IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Traitement du Signal
Vol. 38, No. 4, August, 2021, pp. 947-953
Journal homepage: http://iieta.org/journals/ts

Comparison of the Effectiveness of Deep Learning Methods for Face Mask Detection

Onur Gedik, Ayşe Demirhan*

Department of Electrical-Electronics Engineering, Gazi University, Ankara 06560, Turkey

Corresponding Author Email: ayseoguz@gazi.edu.tr

https://doi.org/10.18280/ts.380404

Received: 25 March 2021
Accepted: 22 July 2021

Keywords:
CNN, deep learning, face mask detection, transfer learning

ABSTRACT

The usage of mask is necessary for the prevention and control of COVID-19 which is a respiratory disease that passes from person to person by contact and droplets from the respiratory tract. It is an important task to identify people who do not wear face mask in the community. In this study, performance comparison of the automated deep learning based models including the ones that use transfer learning for face mask detection on images was performed. Before training deep models, faces were detected within images using multi-task cascaded convolutional network (MTCNN). Images obtained from face mask detection dataset, COVID face mask detection dataset, mask detection dataset, and with/without mask dataset were used for training and testing the models. Face areas that are detected with MTCNN were used as input for convolutional neural network (CNN), MobileNetV2, VGG16 and ResNet50. VGG16 showed best performance with 97.82% accuracy. MobileNetV2 showed the worst performance for detecting faces without mask with 72.44% accuracy. Comparison results show that VGG16 can be used effectively to detect faces without mask. This system can be used in crowded public areas to warn people without mask that may help the reduce the risk of pandemic.

1. INTRODUCTION

COVID-19 is an infectious disease caused by a type of coronavirus which is first discovered in December 2019. The use of mask is necessary for the prevention and control of COVID-19 since it is a respiratory disease that passes from person to person by contact and droplets from the respiratory tract. Droplet contamination occurs when a person is in close contact with an infected person. Respiratory droplets that may cause infection can enter through the mouth, nose, or eyes as a result of coughing, sneezing, or very close contact [1]. Masks are used by people to protect himself or herself while in contact with an infected person. This will also prevent the next transmission of the virus.

World Health Organization (WHO) and UNICEF advise the children aged 5 years and under should not wear masks, children aged 6-11 should make a decision to wear masks for some factors like the transmission in the area, the ability of the child to safely and appropriately use a mask, etc. Children aged 12 and over and all adults are advised to wear masks to suppress transmission and save lives [2]. There are also recommendations on WHO's website for fabric mask materials, how to wear medical, non-medical, and fabric masks safely.

There are penalties such as fines for the people who do not wear face mask at almost all countries. In some countries the punishment for not wearing a mask can be even more severe. For example, UK and Indonesian authorities punished eight people they detected not wearing face masks by digging the graves of those who died from the coronavirus [3]. Therefore, it is an important task to identify people who do not wear face masks in the community.

Creating decision support models is of great importance to prevent and control the spread of the virus [4]. Most of these studies use deep learning methods because of their great success in image and video processing tasks. Deep learning is a branch of machine learning that has very complex layers to process information in a non-linear way. Some factors make deep learning techniques, which have a deeper structure than artificial neural networks, become popular today. The most important of these factors is that the amount of data gets larger every day and the development of computer technologies that can process this huge data. Graphical processing units (GPUs) have made a significant contribution to the development of deep learning with the ability to work with large amount of data and execute parallel operations. One of the most important advantages of deep learning is that it eliminates the need to manually extract features and does feature extraction on its own. Thanks to this automatic feature extraction, the learning patterns of computers have become even more precise.

Chavda et al. [5] have used multi-step convolutional neural network (CNN) model for face mask detection. Their data set included images from RMFRD (Real-world Masked Face Recognition Dataset) and Face Mask Detection dataset. Their dataset included 7,855 in total with 4,415 unmasked and 3,440 masked images. They used NASNetMobile, DenseNet121, MobileNetV2 models. The success rate of this study was 99.40%. Jiang et al. have used multistage CNN models for the face mask. They used ResNet and MobileNet models and Wider Face and Masked Faces data sets in their experimental studies. Success rate of mask recognition was 93.4% in this study [6]. Loey et al. [7] used multistage CNN models for face mask detection, too. They trained the system with 60 epochs and 64 batch size. Stochastic gradient descent and Adam methods were used as optimization algorithms. The data set was divided into 70% training, 10% validation, and 20% test.

