# Masked-Face Classification and Recognition Using CNNs and Transfer Learning

**Jawher DRIDI, Mohamed BOUZID**, Engineering Students at Ecole Polytechnique de Tunisie
Supervised by :Mme **Wided SOUIDENE**

***Abstract*—Nowadays, the mask has become the essential accessory to fight against the spread of the virus. Researchers started developing a lot of Artificial Intelligence algorithms to deal with the problems related to masks. Most of the algorithms and techniques that have been released are related to whether classification problems or recognition problems. In this project, we developed our own algorithm of classification and recognition. Our method will give us information about people who are respecting the sanitary protocol, and will make it easy to recognize people wearing masks. We developed five models that will treat the classification of people wearing masks and not, as well as a technique of classification of people wearing masks by gender, and an algorithm that will allow us to recognize the masked faces. In our methods, we used a model developed from scratch using CNNs and a model using Transfer Learning to fix the case of insufficient amount of data especially with the case of the recognition task. All these models have been trained using self-collected Datasets. The results obtained for the different models prove the robustness and the reliability of our work.**

***Index Terms*—Face Detection, Face Recognition, Masked-Face, Gender Masked-Face,CNNs, Transfer Learning, Machine Learning.**

## I. INTRODUTION AND MOTIVATION

Due to the Coronavirus, the deaths rate is increasing day by day. That is why, authorities made the wear of masks obligatory in the public areas. To make sure that workers in plants, students in schools, and people generally are wearing their masks everywhere, a lot of artificial intelligence programs have been developed since the coming of the coronavirus. Also, to adapt new technologies to the new faces that wear masks, the old techniques of recognition in smart devices have been adapted to suit the new situation. Through the months, researchers have found a lot of techniques that allowed them to detect the presence of masks with high accuracy using image processing and computer vision methods. As mask is distinguished by its color, color analysis technique has been widely used like the use of CNNs to extract relevant features that best define the colors for instance. Deep Learning techniques helped researchers a lot to extract relevant features that best represent the problem. Indeed, Neural Networks have been successfully used in several fields such as image classification, speech recognition, face recognition, self-driving cars, cancer detection, etc. For all these applications, deep learning proved its efficiency in detecting different classes of objects. For the task of detection of the presence of masks a lot of techniques have developed from scratch like our methods that we will introduce them throw these papers. In this context, we present in this project a mask detection via simple classification, masked-face gender classification, and a masked-face recognition system based on deep learning using neural networks and transfer learning. For this purpose, we prepare the data by applying different type of data augmentation to fix the problem of insufficient data. Then, we feed our data to the different type of model depending on the case that we are treating. In this project, we developed our own three datasets. Indeed, our datasets are a combination of a lot of public datasets and self-collected data that have been adapted to the different situations that we will deal with during this project. More specifically, this paper makes three main contributions :

- Five models with high performance on high and small size masked faces.
- A model able to recognize masked faces.
- A model with a capability of distinguishing between male and female wearing masks.

The remainder of the paper is organized as follows:Chapter we give some related works specifically Masked Face Recognition Using Convolutional Neural Network. Next, in Chapter 2 we provide a description of the proposed deep learning architectures. In chapter 3, the experimental results are presented. Finally, we conclude the project.

## II. RELATED WORK : MASKED FACE RECOGNITION USING CONVOLUTIONAL NEURAL NETWORK

A feasible approach has been developed since the coming of coronavirus. Indeed, it consists of training a model to classify images that contain faces. Then, it detects and crops faces using multi-Task Cascade Convolutional Neural Networks (MTCNN). After, it implies some image preprocessing that is needed in order to apply Facenet in the next stage. the Facenet Extracts useful information about each image. Then, we create an embedded system using these features. The final information will be classified using an SVM model to recognize different people[1]. The general architecture is attached below in the figure 1.

In this section we have detailed an architecture that is related to our work, in the next section, we provide a detailed architecture of our original techniques.

## III. PROPOSED METHOD

In this section, we will detail our proposed architectures that will take an original image containing a face as input and provide us with results as output that depends on the case that we are dealing with ( classification masked or not masked, classification male masked or female masked, or face recognition).

