Realtime Head Pose Estimation with Facial Keypoints Prediction On Masked/Unmasked Faces

SAURAV ARORA
School of Computer Science and Engineering
Galgotias University,
Greater Noida , Uttar Pradesh
isauravv110@gmail.com

PRANJAL DUBEY
School of Computer Science and Engineering
Galgotias University
Greater Noida , Uttar Pradesh
pranjaldub1999@gmail.com

SAHAJ KAPOOR
School of Computer Science and Engineering
Galgotias University
Greater Noida , Uttar Pradesh
sahajkapoor412@gmail.com

***Abstract*—Numerous Governing authorities/organizations expect people to utilize the services only if they wear masks, effectively masking both their nose and mouth, according to the rules from the World Health Organization (WHO). Manual screening and distinguishing proof of individuals following/not following this arrangement is an enormous assignment in public places.Keeping in mind these challenges, the ideal methodology is to utilize innovations in Artificial Intelligence and Deep Learning; to be utilized as to make this undertaking straightforward, which is anything but difficult to utilize and robotized. In this paper, we propose "DeepFaceMask", which is a high-precision and efficient face mask classifier. The presented DeepFaceMask is a one-stage identifier, which consists of a Deep Convolutional Neural Network (DCNN) to combine significant level semantic data with different element/feature maps. Other than this, we additionally investigate the chance of actualizing DeepFace-Mask with a light-weighted neural organization MobileNet for cell phones. MTCNN, utilizes the inalienable connection among's recognition and alignment to help boost their performance. Specifically, our frame work uses a cascaded architecture with three phases of diligently planned DCNN to predict the face and its key points or landmarks in a coarse-to-fine way.** [1]

I. Introduction (Heading 1)

To viably stop the spread of COVID-19 pandemic, everyone is required to wear a mask in public places.This nearly makes regular facial recognition techniques ineffective, for example, public access control, face access control, facial recognition, facial security checks at train stations, and so forth.. The science around the utilization of masks by the overall population to prevent COVID-19 transmission is progressing quickly. Policymakers need guidance on how masks should be utilized by everybody to battle the

recognition is to recognize a specific class of objects, for example face. Uses of object and face recognition can be found in numerous territories, for example, self driving vehicles,education,surveillance, etc. Customary object locators are based on handmade feature extractors.[4]

II. Problem Statement

The objective of this project is to prepare 'Object Detection Models' fit for distinguishing facial keypoints for 'Face Recognition' and 'Attention Detection' and the location of Masked and Unmasked faces in static as well as moving(video) images. The detection technique should be robust to the occlusion present in the images for better predictability Preferably, they should be sufficiently quick to function admirably for certifiable programs would like to zero in on in our future executions.

III. Vision

This undertaking was made with the vision of building up a "Real-Time Mask Detection System" accessible for public use, to help general wellbeing authorities and little to huge foundations everywhere on the world viably battle this COVID19 pandemic. We trust that the models created here by the little exploration AI/ML people group empower engineers around the planet to have the option to utilize and convey the equivalent to construct systems that would be fit for withstanding the requests of a real-time, real-world use case. Specifically, it would assist manufacturing plants with guaranteeing mask consistence is followed, help guarantee security for guests in control zones or public spots where it is vital for such measures to be taken, etc. The applications are endless and are of earnest need in this crucial time.[2]

IV. Datasets

COVID-19 pandemic. Furthermore, masks should be worn effectively on the face with the end goal that it masks the