They used ResNet50 model and MMD (Medical Masks Dataset) and FMD (Face Mask Dataset) datasets. Their highest success rate was 81%. Bhuiyan et al. used YOLOv3 for face mask detection. They used 650 images of both mask and no-mask collected with a web-scraping tool from websites. They used 80% of this data set as training and the rest as validation. 4,000 epoch training was performed and 96% success was achieved in their study [8].

In this study, multi-task cascaded convolutional network (MTCNN) was used to detect the face areas in the images. 9848 images obtained from face mask detection dataset [9], COVID face mask detection dataset [10], mask detection dataset [11], and with/without mask dataset [12] were used in this study. Different deep learning models have been utilized to classify the faces in the images as masked and unmasked. In the experimental studies, data preparation, data pre-processing, and transfer learning techniques were used to increase the classification performance. In this study, the pre-trained deep learning models VGG16 and ResNet50 yielded successful results in classifying facial images as masked and unmasked. Figure 1 shows the block diagram of the workflow.

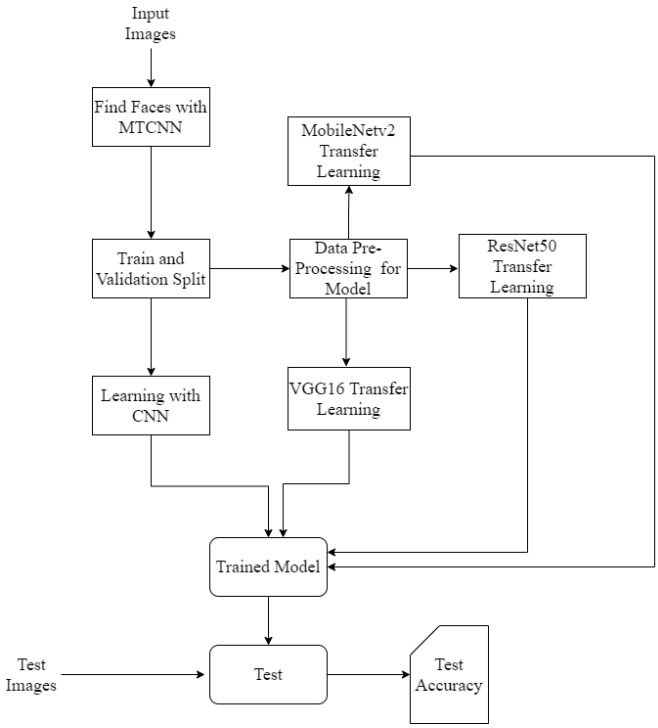


Figure 1. Block diagram of the workflow

2. MATERIALS AND METHODS

2.1 Dataset

8,331 of the total 9,848 images obtained from the face mask detection dataset [9], COVID face mask detection dataset [10], mask detection dataset [11], and with/without mask dataset [12] were used for training.

Face mask detection dataset includes RGB images to detect faces without mask because wearing face mask became mandatory worldwide due to COVID-19 [9]. COVID face mask detection dataset includes handpicked assemble of different images collected from Google and other image datasets. Images in the dataset are unified frontal faced mask or non-mask images [10]. Mask detection dataset was constructed using the images collected from Google image. This dataset includes face images of two classes; with mask and the without mask. In this dataset, there are faces with different ages, sexes and colors, with or without beard [11]. With/without mask dataset contains images of faces and faces with masks. In this dataset, masks are added subsequently to the images to create a masked face dataset [12]. There are approximately 30% people with different skin color, 50% female and 50% male in the dataset. Images that contain faces with mask and without mask are approximately equal in the dataset.