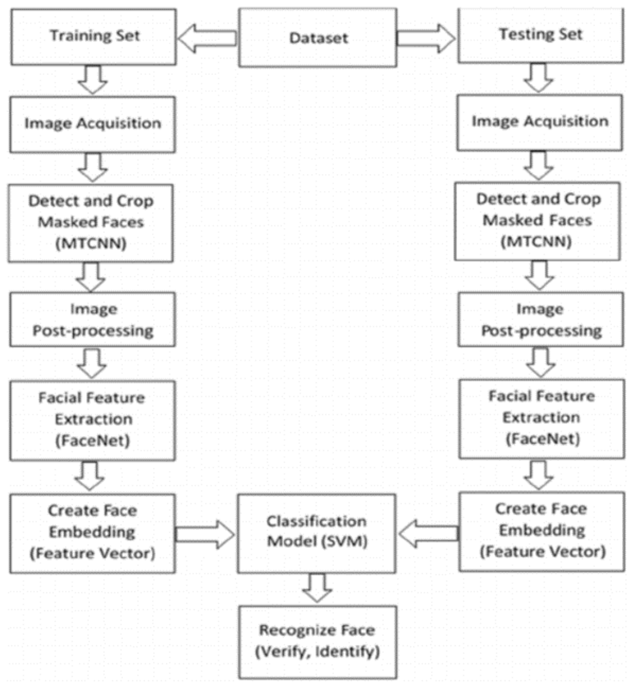


Fig. 1: Image Recognition Architecture

### A. First Proposed Architecture : Neural Network Model using CNNs

*1) Input:* Our model takes in 120x120 pixel RGB images. Indeed, all the initial images have been resized to the desired size(120x120).

*2) Convolutional Layers:* The convolutional layer uses very small filters (3x3). It is the minimum size that can still capture left/right and up/down. All the convolutional layers are followed by a RELU linear activation function.

*3) Pooling Layers:* Max Pooling was employed in this architecture after each convolutional layer with strides of dimension 2x2. The objective of these layers is to down-sample the previous output.

*4) Fully-Convolutional Layers:* This architecture contains two Fully-convolutional layers: the first one contains 3200 channels and ends up with a RELU activation function, and the second one has 2 or 6 channels depending on the case that we are dealing with, one for each class.

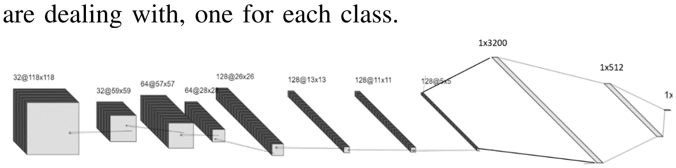


Fig. 2: 1st Model Architecture

### B. Second Proposed Architecture : Neural Network Model using MobileNetV2

*1) Input:* Our model takes in 120x120 pixel RGB images. Indeed, all the initial images have been resized to the desired size(120x120) like the same case with the first architecture.

*2) MobileNetV2:* The input images are fed into a pretrained MobileNetV2. Indeed, we will use the knowledge stored in this model and we will apply it to our case willing to improve the performance of our new model. The MobileNetV2 contains 30 layers distributed as follows :

- Convolutional layer with stride 2
- Depthwise layer
- Pointwise layer that doubles the number of channels
- Depthwise layer with stride 2
- Pointwise layer that doubles the number of channels

Figure 3 gives the general architecture of the MobileNetV2.

| Type / Stride | Filter Shape |
|---|---|
| Conv / s2 | $3 \times 3 \times 3 \times 32$ |
| Conv dw / s1 | $3 \times 3 \times 32$ dw |
| Conv / s1 | $1 \times 1 \times 32 \times 64$ |
| Conv dw / s2 | $3 \times 3 \times 64$ dw |
| Conv / s1 | $1 \times 1 \times 64 \times 128$ |
| Conv dw / s1 | $3 \times 3 \times 128$ dw |
| Conv / s1 | $1 \times 1 \times 128 \times 128$ |
| Conv dw / s2 | $3 \times 3 \times 128$ dw |
| Conv / s1 | $1 \times 1 \times 128 \times 256$ |
| Conv dw / s1 | $3 \times 3 \times 256$ dw |
| Conv / s1 | $1 \times 1 \times 256 \times 256$ |
| Conv dw / s2 | $3 \times 3 \times 256$ dw |
| Conv / s1 | $1 \times 1 \times 256 \times 512$ |
| $5\times$ Conv dw / s1 <br> Conv / s1 | $3 \times 3 \times 512$ dw <br> $1 \times 1 \times 512 \times 512$ |
| Conv dw / s2 | $3 \times 3 \times 512$ dw |
| Conv / s1 | $1 \times 1 \times 512 \times 1024$ |
| Conv dw / s2 | $3 \times 3 \times 1024$ dw |
| Conv / s1 | $1 \times 1 \times 1024 \times 1024$ |
| Avg Pool / s1 | Pool $7 \times 7$ |
| FC / s1 | $1024 \times 1000$ |
| Softmax / s1 | Classifier |

Fig. 3: MobileNetV2 Architecture

*3) Global Average Pooling and Fully Convolutional Layer:* After applying the pretrained model, we feed the output to a Global Average Pooling. Finally, we feed the results to an FCL with 1 or 6 channels depending on the case that we are studying.

