AUTOMATING FACE MASK DETECTION USING CONVOLUTIONAL NEURAL NETWORK

Meenakshi Tomar[1], Surendra Kumar[2]
Department of Computer Science & Engineering[1,2]
Dr. K. N. Modi University, Newai, India.

Abstract: **-** In the absence of any systematic well defined diagnosis process, the well-known safety measure are the best choice in order to prevent from the effect of the COVID-19. From all of these safety measures, wearing a face mask in public places is one of the key prevention to decrease the chances of COVID-19 infection. So, in order to provide an public environment that discover those people who haven't wear a face mask there are a strong demand to develop such an artificially intelligent system that is enough capable to detect such person from the public places or an entry gate. In literature, so many system have been developed for this purposes. Apart from features provided by these system, there are a lacking to detect such system in a variety of environment. By inspiring from the existing system, we are going to develop a system that performs the same task but in a wide range of environments. For this purposes, we will use the Convolutional Neural Network to develop our system. We will use benchmark dataset to train the system and will tune the hyper parameter accordingly. Finally, we will measure the performance of our system in order to verify the accuracy.

Starting from the analysis of a known training dataset, the learning algorithm produces an inferred function to make predictions about the output values. The system will able to provide targets for any new input after sufficient training. The learning algorithm can also compare its output with the correct, intended output and find errors to modify the model accordingly. System will be developed using Supervised Learning paradigm and a Web Camera to detect people with or without masks. The main goal of the project is to implement this system at colleges, airports, hospitals, and offices where chances of spread of COVID-19 through contagion are relatively higher.

Keywords: **- Machine learning, Convolutional neural network, Data augmentation, Classification, Accuracy, OpenCV**

I. INTRODUCTION

There are a huge increase in practice to wear a face mask in open publicly area during the last one and half year because of Covid-19 epidemic throughout the world. Masks are primarily used to keep them protective from the effect of air pollution. But now, the scenario are different, today masks are heavily wear to protect themselves from the transmission of covid-19. As covid-19 spread much faster and severely affected a large population worldwide, covid-19 has been declared as global pandemic by the World Health Organization (WHO). The victim of corona virus became due to the close contact in crowed and over-crowed area. The infection spreads through close contact and in swarmed and stuffed territories.

The Covid plague has offered ascend to an unprecedented level of overall logical and scientific collaboration. Man-made brainpower (AI) in view of Machine learning and deep Learning can assist with battling Covid-19 from various perspectives. AI permits specialists and clinicians assess tremendous amounts of information to figure the appropriation of COVID-19, to fill in as an early notice instrument for possible pandemics, and to order weak populations. The arrangement of medical care needs financing for arising innovation like man-made reasoning, IoT, huge information and AI to handle and foresee new illnesses. To all the more likely comprehend disease rates and to follow and rapidly recognize contaminations, the AI's force is being misused to address the Covid-19 pandemic. Individuals are constrained by laws to wear face covers openly in numerous nations. These standards and laws were created as an activity to the remarkable development in cases and passing in numerous regions. Nonetheless, the way toward checking huge gatherings of individuals is getting more troublesome. The checking interaction includes the recognition of any individual who isn't wearing a face mask.

The hierarchy of concepts allows the computer to learn complicated concepts by building them out of simpler ones. If we draw a graph showing how these concepts are built on top of each other, the graph is deep, with many layers. For this reason, we call this approach to AI deep learning. Deep learning excels on problem domains where the inputs (and even output) are analog. Meaning, they are not a few quantities in a tabular format but instead are images of pixel data, documents of text data or files of audio/video data. Deep learning allows computational models that are composed of multiple processing layers to learn representations of data with multiple levels of abstraction. Authors presented here, a face mask detection model that depends on PC vision and Machine learning. The proposed model can be incorporated with observation cameras to hinder the COVID-19 transmission by permitting the recognition of individuals who are wearing mask or not wearing face mask. The model is coordination between Deep learning with opencv, tensorflow and keras. We have utilized Deep learning for extracting features and joined it with balanced calculations. We

acquainted an examination between them with track down the most reasonable calculation that accomplished the most elevated exactness and devoured minimal time during the time spent preparing and discovery.

