26th International Computer Conference, Computer Society of Iran, Tehran, Iran

A Face-Mask Detection Approach based on YOLO Applied for a New Collected Dataset

Sahand Abbasi
Faculty of Computer Engineering
K. N. Toosi University of Technology
sahand.abs@email.kntu.ac.ir

Haniyeh Abdi
Faculty of Computer Engineering
K. N. Toosi University of Technology
hani_abdi@email.kntu.ac.ir

Ali Ahmadi
Faculty of Computer Engineering
K. N. Toosi University of Technology
ahmadi@kntu.ac.ir

2021 26th International Computer Conference, Computer Society of Iran (CSICC) | 978-1-6654-1241-4/20/$31.00 ©2021 IEEE | DOI: 10.1109/CSICC52343.2021.9420599

***Abstract*— Since the beginning of the COVID-19 pandemic, many lives are in danger. According to WHO (World Health Organization)'s statements, breathing without a mask is highly dangerous in public and crowded places. Indeed, wearing masks reduces the chance of being infected, and detecting unmasked people is a waste of resources if not performed automatically. AI techniques are used to increase the detection speed of masked and unmasked faces. In this research, a novel dataset and two different methods are proposed to detect masked and unmasked faces in real-time. In the first method, an object detection model is applied to find and classify masked and unmasked faces. In the second method, a YOLO face detector spots faces (whether masked or not), and then the faces are classified into masked and unmasked categories with a novel fast yet effective CNN architecture. By the methods proposed in this paper, the accuracy of 99.5% is achieved on the newly collected dataset.**

Keywords—mask detection, object detection, classification, YOLO, covid-19, pandemic, real-time

I. Introduction

During the COVID-19 pandemic in many countries, wearing a mask is mandatory and indeed, it reduces the fatality rates. As long as the vaccine is not widely utilized and is not fully protective for every person, wearing a mask is crucial. It is essential to prevent the spread of infection. The authorities need a way to monitor public places like subway stations and shopping centers. Therefore, a need for masked and unmasked face detection arises. Because face detection is a vital part of this process, it requires a large amount of time and resources if done manually and increases the chance of making mistakes in detecting unmasked faces. Machine learning and computer vision techniques may help to automate this process.

In this work, we have proposed a dataset that focuses on Iranian men and women with different clothing. Considering the differences in women's clothing in Iran, most of the existing datasets perform poorly in real-world usage. The collected dataset mostly consists of Iranian women faces with hijab and masks. Benefiting from this dataset, the proposed methods are adapted for mask detection in Iran and other Muslim countries. The first proposed mask detection method uses YOLOv4 object detection architecture to find and classify masked and unmasked faces. In the second proposed method, only face detection is based on YOLOv4. This method is applied to detect both masked and unmasked faces. The detected cropped faces are classified by two simple, yet effective CNN architectures to masked and unmasked classes.

This paper is divided into 6 sections. In the following part, related works are discussed (section 2), then the dataset which was used has been explained in detail (section 3). Illustrating the proposed methods stand in the fourth section (section 4). The fifth section analyzes the results (section 5) and, at last, the proposed methods are summarized, and winded up as a conclusion (section 6).

II. Related Works

A. Face detection

Face detection has long been a research field. Many algorithms have been applied for this problem. The traditional algorithms are based on designed features like HAAR [1] and HOG [2]. Also, object detection algorithms are widely used mainly in recent years, especially RCNN [3-5], SSD [6] and, YOLO [7]. Domain-specific models like MTCNN [21] also perform well on face detection.

YOLO is an object detection architecture that was introduced in 2015. YOLO is real-time and has established state-of-the-art speed for object detection problem. It finds bounding boxes as a regression problem and, it classifies simultaneously. YOLO9000[8] is an improved version of YOLO but, it has a downside of low time efficiency. Besides, YOLO v3[9] and 4[10] were introduced, with improvements in speed and accuracy.