The dataset we will be using primarily is the MaskPascalVOC zip file taken from the website:
nose and mouth totally, which is frequently not being followed. Consequently, it is dire to improve the recognition capabilities of the current face/mask recognition technology. Face mask identification alludes to distinguish if an individual is using mask and amount of area covered, which [3]

https://makeml.app/datasets/mask The dataset contains 853 images of the following classes: With mask, Without mask, and Mask weared incorrect. It is labeled with bounding box annotations for object detection. But the number of images

we identify by including the facial keypoints too. The issue is firmly identified with general object identification to distinguish the classes of items (here we manage primarily belonging to the class of mask worn incorrectly are too less in quantity compared to the other two classes in the dataset, which was creating class imbalance so, we collected data

three classes specifically: wearing mask accurately, wearing from additional sources having the class name as None for mask erroneously, and not wearing mask) and face people not wearing masks correctly, which we have combined separately and uploaded on the web, So, finally our combined dataset overall has the following four labels namely: with_mask, without_mask, mask_weared_incorrect and none. This is divided finally into 3 classes, first one having the label “with_mask” which we signify later by a green colour bounding box on the face with a text label over it as “Correctly Masked”, second having the label “without_mask” which we signify later by a red colour bounding box on the face with a text label over it as “Unmasked”, and the third one having either of the two labels, “mask_weared_incorrect” or “none” which we signify later by a blue colour bounding box on the face with a text label over it as “Incorrectly Masked". . 3 Additionally, we are also implementing the main facial keypoints inside the bounding box while detecting the face and the dataset used to train this model is taken from the website: https://ibug.doc.ic.ac.uk/download/annotations/ xm2vts.zip/ The data is in the format of a CSV (Comma Separated Values) file where there are sixty- eight key points of images representing x, y coordinates. This data is being fed into a deep CNN or ConvNet model with the final layer having 68*2=136 dimensions output predicting the X and Y coordinates of those sixty eight key points. Smooth L1 loss and MSE (Mean Squared Error) loss metrics resulted in the best accuracy outputs, we choose Smooth L1 loss metric for our final model as it performed better in real-time comparatively. [5]

V. RELATED WORK

A. Object Detection

The face detection technique used here is MTCNN (Multi-task Cascaded Convolutional Networks). Humanface classification and arrangement in unconstrained climate Ongoing investigations show that profound learning approaches can accomplish great execution on these two errands. In this paper, we have utilized a Deep Cascaded perform various tasks system which abuses the inalienable relationship among discovery and arrangement to help up their exhibition. Specifically, this casing work uses a fell engineering with three phases of painstakingly planned Deep Convolutional Neural Networks to anticipate face and milestone area in a coarse-to-fine way. What's more, it proposes another online hard example mining technique that further improves the presentation practically speaking..

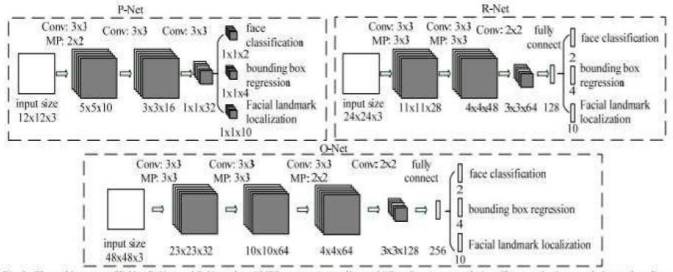
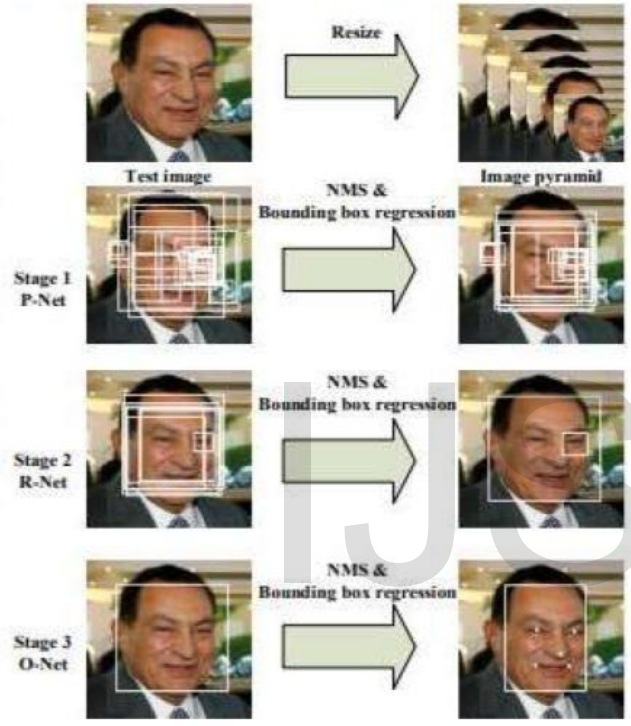


Fig. 2. The architectures of P-Net, R-Net, and O-Net, where “MP” means max pooling and “Conv” means convolution. The step size in convolution and pooling is 1 and 2, respectively.