The layered of concept in form of convolution permits the computer to gain the crucial features by building them out less complex ones. A diagrammatic presentation is shown to explore how these ideas are based on top of one another, the graph is deeper, with numerous layers. Hence, we call this way to deal with AI based Deep learning. Furthermore, the techniques of deep learning performs well in a domain where mostly inputs are in analog as pixel information etc. Profound learning permits computational models that are made out of different preparing layers to learn portrayals of information with numerous degrees of deliberation.

II. LITERATURE STUDY

In the recent past, numerous researchers and analysts in the main centered on gray-scale face image [29]. While some were utterly engineered on pattern identification models, possessing initial data of the face model whereas others were victimization AdaBoost [17] that was a wonderful classifier for coaching functions. Then came the Viola-Jones Detector, that provided a breakthrough in face detection technology, and time period face detection got attainable. It Janus-faced numerous issues just like the orientation and brightness of the face, creating it exhausting to intercept. Therefore essentially, it failed to work in uninteresting and dim light weight. Thus, researchers started looking for a new various model that might simply sight faces still as masks on the face.

In the studied literature, several datasets for the purpose of face detection were developed to make an impression of mask detection models. Earlier datasets consisted of pictures fetched in supervised surroundings, whereas recent datasets square measure constructed by taking on-line pictures like WiderFace [32], IJB-A [18], MALF [33], and CelebA [19]. Annotations square measure provided for present faces in these datasets as compared to earlier ones. Datasets on large scale are rather more required for creating higher coaching and testing information and perform real-world applications in an exceedingly a lot of less complicated manner. This calls for numerous deep learning algorithms which may browse faces and mask straight from the information provided by the user.

A wide range of variation have available on the model for face mask detection. These can be categorize into various division. In Boosting-based order, supported falls with simple haar highlights were accepted utilizing the Viola-Jones face locator [15], which was talked about above in this segment. At that point a Multiview face mask indicator was made roused by the Viola-Jones indicator model. Also, a face cover locator model was settled on utilizing choice trees calculations. Face cover locators in this class were exceptionally compelling in distinguishing face masks.

In Deformable Part Model-based characterization, the design and directions of a few unique countenances are displayed utilizing DPM. In 2006 Ramanan proposed a Random backwoods tree model in face mask discovery, which precisely surmises face designs and facial postures. In [34], one of the famous specialists made a DPM-based face cover finder utilizing around 30, 000 appearances separated into covers and without masks classification. His work accomplished an uncommon precision of 97.14 %. Further models of face mask finders were made by [5]. Commonly, DPM-based face cover location models can accomplish superb precisions, yet it could be lenient from the transcending cost of calculation because of the utilization of DPM.

In Convolutional Neural Network-based arrangement, face indicator models gain straightforwardly from the client's information and afterward apply a few profound learning calculations on it [35]. In the year 2007, [23] explore concocted Cascade CNN.

In [33], Yang et al. concocted highlights conglomeration of appearances in the face location model. In additional examination works, [29] overhauled the AlexNet design for adjusting the picture dataset. For uninhibited conditions, Zhu et al.

III. APPLIED TOOLS & METHODLOGIES

1. OpenCV

OpenCV (Open Source Computer Vision Library) is an open source computer vision and machine learning software library. OpenCV was built to provide a common infrastructure for computer vision applications and to accelerate the use of machine perception in the commercial products. Being a BSD-licensed product, OpenCV makes it easy for businesses to utilize and modify the code.

The library has more than 2500 optimized algorithms, which includes a comprehensive set of both classic and state-of-the-art computer vision and machine learning algorithms. These algorithms can be used to detect and recognize faces, identify objects, classify human actions in videos, track camera movements, track moving objects, extract 3D models of objects, produce 3D point clouds from stereo cameras, stitch images together to produce a high resolution image of an entire scene, find similar images from an image database, remove red eyes from images taken using flash, follow eye movements, recognize scenery and establish markers to overlay it with augmented reality, etc. The library is used extensively in companies, research groups and by governmental bodies.Along with well-established companies like Google, Yahoo, Microsoft, Intel, IBM, Sony, Honda, Toyota that employ the library, there are many startups such as Applied Minds, Video Surf, and Zeitera, that make extensive use of OpenCV.

