Journal of Advanced Zoology

ISSN: 0253-7214
Volume 44 Issue S-6 Year 2023 Page 835:843

Mask Detection Using CNN and OpenCV

Amogh J Tambad[1*], Aditya G Tirumale[2], Aman Kumar Srivastava[3], Akash Hadagali[4], Thirumagal E[5]

[1,2,3,4,5]Department of Computer Science Engineering, REVA University, Bangalore, Karnataka, India
Email: r17cs025@cit.reva.edu.in[1], r17cs009@cit.reva.edu.in[2], r17cs019@cit.reva.edu.in[3], r17cs014@cit.reva.edu.in[4], thirumagal.e@reva.edu.in[5]

**Corresponding author's E-mail: r17cs025@cit.reva.edu.in*

Article History

Received: 06 June 2023
Revised: 05 Sept 2023
Accepted: 30 Nov 2023

Abstract

With Covid-19 tormenting the entire world, masks have become an essential item when stepping out of the house as it greatly prevents the spread of disease-causing germs and viruses. However, people often seem to take it lightly and not wear their masks when they step out. Therefore, we propose to design a mask detection system that alerts the concerned people with an alarm when the person in question is not wearing a mask. For this, we shall be using the medical imaging dataset obtained on Kaggle. In addition to this, we incorporate CNN and ResNet to train the model and OpenCV to test it with real-time data. We aim to obtain a high accuracy to ensure perfect face mask detection.

Keywords: *Covid-19, CNN, ResNet, medical imaging, Kaggle, OpenCV*

1. Introduction

With the increasing torment and influence of COVID-19 across the world, there have been significant changes in people's lifestyles. The biggest change, arguably, has been the frequent use of masks in public spaces. Almost all the countries recommend the use of masks whenever an individual is stepping out of his/her house or meeting people. Tom Li et al. [1] have evaluated the potential impact of wearing masks through mathematical modeling and the use of effective scientific evidence. They compared Asian countries that have made masks mandatory to other countries with no such rule. Finally, they concluded that the Asian countries had much fewer COVID cases when compared to other countries.

However, despite the government's strict policy and various awareness campaigns across the world, people have been extremely negligent in terms of wearing masks and have been flouting rules quite frequently. This is seen especially in countries like India due to which COVID cases increased by a huge number and still on the rise despite the vaccine drive. Hence, India still cannot abandon masks even though it happens to be the largest producer of vaccines. So, it becomes exceedingly important to ensure that every citizen of the country wears a mask and follows social distancing norms to ensure the safety of themselves as well as the people around them. For this purpose, we are making a mask detection system to correctly detect the presence or absence of masks on people's faces.

For image classification, we are using a type of Convolutional Neural Networks (CNNs) called ResNets (Residual Networks). CNN is a highly effective Deep Learning technique for image classification and object detection. This is mainly because of the feature extraction process done through convolution and pooling.

Convolution is the process of running a kernel or a filter of a preferred size over the image to extract the pixel-by-pixel features of the image. This will enhance the important features of the image by making it more prominent and hence, will improve the accuracy of prediction. We are using a 3x3 kernel in our convolutional layer. Xie D et al. [2] have proven the importance of a convolution kernel by generating a pencil sketch version of many images by modifying the values on the kernel or filter.

Pooling is the process of reducing the number of parameters in the resultant image obtained after convolution. This makes computation easier, faster, and more efficient as the computational time is directly proportional to the number of parameters. We are using a Max Pooling layer with a 2x2-feature map to reduce the parameters. Max-pooling performs better than the opposite alternatives because it is the sole one that is invariant to the special pad tokens that are appending to the shorter sentences known as padding.[3] In this paper, we aim to classify the images correctly into the categories 'wearing a mask' and 'not wearing a mask'.

Related Works

Jiahui Yu et al. [4] in the paper "UnitBox: An Advanced Object Detection Network" have identified the need for accurate bounding boxes to achieve the best level of face detection. They have adopted the Intersection over Loss (IoU) loss function for bounding box prediction which regresses the four points of a predicted box as a whole unit. Moreover, it also performs accurate and efficient localization, shows robustness to various sizes and shapes, and converges fast. Using the above statements as proof of the importance of bounding boxes, we have also chosen to implement bounding boxes for better localization and face detection.