Real Time analysis of MTCN (its workflow which is being followed by all the three sequential models which are the P , R and O model respectively) :

Sliding window process of MTCNN:



(a) Examples of results on FDDB

(b) Examples of results on WIDER FACE

(a) Results on AFLW. Faces with occlusion (row 1), pose variation (row 2), different lighting conditions (columns 1-2 in row 3), low image quality (column 3 in row 3), different expressions (column 4-5 in row 3), more inaccurate cases are shown in column 6-8 in row 3.

(b) Results on AFW

N-face and keypoints detection:

MTCNN is a technique comprising of three stages, which can predict basic facial keypoints and perform basic face alignment . To avoid detection errors , it uses a technique called Non Max Suppression .[6][7]

The MTCNN framework / Architecture uses three separate networks:

- "P" – Network
- "R" – Network
- "O" – Network

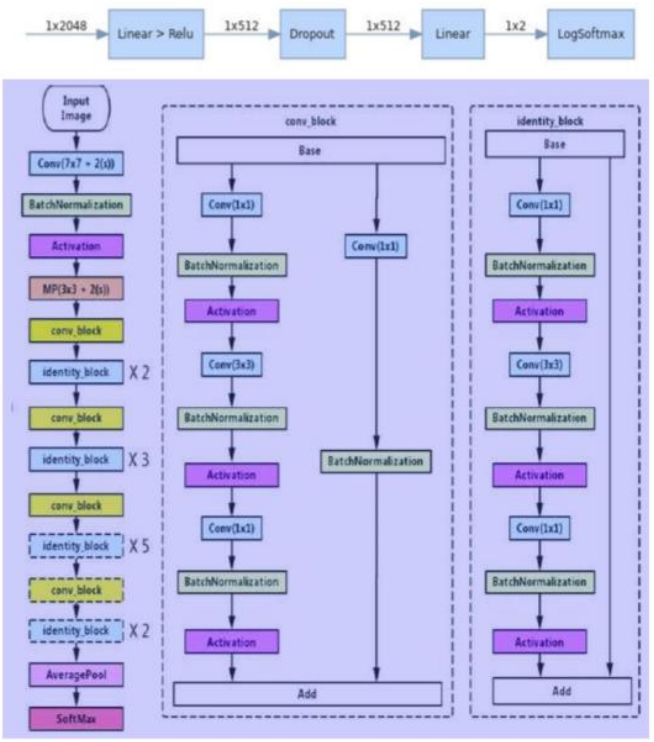
• Structure of P-Net:
P-Net predicts bounding box using sliding a 12*12 size kernel/filter across the image.

• Structure of R-Net:
R-Net has similar structure, but uses more layer, thus predicting more accurate bounding box coordinates.

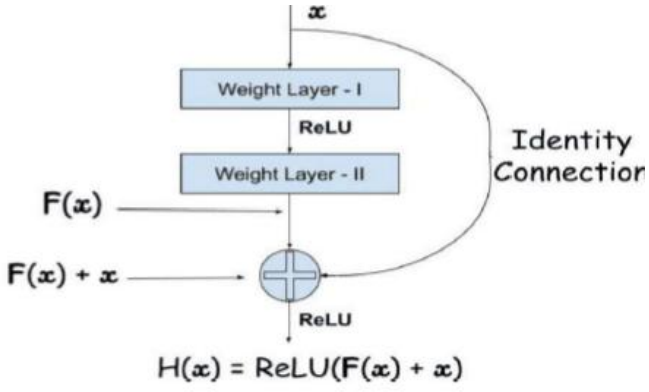
• Structure of O-Net:
O-Net takes the output of R-Net and predicts three sets of data namely - the probability of face being in the box, bounding box, and facial keypoints.[8][9]