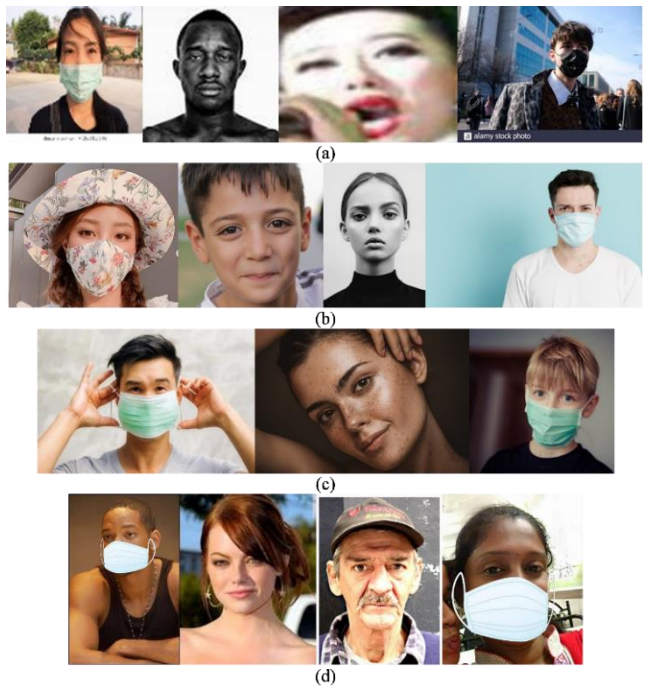


Figure 2. Images from the (a) face mask detection dataset, (b) COVID face mask detection dataset, c) mask detection dataset and (d) with/without mask dataset

All the images were resized as 224 x 224 x 3 before using as input in the model. 3,993 of the 8,331 training images include faces with masks and 4,338 of them include faces without masks.

Images in the training data set were divided into two parts as training and validation. 78% of the training data set was used for training while the remaining 22% was used for validation.

A separate test data set was created using the same reference datasets. Number of images in the test dataset was 1,517. Examples of images used for training and test of the models are given in Figure 2.

2.2 Multi-task cascaded convolutional network (MTCNN)

The image of the face is sufficient to identify whether a person is wearing a mask or not. Besides, when training a model, areas other than the face in the image may cause the model to be trained incorrectly. For this reason, detecting faces in the images and using only faces in training increases the performance of the model. For face detection MTCNN which is a pre-trained model was used in this study [13].

With MTCNN, facial positions can be determined from the images that contain more than one person. With this information it can be detected that if a person is wearing a

mask or not for each person in the image. When using the MTCNN, it returns the starting coordinates of the faces, width, height, position of the right and left eyes, where the left and right side of the mouth begins, and the coordinates of the nose. In this study, the positions of the faces were determined by using the initial coordinates, width, and height values.

2.3 Convolutional neural network

Convolutional neural networks (CNNs) are a type of artificial deep neural network specialized to process multidimensional, big data. Convolutional networks use the convolution process in at least one layer instead of the general matrix multiplication. Convolution layer contains the learnable filter set. Filter size, number of filters, and step range are important parameters for this layer. These filters are applied to the image and extract the properties of the image. The resulting filter values are sent to the pooling layer. In the pooling layer, the number of filters does not change, only size reduction occurs in terms of height and width. Max pool and mean pool are the most common pooling methods that give the maximum and mean values as output in the area covered by the filter [14]. Schematic diagram of the CNN is given Figure 3.

Figure 3. Schematic diagram of the CNN

Activation function was used to add nonlinearity to the neural networks since the real-world features are usually nonlinear. Activation functions process the incoming data and transform it to different values. This process significantly affects the performance of CNN. Therefore, the activation function selection should be made correctly. The most commonly used activation functions are hyperbolic tangent, sigmoid, rectified linear unit (ReLU), and leaky ReLU. ReLU is one of the most commonly used activation function because of its speed and high success. It works as a linear function if the input is positive, and it returns zero if the input is negative [15].

Dropout is the process of forgetting. It is used to prevent overfitting by forgetting some neurons (assigning zero) during training. Batch normalization is a method used to make neural networks faster and more stable. It normalizes the input layer by re-centering and rescaling. The fully connected layer makes the tensor data flat by vectorization. It is the layer where the tensor formed by n matrices of f x f dimensions is converted into a vector of f x f x n. Finally, as many neurons as our class number were used in the output layer because this is the layer that performs classification. Softmax is the most used activation function in the output layer. Optimization methods are used to find the optimum value in the solution of nonlinear problems. Optimization algorithms such as stochastic gradient descent, AdaGrad, AdaDelta, Adam, AdaMax are widely used in deep learning applications. These algorithms differ in performance and speed. In this study, Adam was used as an optimizer.

