Masked Face Detection and Calibration with Deep Learning Models

Xingze Li*
School of Electronic Engineering, Xidian University
*Corresponding Author’s E-mail: edmond4190913@163.com

Abstract. Under the COVID-19 pandemic, the demand that face detection devices should be enhanced to detect masked faces is imperative. In this study, we utilize several state-of-the-art face detection models and compare them on various unmasked and masked human face datasets. Moreover, by analyzing the results we obtain, we evaluate these disparate models and discover some problems. Attempting to overcome the problems discovered, we propose and implement several improvements, and acquire more results for analysis. At length, we propose some ideas for future research directions.

1. Introduction
Since the outbreak of corona virus in January 2020, there are more than 100 million people infected the COVID-19 virus1, epidemiologists claim that wearing a mask can preclude the virus from spreading efficiently2 on account of its capacity of airborne transmission. Therefore, the authorities worldwide begin to request the public to wear a mask when they stay in public areas. However, such policies cause serious challenges to existing face recognition systems, since facial features of nose and mouth are covered, which results in fewer features for recognition instruments to detect.

Traditionally, adaptive boosting and cascade classifier are used for face detection, which is also used for masked cases [1]. However, more recently, deep learning models are dominant in masked face detection, as we would discuss in Section 2, e.g., fine-tuned InceptionV3 in a transfer based approach and pre-trained VGG or ArcFace based models. Our aim belongs to compare various models on a number of different datasets, e.g., YOLO and SSDFace detect model, on Real-World-Masked-Face-Dataset, released by Wuhan University and relevant laboratory and simulated masked dataset like masked LFW (Labeled Faces in the Wild). Many studies contribute to establish new datasets or to masked face recognition. Datasets of masked faces are respectively rare, which impedes the training of proposed models, and these studies make considerable contribution. There are plenty of machine learning and deep learning methods used to detect the mask on face, some of which are based on fine-tuned existing and pre-trained models, all these models achieve good performances. As for face recognition, there are two research direction, namely, using remnant faces in images to finish recognition or recovering the whole face to provide as many as features for models to recognize. In general, methods above use different amounts of features in the face. Therefore, the performance vary a lot as well.

Nonetheless, there are still few researches focusing on the condition that people wear masks. Some studies propose several new datasets, there are still many studies intended to detect masked face

https://www.who.int/emergencies/diseases/novel-coronavirus-2019/situation-reports
https://www.who.int/emergencies/diseases/novel-coronavirus-2019/advice-for-public
directly or utilize some different approaches to detect the masked faces. However, this approach is time-consuming and costly, and researchers have to invest a large quantity of resources to collect data to train the novel models. In addition, plenty of existing models that are already deployed in lots of systems are required to be displaced because their low performance in masked face detection. The problem we need to tackle is whether we can detect the masked face based on the combination of existing models.

In this study, we attempt models including yoloface, ssdface, retinoface, centerface and lffdface on datasets AFDB, AgeDB-30, CASIA-WebFace, CFP-FP, LFW, Real-World-Masked-Face-Detection, and WHN. The results show that:

1. The evaluated models have brilliant performances on those unmasked datasets, all of which achieve nearly 100% accuracy.
2. The performances of these models on masked datasets decrease in various degrees. For example, the accuracy of yoloface model decreases by 30-40% (from more than 90% to 50-60% or so) when they detect masked faces from datasets listed in Section 3 and the accuracy of ssdface model decreases by 5-10% (from nearly 100% to 90-95%). Moreover, the accuracy of retinoface model decreases by 20% or so (from nearly 100% to 80%), the accuracy of centerface model decreases with remarkable variation, and the accuracy of lffdface decreases by 2-5% (from nearly 100% to 95%).

The rest of our study is as follows:

1. We analyze and compare the performances of different deep learning models that are trained on several datasets.
2. We then make some judgments in Section 4 aiming to make further exploration of the results. Some histograms contribute to illustrate the situation. Furthermore, we expose some problems, in general and detailed, of yoloface and ssdface detection model.
3. We achieve some improvement on certain models we utilize, namely, YOLOv2 detection model and try to use the ameliorated YOLOv3 and YOLOv4 model to generate new results for general judgment and contrast.
4. In the end, we propose several problems and give some possible solutions, looking forward to inspire further research.