B. Image Classification

Image classification refers to extracting specific desired features from a static or a real time image and classifying it to solve a specific problem at hand. This objective was accomplished by using a transfer learning approach. The ResNet-50 pre-trained model was used as a feature extractor connected with a custom fully connected layer for robust and efficient image classification. The model was trained on a dataset consisting three classes, masked, not masked, not properly masked respectively. The problem with the dataset was that it didn't represent the same amount of each class i.e. it was an imbalance of data, so the model was trained on two datasets combined. To achieve more robust results, custom image augment- ation techniques were implemented during the training process. The convolutional layers of ResNet-50 were used as feature extractor (last convolutional layers), rest all were frozen during training. Thus, fine tuning the model gave much better results from traditional state-of-the-art architectures. It also helped in tackling vanishing gradients problem by leveraging the use of skip connections and strong robust feature extractor proved to be efficient enough to extract features from a relatively small dataset. ResNet-50 layers were connected to linear layers before end-to-end result prediction.



Recently DNN community started experimenting with deeper networks because they were able to achieve high accuracy values. All in all, the underlying layers of the organization won't adapt successfully. Thus, profound organization preparing won't combine and precision will either begin to corrupt or immerse at a specific worth. In spite of the fact that the disappearing angle issue tended to utilizing the standardized instatement of loads, further organization exactness was as yet not expanding. Profound Residual Network is practically like the organizations which have convolution, pool-ing, activation and completely associated layers stacked one over the other. Skip connections used by ResNet-50.[14][15]



Key Features of ResNet:

• Resnet utilizes the layer called Batch normalization which has a sole purpose of adjusting the input of the next layer hence increasing the performance. The problem of covariate shift is mitigated.

• Resnet uses skip connection to overcome the gradient diminishing problems.[17]

3232, number of images labelled 2 i.e. not wearing mask are 717, number of images labelled 3 i.e. Wearing a mask incorrectly is 249.

The dataset was then divided into training data,

C. HEAD POSE ESTIMATION :

Alignment of any object suggests its general direction and position with respect to a camera. We can change the posture by either moving the thing regarding the camera, or the camera concerning the article.[10]

The posture estimation issue portrayed in this paper is often alluded to as Perspective-n-Point issue or PNP . In this issue the objective is to determine the inclination or posture of an article as for the camera , and we know the coordinates of n 3D points on the item and the corresponding 2D projections in the picture.[11]

Motions performed by a third dimensional rigid object : 1. Translation : Change in the pixel values such that there is a motion caused to the image in either x axis or the y axis.

2. Rotation : In this type of movement the image is translated with respect to a single pivot point .

So, estimating the pose of a 3D object means finding 6 numbers — three for translation and three for rotation. To calculate the 3D pose of an object in an image you need the following information[12][13]

1. 2D coordinates of a couple of points : You need the 2D (x,y) locations of a couple of points in the picture. For the situation of a face, you could pick the corners of the eyes, the tip of the nose, corners of the mouth and so on .

2. 3D locations of the same points : We need the 3D coordinates of the 2D feature points. Primary 3d coordinates refer to : Nose tip , Chin , right corner of mouth , left corner of mouth , left eye , right eye.