B. Mask detection

Bosheng Qin and Dongxiao Li [11] have used image super-resolution to identify faces and classify them with SRCNet into 3 categories :1) correct face-mask wearing, 2) incorrect face-mask wearing, 3) no face-mask wearing. PCA-based methods [12] perform efficiently in detecting faces without masks (95.25%) but, they don't perform well in identifying masked faces (68.75%).

In some papers [13-15], the methods were to locate the masks, microphones, and glasses first, and remove them in the next steps. They seek to fill the removed parts with GAN-based networks. In [16], they used SVM, decision tree, and ensemble learning to classify images into masked and unmasked classes by the output features gained from a feature extraction network (resnet50[24]). Also, in [17], they have used inceptionv3[50] as the backbone and added 4 layers (average pool, flatting layer, dense (128), dense (2)) to classify masked and unmasked faces.

978-1-6654-1241-4/21/$31.00 ©2021 IEEE

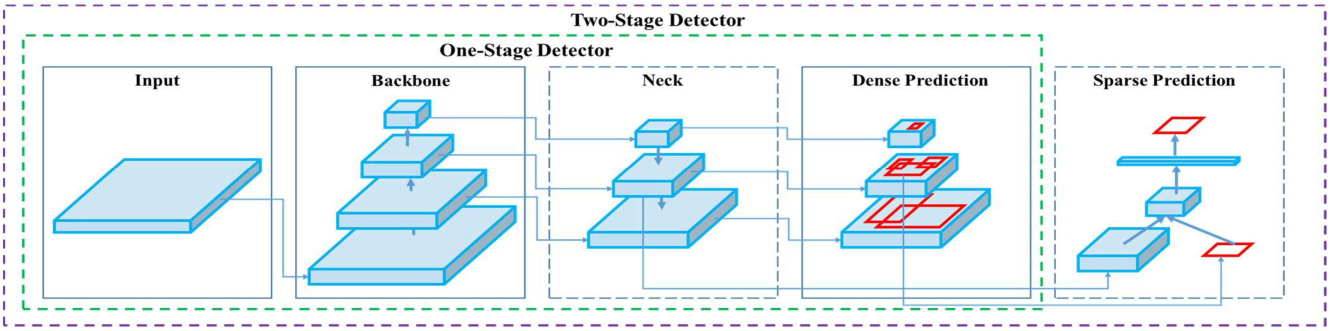

Fig. 1. The architecture of the YOLOV4 object detector.
The input can be either Images, Patches or Image Pyramids and Backbone can be either VGG16[23], ResNet-50, SpineNet [25], EfficientNet-B0/B7[26], CSPResNeXt50 and CSPDarknet53.
The Neck module can be Additional blocks (SPP [27], ASPP [28], RFB [29], SAM [30]) or Path-aggregation blocks (FPN [31], PAN [32], NAS-FPN [33], Fully-connected FPN, BiFPN [34], ASFF [35], SFAM [36])
For Head modules, Dense Prediction (RPN [37], SSD [38], YOLO, RetinaNet [40], CornerNet [41], CenterNet [42], MatrixNet [43], FCOS [44]) or Sparse Prediction (Faster R-CNN [45], R-FCN [46], Mask RCNN [47], RepPoints [48]) can be used.

III. Dataset

A. Dataset structure

The proposed dataset is a product of accumulated images, samples from MAFA [19], and samples from WIDER FACE [18].

A total number of 4066 images are selected from the MAFA dataset. MAFA dataset is a masked face detection benchmark dataset. MAFA's images are collected from the internet and, it has a total number of 30811 images and 35806 masked faces. Faces in this dataset have various orientations and occultation degrees. Also, 3894 images are from the WIDER FACE dataset. WIDER FACE is a face detection dataset and contains images selected from the publicly available WIDER dataset. It contains 32203 images and 393703 faces with a high degree of variability in scale, poses, and occultation. Also, 1500 images of Iranian people are collected and labeled. There are 500 faces of men and 1000 of women. Images are captured from both indoor and outdoor environments with variant weather, brightness, and angles. These images are labeled by drawing a bounding rectangle around each face. For each face, 5 numbers are stored. The first number determines that this face is masked or not and, the other 4 numbers are the coordinates of the bounding rectangle.