2. Related Work

In this section, we discuss the related work from the three aspects, namely, masked face datasets, masked face detection and masked face recognition.

2.1. Masked Face Datasets

There are two approaches of creating masked face datasets. The first approach is to collect real-world face images on the Internet with masks. However, this approach is time-consuming and costly. It is also difficult to build a large image dataset in this approach. The alternative approach is to detect the anchor points in the face images and append fake masks. In this approach, it is easy to create a large masked image face dataset because we have already many open face image datasets, e.g., WebFace and LFW, etc. Some competitions are also related to masked face recognition, which releases the corresponding dataset for research [2].

In this part, we introduce some of these masked face datasets.

In [3], the authors present a novel dataset focusing on Indian, considering that those existing datasets lack the cultural diversity and are collected in unrestricted settings. The proposed dataset contains 1,574 images of several kinds of unconstrained circumstances. The authors implement two different experiments to evaluate the proposed IMFW dataset. The first one evaluates the performance
of four pre-trained models, namely, VGGFace, ResNet-50 (Trained on the VGGFace2 dataset), LightCNN29, and ArcFace. The second utilizes existing loss functions, i.e., contrastive loss and triplet loss for masked face recognition. The result shows that the model performs poorly in both subsets and the complete set, while the lowest accuracy corresponds to 20% in VGGFace model and the best accuracy corresponds to 60% in ArcFace model.

Many face recognition instruments are required to improve their performance on masked face recognition. While most existing face recognition approaches base on deep learning, we need larger publicly available datasets. In [4], the authors propose to construct masked face datasets by different means, they propose three types of masked face datasets. The first one is Masked Face Detection Dataset (MFDD), which contains 24,771 masked face images. The second one is Real-world Masked Face Recognition Dataset (RMFRD), which contains 5,000 masked people images and 90,000 unmasked images of same subjects. The last one is Simulated Masked Face Recognition Dataset (SMFRD) with 500,000 face images of 10,000 subjects.

In [5], the authors propose three types of masked face detection dataset, namely, Correctly Masked Face Dataset (CMFD), Incorrectly Masked Face Dataset, and their combination MaskedFace-Net. There are 137,016 images divided into correctly worn and incorrectly worn masked face categories.

### 2.2. Masked Face Detection

With masks, it would become more difficult to detect either faces or masks, both of which have been considered in the literature. During the COVID-19 period, wearing a mask has become a common approach to prevent the virus from spreading. And to comply with the social distance policy, many restaurants and markets have to ask the customers to wear masks. With the existing cameras, some tech companies have proposed the use of automatic mask detection solutions. Furthermore, we want to detect the existence of faces to make sure the previous relevant systems still work even with masks. In this part, we introduce some relevant studies in this direction.

There are many deep learning models [6] that aid in identifying a masked face. In [7], the aim is to develop a deep learning model for detecting unmasked people. The work implements image augmentation firstly, then utilizes the fine-tuned InceptionV3 as the pre-trained model in a transfer learning based approach. The model is further trained on Simulated Masked Face Dataset, which consists of 785 masked faces and 785 unmasked faces. The performance shows that the InceptionV3 model attains an accuracy of 99.92% in training phase and 100% in testing phase, a precision of 99.9% in training phase and 100% in testing phase. This model has a remarkable performance and exceeds many other models.

There are increasing need to detect subjects with masks to preclude the virus from spreading during the COVID-19 pandemic. A full training pipeline based on the ArcFace work is proposed in [8] to overcome this problem. The network bases on a modified ResNet-50 model. This work selects the MS1MV2 dataset that contains 5.8 million images and 85,000 identities for training. In the testing phase, the model achieves nearly 100% accuracy for mask-usage verification. The results surpass all others with a 12% accuracy increase than the original.

For helping existing recognition system of monitoring public areas to identify whether a person wears mask, the authors in [9] propose CNN and VGG16 based models to detect the mask, the model are trained on Simulated Masked Face Dataset (SMFD) with 1509 images. The results show that the CNN model achieves training, validation and testing accuracy of 96.35%, 96.35%, 97.42%, while another model VGG-16 achieved 99.47%, 98.59%, 98.97% respectively. The results are better in just 25 epochs with SMFD dataset.