It has C++, Python, Java and MATLAB interfaces and supports Windows, Linux, Android and Mac OS. OpenCV leans mostly towards real-time vision applications and takes advantage of MMX and SSE instructions when available. A full-featured CUDA and OpenCL interfaces are being actively developed right now. There are over 500 algorithms and about 10 times as many functions that compose or support those algorithms. OpenCV is written natively in C++ and has a template interface that works seamlessly with STL containers.

2. KERAS

Keras is an API designed for human beings, not machines. Keras follows best practices for reducing cognitive load: it offers consistent & simple APIs, it minimizes the number of user actions required for common use cases, and it provides clear & actionable error messages. It also has extensive documentation and developer guides. Keras contains numerous implementations of commonly used neural-network building blocks such as layers, objectives, activation functions, optimizers, and a host of tools to make working with image and text data easier to simplify the coding necessary for writing deep neural network code. The code is hosted on GitHub, and community support forums include the GitHub issues page, and a Slack channel. Keras is a minimalist Python library for deep learning that can run on top of Theano or Tensor Flow. It was developed to make implementing deep learning models as fast and easy as possible for research and development. It runs on Python 2.7 or 3.5 and can seamlessly execute on GPUs and CPUs given the underlying frameworks. It is released under the permissive MIT license.

Keras was developed and maintained by François Chollet, a Google engineer using four guiding principles:

- **Modularity**: A model can be understood as a sequence or a graph alone. All the concerns of a deep learning model are discrete components that can be combined in arbitrary ways.

- **Minimalism**: The library provides just enough to achieve an outcome, no frills and maximizing readability.

- **Extensibility**: New components are intentionally easy to add and use within the framework, intended for researchers to trial and explore new ideas.

- **Python**: No separate model files with custom file formats. Everything is native Python. Keras is designed for minimalism and modularity allowing you to very quickly define deep learning models and run them on top of a Theano or TensorFlow backend.

3. DATASETS

We used well famous and benchmarked datasets prepared by Prajana's datasets. This datasets contain total 1491 images, in which 747 images are available in form of faces with mask while remaining 744 images are available in form of faces without mask. We used this set of images for the training purposes. Apart from this, the datasets also provide additional 194 images in both form for the purposes of validating our model.

IV. PROPOSED APPROACH

Here, we are going to elaborate our approach for detecting the faces that who is wear the face mask or not. The overall system have been divided into two phase. The first phase has been used for the model training using datasets with the Deep learning network. The proposed Deep learning network model have two hidden layer, one input layer, and one output layer. In order to increase the system reliability and to make a robust system we have done some preprocessing on the datasets. The preprocessing steps involve removing the noise from the images, resizing them as all images are available with the same size before they fed into the training phase. The preprocessing process have also involved the operation of image augmentation using the various available algorithm. These also helps to increase our datasets. After that we feed these the training datasets to train the proposed model, which is then validated by using the validation datasets. Furthermore, for the purpose of training our deep learning network model we used well famous and benchmarked datasets prepared by Prajana's datasets. This datasets contain total 1491 images, in which 747 images are available in form of faces with mask while remaining 744 images are available in form of faces without mask. We used this set of images for the training purposes. Apart from this, the datasets also provide additional 194 images in both form for the purposes of validating our model. The second phase involves to provide a user interface that capturing the real time face from real environment and classify them whether the corresponding faces uses a mask are not. In this phase, we used the web camera to capturing the real face in form of video and then this streaming are feed into the pre-trained Deep learning network to detect the faces with mask or without mask. The process flow of proposed model are shown in figure-1.