A total number of 7320 images are used for training and, 2139 images are used for testing, which makes 9459 images.

Some samples of the proposed dataset are shown in Figures 2 and 3.

Fig. 2. Samples of the proposed dataset for unmasked faces

Fig. 3. Samples of the proposed dataset for masked faces

B. Data augmentation

These images are augmented using random perspective transformations, brightness manipulation, and addition of Gaussian noise which increased the size of the dataset to around 25000 images.

IV. PROPOSED METHOD

YOLO (you only look once) is one of the fastest object detection algorithms. In April 2020 the 4th generation of YOLO was released. YOLO v4 has obtained an AP value of 43.5% on COCO [49] dataset along with 65fps real-time speed on the TESLA v100. It also uses BoF (bag of freebies) and BoS (bag of specials). YOLO v4 can predict up to 9000 classes. YOLO v4 is based on a single convolution neural network. The CNN divides an image into regions and, predicts boundary boxes and, probabilities of each region. YOLO sees the entire image during training and test time and, as a result, it implicitly encodes contextual information about classes as well as their appearances. YOLO v4 uses CSPDarknet53[22] as a backbone and, it also uses spatial pyramid pooling and PANet path-aggregation neck. Besides, average precision has increased 10% compared to YOLOv3 on the COCO dataset.

YOLOv4 is adapted as an object detector in the first proposed method. Both masked and unmasked faces are detected as two different classes. Figure 1 shows YOLO v4's architecture. The input size is 416*416 and, the outputs are 7-dimensional vectors. The first dimension is to define whether an object is present in the grid or not. The next four elements specify a bounding box and the last two dimensions represent the classes' probabilities.

The second proposed method detects faces in an image and classifies the cropped faces into masked and unmasked categories. In this method, YOLO v4 is also applied as the face detector. Both masked and unmasked faces are detected as a single class. This single class object detector is trained to detect faces whether they are masked or not. In this method

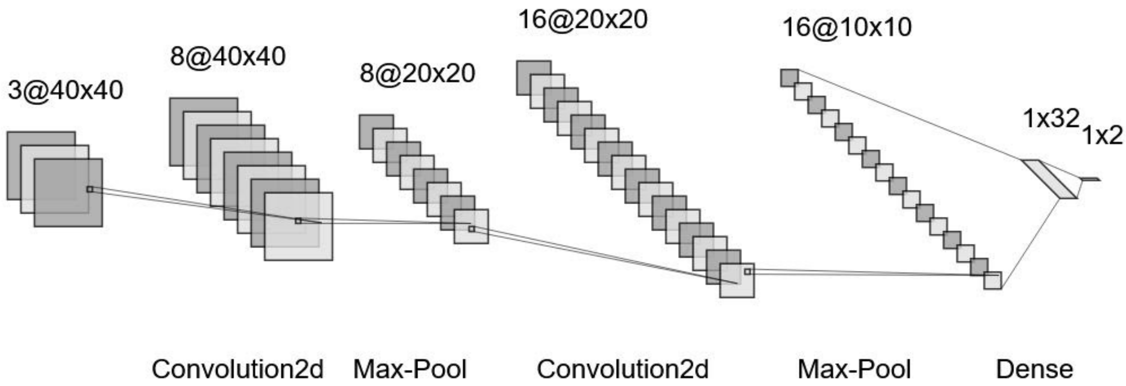

Fig. 4. The architecture of the second proposed method with the first classifier. The inputs of this network are the cropped faces that were detected by YOLO (with 40*40*3 dimensions). There are 2 Convulotion2d and max-pooling blocks followed by 2 dense layers.