During the COVID-19 pandemic, people are required to wear masks in public areas, which challenges those existing monitoring instruments to detect people whether wear a mask or not. The authors in [10] propose a two-stage approach to detect wearing masks, bases on the transfer model of Faster-RCNN and InceptionV2 that trains on a dataset of 7,804 images with 26,403 wearing masks. The results show that this approach achieves an accuracy of 97.32% for simple scene and 91.13% for complex scene, exceeding other methods.
While people are required to wear masks in public areas under COVID-19 pandemic, existing monitoring instruments are facing challenges of detecting people with masks efficiently. The authors in [11] propose a deep learning model based on multi-graph convolutional networks (MGCN). The model trains on the real world masked face dataset which contains more than 7600 images and achieves an accuracy of 97.9%.

3. Dataset Exhibition

In this study, we utilize all the public datasets as we can find from the literature, to compare models for unmasked face detection and masked face recognition. Table 1 describes the datasets we select with the data size. This comprehensive collection would also be inspiring for follow-up studies.

| Dataset               | Unmasked Size | MFRD-masked Size | AgeDB-30-masked Size | LFW-masked Size | whn-masked Size | CASIA-WebFace (masked) Size |
|-----------------------|---------------|-------------------|----------------------|-----------------|-----------------|-----------------------------|
| MFRD                  | 90,468        | 6,000             | 6,000                | 358             | 494,414         |                             |
| AgeDB-30              | 6,000         | 6,000             |                      |                 |                 |                             |
| LFW                   | 6,000         | 6,000             |                      |                 |                 |                             |
| whn                   | 358           |                   |                      |                 |                 |                             |
| CASIA-WebFace (masked)| 494,414       |                   |                      |                 |                 |                             |

There are six datasets in total, four of which have both unmasked part and masked part, namely, MFRD dataset, AgeDB-30 dataset, LFW dataset and WHN dataset. While CASIA-WebFace and RMFD (Real-World-Masked-Face-Detection) are merely masked datasets.

AFDB, whn-masked and RMFD are all created by Wuhan University during corona virus pandemic, aiming to help deal with hygienic emergency and contribute to face detection research [12]. AFDB consists of 90,000 unmasked images and 2,000 masked faces or so, each masked face correlated to several unmasked face images. In addition, we use RMFD as supplementary masked dataset that contains about 4,000 images. The designer describes WHN as a verification dataset, containing more than 4,000 images of about 300 actors, similarly, each unmasked person correlated to various masked images. Particularly, images in RMFD dataset differ with each other significantly, some of which are stretched with relatively abnormal proportion whereas others may have various background and dim light conditions. AgeDB-30 [13], CASIS-WebFace [14] and LFW [15] datasets are all simulated masked datasets.

AgeDB-30 consists of images of 16,488 celebrities, like scientists, politicians, writers etc. We divide them into unmasked part of 6,000 images and masked part of equal 6,000 images. CASIS-WebFace dataset contains nearly 500,000 images of more than 10,000 celebrities, all of which are collect from IMDB website. LFW is a dataset often used for face recognition, in which the images derive from natural scenes with variety of factors like different postures, light conditions, expression, occlusion etc. Containing more than 13,000 pictures, whose sizes are all for 250x250. We select 6,000 for unmasked dataset and 6,000 for masked dataset.

During the research process, we have also tried the CFP (Celebrities in Frontal-Profile in the Wild) dataset. Nonetheless, we abandon this dataset since images in unmasked part are front sides whereas in masked part are profiles.

4. Models utilized

In this work, we utilize various networks to test on several datasets, all of which are highly advanced models that have been used and perfected by amounts of researchers. Here the synthetic description and detailed facet of contribution in these works are listed.

4.1. YOLO 9000 [16]

Advanced from YOLOv1, the researchers make some improvements to augment accuracy of the model while keep the advantage in speed. Ameliorations like Batch Normalization (BN) and fine-tune of dealing resolution, the former enable YOLOv2 to gain an immense augment of convergence meanwhile decreases the dependence on other regularization methods (Abandoning dropout
regularization without overfitting), at length achieves more than 2% improvement in mAP. In the latter progress, the authors fine tune the input resolution in classification network for 10 epochs, considering the problem that the previous YOLO model [17] has to switch directly to new resolution after pre-trained on ImageNet, from 224 x 224 to the resolution of 448. By which the network has time to adjust to higher input resolution and attains an improvement on mAP for 4%.