The epoch number determines how many times the data set will be passed and the batch size value determines how many times data is given to the model [14, 15]. Parameters used in this study are given in Table 1.

Table 1. CNN parameters

Parameter	Value
Input size	224 x 224 x 3
Number of layers	7
Activation function	ReLU Softmax for output layer
Normalization	Batch
Dropout	Pooing layers 60% Joint layer 70%
Optimization method	Adam
Loss function	Categorical cross entropy

2.4 Transfer learning

Transfer learning is freezing certain layers of a trained model and using it as a starting point for another model for a different task. It increases performance significantly, especially for small data sets [16]. In this study pre-trained VGG16, ResNet50, and MobileNetV2 were used to detect faces without mask.

In the VGG16 model, the input images were 224 x 224 x 3 RGB images. The most important feature of VGG16 is that the convolution is performed more than once after pooling. The ReLU activation function is used in the convolution layer. Maximum pooling is applied in the pooling layer. VGG16 model achieves the top-5 test accuracies in ImageNet 2014 competition. ImageNet contains 14 million images belonging to 1,000 classes [17].

ResNet50 is a deep residual network model that is 50 layers deep. It is also pre-trained on ImageNet dataset and allows to train extremely deep neural networks without vanishing gradients problem. During backpropagation of the network, taking the partial derivative (gradient) of the error function relative to the weights available on each training epoch requires the multiplication of n of these numbers to compute gradients in the first layers of an n-layer network. When the network is deep and these numbers are small, the product of n of the numbers becomes zero which is called vanishing gradients. ResNet50 was the winner of ImageNet 2015 challenge [18].

MobileNetV2 has a general purpose deep CNN architecture that aims to perform well on mobile vision applications. Classification, segmentation and object detection are the supported mobile visual recognition tasks of MobileNetV2. MobileNetV2 is an improved version of MobileNetV1 that can run deep neural networks on mobile devices [19]. MobileNetV2 have linear bottlenecks between the layers. There are also connections between the bottleneck layers. Bottlenecks encode the intermediate inputs and outputs of the model. The intermediate layer uses filters to transform the lower level features to higher level descriptors. Inverted residual structure of MobileNetV2 makes it faster and successful [20].

3. RESULTS AND DISCUSSION

The first step of this study was finding the facial regions in the images using MTCNN. MTCNN returns the pixel

coordinates of the faces found in the given input RGB image. The pixels that are in these returned coordinates were taken as faces and presented to our deep model for training.

In the second step, it was found out whether there was a mask in the face area or not by using the CNN, MobileNetV2, VGG16, and ResNet50 models.

While training CNN, 20% of 8,331 images were used for validation and 80% for training. Parameters given in Table 1 were used for training. The model has 7 layers including the output layer. In layer 1, convolution process with 64 filters with 3 x 3 dimension was applied. Batch normalization and ReLU activation function were used. In layer 2, convolution process with 64 filters with 3 x 3 dimension was applied. After applying batch normalization and ReLU activation function, a max pooling of 2 x 2 with stride 2 was performed. After pooling, overfitting was prevented with 0.6 dropout. In layer 3, convolution process with 32 filters with 3 x 3 dimensions was applied. Batch normalization and ReLU activation function were applied. In layer 4, convolution process with 32 filters with 3 x 3 dimensions was applied. Batch normalization and ReLU activation function were applied. In layer 5, convolution process with 32 filters with 3 x 3 dimensions was applied. After the batch normalization and ReLU activation function, a max pooling of size 2 x 2 with stride 2 was performed. After the pooling process, 60% forgetting was applied with dropout. Over-learning was tried to be prevented with dropout. Layer 6 was a full connection layer where the vectorization was done with flatten. Data with dimensions of 52 x 52 x 32 has become 86,528 by vectorization. Then, it was reduced to 128 neurons, and batch normalization and ReLU activation function were applied. Finally, with dropout, forgetting process was applied with 70% ratio. Layer 7 was the output layer. Since there are two classes as masked and unmasked, the number of neurons in this layer was 2. Softmax activation function was applied for the classification process.

In the training of the CNN, the number of epochs was 40 and the batch size was 100. The best result was recorded at epoch 14 of 40 epoch training with 0.9556 validation accuracy and 0.1223 validation loss. The loss and accuracy curves of the training per epoch are given in Figure 4 (a) and (b), respectively.

