± 0.06(1)	99.82 ± 0.03(1)	99.63 ± 0.06(1)	99.63 ± 0.06(1)	99.64 ± 0.05(1)	1.0
MobileNetV2	96.12 ± 0.25(2)	96.57 ± 0.13(2)	96.12 ± 0.25(2)	96.12 ± 0.25(2)	96.14 ± 0.25(2)	2.0
MobileNetV2 ⁺	98.53 ± 0.15(2)	98.77 ± 0.07(2)	98.52 ± 0.14(2)	98.53 ± 0.14(2)	98.54 ± 0.13(2)	2.0
NasNetMobile	94.60 ± 0.66(5)	94.83 ± 0.31(5)	94.61 ± 0.64(5)	94.59 ± 0.68(5)	94.64 ± 0.64(5)	5.0
NasNetMobile ⁺	97.79 ± 1.81(5)	98.42 ± 0.85(5)	97.88 ± 1.60(5)	97.80 ± 1.78(5)	97.84 ± 1.71(5)	5.0
ShuffleNet	95.62 ± 0.34(4)	96.32 ± 0.16(3)	95.62 ± 0.36(4)	95.62 ± 0.35(4)	95.64 ± 0.34(4)	3.8
ShuffleNet ⁺	98.49 ± 0.23(3)	98.76 ± 0.11(3)	98.48 ± 0.22(3)	98.48 ± 0.22(3)	98.50 ± 0.21(3)	3.0
SqueezeNet	95.88 ± 0.56(3)	95.45 ± 0.28(4)	95.88 ± 0.57(3)	95.88 ± 0.57(3)	95.90 ± 0.56(3)	3.2
SqueezeNet ⁺	97.93 ± 0.64(4)	98.47 ± 0.32(4)	97.92 ± 0.66(4)	97.92 ± 0.65(4)	97.94 ± 0.64(4)	4.0
Avg. on CNN	95.74 ± 0.71	95.98 ± 0.81	95.74 ± 0.71	95.74 ± 0.71	95.76 ± 0.70	
Avg on CNN⁺	98.48 ± 0.73	98.84 ± 0.57	98.49 ± 0.71	98.47 ± 0.72	98.49 ± 0.72	

*Bold numbers indicate the best performance

significant difference between the efficiency of the different methods.

3) Visual Analysis

Gradient-weighted class activation mapping (Grad-CAM) technique [15] was used for detailed visual analysis, which provides a visualization of the extracted deep features through the fine-tuned EfficientNetB0, as shown in Fig. 6. Grad-CAM is a technique to interpret deep CNN predictions

and check whether the CNN is focusing on the right parts of the input image. Prediction regions can be investigated using heat maps. The spatial parts with the greatest impact on the network score were identified by Grad-CAM heat mapping, as shown in Fig. 6. The standard jet map was used in which red and yellow indicate regions with high contribution to the right predictions and blue denotes regions with low contribution. As can be seen, the fine-tuned deep EfficientNetB0 well identified the effective regions in the classification predictions.

Table 4 Comparison of well-known classifiers with the proposed LMPL classifier in experiment 1: correctly face mask (CFM), incorrectly face mask (IFM), and not face mask (NFM) wearing