Using Faster R-CNN for reference [18], the authors remove the fully connected layers from previous YOLO whereas predict the bounding boxes by utilizing anchor boxes which results in tiny declination of mAP whereas significant augment of recall ratio. To get better priors, the authors utilize k-means method in which the authors also supersede the original Euclidean distance by using their certain distance metric related to IOU. To ensure the stability of the model, the authors follow location prediction method in YOLO, which means to predict location coordinates related to the grid cell location.

Additionally, there are tiny approaches as by adding a passthrough layer to provide a way to utilize better feature maps to localize some smaller items and training YOLOv2 model on images of various sizes, which improves the accuracy and is able to achieve relatively fast detection in small resolution at the same time.

4.2. **SSD** [19]

SSD is a light-weight algorithm with brilliant accuracy, faster than YOLO, nearly with same accuracy as Faster RCNN. Comparing with Faster RCNN [20], SSD has no process of generating the proposal region, which improve detect speed significantly. Different with traditional methods for detection of objects with a variety of sizes that detect separately at length synthesize the results by Non-maximum suppression, the SSD algorithm achieves equal result by utilizing feature maps in disparate layers. The backbone network is VGG16, the authors make some fine-tunes by eliminating and converting and adding some fully connected layers.

SSD utilizes convolutional layers with different depths to predict objects with disperse sizes. High-resolution correspondent to small object, i.e. feature maps in lower layers settled with smaller anchor, higher layers vice versa. There are two different convolution kernels that deal the output of feature maps with disparate sizes, one outputs confidence for classification while the other outputs location for regression. The authors introduce default box involved in prediction and prior box similar to anchor in Faster RCNN, by matching prior box and ground truth through IOU the process generate positive and negative examples, which are controlled on certain ratio. To some extent, training process becomes easier and

The data augmentation the authors implement has significant effects on model performance. For each sample, it will be sampled to generate several samples among which to be selected stochastically for training.

4.3. **Retinaface** [21]

There are two backbone networks can be used for feature extraction during training process, Resnet and MobilenetV1-0.25, the former contribute to higher accuracy and the latter enable the model to realize real-time detection on CPU.

As a key method in this work, depthwise separable convolution greatly reduce the amount of parameters. Similar to Retinanet, Retinaface adopts Feature-Pyramid Network which focus on these rear feature layers. Using 1x1 convolution to adjust its channels then utilize up sample to fuse the features. For further feature extract, the authors utilize SSH [22] with parallel structure to enhance receptive field by piling up 3x3 convolution to supersede 5x5 and 7x7 convolution.

The results of prediction can be categorized into face classification prediction, face regression prediction and facial landmark regression prediction which are separately utilized to judge whether there is object included in anchor, adjust the anchor to generate predicted box, adjust the anchor to acquire facial landmarks. Non-maximum suppression is also implemented after steps above to
eradicate repeat anchors. By annotating five facial landmarks and adding extra supervised and self-supervised mesh decoder branch the authors minimize the multi-task loss function to get a brilliant one.

Comparing with MTCNN [23], the accuracy on face verification has a significant improvement. Additionally, on single CPU, it can achieve a real-time detection on VGA-resolution images.

4.4. LFFD [24]
Light and Fast Face Detector (LFFD), is an anchor-free and one-stage algorithm suitable to edge devices. Enlightened by correlation of RF, ERF (receptive field, effective receptive field) play the core new design of authors’ work.

Considering the drawbacks of anchors, the authors put forward that RF in feature maps are natural anchors to deal the shortages of anchor based methods and various RF strategies that various image sizes correspond to disperse ERFs and EFs. By controlling the stride of RF, it can attain 100% coverage of face in theory. Additionally, RF is able to handle consecutive face sizes, by which means the imbalance of anchor can be solved.

The authors design the network architecture with four parts with different sizes, each of which has several loss branches since it has to ensure the features of faces to be robust and 100% coverage. They also utilize data augmentation including sampling stochastically, horizontal flip and colour distortion. The key part loss function in each branch includes two sub-branches for face classification and bounding box regression by which it can deal with RFs of various sizes.