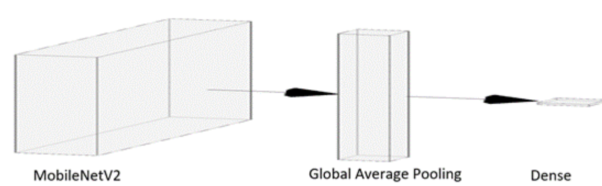


Fig. 4: 2nd Architecture

### C. The studied Cases

*1) Mask Detection via simple Classification:* When dealing with the classification of images whether they contain masked faces or not masked faces, we used the two architectures explained before. We wanted to study the difference between the two architectures in this case by comparing the efficiency of each model.

*2) Masked-Face Gender Classification:* In this case, the task started to be a bit challenging. The non-availability of the data was our first challenge. Second, extracting features to classify people whether they are male or female, while they are wearing masks, was not that evident. Indeed, a lot of principle features will be hidden from the model. During this task, we also examined both architectures to see the difference and to improve the performance to the maximum.

*3) Face Recognition:* This task was the most challenging compared to the previous ones. Indeed, the small amount of data and the best choice of the model were our blocks. We chose to use transfer learning (second architecture) to solve the problem of the small amount of data.

In this section, we explained the different architectures and cases that we will be dealing with during this project.

## IV. Implementation and Results

In this section, we detail the implementation settings we followed to train and test our proposed techniques. Namely, the data preparation step (dataset and data augmentation), then finally the overall training materials used in our framework.

### A. Datasets

In this project, we created our own datasets whether by collecting data from other datasets such as â€œCOVID19 Mask Image Datasetâ€ or by looking for photos from Google Images. We prepared three datasets:

- 1st dataset : containing photos of masked faces and not masked faces for the mask classification task.
- 2nd dataset : containing masked faces by gender for the gender classification task.
- 3rd dataset : containing masked faces of some famous people for the face recognition task.

(a) (b) (c)

Fig. 5: Examples from the Dataset: (a) : 1st dataset - (b) : 2nd dataset - (c) : 3rd dataset

### B. Data Augmentation

It is important to use data augmentation techniques if want to improve the performance of our model and avoid overfitting. Mainly, it consists of applying transformations on the image such as geometrical transformations (rotation, scaling, zooming, shifting, and flip translation) and photometric transformation (brightness, contrast, and shear) :

- Flipping: The main applications have three horizontal flips, vertical flips, and horizontal flips.
- Cropping or Zooming: makes the network less sensitive to scales, so it can identify smaller objects.
- Image shifting: Consists of moving elements of a ndarray along any dimension.
- Rotation, shear, image scale, etc.

In our project, we used a lot of data augmentation techniques. Indeed, using these data augmentation techniques helped increase the performance.

### C. Training

During the implantation, we trained the five models using Tensorflow on a machine with GPU NVIDIA Tesla P100 16 GB. Moreover, we divided our datasets into 2 subsets each one as presented in the Tables below.

| | Number of Positive Images | Number of Negative Images |
|---|---|---|
| Training set | 1709 | 1930 |
| Testing set | 304 | 300 |

Fig. 6: 1st Dataset statistics

| | Number of Positive Images | Number of Negative Images |
|---|---|---|
| Training set | 770 | 770 |
| Testing set | 192 | 192 |

Fig. 7: 2nd Dataset statistics

| | Ronaldo | Neymar | Salah | Kendall | Hailey | Serena |
|---|---|---|---|---|---|---|
| Training set | 52 | 87 | 59 | 81 | 69 | 52 |
| Testing set | 4 | 5 | 6 | 20 | 5 | 4 |

Fig. 8: 3rd Dataset statistics

### D. Neural Network Model using CNNs

The input data are positive and negative PNG images. Our training was conducted using Sparse Categorical Cross entropy, an â€œadamâ€ optimizer, a batch size of 100, a number of epochs set to 20 epochs, due to the lack of high resources especially of GPU, and an image size set to 120x120. Note that the training time changes depending on the models since we trained this model on a different type of datasets.