Method	Performance Metrics (%)					
	Sensitivity (Recall)	Specificity (TNR)	Precision (PPV)	F1-score	Accuracy	Average rank
NaiveBayes	93.59 ± 0.80(13)	96.30 ± 0.41(13)	93.89 ± 0.76(13)	93.66 ± 0.79(13)	93.62 ± 0.82(13)	13.0
<i>k</i> NN (<i>k</i> = 3)	96.10 ± 1.25(6)	97.54 ± 0.63(6)	96.15 ± 1.18(7)	96.08 ± 1.31(6)	96.08 ± 1.28(6)	6.2
<i>k</i> NN (<i>k</i> = 5)	96.17 ± 1.44(4)	97.58 ± 0.57(4)	96.22 ± 1.08(4)	96.15 ± 1.19(4)	96.16 ± 1.17(4)	4.0
<i>k</i> NN (<i>k</i> = 7)	96.14 ± 1.12(5)	97.56 ± 0.56(5)	96.19 ± 1.04(5)	96.12 ± 1.17(5)	96.13 ± 1.14(5)	5.0
OvO SVM	96.08 ± 1.26(7)	97.53 ± 0.63(7)	96.16 ± 1.14(6)	96.06 ± 1.32(7)	96.06 ± 1.28(7)	6.8
OvA SVM	96.02 ± 1.45(8)	97.50 ± 0.73(8)	96.10 ± 1.31(8)	96.00 ± 1.51(8)	96.00 ± 1.48(8)	8.0
Decision Tree	95.19 ± 1.01(11)	97.08 ± 0.52(11)	95.28 ± 0.87(11)	95.19 ± 1.00(11)	95.16 ± 1.02(11)	11.0
AdaBoostM2	95.47 ± 0.59(9)	97.23 ± 0.30(9)	95.47 ± 0.60(9)	95.43 ± 0.61(9)	95.46 ± 0.60(9)	9.0
TotalBoost	95.36 ± 0.71(10)	97.18 ± 0.36(10)	95.38 ± 0.68(10)	95.33 ± 0.73(10)	95.35 ± 0.72(10)	10.0
LP Boost	94.64 ± 0.34(12)	96.81 ± 0.17(12)	94.69 ± 0.35(12)	94.58 ± 0.34(12)	94.61 ± 0.34(12)	12.0
Random Forrest	96.28 ± 1.11(3)	97.64 ± 0.56(2)	96.34 ± 1.00(3)	96.26 ± 1.17(3)	96.27 ± 1.13(3)	2.8
SoftMax	97.24 ± 0.22(2)	97.62 ± 0.11(3)	97.25 ± 0.22(2)	97.24 ± 0.22(2)	97.24 ± 0.22(2)	2.2
Novel LMPL	99.54 ± 0.16(1)	99.77 ± 0.08(1)	99.54 ± 0.16(1)	99.54 ± 0.16(1)	99.53 ± 0.16(1)	1.0

*Bold numbers indicate the best performance

Table 5 Comparison of well-known classifiers with the proposed LMPL classifier in experiment 2: uncovered chin IFM, uncovered nose IFM, and uncovered nose and mouth IFM

Method	Performance metrics (%)					
	Sensitivity (Recall)	Specificity (TNR)	Precision (PPV)	F1-score	Accuracy	Average Rank
NaiveBayes	92.05 ± 0.62(11)	95.13 ± 0.29(10)	92.11 ± 0.62(11)	92.07 ± 0.62(10)	92.21 ± 0.58(10)	10.4
<i>k</i> NN (<i>k</i> = 3)	92.54 ± 0.92(9)	95.36 ± 0.46(9)	92.51 ± 0.94(9)	92.52 ± 0.93(9)	92.65 ± 0.93(9)	9.0
<i>k</i> NN (<i>k</i> = 5)	92.73 ± 0.75(7)	95.46 ± 0.37(7)	92.71 ± 0.76(7)	92.71 ± 0.76(7)	92.82 ± 0.74(7)	7.0
<i>k</i> NN (<i>k</i> = 7)	93.08 ± 0.69(6)	95.63 ± 0.34(6)	93.06 ± 0.71(6)	93.07 ± 0.70(6)	93.19 ± 0.70(6)	6.0
OvO SVM	93.34 ± 1.05(4)	95.75 ± 0.51(4)	93.34 ± 1.04(4)	93.33 ± 1.05(4)	93.45 ± 1.03(4)	4.0
OvA SVM	93.30 ± 1.06(5)	95.73 ± 0.51(5)	93.31 ± 1.05(5)	93.30 ± 1.05(5)	93.41 ± 1.03(5)	5.0
Decision Tree	90.67 ± 0.61(13)	94.47 ± 0.29(13)	90.69 ± 0.59(13)	90.68 ± 0.60(13)	90.85 ± 0.59(13)	13.0
AdaBoostM2	92.72 ± 0.56(8)	95.40 ± 0.27(8)	92.66 ± 0.56(8)	92.68 ± 0.56(8)	92.75 ± 0.54(8)	8.0
TotalBoost	91.04 ± 1.13(12)	94.56 ± 0.56(12)	91.39 ± 1.16(12)	91.00 ± 1.14(12)	91.01 ± 1.13(12)	12.0
LP Boost	92.10 ± 1.72(10)	95.07 ± 0.86(11)	92.15 ± 1.63(10)	92.03 ± 1.74(11)	92.05 ± 1.76(11)	10.6
Random Forrest	94.69 ± 1.06(3)	96.38 ± 0.52(3)	94.69 ± 1.07(3)	94.66 ± 1.07(3)	94.72 ± 1.05(3)	3.0
SoftMax	96.47 ± 0.19(2)	96.74 ± 0.09(2)	96.48 ± 0.18(2)	96.47 ± 0.18(2)	96.48 ± 0.18(2)	2.0
Novel LMPL	99.64 ± 0.06(1)	99.82 ± 0.03(1)	99.63 ± 0.06(1)	99.63 ± 0.06(1)	99.64 ± 0.05(1)	1.0