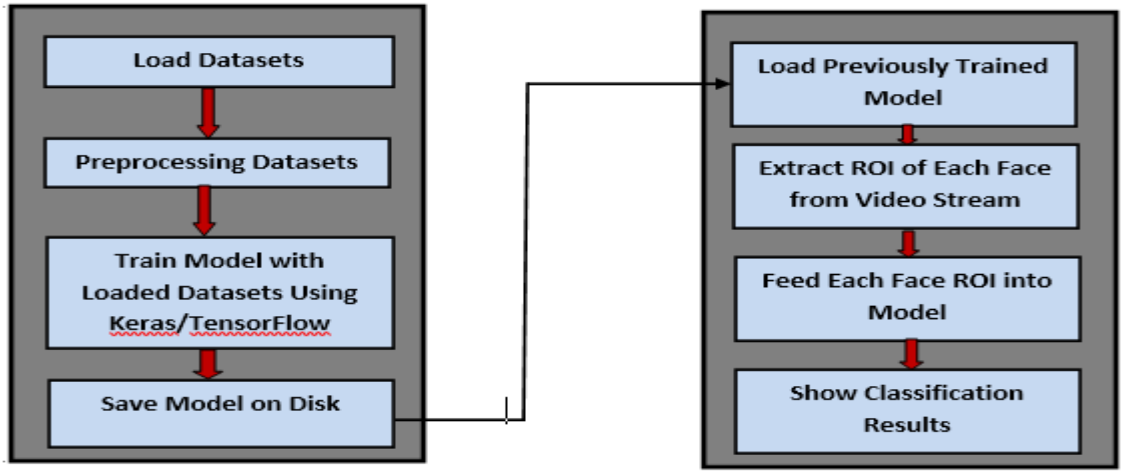

Figure-1: Training and deployment phase for the proposed model

For the purpose of training, we used the python based Keras/Tensorflow library routine. The routine in Keras/Tensorflow provide a wide range of facilities to implement the system model. After the training has been completed we save the generated model on the disk for future use. During the deployment, we load our saved model with an interface that have been developed in OpenCV, capturing the real time video stream, feed into the loaded model, and then we get the appropriate results. The whole implementation have been done in Python. The proposed model summary is shown in Table-1.

```
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_1 (Conv2D)            (None, 148, 148, 100)     2800
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 74, 74, 100)       0
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 72, 72, 100)       90100
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 36, 36, 100)       0
_________________________________________________________________
flatten_1 (Flatten)          (None, 129600)            0
_________________________________________________________________
dropout_1 (Dropout)          (None, 129600)            0
_________________________________________________________________
dense_1 (Dense)              (None, 50)                6480050
_________________________________________________________________
dense_2 (Dense)              (None, 2)                 102
=================================================================
Total params: 6,573,052
Trainable params: 6,573,052
Non-trainable params: 0
_________________________________________________________________
```

Table-1: Proposed Model Summary

V. EXPERIMENTAL RESULT & ANALYSIS

1. Classification Accuracy

Classification accuracy is perhaps the simplest metrics one can imagine, and is defined as the number of correct predictions divided by the total number of predictions**,** multiplied by 100**.**

2. Precision

There are many cases in which classification accuracy is not a good indicator of your model performance. One of these scenarios is when your class distribution is imbalanced. In this case, even if you predict all samples as the most frequent class you would get a high

accuracy rate, which does not make sense at all. Therefore we need to look at class specific performance metrics too. Precision is one of such metrics, which is defined as:

Precision= True_Positive / (True_Positive + False_Positive)

3. Recall

Recall is another important metric, which is defined as the fraction of samples from a class which are correctly predicted by the model. More formally:

Recall = True_Positive / (True_Positive + False_Negative)

4. F1 Score

Depending on application, you may want to give higher priority to recall or precision. But there are many applications in which both recall and precision are important. Therefore, it is natural to think of a way to combine these two into a single metric. One popular metric which combines precision and recall is called F1-score, which is the harmonic mean of precision and recall defined as:

F1-score = 2 * Precision * Recall / (Precision + Recall)

Classification Report

	precision	recall	f1-score	support
Mask	0.98	0.99	0.98	97
No Mask	0.99	0.98	0.98	97
accuracy			0.98	194
macro avg	0.98	0.98	0.98	194
weighted avg	0.98	0.98	0.98	194

Table-2: Classification Report of Proposed Model

From table-2, we can observe that the proposed model achieve the 98 percent of accuracy. Table-2 reveals the well form characteristics about the prposed model.

5. Confusion Matrix

One of the key concept in classification performance is confusion matrix also known as error matrix, which is a tabular visualization of the model predictions versus the ground-truth labels. Each row of confusion matrix represents the instances in a predicted class and each column represents the instances in an actual class.