Mohamed Loey et al. [5] in the paper "Fighting against COVID-19: A novel deep learning model based on YOLO-v2 with ResNet-50 for medical face mask detection" used two different medical imaging datasets to train the model better. Moreover, they used ResNet-50 to make the feature extraction more efficient and YOLO v2 for the actual mask detection. Also, they used the Adam optimizer and IoU to finally obtain a result of 81% for their model. Hence, ResNet-50 was the ideal model architecture for us to use in our project.

Md. Sabbir Ejaz et al. [6] in the paper "Implementation of Principal Component Analysis on Masked and Non-masked Face Recognition" implemented Principal Component Analysis (PCA) on masked and non-masked faces. They try face detection and identification with masked and unmasked faces and compare the models to see which one works the best. Ultimately, they came up with the conclusion that PCA was the best technique in case of a person's identification regardless of whether he/she is wearing a mask. Since PCA is a feature extraction technique, we also emphasized the importance of feature extraction through our project.

Zhigou Zhu et al. [7] in the paper "Application of attitude tracking algorithm for face recognition based on OpenCV in the intelligent door lock "have used OpenCV for attitude tracking. They have identified the various problems that arise in the field of automation, especially when it comes to controlling the operation of certain equipment that depends heavily on the quality of face detection. Hence, they have concluded the paper by stating the importance of OpenCV in terms of real-time face detection which can potentially power various tasks. Therefore, we have chosen OpenCV for real-time testing and detection of our faces.

In this paper, we propose to detect the presence and absence of face masks on people accurately with the use of 50 layered residual network and OpenCV for real-time face detection.

2. Materials And Methods

A. Dataset Collection

We have made use of a medical imaging dataset available on kaggle.com which comprises 682 images in which there are 3000 faces of people wearing masks properly and around 700 faces of people either not wearing masks or wearing them improperly. It also contains XML files with the labels – 'Good', 'Bad', and 'None' representing the presence, absence, and improper use of masks, respectively.

Fig 1. Overview of dataset

B. Extracting faces using bounding boxes

It is insufficient to train the model using the images in their original form because a lot of the images contain a mix of people who are wearing masks and people who are not. Therefore, we need to extract each face from every image and make it into a separate dataset. For this task, we use the bounding box coordinates provided in the XML files along with the respective label for each face. It is as shown in Fig 2.

```
<annotation>
	<folder>images</folder>
	<filename>Xiaomi-mi-Purely-Mask-with-PM2-5.jpg</filename>
	<path>P:\mask_mouth\images\Xiaomi-mi-Purely-Mask-with-PM2-5.jpg</path>
	<source>
		<database>Unknown</database>
	</source>
	<size>
		<width>800</width>
		<height>800</height>
		<depth>3</depth>
	</size>
	<segmented>0</segmented>
	<object>
		<name>good</name>
		<pose>Unspecified</pose>
		<truncated>0</truncated>
		<difficult>0</difficult>
		<bndbox>
			<xmin>346</xmin>
			<ymin>328</ymin>
			<xmax>535</xmax>
			<ymax>542</ymax>
		</bndbox>
	</object>
</annotation>
```

Fig 2. XML file for each image.

Fig 3 and Fig 4 represent the dataset before and after the face extraction is done using bounding box co-ordinates,

Fig 3. Before face cropping/extraction.

Fig 4. After face cropping/extraction.

C. Training the model

To train our model, we use the ResNet-50 architecture. The main feature of this architecture is that it strikes a good balance between computational requirements and model accuracy. Therefore, we have chosen this to be the architecture [11-18].

Moreover, we have chosen the Cross-Entropy loss function and Adam optimizer to update weights and biases. Further details about the same are as follows.

1) *ResNets:* In the case of any neural network, the normal convention states that the model accuracy should improve with

the increase in the number of layers. However, this only holds good for a limited number of layers after which the error starts increasing with every single layer.

Therefore, we use a particular variant of CNN called ResNet, which is short for Residual Neural Network. A ResNet has something called 'skip connections' as shown in Fig 5. which prevents saturation and degradation of accuracy.