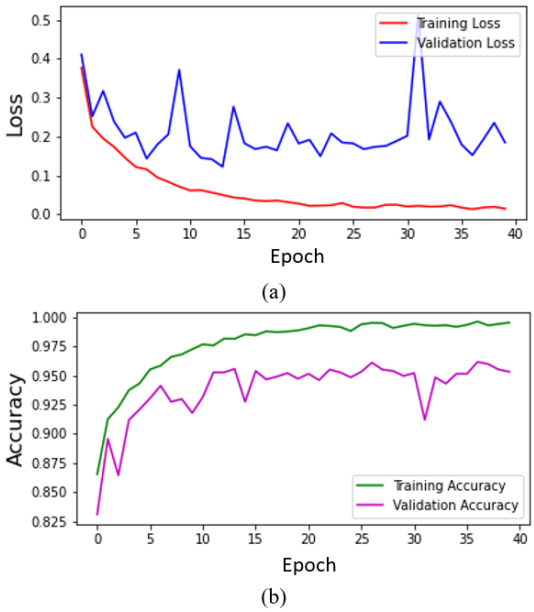


Figure 4. (a) Loss and (b) accuracy curves of CNN

MobileNetV2 training was performed using 22% of 8,331 images for validation and 78% for training. MobileNetV2 uses ImageNet weights. Layers before the output and full connection layer were frozen. The output and full connection layer of the model were removed. The average pooling operation was performed using 7 x 7 filters then vectorization was performed using flatten. Then the neuron size was reduced to 128 and ReLU activation function was applied. With 50% dropout, forgetting of the model was performed. Softmax activation function was applied with 2 neurons for output layer. Loss function of the model was categorical cross entropy, optimization function was Adam, and evaluation metric was accuracy. Training epoch number was chosen as 20 and the batch size was 32. Best validation accuracy and loss were obtained at the 15th epoch of the total 20 epochs as 0.9231 and 0.1997, respectively. The loss and accuracy curves of the MobileNetV2 training are given in Figure 5 (a) and (b), respectively.