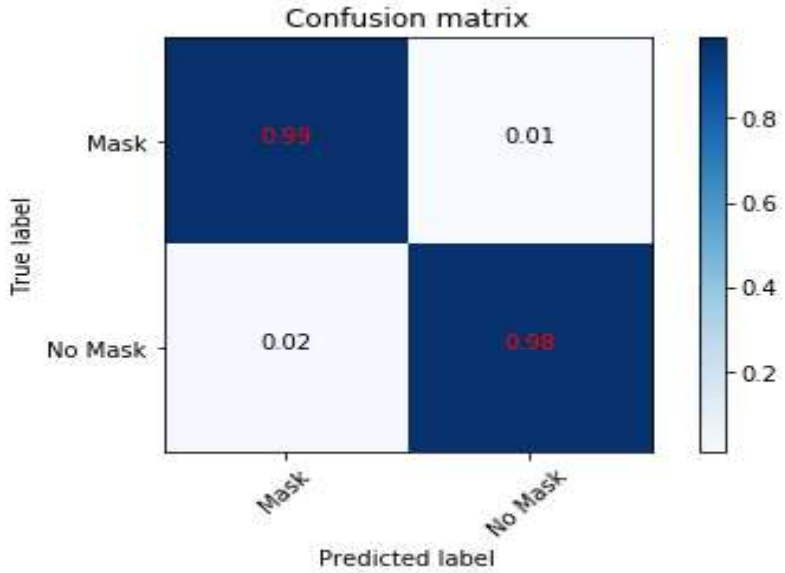

Figure-2: Confusion Matrix of the Proposed Model

Figure-2 express about the ground truth against the model's prediction. The proposed model is able to recognize 99 percent of faces with mask accurately, while the proposed model is able to recognize 98 percent of faces without mask accurately.

VI. CONCLUSION

Authors built and deployed a model to automatically detect the face with mask or without mask. Authors attempts to keep the model simple but not make any compromise with the performance. The proposed model achieve the accuracy by 98 percent. The accuracy can be further improve by hyper parameter tuning. The model has also been tested against the benchmark datasets.

REFRENCES

[1] Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., … Ghemawat, S. (2016). *Tensorflow: Large-scale machine learning on heterogeneous distributed systems*. arXiv preprint arXiv:1603.04467.

[2] Ahmed, I., Ahmad, M., Rodrigues, J. J., Jeon, G., & Din, S. (2020). A deep learning-based social distance monitoring framework for COVID-19. *Sustainable Cities and Society*, Article 102571.

[3] Alom, M. Z., Taha, T. M., Yakopcic, C., Westberg, S., Sidike, P., Nasrin, M. S.,Asari, V. K. (2018). *The history began from alexnet: A comprehensive survey on deep learning approaches*. arXiv preprint arXiv:1803.01164.

[4] Anisimov, D., & Khanova, T. (2017). Towards lightweight convolutional neural networks for object detection. August. In *2017 14th IEEE international conference on advanced video and signal based surmasklance (AVSS)* (pp. 1–8).

[5] Chen, D., Ren, S., Wei, Y., Cao, X., & Sun, J. (2014). Joint cascade face detection and alignment. September. *European conference on computer vision* (pp. 109–122). Cham: Springer.

[6] Chen, D., Hua, G., Wen, F., & Sun, J. (2016). Supervised transformer network for efficient face detection. October. *European conference on computer vision* (pp. 122–138). Cham: Springer.

[7] Ge, S., Li, J., Ye, Q., & Luo, Z. (2017). Detecting masked faces in the wild with lle-cnns. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2682–2690.

[8] Ge, X. Y., Pu, Y., Liao, C. H., Huang, W. F., Zeng, Q., Zhou, H., … Chen, H. L. (2020). Evaluation of the exposure risk of SARS-CoV-2 in different hospital environment. *Sustainable Cities and Society, 61*, Article 102413.

[9] Ghiasi, G., & Fowlkes, C. C. (2014). Occlusion coherence: Localizing occluded faces with a hierarchical deformable part model. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2385–2392.

[10] Gulcehre, C., Moczulski, M., Denil, M., & Bengio, Y. (2016). Noisy activation functions. June. *International conference on machine learning* (pp. 3059–3068).

[11] Hahnloser, R. H., Sarpeshkar, R., Mahowald, M. A., Douglas, R. J., & Seung, H. S. (2000). Digital selection and analogue amplification coexist in a cortex-inspired silicon circuit. *Nature, 405*(6789), 947–951.

[12] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 770–778.