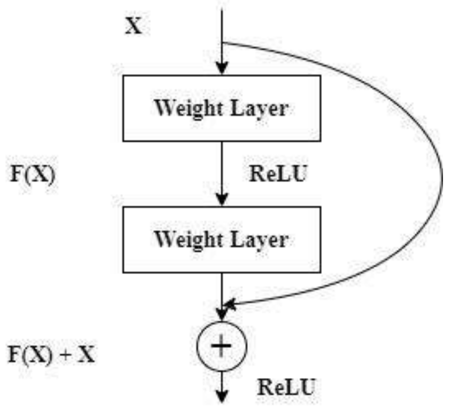


Fig 5. Pictorial representation of skip connection.

There are many variants of ResNets like ResNet-18, ResNet-22, ResNet-34, etc. We have made use of ResNet-50 as stated earlier.

2) *ReLU Activation Function:* Activation functions activate certain connections of the neural network to push the results

in favor of the class with the highest probability. There are many activation functions, but ResNets make use of ReLU or Rectified Linear Unit, which has the following formula: -

$y = \max(0, x)$

Where y refers to the output after applying the activation function and x refers to the input fed to the activation function, it outputs 0 when $x < 0$, and conversely, it outputs a linear function when $x \geq 0$ [8].

Activation functions like Sigmoid, TanH, and Softmax have a common problem called vanishing gradient which halts the optimization of weights and biases. Due to this, the model stops training, and the accuracy stagnates. ReLU eliminates this.

3) *Cross-Entropy loss function:* Cross-Entropy is a very important loss function and is generally preferred in the case of

classification problems.

Here, the term entropy refers to the level of uncertainty inherent in the variable's possible outcome. The higher the value of entropy, the greater is the level of uncertainty and vice versa.

In the case of the cross-entropy function, each predicted class probability is compared to the actual class and an error is generated based on that. Since it is logarithmic, it penalizes the model for large differences nearing the value 1 and is soft in case of small differences nearing the value 0.

The formula for cross-entropy loss function is as follows:

$$\mathrm{CE} = -\sum_{i}^{C} t_i \log(s_i)$$

4) *OpenCV:* OpenCV is a computer vision library containing a wide variety of algorithms for the AI community [9]. This

was built to accelerate the use of machine perception in day-to-day life. It has well over 2500 algorithms which include both classic and state-of-the-art machine learning algorithms.

In our project, we are using OpenCV for real-time testing. It makes use of the connected camera to extract the live video frame by frame, where it specifically narrows down on the face by using the *get_frontal_face_detector* function. Then, it feeds each frame to the model and displays the relevant output on the screen based on the model's prediction.

5) *Adam Optimizer:* Adam, just like the Stochastic Gradient Descent algorithm, is used for updating weights iteratively

when working with training data. It combines the advantages of two other extensions of SGD, namely AdaGrad and RMSProp.

While SGD keeps the learning rate constant throughout the training process, Adam works differently. It maintains a per-parameter learning rate that improves performance on problems with sparse gradients like Computer Vision and Natural Language Processing. This is a property of the Adaptive Gradient Algorithm.

RMSProp maintains a per-parameter learning rate that changes based on the average of recent magnitudes of gradients, i.e., how quickly it is changing. Therefore, the algorithm performs well on non-stationary problems.

However, instead of adapting parameter-wise learning rates based on mean, Adam optimizer makes use of the average of the uncentered variance.