4.5. CenterFace [25]
CenterFace is a light-weight anchor-free algorithm with ability to predict face box and face landmark in real-time and with brilliant accuracy, which takes advantage of contextual maps and facial landmarks in learning. By utilizing the centre point of face bounding box, the facial landmark can be regressed straight to features.

The authors select Mobilenetv2 as the backbone network, by which the structure can be small but powerful. Similar to Retinaface we have discussed above, CenterFace also adopt Feature Pyramid Network to adjust channel and make further feature extraction, from where features can be regressed directly.

4.6. YOLOv3 [26]
While still utilizing dimension clusters as anchor boxes, the author replaces the IOU of bounding box and a ground truth with confidence 1 using logistic regression, by which recall rate can be relative high for small object detection. YOLOv3 extracts features from 3 various scales by utilizing feature pyramid networks [27]. By using residual network and upsampling, this YOLOv3 work is able to deal semantic information with further meaning, containing more information from previous layers.

5. Evaluation and Improvement
In this section, we focus on analysis of these results. Table 2 below shows the results we get from models tested on several datasets we used. The wide line divides the results into datasets with unmasked-masked pairs and masked datasets. Furthermore, we indicate some advantages and drawbacks of models concerning some certain problems.

| Dataset         | Model       | yolo | ssdface | retinaface | centerface | lffdface |
|-----------------|-------------|------|---------|------------|------------|---------|
| AFDB-unmasked   | yolo        | 88.93| 99.95   | 99.80      | 99.44      | 99.81   |
| AFDB-masked     | yolo        | 41.81| 90.92   | 72.99      | 81.07      | 93.60   |
| AgeDB-30-unmasked | yolo     | 99.93| 100.00  | 99.73      | 99.98      | 100.00  |
| AgeDB-30-masked | yolo       | 49.15| 94.48   | 79.63      | 86.63      | 94.03   |
| LFW-unmasked    | yolo        | 99.97| 100.00  | 100.00     | 99.98      | 100.00  |
| LFW-masked      | yolo        | 61.17| 91.63   | 93.18      | 93.4       | 98.13   |
5.1. General Analysis
For unmasked datasets, the results indicate that all of models we used manage to achieve brilliant results, with nearly 100% accuracies. Since these unmasked images are with a good quality, clear and has few occlusion, these algorithms face only a few challenges.

We focus our analysis on results of masked datasets. All of the models we use achieve a worse performance at various extents. Moreover, their performances differ significantly with each other. For example, ranging from 50% to 95%, on AgeDB-30-masked dataset.

5.2. Analysis of significant abnormalities
We focus our analysis on results of masked datasets, particularly those marked in red shown in Table 2. There are two abnormal parts we need to address. The first one is the performance of yoloface on masked datasets, on AFDB, AgeDB, LFW and CASIA-WebFace particularly. The second is that ssdface attains an aberrant high accuracy on AFDB and AgeDB contrary to its low performances on other masked datasets.

For the first problem, YOLOv2 itself is a relatively less accurate algorithm whereas has high speed. We propose that such performances result from the weak adaptability of YOLOv2 on some datasets or there are mistakes in the processes of algorithm. To verify our conjecture, we output the aspect ratios of YOLOFace tested on AFDB, AgeDB, LFW and CASIA-WebFace. Figure 1 shows these relevant aspect ratios. The intervals that aspect ratios fall down are relatively normal, indicating it is actually the relatively low precision of YOLOv2.

Concerning the second problem, we propose that SSDFace may detect some irrelevant region as facial region or detect facial landmark falsely, which result in inflation of the precision. To verify our conjecture, we draw the ground box of results of two models on AFDB-masked and AgeDB-30 masked datasets, finding that there is truly problem with detection process of SSDFace. In detection, there are some ground boxes cover merely one or no facial landmark but SSDFace may make false detection. It is hard to judge whether this image is properly detected. After attempting to take advantage of aspect ratio histogram to judge the correctness of detection like before, we rectify the result at some extent. Figure 2 shows the histogram of ratios of AFDB and AgeDB datasets results. We qualify the referential ratio of 0.6-1.0 on AFDB and 0.7-1.1 on AgeDB to rectify the results and generate new ones of 78.73% on AFDB (90.92% originally) and 87.64% n AgeDB (94.48% originally).