[13] Huang, C., Ai, H., Li, Y., & Lao, S. (2007). High-performance rotation invariant multiview face detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 29*(4), 671–686.

[14] Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R. Darrell, T. (2014). Caffe: Convolutional architecture for fast feature embedding. November *Proceedings of the 22nd ACM international conference on multimedia*, 675–678.

[15] Jones, P., Viola, P., & Jones, M. (2001). *Rapid object detection using a boosted cascade of simple features*. Charles Rich: University of Rochester.

[16] Jun, B., Choi, I., & Kim, D. (2012). Local transform features and hybridization for accurate face and human detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 35*(6), 1423–1436.

[17] Kim, T. H., Park, D. C., Woo, D. M., Jeong, T., & Min, S. Y. (2011). Multi-class classifier based adaboost algorithm. October. *International conference on intelligent science and intelligent data engineering* (pp. 122–127). Berlin, Heidelberg: Springer.

[18] Klare, B. F., Klein, B., Taborsky, E., Blanton, A., Cheney, J., Allen, K., Jain, A. K. (2015a). Pushing the frontiers of unconstrained face detection and recognition: Iarpajanus benchmark a. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1931–1939.

[19] Klare, B. F., Klein, B., Taborsky, E., Blanton, A., Cheney, J., Allen, K., Jain, A. K. (2015b). Pushing the frontiers of unconstrained face detection and recognition: Iarpajanus benchmark a. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1931–1939.

[20] Kumar, P., Hama, S., Omidvarborna, H., Sharma, A., Sahani, J., Abhijith, K. V., Tiwari, A. (2020). Temporary reduction in fine particulate matter due to 'anthropogenic emissions switch-off' during COVID-19 lockdown in Indian cities. *Sustainable Cities and Society, 62*, Article 102382.

[21] Lawrence, S., Giles, C. L., Tsoi, A. C., & Back, A. D. (1997). Face recognition: A convolutional neural-network approach. *IEEE Transactions on Neural Networks, 8*(1), 98–113.

[22] LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE, 86*(11), 2278–2324.

[23] Li, H., Lin, Z., Shen, X., Brandt, J., & Hua, G. (2015). A convolutional neural network cascade for face detection. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 5325–5334.

[24] Li, Y., Sun, B., Wu, T., & Wang, Y. (2016). Face detection with end-to-end integration of a convnet and a 3d model. October. *European conference on computer vision* (pp. 420–436). Cham: Springer.

[25] Li, C., Wang, R., Li, J., & Fei, L. (2020). Face detection based on YOLOv3. *Recent trends in intelligent computing, communication and devices* (pp. 277–284). Singapore: Springer.

[26] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., Berg, A. C. (2016). Ssd: Single shot multibox detector. October. *European conference on computer vision* (pp. 21–37). Cham: Springer.

[27] Megahed, N. A., & Ghoneim, E. M. (2020). Antivirus-built environment: Lessons learned from Covid-19 pandemic. *Sustainable Cities and Society, 61*, Article 102350.

[28] Nguyen, h. (2020). Fast object detection framework based on mobilenetv2 architecture and enhanced feature pyramid. *Journal of Theoretical and Applied Information Technology*, *98*(05).

[29] Ojala, T., Pietikainen, M., & Maenpaa, T. (2002). Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *24*(7), 971–987.

[30] Opitz, M., Waltner, G., Poier, G., Possegger, H., & Bischof, H. (2016). Grid loss: Detecting occluded faces. October. *European conference on computer vision* (pp. 386–402). Cham: Springer.

[31] Opschoor, J. A., Schwab, C., & Zech, J. (2019). Exponential ReLU DNN expression of holomorphic maps in high dimension. *SAM research report, 2019*.

[32] Yang, S., Luo, P., Loy, C. C., & Tang, X. (2016). Wider face: A face detection benchmark. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 5525–5533.

[33] Yang, B., Yan, J., Lei, Z., & Li, S. Z. (2015). Fine-grained evaluation on face detection in the wild. May. In *2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)* (Vol. 1, pp. 1–7).

[34] Zhang, K., Zhang, Z., Li, Z., & Qiao, Y. (2016a). Joint face detection and alignment using multi-task cascaded convolutional networks. *IEEE Signal Processing Letters, 23*(10),1499–1503.

[35] Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems* (pp. 91–99).