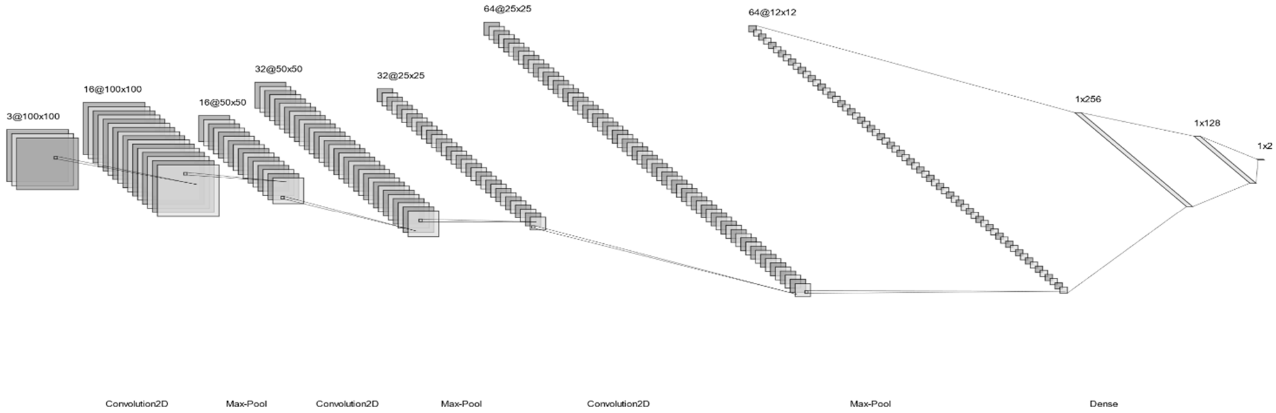

Fig. 5. The architecture of the second proposed method with the second classifier. The inputs of this network are the cropped faces that were detected by YOLO (with 100*100*3 dimensions). There are 3 Convulotion2d and max-pooling blocks followed by 3 dense layers.

the previous object detector architecture is used with the exception that it has been trained with only one class (faces).

After detecting faces, they are cropped and fed into a fast, yet effective classifier. At first, the approach was to combine both classification loss and object detector loss, however, it didn't perform well enough. Thus, the problem was divided into two different stages. First, to train a YOLO based face detector and then to create a masked/unmasked classifier. Therefore, in the second method, two different architectures are proposed for the classifiers.

In the first architecture which is shown in Figure 4, the input shape is 40*40*3, which are the width, height, and channel shapes for cropped faces, respectively. The input is normalized. Thus, the pixels will be bounded to values between (0,1). A 3*3*8 2D convolution is applied with Relu activation followed by a 2*2 max pooling. Then there is a 3*3*16 2D convolution with Relu activation followed by 2*2 max pooling. The next layer is a flatten layer followed by a 32-neuron dense layer with Relu activation and finally, a dense layer with 2 neurons and SoftMax activation. The second proposed architecture (Figure 5) is similar to the first one, but has another conv-MaxPool block and an additional 256-neuron dense layer and, 128 neurons instead of 32 neurons at the layer before the SoftMax.

V. Experimental result

For the object detection models, a batch size of 64 was used and, the initial learning rate was 0.0013. The proposed YOLO v4 object detection models were trained from scratch for 1000 epochs.

In the classifiers, a batch size of 8, a validation split of 0.2, and, an Adam optimizer were used. The initial learning rate was 0.001 and the model was trained for 200 epochs.

Precision, recall, and mAP are some used metrics in measuring the accuracy of object detectors. Precision measures how accurate the predictions are and recall measures how well all the positive classes were detected.