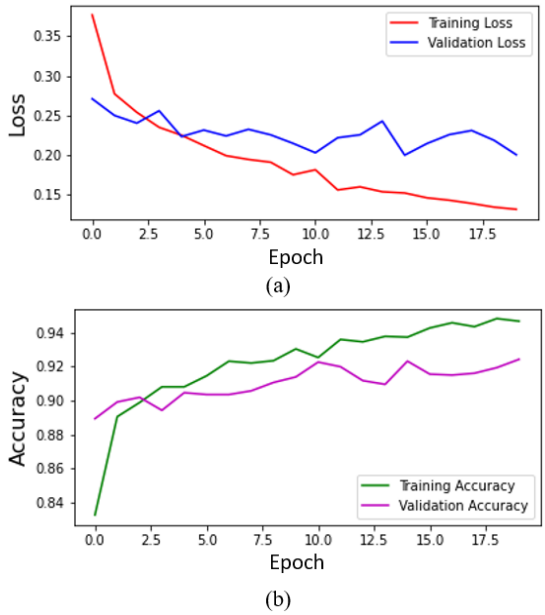


Figure 5. (a) Loss and (b) accuracy curves of MobileNetV2

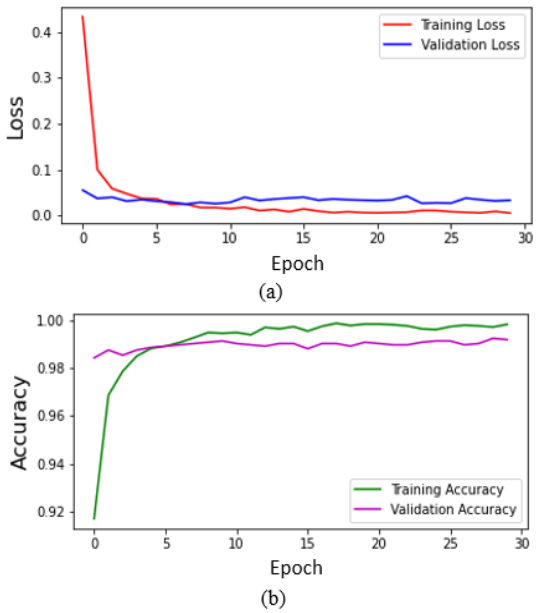


Figure 6. (a) Loss and (b) accuracy curves of VGG16

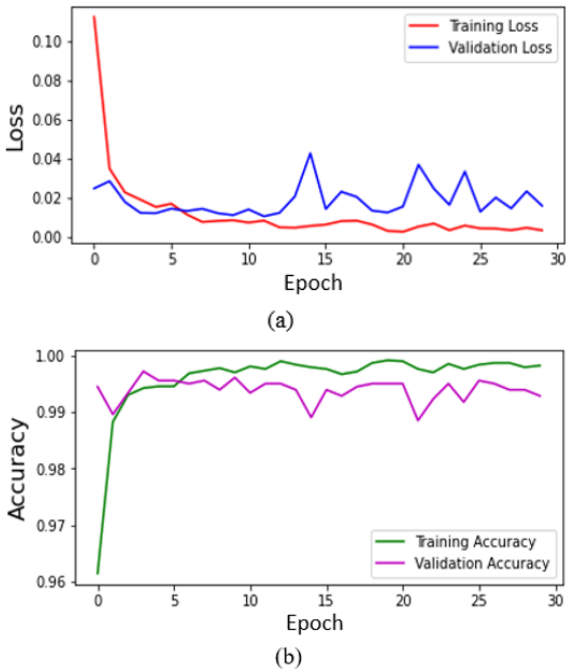


Figure 7. (a) Loss and (b) accuracy curves of ResNet50

For training VGG16, the same parameters and the number of validation and training images with MobileNetV2 were used. Training epoch number was 30 and the batch size was 80 for VGG16. Best validation accuracy and loss were obtained at the 8th epoch of the total 30 epochs as 0.9902 and 0.0247, respectively. The loss and accuracy curves of the VGG16 training are given in Figure 6 (a) and (b), respectively.

Parameters and number of validation and training images used training of the ResNet50 was same with MobileNetV2 and VGG16. Training epoch number was 30 and the batch size was 80 for ResNet50. Best validation accuracy and loss were obtained at the 12th epoch of the total 30 epochs as 0.9951 and 0.0103, respectively. The loss and accuracy curves of the ResNet50 training are given in Figure 7 (a) and (b), respectively.

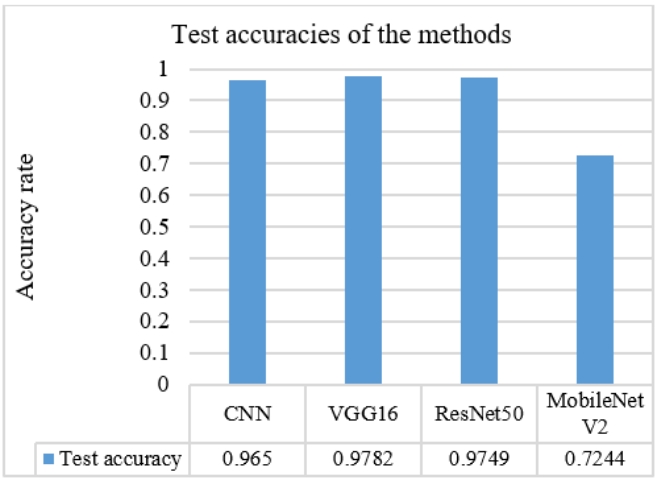


Figure 8. Test accuracies of the CNN, VGG16, ResNet50 and MobileNetV2

Performance of the all used methods were evaluated on a separate test set that has 1,517 images. Test accuracies for CNN, MobileNetV2, VGG16 and ResNet50 were 96.5%, 72.44%, 97.82%, 97.49%, respectively. Figure 8 gives test accuracies of the used methods as a graph. According to the test accuracies, the most successful method for detecting the faces without mask was VGG16 and the least successful method was MobileNetV2. Accuracies of VGG16 and ResNet50 were very close. CNN was more successful than MobileNetV2 but could not perform better than VGG16 and ResNet50.