OpenCV solvePnP

The capacity solvePnP and solvePnPRansac can be utilized to gauge pose.[16]

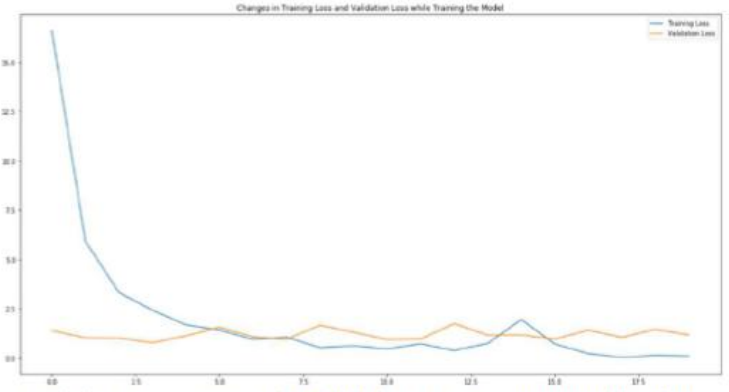
solvePnP actualizes a few calculations for pose estimation which can be chosen utilizing the boundary flag. As a matter of course it utilizes the check solve pnp iteration to true and its basically distributed ledger technology arrangement trailed by LM algorithm. Solve pnp p3p function utilizes just three focuses ascertaining the alignment and it must be utilized just when utilizing solve pnp pransac.

VI. TRAINING

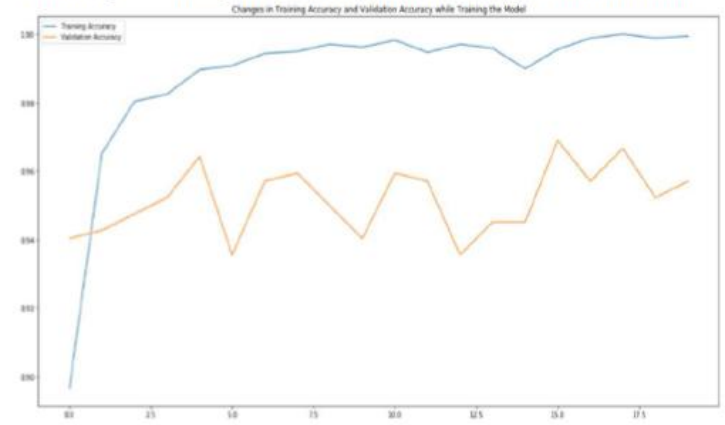
After preprocessing the data, our combined dataset consists a total number of 4198 images. Number of images labelled 1 i.e. wearing mask correctly are

validation data and test data. It was split into 8:1:1 ratio i.e. train set size is 3358, validation set size is 419 and test set size is 421. The difference in the validation and test[18]

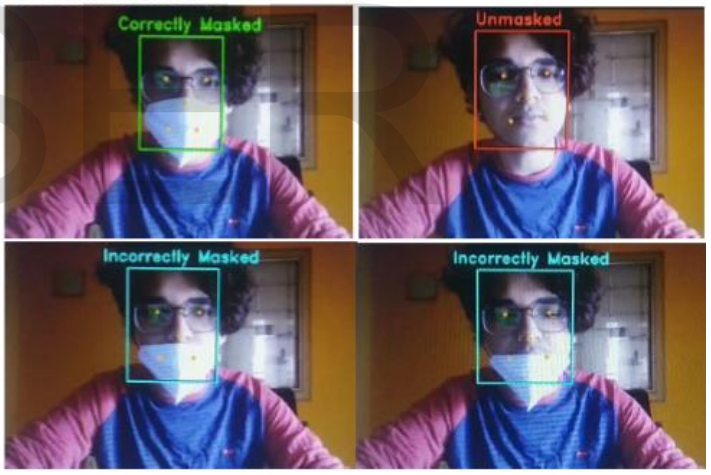
size, despite the same ratio, is because test set size was calculated after calculating the train set size and validation set size and their summation was subtracted from the total number of images. Also, images were randomly shuffled for no imbalance of class and robust performance of model, in batch size of 64 for faster computation. We trained the model using cross-entropy loss and Adam [19]
optimizer (an upgrade to stochastic gradient descent with momentum capabilities). In addition, the learning rate was set as 10-3 i.e. 0.001 and the number of epochs as 20, post this the model stopped learning based on earlier observations during training.
Challenges faced during the training process was that single data source wasn't enough to provide sufficient number of images belonging to each class. So, many data sources were considered and a robust, balanced, sufficiently large dataset was created that would provide enough data for the model to adapt to variances in data. GPU was used in training the model due to the large data. Training on GPU proved to be about 3x times faster than training on the CPU. GPU model used while training: NVIDIA GeForce GTX 1050 2GB GDDR5. Lighting and camera settings play a major role in model performance. Thus, we used MTCNN, which easily tackles such problems. Total params: 24,558,146
Trainable params: 16,014,850
Non-trainable params: 8,543,296