Precision and Recall are calculated with the below formulas. In Equations 1 and 2, TP, FP, and FN are short forms of true positive, false positive, and false negative, respectively.

$$Precision = \frac{TP}{TP+FP} \qquad (1)$$

$$Recall = \frac{TP}{TP+FN} \qquad (2)$$

The mean average of precision or mAP is the average of the AP calculated for all the classes that is illustrated in Equation 3:

$$MAP = \frac{\sum_{q=0}^{Q} \mathrm{AveP(q)}}{Q} \qquad (3)$$

Accuracy is used as another metric to measure the performance of the classification architectures. In Equation 4, TN stands for true negative.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \qquad (4)$$

The metrics for all methods are shown in Table I. Accuracy of 99.5% has been achieved by the second masked-unmasked classifier. The final mAP 0.5 is 98% after 1000 epochs for the first proposed method. The FPS column is measured on TITAN X Nvidia GPU.

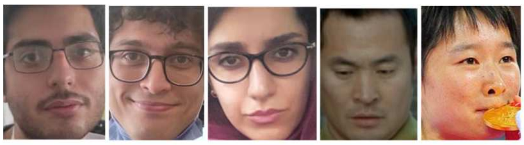

Fig. 6. Samples of the dataset that were classified as unmasked faces (second method).

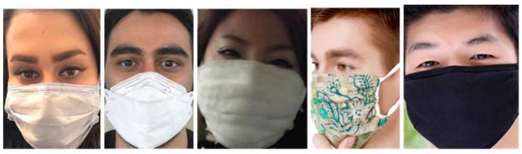

Fig. 7. Samples of the dataset that were classified as masked faces (second method).

Pictures in Figure 6 are some samples which were detected as unmasked faces and Figure 7, plots samples of the faces that were detected as masked ones, both with the second method. In Figure 10, detected bounding boxes for masked and unmasked faces along with their probabilities are plotted.

In Figure 8 the epoch-precision and the epoch-recall curves along with epoch-mAP curves are plotted. Also, in Figures 9 and 11 epoch-loss and epoch-accuracy plots for the first and the second classifier are illustrated, respectively.

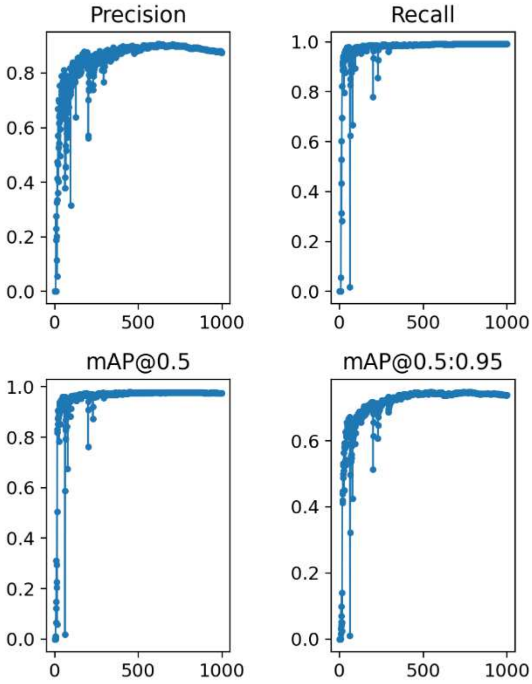

Fig. 8. The metrics for the first method.

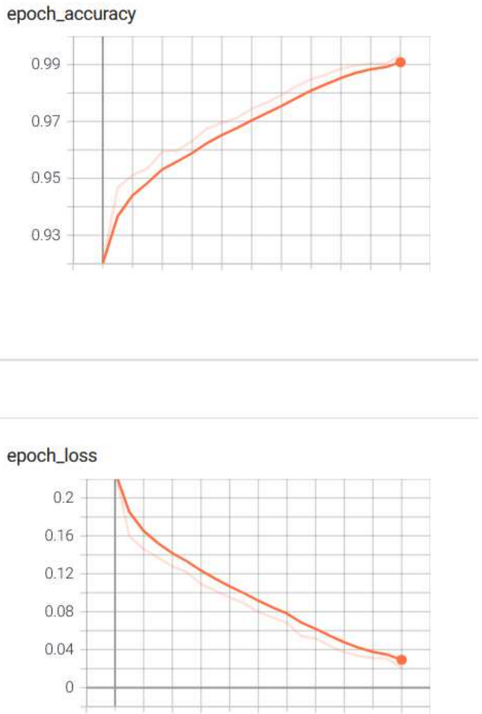

Fig. 9. The metrics for the second method with the first classifier.