Table 2. Confusion matrix of CNN

		Predicted	
		With Mask	**Without Mask**
Actual	**With Mask**	602	34
	Without Mask	19	862

Table 3. Confusion matrix of MobileNetV2

		Predicted	
		With Mask	**Without Mask**
Actual	**With Mask**	556	93
	Without Mask	325	543

Table 4. Confusion matrix of VGG16

		Predicted	
		With Mask	**Without Mask**
Actual	**With Mask**	625	11
	Without Mask	22	859

Table 5. Confusion matrix of ResNet50

		Predicted	
		With Mask	**Without Mask**
Actual	**With Mask**	625	11
	Without Mask	22	859

Figure 9. Result image that shows people with and without mask obtained from CNN

Figure 10. Result image that shows people with and without mask obtained from MobileNetV2

Figure 11. Result image that shows people with and without mask obtained from VGG16

Figure 12. Result image that shows people with and without mask obtained from ResNet50

Confusion matrixes obtained from the used methods are given in Table 2-5. Table 2 shows that CNN tends to classify people with mask as without mask. False negative rate of MobileNetV2 was very high as can be seen from Table 3 that leads it to label people without mask as with mask. Table 4 shows that VGG16 had the best performance for identifying people who do not wear face mask. It can be seen that ResNet50 was the best method for classifying people with mask from Table 5.

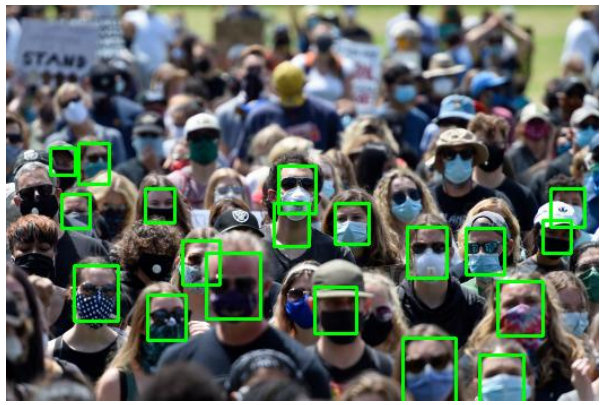
Results obtained from the methods using the same test image are given in Figures 9-12. People with mask are marked with green boxes around their faces while people without mask are marked with red boxes. In the test image, MTCNN fails to detect faces of 3 people with mask whose faces can be seen fully in the image and 1 person whose face can be seen partially. In Figure 9, CNN misclassifies 2 people near the lower left corner of the image as without mask. MobileNetV2 shows the worst performance that can also be seen from Figure 10. It misclassifies 13 people with mask as without mask. Figure 11 shows the performance of VGG16. It misclassifies a person who wears a black mask as without mask. In Figure 12, the same person with black mask was misclassified by ResNet50, too. VGG16 and ResNet50 classified correctly all other faces detected in the test image.

The best result obtained from this study is compared to the other studies that detect people with and without mask using deep learning techniques in Table 6. It can be seen from the table that DenseNet121 shows the best performance for face mask detection and VGG16, which is used in this study, gives the second best result compared to the other methods.

The trained models were also evaluated for the dark colored and printed masks since the diversity of the face masks increase every day. Evaluation results show that our deep models were successful for determining the masks with different colors and prints. The results obtained from VGG16 for two different images that people wear dark color and printed face masks are given in Figure 13.

Table 6. Accuracy comparison of the methods

Reference	Method	Accuracy
Chavda et al. [5]	DenseNet121	99.40%
Jiang et al. [6]	ResNet	93.40%
Loey et al. [7]	YOLOv2 with ResNet	81.00%
Bhuiyan et al. [8]	YOLOv3	96.00%
This study	VGG16	97.82%



(a)

(b)

Figure 13. Result images obtained from VGG16 for (a) dark color (b) printed face masks

4. CONCLUSIONS

Performance comparison of the deep learning methods for face mask detection was performed in this study. Identifying people who do not wear face masks in the community is an important task since COVID-19 is a respiratory disease that passes from person to person by droplets from the respiratory tract where mask protect people from the infected ones. Deep learning methods were used in this study because of their great success in image and video processing tasks. CNN, MobileNetV2, VGG16 and ResNet50 were the used deep leaning models. CNN was trained from the scratch and transfer learning is applied for the other methods. 9,848 images obtained from four different face mask dataset were used. The best face mask detection performance was obtained from VGG16 with 97.82% accuracy. It is shown that VGG16 can be used effectively to detect faces without mask. This system can be used in crowded public areas to warn people without mask that may help the reduce the risk of pandemic.