*Bold numbers indicate the best performance

4) Comparison with State-of-the-Art Studies

According to Table 6, the developed method showed superior performance in comparison to several recent studies. It can be concluded that the proposed Efficient-Mask Net can be useful in face mask-wearing monitoring systems, especially in public places, to control coronavirus spreading, as

well as face authentication services in lightweight devices like mobile phones.

5 Conclusion and future work

The proposed EfficientMask-Net model is lightweight and needs low power resources. Hence, the method can be useful

Table 6 Comparison of the proposed method with state-of-the-art deep models in face mask detection (CFM = correctly face mask-, IFM = incorrectly face mask-, NFM = not face mask-wearing)

Study	No. of cases	Method	Accuracy (%)
Qin et al. [5]	3030 CFM 134 IFM 671 NFM	SRCNet	98.70
Nagrath et al. [14]	690 Masked 686 Not Masked	MobilenetV2	92.64
Fasfous et. al [3]	68,229 CFM 6,689 Chin 1608 Nose 52,175 Nose & Mouth	BinaryCOP	98.10
Jiang et al. [7]	34,806 Masked 393,703 Not Masked	RetinaFaceMask	93.4 (Precision)
	1916 Masked 1930 Not Masked	MobileNetV2	96.85
Inamdar et al. [21]	10 CFM 15 IFM 10 NFM	Facemasknet	98.6
Loey et al. [19]	785 Masked 785 Not Masked	ResNet-50	99.49
Mercaldo and Santone [15]	2165 Masked 1930 Not Masked	MobilenetV2	98.0
Zhang et al. [20]	636 CFM 48 IFM 3988 NFM	Context-Attention R-CNN	84.1 (mAP)
Batagelj et al. [16]	29,532 CFM 1528 IFM 32,012 NFM	ResNet-152	98.93
Jiang et al. [7]	7695 CFM 366 IFM 10,471 NFM	SE-YOLOv3 and DarkNet-53	98.6 (AP)
Militante and Dionsio [17]	12,500 Masked 12,500 Not Masked	CNN	96.0
Dey et. al [6]	1916 Masked 1919 Not Masked	MobileNet Mask	93.0
Efficient-Mask Net (Proposed Method)	4792 CFM 4991 IFM 5000 NFM 1815 Chin 1608 Nose 1569 Nose & Mouth	Efficient -NetB0	99.53 99.64

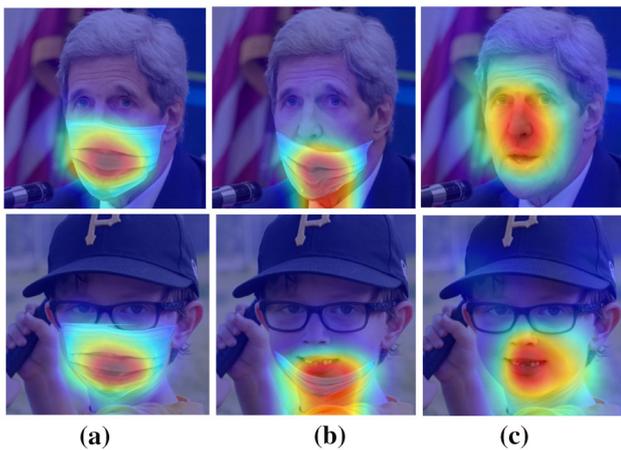


Fig. 4 Grad-CAM visualization results of different class face images. **a** Correctly Face Mask (CFM), **b** Incorrectly Face Mask (IFM), and **c** Not Face Mask (NFM) wearing. (Original images are shown in Fig. 1)

in real-time face mask-wearing systems to identify mask-wearing conditions in public places for epidemic prevention. Two experiments were conducted to evaluate the proposed method on various deep CNNs. The EffientNetB0 with the novel LMPL classifier showed the best average accuracy in both experiments, equal to 99.53 and 99.64%, respectively. The face unmasking was also performed on masked faces and showed promising results that can be useful in face authentication systems.