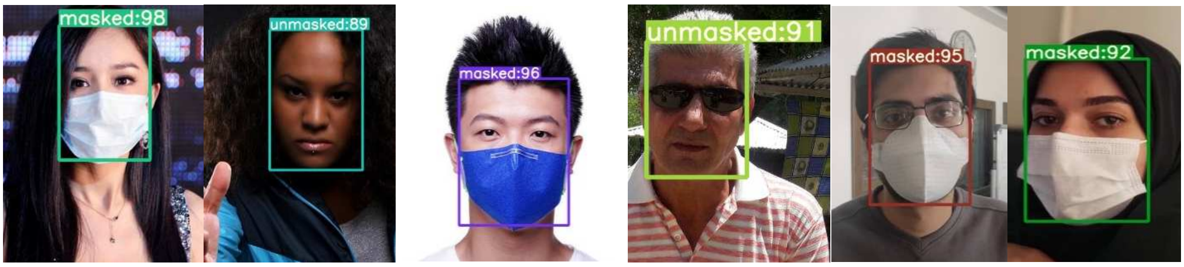

Fig. 10. Samples of the first method's outputs (YOLO object detection).

TABLE I. MODELS COMPARISON

Model/Metric	Precision	Recall	mAP 0.5	Acc	FPS
Method1	92%	97%	98%	-	49
Face Detection	92%	97%	97%	-	50
Classification 1	97%	97%	-	99%	330
Classification 2	**99%**	**99%**	-	**99.5%**	200

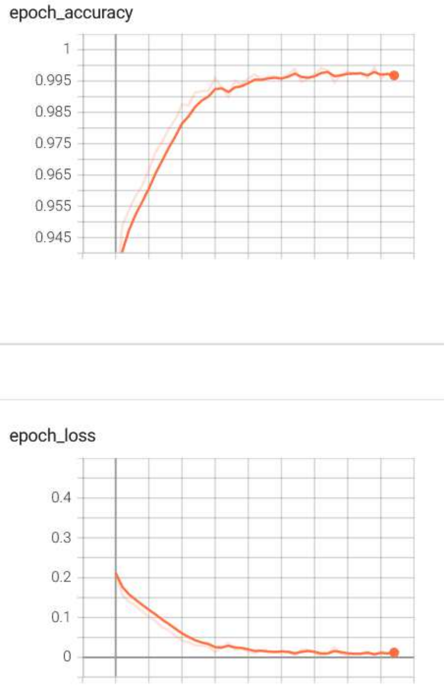

Fig. 11. The metrics for the second method with the second classifier.

VI. CONCLUSION

Regarding the fact of spreading the COVID-19 virus, wearing a face mask is one of our new priorities. Thus, detecting masked-unmasked faces is essential in this period.

In this paper, a novel masked and unmasked dataset has been collected, containing 1500 images from Iranian masked-unmasked faces. This dataset fits Iranian culture and, the proposed models perform quite efficiently in real-world usage (especially for faces wearing hijab).

To detect masked-unmasked faces, two different methods were proposed. The first method is a masked-unmasked object detector and, the second method contains a face detector and two different classifiers. This model achieves 99.5% accuracy on the masked and unmasked collected dataset. This classification can be performed with a real-time speed for real-world usage.