Furthermore, drawing support from brilliant LFFDFace algorithm, we utilize LFFDFace to re-examine the outputted ground boxes of SSDFace to check their validity. The results show a 30.84% accuracy on ground boxes of results of SSDFace on AFDB dataset and 21.06% on AgeDB-30 dataset. Given such low precisions, SSDFace truly has a virtually high accuracy comparing with other models, despite some lost detection of LFFDFace may pull down the accuracy.

| CASIA-WebFace | 49.62 | 82.81 | 81.14 | 83.15 | 92.12 |
| Real-World-Masked-Face-Detection | 84.09 | 70.52 | 69.00 | 90.35 | 96.38 |
| whn-masked | 95.95 | 95.71 | 97.74 | 99.32 | 99.91 |
5.3. Improvements
We replace YOLOv2 with YOLOv3 to check if there is any improvement of accuracy. Still on datasets AFDB, AgeDB-30, LFW and CASIA-WebFace, we keep original class labels of YOLOv3 to see various labels of the results. The results are listed in Table 3.

Table 3. Comparison of results of YOLOv2 and YOLOv3 (in percentage)

| Dataset                  | Model     | AFDB-masked | AgeDB-30-masked | LFW-masked | CASIA-WebFace |
|--------------------------|-----------|-------------|-----------------|------------|---------------|
| AFDB                     | YOLOv2    | 41.81       | 49.15           | 61.17      | 49.62         |
| AFDB                     | YOLOv3    | 65.96       | 53.15           | 32.55      | 38.93         |

The results indicate that YOLOv3 does achieve a significant improvement over YOLOv2 on AFDB and AgeDB-30. However, on datasets LFW and CASIA, the performance decreases greatly. A reasonable explanation is that the YOLOv3 model have less competence on simulated masked images, otherwise the accuracy on AgeDB-30, LFW and CASIA dataset will be higher. Meanwhile, all of these datasets have large-scale face region, which for YOLOv3 thwart its remarkable advantage on tiny scale object detection.

However, there are confusing results that some of detected images are falsely classified as other objects like cat, dog, etc. Moreover, facial landmarks are important for detection since images with sunglass and those profile images are rarely detected at all.

Generally, YOLOv3 should generate much more accurate results, being limited by simulated datasets and large-scale image size. In futurity, if there are more real-world masked dataset, this problem can be well verified.

6. Conclusion
In general, we test YOLOv2, SSDFace, LFFDFace, RetinaFace and CenterFace models on several datasets, namely, AFDB, AgeDB-30, LFW, CASIA-WebFace, RMFD and whn. Then we acquire a series of results, by evaluating what we find that algorithm like LFFDFace, RetinaFace and CenterFace achieve remarkable performance whereas there are also problems with YOLOFace and SSDFace. By analyzing aspect ratios of results from YOLOFace and SSDFace, we discover YOLOv2 actually has limited adaptability dealing with masked datasets. Moreover, SSDFace is a far less suitable model to be competent in masked face detection. Beyond YOLOv2, we further utilize YOLOv3 and YOLOv4 to test previous datasets additionally. We get unexpected results, which may result from simulated masked dataset but not real masked dataset and large-scale facial images, as conclusion of analysis afterward.

There are still several drawbacks in this work. We have limited time to formulate our own datasets, and lack of sufficient resources to train gigantic models to make the work more complete. So we have...
to choose those well-designed, more official datasets and utilize well trained networks directly to make assessment. In future work, we expect that we can design our own network and enable which to be well trained to propose a more comprehensive work.