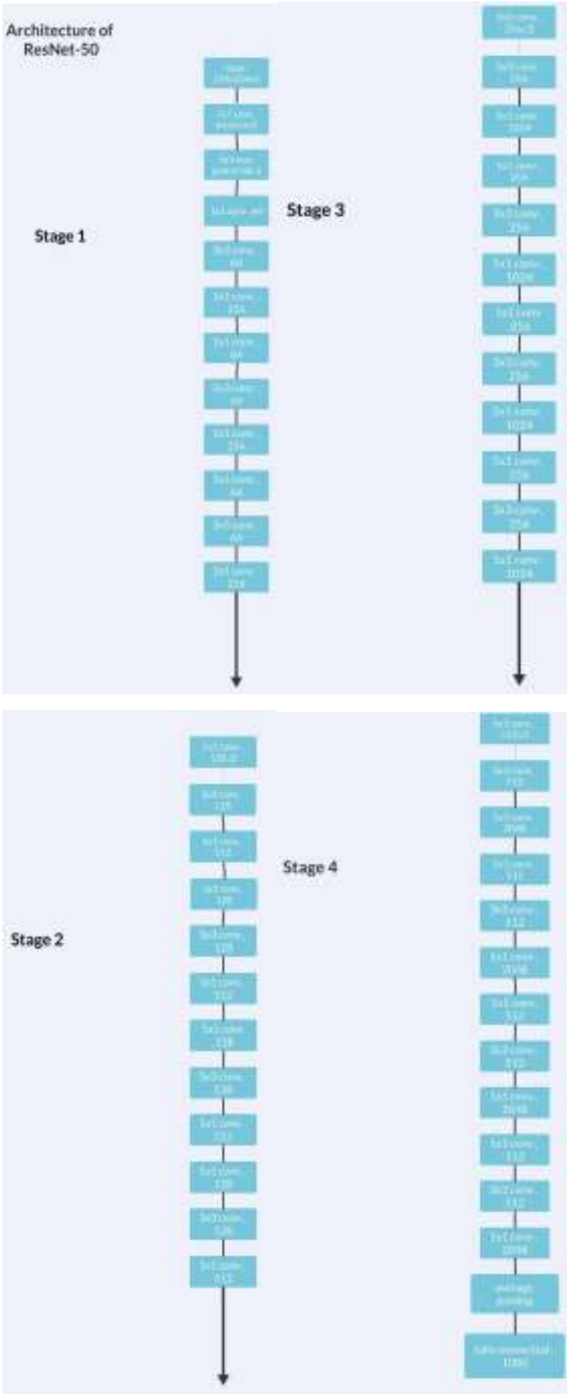


Fig 6. Architecture of ResNet-50

The entire process can be summarized using a flowchart shown in Fig 7.

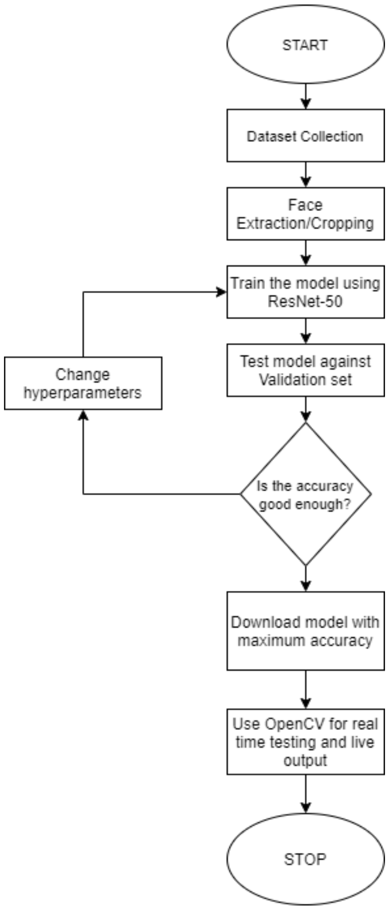


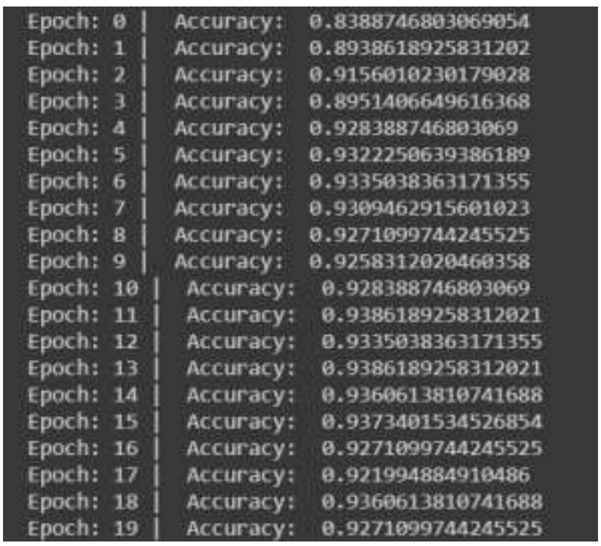
Fig 7. Flowchart depicting implementation procedure.