REFERENCES

[1] Fahmi, I. (2019). World Health Organization coronavirus disease 2019 (COVID-19) situation report. DroneEmprit. https://www.who.int/docs/default-source/coronaviruse/situation-reports/20200402-sitrep-73-covid-19.pdf?sfvrsn=5ae25bc7_6, accessed on Sept. 04, 2021.

[2] World Health Organization. Coronavirus disease (COVID-19) advice for the public: When and how to use masks. https://www.who.int/emergencies/diseases/novel-coronavirus-2019/advice-for-public/when-and-how-to-use-masks, accessed on Sept. 04, 2021.

[3] Punishment for not wearing a face mask? Dig COVID-19 victims' graves. https://www.france24.com/en/20200915-punishment-for-not-wearing-a-face-mask-dig-covid-19-victims-graves, accessed on Sept. 04, 2021.

[4] Sousa, J., Barata, J. (2021). Tracking the wings of COVID-19 by modeling adaptability with open mobility data. Applied Artificial Intelligence, 35(1): 41-62. https://doi.org/10.1080/08839514.2020.1840196

[5] Chavda, A., Dsouza, J., Badgujar, S., Damani, A. (2020). Multi-stage CNN architecture for face mask detection. arXiv preprint arXiv:2009.07627.

[6] Jiang, M., Fan, X., Yan, H. (2020). Retinamask: A face mask detector. arXiv preprint arXiv:2005.03950.

[7] Loey, M., Manogaran, G., Taha, M.H.N., Khalifa, N.E.M. (2021). Fighting against COVID-19: A novel deep learning model based on YOLO-v2 with ResNet-50 for medical face mask detection. Sustainable Cities and Society, 65: 102600. https://doi.org/10.1016/j.scs.2020.102600

[8] Bhuiyan, M.R., Khushbu, S.A., Islam, M.S. (2020). A deep learning based assistive system to classify COVID-19 face mask for human safety with YOLOv3. In 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT), pp. 1-5. https://doi.org/10.1109/ICCCNT49239.2020.9225384

[9] Kaggle, Gurav, O. (2021). Face Mask Detection Dataset. https://www.kaggle.com/omkargurav/face-mask-dataset, accessed on Sept. 04, 2021.

[10] Kaggle Mitra, P. (2021). COVID Face Mask Detection Dataset. https://www.kaggle.com/prithwirajmitra/covid-face-mask-detection-dataset, accessed on Sept. 04, 2021.

[11] Kaggle Abdelatif, M. (2021). Mask Detection. https://www.kaggle.com/moussaid/mask-detection?select=face_mask_data, accessed on Sept. 04, 2021.

[12] Kaggle Pandit, N. (2021). With/Without Mask. https://www.kaggle.com/niharika41298/withwithout-mask, accessed on Sept. 04, 2021.

[13] Zhang, K., Zhang, Z., Li, Z., Qiao, Y. (2016). Joint face detection and alignment using multitask cascaded convolutional networks. IEEE Signal Processing Letters, 23(10): 1499-1503. https://doi.org/10.1109/LSP.2016.2603342

[14] Chollet, F. (2018). Deep Learning with Python. New York, Manning, pp. 119-177.

[15] Towards Data Science, Kızrak, A. (2021). Comparison of Activation Functions for Deep Neural Networks. https://towardsdatascience.com/comparison-of-activation-functions-for-deep-neural-networks-706ac4284c8a, accessed on Sept. 04, 2021.

[16] Song, X.R., Gao, S., Chen, C.B., Wang, S.L. (2020). A novel face recognition algorithm for imbalanced small samples. Traitement du Signal, 37(3): 425-432. https://doi.org/10.18280/ts.370309

[17] Simonyan, K., Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556.

[18] He, K., Zhang, X., Ren, S., Sun, J. (2016). Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770-778. https://doi.org/10.1109/10.1109/CVPR.2016.90

[19] Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M., Adam, H. (2017). MobileNets: efficient convolutional neural networks for mobile vision applications. arXiv preprint arXiv:1704.04861.

[20] Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., Chen, L.C. (2018). MobileNetV2: Inverted residuals and linear bottlenecks. IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, Utah, pp. 4510-4520. https://doi.org/10.1109/CVPR.2018.00474