References
[1] Dewantara, B. S. B., & Rhamadhaningrum, D. T. (2020). Detecting multi-pose masked face using adaptive boosting and cascade classifier. In: 2020 International Electronics Symposium (IES). Online. pp. 436-441.
[2] Boutros, F., Damer, N., Kolf, J. N., Raja, K., Kirchbuchner, F., & Ramachandra, R., et al. (2021). Mfr 2021: masked face recognition competition.
[3] Mishra, S., Majumdar, P., Singh, R., & Vatsa, M. (2021). Indian Masked Faces in the WildDataset. arXiv preprint arXiv:2106.09670.
[4] Wang, Z., Wang, G., Huang, B., Xiong, Z., Hong, Q., Wu, H., ... & Liang, J. (2020). Masked face recognition dataset and application. arXiv preprint arXiv:2003.09093.
[5] Cabani, A., Hammoudi, K., Benhabiles, H., & Melkemi, M. (2021). MaskedFace-Net–A dataset of correctly/incorrectly masked face images in the context of COVID-19. Smart Health, 19, 100144.
[6] Jiang W, Zhang L. Edge-siamnet and edge-triplenet: New deep learning models for handwritten numeral recognition[J]. IEICE Transactions on Information and Systems, 2020, 103(3): 720-723.
[7] Chowdary, G. J., Punn, N. S., Sonbhadora, S. K., & Agarwal, S. (2020). Face mask detection using transfer learning of inceptionv3. In: International Conference on Big Data Analytics. Sonnepat, India. (pp. 81-90).
[8] Monterro, D., Nieto, M., Leskovsky, P., & Aginako, N. (2021). Boosting Masked Face Recognition with Multi-Task ArcFace. arXiv preprint arXiv:2104.09874.
[9] Negi, A., Kumar, K., Chauhan, P., & Rajput, R. S. (2021, February). Deep Neural Architecture for Face mask Detection on Simulated Masked Face Dataset against Covid-19 Pandemic. In: 2021 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS). Greater Noida, India. (pp. 595-600).
[10] Wang, B., Zhao, Y., & Chen, C. P. (2021). Hybrid Transfer Learning and Broad Learning System for Wearing Mask Detection in the COVID-19 Era. IEEE Transactions on Instrumentation and Measurement, 70: 1-12.
[11] Alguzo, A., Alzu’bi, A., & Albalas, F. (2021, May). Masked Face Detection using Multi-Graph Convolutional Networks. In: 2021 12th International Conference on Information and Communication Systems (ICICS). Valencia, Spain. (pp. 385-391).
[12] Wang, Z., Wang, G., Huang, B., Xiong, Z., Hong, Q., Wu, H., ... & Liang, J. (2020). Masked face recognition dataset and application. arXiv preprint arXiv:2003.09093.
[13] Moschoglou, S., Papaioannou, A., Sagonas, C., Deng, J., Kotsia, I., & Zafeiriou, S. (2017). Agedb: the first manually collected, in-the-wild age database. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (pp. 51-59).
[14] Yi, D., Lei, Z., Liao, S., & Li, S. Z. (2014). Learning face representation from scratch. arXiv preprint arXiv:1411.7923.
[15] Huang, G. B., Mattar, M., Berg, T., & Learned-Miller, E. (2008, October). Labeled faces in the wild: A database for studying face recognition in unconstrained environments. In Workshop on faces in ‘Real-Life’Images: detection, alignment, and recognition.
[16] Redmon, J., & Farhadi, A. (2017). YOLO9000: better, faster, stronger. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 7263-7271).
[17] Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 779-788).
[18] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with
region proposal networks. arXiv preprint arXiv:1506.01497, 2015.
[19] Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., & Berg, A. C. (2016, October). Ssd: Single shot multibox detector. In European conference on computer vision (pp. 21-37). Springer, Cham.
[20] 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, 28, 91-99.
[21] Deng, J., Guo, J., Zhou, Y., Yu, J., Kotsia, I., & Zafeiriou, S. (2019). Retinaface: Single-stage dense face localisation in the wild. arXiv preprint arXiv:1905.00641.
[22] Najibi, M., Samangouei, P., Chellappa, R., & Davis, L. S. (2017). Ssh: Single stage headless face detector. In Proceedings of the IEEE international conference on computer vision (pp. 4875-4884).
[23] Zhang, K., Zhang, Z., Li, Z., & Qiao, Y. (2016). Joint face detection and alignment using multitask cascaded convolutional networks. IEEE Signal Processing Letters, 23(10), 1499-1503.
[24] He, Y., Xu, D., Wu, L., Jian, M., Xiang, S., & Pan, C. (2019). LFFD: A light and fast face detector for edge devices. arXiv preprint arXiv:1904.10633.
[25] Xu, Y., Yan, W., Yang, G., Luo, J., Li, T., & He, J. (2020). CenterFace: joint face detection and alignment using face as point. Scientific Programming, 2020.
[26] Redmon, J., & Farhadi, A. (2018). Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767.
[27] T.-Y. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie. Feature pyramid networks for object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 2117–2125, 2017. 2, 3