### E. Neural Network Model using MobileNetV2

The input data are positive and negative PNG images (6 classes of famous people in the case of face recognition). We used Binary Cross as a loss function. We resized the images to 120x120 and we used 15 Epochs of training due to the lack of high resources especially of GPU. We used a learning rate of 0.01 for the Adam optimizer with a batch size of 8.

### F. Metrics

We used two metrics during this project. The accuracy for all models and the AUC for the transfer learning-based models.

- Accuracy: Accuracy is the quintessential classification metric : (TP+TN)/(TP+FP+FN+TN)
- AUC curve is defined by plotting the true positive rate (TPR) against the false positive rate (FPR) at various threshold settings.

### G. Results

In this section, we will present the performance of the different models in the different scenarios, and we will make some comparisons between the different methods in the same case.

*1) 1st Case : Mask Detection via simple Classification:* The table below shows the best scores that we got in this case.

| Model | Accuracy | AUC |
|---|---|---|
| NN using CNNs | 99% | -- |
| NN using MobileNetV2 | 100% | 100% |

Fig. 9: Mask Detection via simple Classification Results

The results of this case study are very excellent, and this can be explained by the clear difference between a masked face and a normal face. The use of transfer learning has improved further the performance.

*2) 2nd Case : Masked-Face Gender Classification:* The table below shows the best scores that we got in this case.

| Model | Accuracy | AUC |
|---|---|---|
| NN using CNNs | 75% | -- |
| NN using MobileNetV2 | 87% | 86% |

Fig. 10: Masked-Face Gender Classification Results

Having a model that classifies between masked people whether they are male or female is very challenging, and that is why the performance that we got is less than the previous case of simple mask classification. We can notice the impact of transfer learning on our results. Indeed, having a pretrained model that uses a stored knowledge from other data is very helpful in our case.

*3) 3rd Case : Face Recognition:* The table below shows the best scores that we got in this case.

| Model | Accuracy |
|---|---|
| NN using MobileNetV2 | 88% |

Fig. 11: Face Recognition Results

Using a self-made dataset about famous people, and using just transfer learning because of the small amount of data, we were able to get an acceptable performance in the face recognition task.

*4) Evolution of Metrics:* The following graphs show the evolution of the accuracy and the loss with the number of epochs.

The accuracy and the loss tend, with time, to converge to the desired values. We can notice here a lot of fluctuation due to the small number of Epochs. Indeed, we did not train the model for a big number of Epochs because of the limitation on the GPU that we used.