In the future, the proposed method can be extended to work on real-world masked face datasets. In order to improve face unmasking, the existing face completion methods under occlusion can be applied to masked faces. Besides, the impact of unmasking on present face recognition methods can be investigated.

References

1. Azouji, N., Sami, A., Taheri, M., Müller, H.: A large margin piecewise linear classifier with fusion of deep features in the diagnosis of COVID-19. *Comput. Biol. Med.* **139**, 104927 (2021)
2. Prasad, S., Li, Y., Lin, D., Sheng, D.: maskedFaceNet: a progressive semi-supervised masked face detector. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 3389–3398 (2021)
3. Fafous, N., Vemparala, M.-R., Frickenstein, A., Frickenstein, L., Stechele, W.: BinaryCoP: Binary neural network-based COVID-19 face-mask wear and positioning predictor on edge devices. *arXiv Prepr. arXiv2102.03456* (2021)
4. Cabani, A., Hammoudi, K., Benhabiles, H., Melkemi, M.: MaskedFace-net—a dataset of correctly/incorrectly masked face images in the context of COVID-19. *Smart Heal.* **19**, 100144 (2021)
5. Qin, B., Li, D.: Identifying facemask-wearing condition using image super-resolution with classification network to prevent COVID-19. *Sensors* **20**, 5236 (2020)
6. Dey, S.K., Howlader, A., Deb, C.: MobileNet Mask: A multi-phase face mask detection model to prevent person-to-person transmission of SARS-CoV-2. In: Proceedings of International Conference on Trends in Computational and Cognitive Engineering, pp. 603–613. Springer (2021)
7. Jiang, M., Fan, X., Yan, H.: RetinaMask: a face mask detector. *arXiv Prepr. arXiv2005.03950* (2020)
8. Tan, M., Le, Q.: Efficientnet: Rethinking model scaling for convolutional neural networks. In: International Conference on Machine Learning, pp. 6105–6114. PMLR (2019)
9. Yu, J., Hu, C.-H., Jing, X.-Y., Feng, Y.-J.: Deep metric learning with dynamic margin hard sampling loss for face verification. *Signal Image Video Process.* **14**, 791–798 (2020)
10. Li, Y., Guo, K., Lu, Y., Liu, L.: Cropping and attention based approach for masked face recognition. *Appl. Intell.* **51**, 3012–3025 (2021)
11. Anwar, A., Raychowdhury, A.: Masked face recognition for secure authentication. *arXiv Prepr. arXiv2008.11104* (2020)
12. Kingma, D.P., Ba, J.: Adam: a method for stochastic optimization. *arXiv Prepr. arXiv1412.6980* (2014)
13. Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-cam: visual explanations from deep networks via gradient-based localization. In: Proceedings of the IEEE International Conference on Computer Vision, pp. 618–626 (2017)
14. Nagrath, P., Jain, R., Madan, A., Arora, R., Kataria, P., Hemanth, J.: SSDMNV2: a real time DNN-based face mask detection system using single shot multibox detector and MobileNetV2. *Sustain. Cities Soc.* **66**, 102692 (2021)
15. Mercaldo, F., Santone, A.: Transfer learning for mobile real-time face mask detection and localization. *J. Am. Med. Informatics Assoc.* (2021)
16. Batagelj, B., Peer, P., Štruc, V., Dobrišek, S.: How to correctly detect face-masks for COVID-19 from visual information? *Appl. Sci.* **11**, 2070 (2021)
17. Militante, S. V., Dionisio, N. V.: Real-time facemask recognition with alarm system using deep learning. In: 2020 11th IEEE Control and System Graduate Research Colloquium (ICSGRC), pp. 106–110. IEEE (2020)
18. Jiang, X., Gao, T., Zhu, Z., Zhao, Y.: Real-time face mask detection method based on YOLOv3. *Electronics* **10**, 837 (2021)
19. Loey, M., Manogaran, G., Taha, M.H.N., Khalifa, N.E.M.: A hybrid deep transfer learning model with machine learning methods for face mask detection in the era of the COVID-19 pandemic. *Measurement* **167**, 108288 (2021)
20. Zhang, J., Han, F., Chun, Y., Chen, W.: A novel detection framework about conditions of wearing face mask for helping control the spread of COVID-19. *IEEE Access.* **9**, 42975–42984 (2021)
21. Inamdar, M., Mehendale, N.: Real-time face mask identification using facemasknet deep learning network. *Avail. SSRN* (2020)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.