TABLE I System Requirements

Software Requirements	**Hardware Requirements**
Anaconda 3.5.6 Pytorch 1.0.0 Online GPU (Google Colab/Kaggle Kernel)	Laptop with minimum specifications: 4GB Ram CPU at 1.60 GHz Intel core i5 64-bit Operating System
IDE - Jupyter Notebook	

3. Results and Discussion

Here, we trained the model with 80% of the images as a training set and the remaining 20% as the validation set. We did not use a separate test set since we used OpenCV for real-time testing of our model. Finally, we ended up with an accuracy of around 93% for our best performing epoch, which is shown in Fig 8.



```
Epoch: 0 | Accuracy: 0.8388746803069054
Epoch: 1 | Accuracy: 0.8938618925831202
Epoch: 2 | Accuracy: 0.9156010230179028
Epoch: 3 | Accuracy: 0.8951406649616368
Epoch: 4 | Accuracy: 0.928388746803069
Epoch: 5 | Accuracy: 0.9322250639386189
Epoch: 6 | Accuracy: 0.9335038363171355
Epoch: 7 | Accuracy: 0.9309462915601023
Epoch: 8 | Accuracy: 0.9271099744245525
Epoch: 9 | Accuracy: 0.9258312020460358
Epoch: 10 | Accuracy: 0.928388746803069
Epoch: 11 | Accuracy: 0.9386189258312021
Epoch: 12 | Accuracy: 0.9335038363171355
Epoch: 13 | Accuracy: 0.9386189258312021
Epoch: 14 | Accuracy: 0.9360613810741688
Epoch: 15 | Accuracy: 0.9373401534526854
Epoch: 16 | Accuracy: 0.9271099744245525
Epoch: 17 | Accuracy: 0.921994884910486
Epoch: 18 | Accuracy: 0.9360613810741688
Epoch: 19 | Accuracy: 0.9271099744245525
```

Fig 8. The training loss was obtained after each epoch.

To get a clearer picture, the progression of accuracy with each epoch can be observed through the graph in Fig 9.

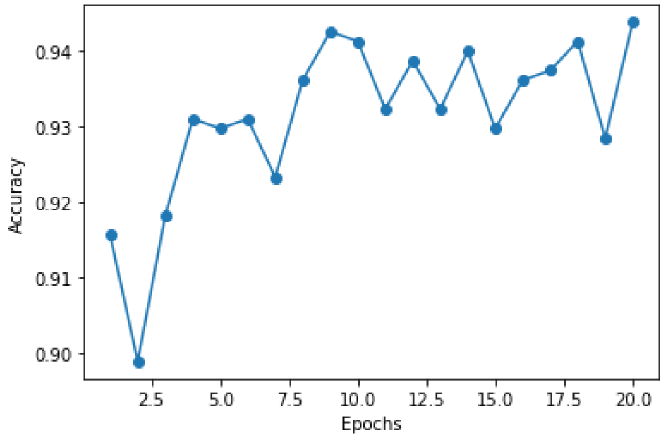


Fig 9. Graph describing the progression of accuracy with each epoch.

4. Conclusion

Preethi Nagrath et al. [10] have used deep learning with TensorFlow and Keras as deep learning frameworks for model training, OpenCV for real-time testing, and Single Shot multibox detector for face detection. Ultimately, they were able to successfully obtain an accuracy of 92.64%. We have used PyTorch as the deep learning framework, OpenCV for real-time testing, and get_frontal_face_detector from the dlib library for face detection. In this process, we have obtained an even higher accuracy of 93%.

As part of the future work, we would want to integrate and control devices based on our model's predictions. Hence, we would be able to integrate our system with doors and alarms to make it work seamlessly.

Acknowledgment

We would like to take this opportunity to express our gratitude to our Guide, Prof. Thirumagal E, Assistant Professor, REVA University, for continuously supporting and guiding us in our every endeavor as well for taking a keen and active interest in the progress of every phase of our project. We would also like to extend our sincere thanks to Dr. Sunilkumar Manvi, Director, REVA University, for providing us with all the support.

References:

Agarap, Abien Fred. *"Deep Learning using Rectified Linear Units (ReLU)"*, Feb 2019.

AH Omar Baabood, Prajoona Valsalan, Tariq Ahmed Barham Baomar, IoT Based Health Monitoring System, Journal of Critical Reviews , Vol. 7, Issue. 4, pp. 739-743, 2020.