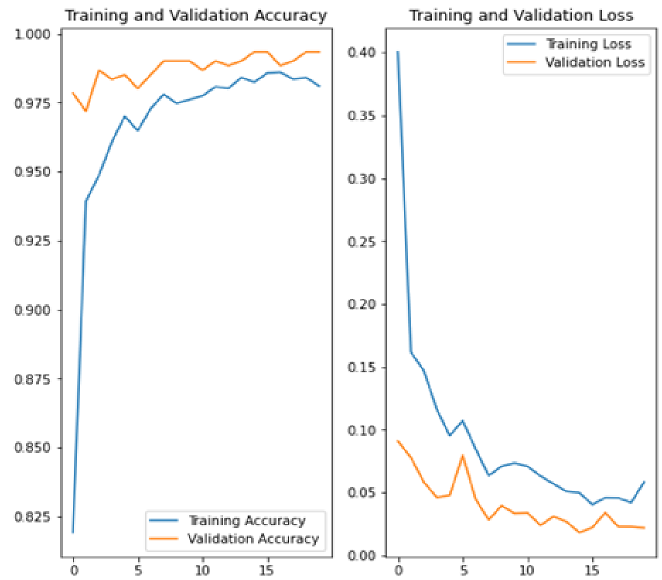


Fig. 12: 1st Model 1st Case

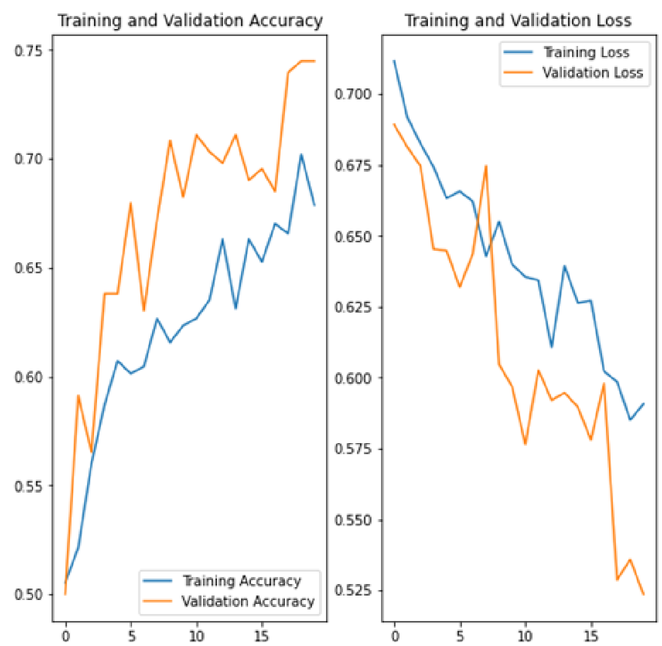


Fig. 13: 1st Model 2nd Case

*5) Evaluating the Models on Random Images:* In this final part, we will see the performance of our models in random images such as our own images and famous people images.

The figure 15 shows the output of the 2 model in the case of the mask detection via simple classification.The figure 16 shows the output of the 2 model in the case of the mask detection via simple classification. And, The figure 17 shows the output of the 2 model in the case of the face recognition.

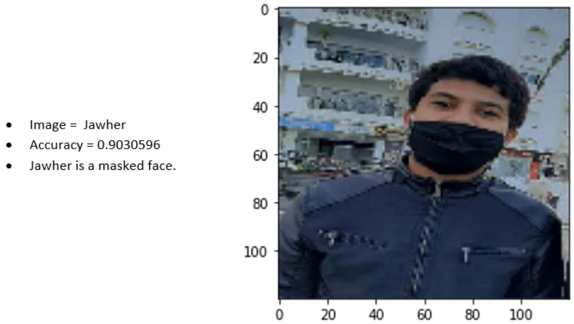


Fig. 14: Mask Detection via simple Classification

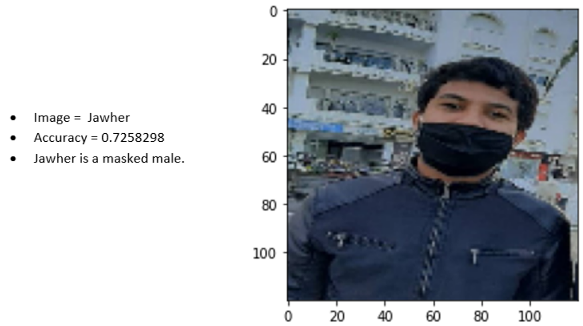


Fig. 15: Masked-Face Gender Classification

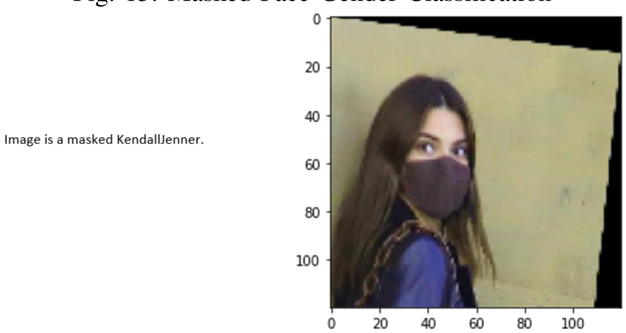


Fig. 16: Face Recognition

## V. Conclusion

In this project, we introduced mask detection via simple classification, masked-face gender classification, and masked-face recognition using five models based on CNNs and Transfer Learning. We got good performance, especially for the transfer learning models. Moreover, the mask detection via simple classification case was the easiest to deal with due to the simplicity of the classification. The face recognition task was the most challenging even though we got acceptable scores. We faced some challenges such as the limited amount of GPU and the lack of data for the face recognition task. We created our datasets for the different study cases using whether a combination of other datasets of by collecting images from Google Images.

## VI. Acknowledgement

I would like to express my gratitude and thanks to the person named below who gave us the opportunity to do this project, Mrs. Wided SOUIDENE, my professor at Ecole Polytechnique de Tunisie who taught us and gave us a lot of information about the different Image Processing Methods and Deep Learning Techniques.

## References

[1] Md Ejaz and Md Islam, "Masked face recognition using convolutional neural network," 12 2019, pp. 1–6.