REFERENCES

[1] Viola, P., Way, O.M., Jones, M.J.: Robust real-time face detection. Int. J. Comput. Vision 57(2), 137–154 (2004)

[2] Forsyth, D.: Object detection with discriminatively trained part-based models. IEEE Trans.Pattern Anal. Mach. Intell. 32(9), 1627–1645 (2010).

[3] Girshick, R., Donahue, J., Darrell, T., et al.: Rich feature hierarchies for accurate objects. detection and semantic segmentation. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 580–587. IEEE Computer Society (2014)

[4] Girshick, R.: Fast R-CNN. Comput. Sci. (2015)

[5] Ren, S., He, K., Girshick, R., et al.: Faster R-CNN: towards real-time object detection with region proposal networks (2015)

[6] Liu, W., Anguelov, D., Erhan, D., et al.: SSD: Single Shot MultiBox Detector. In: European Conference on Computer Vision, pp. 21–37. Springer, Cham (2016)

[7] Redmon, J., Divvala, S., Girshick, R., et al.: You only look once: unified, real-time object detection (2015)

[8] Redmon, J., Farhadi, A.: YOLO9000: better, faster, stronger. In: 2017 IEEE Conference onComputer Vision and Pattern Recognition (CVPR), pp. 6517–6525 (2017)

[9] Redmon J, Farhadi A.: YOLOv3: an incremental improvement (2018)

[10] Bochkovskiy, Alexey, Chien-Yao Wang, and Hong-Yuan Mark Liao. "YOLOv4: Optimal Speed and Accuracy of Object Detection." arXiv preprint arXiv:2004.10934 (2020).

[11] QIN, B., Li, D.: Identifying facemask-wearing condition using image superresolution with classification network to prevent covid-19 (2020)

[12] Ejaz, M.S., Islam, M.R., Sifatullah, M., Sarker, A.: Implementation of principal component analysis on masked and non-masked face recognition. In: 2019 1st International Conference on Advances in Science, Engineering, and Robotics Technology (ICASERT). pp. 1–5 (2019)

[13] Jeong-Seon Park, You Hwa Oh, Sang Chul Ahn, Seong-Whan Lee: Glasses removal from the facial image using recursive error

[14] compensation. IEEE Transactions on Pattern Analysis and Machine Intelligence 27(5), 805–811 (2005)

[15] Khan MKJ, Ud Din N, B.S.Y.J.: Interactive removal of microphone object in facial images. Electronics 8(10) (2019)

[16] N. Ud Din, K. Javed, S. Bae, J. YiA novel GAN-based network for the unmasking of masked face IEEE Access, 8 (2020), pp. 44276-44287,

[17] Loey, Mohamed, et al. "A hybrid deep transfer learning model with machine learning methods for face mask detection in the era of the COVID-19 pandemic." Measurement 167 (2020): 108288.

[18] Chowdary, G. Jignesh, et al. "Face Mask Detection using Transfer Learning of InceptionV3." arXiv preprint arXiv:2009.08369 (2020).

[19] 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, pages 5525–5533, 2016.

[20] S. Ge, J. Li, Q. Ye, and Z. Luo. Detecting masked faces in the wild with lle-cnns. In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 2682–2690, 2017.

[21] Zhang, K., Zhang, Z., Li, Z., et al.: Joint face detection and alignment using multitask cascaded convolutional networks. IEEE Signal Process. Lett. 23(10), 1499–1503 (2016)

[22] Chien-Yao Wang, Hong-Yuan Mark Liao, Yueh-Hua Wu, Ping-Yang Chen, Jun-Wei Hsieh, and I-Hau Yeh. CSPNet: A new backbone that can enhance the learning capability of CNN. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshop (CVPR Workshop), 2020

[23] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556, 2014.