Training Accuracy and Validation Accuracy Visualization



Tested in real-time



VII. RESULTS

The best model saved during training resulted in a validation loss of 0.9591 and validation accuracy of 0.9689 which was

VIII. Real-Time Applications:

• Mall security checks / Super markets • Offices spaces / Schools
• Hospitals
• Mobile applications for alerts

IX. FURTHER IMPLEMENTATIONS

It is evident that one of our biggest obstacles during the COVID-19 pandemic is to make sure people follow the safety regulations especially in public places for his/her own safety and the safety of others around. Our DeepFaceMask model will thus detect if people are wearing masks or not, correctly, when deployed to the CCTVs in the public places and can alert the admin as and when people are not wearing masks or wearing masks incorrectly. Additionally, it can be used in head pose estimation, attention detection in classrooms and lectures on masked faces, drowsiness detection on masked faces using facial keypoints tracking the driver's eyes, and so on. [20]

X. REFERENCES

[1] P. Viola and M. J. Jones, "Robust real-time face detection", Int. J. Comput. Vision, vol. 57, no. 2, pp. 137-154, May 2004.

[2] Z. A. Memish, A. I. Zumla, R. F. Al-Hakeem, A. A. Al-Rabeeah, and G. M. Stephens, "Family cluster of middle east respiratory syndrome coronavirus infections," New England Journal of Medicine, vol. 368, no. 26, pp. 2487–2494, 2013.

[3] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," 2017.

[4] A. Shrivastava, A. Gupta, and R. Girshick, "Training region-based object detectors with online hard example mining," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 761–769.

[5] S. Ge, J. Li, Q. Ye, and Z. Luo, "Detecting masked faces in the wild with lle-cnns," in Proceedings of the IEEE.

[6] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G.Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga et al.,"Pytorch: An imperative style, high-performance deep learning library," in Advances in Neural Information Processing Systems, 2019, pp. 8024–8035.

[7] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 770–778

[8] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W.Wang, T. Weyand, M. Andreetto, and H. Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," arXiv preprint arXiv:1704.04861, 2017.

[9] J. Deng, J. Guo, Y. Zhou, J. Yu, I. Kotsia, and S.Zafeiriou, "Retinaface: Single-stage dense face localization in the wild," arXiv preprint arXiv:1905.00641, 2019.

[10] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2014, pp. 580–587.

[11] Krizhevsky A, Sutskever I, Hinton GE (2012) ImageNet classification with deep convolutional neural networks. Adv Neural Inf Process Syst 25.

[12] S. Yang, P. Luo, C.-C. Loy, and X. Tang, "Wider face:A face detection benchmark," in Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 5525–5533.

[13] R. Girshick, "Fast r-cnn," in Proceedings of the IEEE international conference on computer vision, 2015, pp.1440–1448.

[14] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in 2009 IEEE conference on computer vision and pattern recognition. Ieee, 2009, pp. 248–255.

[15] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in Proceedings of the IEEE conference on computer vision, 2016, pp. 779–788.

[16] A. Krizhevsky, I. Sutskever and G. E. Hinton, "Imagenet classification with deep convolutional neural networks", Advances in Neural Information Processing Systems 25, pp. 1097-1105, 2012.

[17] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition",CoRR, 2014.

[18] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D.
Anguelov, et al., "Going deeper with convolutions", 2015.

[19] K. He, X. Zhang, S. Ren and J. Sun, "Deep residual learning for image recognition", 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770-778, 2016.

[20] P. Viola and M. J. Jones, "Robust real-time face detection", Int. J. Comput. Vision, vol. 57, no. 2, pp. 137-154, May 2004