Bindhia K Francis, Suvanam Sasidhar Babu, Predicting academic performance of students using a hybrid data mining approach, Journal of Medical Systems, 43:162, 2019. https://doi.org/10.1007/s10916-019-1295-4

Bruce, B.R., Aitken, J.M. and Petke, J. (2016) *"Deep parameter optimization for face detection using the Viola-Jones algorithm in OpenCV"*. International Symposium on Search Based Software Engineering, Raleigh, NC, USA. Lecture Notes in Computer Science (9962). Springer International Publishing, pp. 238-243, Oct 2016.

Jiahui Yu, Yuning Jiang, Zhangyang Wang, Zhimin Cao, and Thomas Huang,*" UnitBox: An Advanced Object Detection Network."* In Proceedings of the 24th ACM international conference on Multimedia (MM '16). Association for Computing Machinery, New York, NY, USA, pp. 516–520, 2016.

Li, Tom, et al. *"Mask or no mask for COVID-19: A public health and market study."* PLoS ONE, vol. 15, no. 8, 2020, p. e0237691. Accessed 5 May 2021.

Loey M, Manogaran G, Taha MHN, Khalifa NEM. Fighting against COVID-19: A novel deep learning model based on YOLO-v2 with ResNet-50 for medical face mask detection, *Sustain. Cities Soc*, doi: 10.1016, pp.12-16, 2021

M. S. Ejaz, M. R. Islam, M. Sifatullah and A. Sarker, *"Implementation of Principal Component Analysis on Masked and Non-masked Face Recognition"* 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT), Dhaka, Bangladesh, pp. 1-5, 2019.

Nagrath, Preeti et al. *"SSDMNV2: A real time DNN-based face mask detection system using single shot multibox detector and MobileNetV2."* Sustainable cities and society vol. 66, 2021.

PK Sadineni, Comparative Study on Query Processing and Indexing Techniques in Big Data, 2020 3rd International Conference on Intelligent Sustainable Systems (ICISS), pp. 933-939, 2020.

PM Surendra, S Manimurugan, A New Modified Recurrent Extreme Learning with PSO Machine Based on Feature Fusion with CNN Deep Features for Breast Cancer Detection, Journal of Computational Science and Intelligent Technologies, Vol. 1, Issue. 3, Pp. 15-21, 2020.

Sajay KR, Suvanam Sasidhar Babu, Vijayalakshmi Yellepeddi, Enhancing The Security Of Cloud Data Using Hybrid Encryption Algorithm, Journal of Ambient Intelligence and Humanized Computing, 2019. https://doi.org/10.1007/s12652-019-01403-1

Suárez-Paniagua, V, Segura-Bedmar, I, "Evaluation of pooling operations in convolutional architectures for drug-drug interaction extraction". *BMC Bioinformatics*, vol. 19, pp. 44-47, Jun 2018.

Sudhan Murugan Bhagavathi, Anitha Thavasimuthu, Aruna Murugesan, Charlyn Pushpa Latha George Rajendran, A Vijay, Raja Laxmi, Rajendran Thavasimuthu, Weather forecasting and prediction using hybrid C5.0 machine learning algorithm International Journal of Communication Systems, Vol. 34, Issue. 10, Pp. e4805, 2021.

Teena Jose, Dr. Suvanam Sasidhar Babu – Detecting Spammers on Social Network Through Clustering Technique, Journal of Ambient Intelligence and Humanized Computing, pp.1-15, 2019. https://doi.org/10.1007/s12652-019-01541-6

Vijayalakshmi Y, Manimegalai P, Suvanam Sasidhar Babu, Accurate Approach towards Efficiency of Searching Agents in Digital Libraries using Keywords, Journal of Medical Systems, 43:164, 2019. https://doi.org/10.1007/s10916-019-1294-5

Xie D., Zhao Y. and Xu D., *"An Efficient Approach for Generating Pencil Filter and Its Implementation on GPU"* Computer Aided Design and Computer Graphics, pp. 185-190, 2007.

Zhiguo Zhu, Yao Cheng, "Application of attitude tracking algorithm for face recognition based on OpenCV in the intelligent door lock", Computer Communications, vol. 154, pp. 390-397, 2020.