[24] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun.Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 770–778, 2016

[25] Xianzhi Du, Tsung-Yi Lin, Pengchong Jin, Golnaz Ghiasi, Mingxing Tan, Yin Cui, Quoc V Le, and Xiaodan Song. SpineNet: Learning scale-permuted backbone for recognition and localization. arXiv preprint arXiv:1912.05027, 2019

[26] Mingxing Tan and Quoc V Le. EfficientNet: Rethinking model scaling for convolutional neural networks. In Proceedings of International Conference on Machine Learning (ICML), 2019

[27] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Spatial pyramid pooling in deep convolutional networks for visual recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 37(9):1904–1916, 2015

[28]] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L Yuille. DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 40(4):834–848, 2017

[29] Songtao Liu, Di Huang, et al. Receptive field block net for accurate and fast object detection. In Proceedings of the European Conference on Computer Vision (ECCV), pages 385–400, 2018

[30] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. CBAM: Convolutional block attention module. In Proceedings of the European Conference on Computer Vision (ECCV), pages 3–19, 2018

[31] Tsung-Yi Lin, Piotr Dollar, Ross Girshick, Kaiming He, ´ Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 2117–2125, 2017

[32] Shu Liu, Lu Qi, Haifang Qin, Jianping Shi, and Jiaya Jia. Path aggregation network for instance segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 8759–8768, 2018.

[33] Golnaz Ghiasi, Tsung-Yi Lin, and Quoc V Le. NAS-FPN: Learning scalable feature pyramid architecture for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 7036–7045, 2019

[34] Mingxing Tan, Ruoming Pang, and Quoc V Le. EfficientDet: Scalable and efficient object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2020

[35] Songtao Liu, Di Huang, and Yunhong Wang. Learning spatial fusion for single-shot object detection. arXiv preprint arXiv:1911.09516, 2019

[36] Qijie Zhao, Tao Sheng, Yongtao Wang, Zhi Tang, Ying Chen, Ling Cai, and Haibin Ling. M2det: A single-shot object detector based on multi-level feature pyramid network. In Proceedings of the AAAI Conference on Artificial Intelligence (AAAI), volume 33, pages 9259–9266, 2019

[37] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In Advances in Neural Information Processing Systems (NIPS), pages 91–99, 2015

[38] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C Berg. SSD: Single shot multibox detector. In Proceedings of the European Conference on Computer Vision (ECCV), pages 21–37, 2016

[39] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 779–788, 2016

[40] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollar. Focal loss for dense object detection. In ´Proceedings of the IEEE International Conference on Computer Vision (ICCV), pages 2980–2988, 2017

[41] Hei Law and Jia Deng. CornerNet: Detecting objects as paired keypoints. In Proceedings of the European Conference on Computer Vision (ECCV), pages 734–750, 2018

[42] Kaiwen Duan, Song Bai, Lingxi Xie, Honggang Qi, Qingming Huang, and Qi Tian. CenterNet: Keypoint triplets for object detection. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), pages 6569–6578,2019

[43] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. MobileNetV2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 4510–4520, 2018

[44] Zhi Tian, Chunhua Shen, Hao Chen, and Tong He. FCOS: Fully convolutional one-stage object detection. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), pages 9627–9636,2019

[45] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In Advances in Neural Information Processing Systems (NIPS),pages91–99,2015

[46] Jifeng Dai, Yi Li, Kaiming He, and Jian Sun. R-FCN: Object detection via region-based fully convolutional networks. In Advances in Neural Information Processing Systems (NIPS), pages 379–387, 2016

[47] Kaiming He, Georgia Gkioxari, Piotr Dollar, and Ross Gir- ´shick. Mask R-CNN. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), pages2961–2969, 2017

[48] Ze Yang, Shaohui Liu, Han Hu, Liwei Wang, and Stephen Lin. RepPoints: Point set representation for object detection. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), pages 9657–9666, 2019

[49] Lin, Tsung-Yi, et al. "Microsoft coco: Common objects in context." European conference on computer vision. Springer, Cham, 2014.

[50] Szegedy, Christian, et al. "Rethinking the inception architecture for computer vision." Proceedings of the IEEE conference on computer vision and pattern recognition. 2016.