Optik - International Journal for Light and Electron Optics 239 (2021) 166744

Contents lists available at ScienceDirect

Optik

journal homepage: www.elsevier.com/locate/ijleo

Original research article

Scaling up face masks detection with YOLO on a novel dataset

Akhil Kumar [a], Arvind Kalia [a], Kinshuk Verma [b], Akashdeep Sharma [b,*], Manisha Kaushal [c]

[a] Department of Computer Science, Himachal Pradesh University, Shimla, India
[b] CSE, UIET, Panjab University, Chandigarh, India
[c] CSED, Thapar Institute of Engineering & Technology, Patiala, Derabassi Campus, India

ARTICLE INFO

Keywords:
Face masks detection
Deep learning
YOLO
Face masks dataset

ABSTRACT

Face mask detection is a challenging research problem of computer vision because of the small sized area of face mask. The unavailability of proper datasets makes this problem even harder to crack. To address this bottleneck, we propose a novel face masks detection dataset consisting of 52,635 images with more than 50,000 tight bounding boxes and annotations for four different class labels namely, with masks, without masks, masks incorrectly, and mask area, which makes it a novel contribution for variety of face masks classification and detection tasks. Further, this dataset is tested with eight variants of the YOLO algorithm to determine its effectiveness. On the proposed dataset, original YOLO v4 achieved a mAP value of 71.69 % which was highest among all the original YOLO variants, tiny YOLO v4 achieved a mAP value of 57.71 % which was highest among all tiny variants. To propose new face masks detection algorithms that can perform with high accuracy in a limited computational resources environment, we selected four tiny variants of the YOLO algorithm and proposed new architectures modifications in their feature extraction networks that increased the overall performance and specifically, improved mAP by 4.12 % for tiny YOLO v3 and 2.54 % for tiny YOLO v4.

1. Introduction

More crimes are committed by criminals by wearing face masks and hyper-realistic face masks [1]. Amid the COVID-19 pandemic in the last one year people across the globe are wearing face masks and earlier also people used to wear face masks to protect themselves from pollution and other communicable diseases. Due to the presence of face masks on the face area, in places such as ATMs, banks, airport security checks, facial-biometric attendance systems it has become a difficult task for face recognition systems to identify the person behind the mask. Face recognition systems work by identifying the facial features such as size and shape of eyes, nose, cheekbones, and jaw which tend to hide in an environment where all the people are wearing face masks. Present face recognition systems are trained on such facial features and their effectiveness is degraded due to the presence of the face masks on the face area which hides most of the crucial features to perform recognition tasks. To address this bottleneck and develop effective face masks recognition systems, in the current scenario we need face detectors that can detect the identity of people wearing face masks and specifically the presence of face masks on the face area which is a very small object to detect and a challenge in object detection. The

* Corresponding author at: CSE, UIET, Panjab University, Chandigarh, India.
E-mail addresses: akhil.hpucs@gmail.com (A. Kumar), arvkalia@gmail.com (A. Kalia), chd.kinshuk@gmail.com (K. Verma), akashdeep@pu.ac.in (A. Sharma), manisha.kaushal@thapar.edu (M. Kaushal).

https://doi.org/10.1016/j.ijleo.2021.166744
Received 22 January 2021; Received in revised form 1 March 2021; Accepted 10 March 2021
Available online 18 March 2021

first step to achieve this objective is to develop computer vision algorithms that can detect the presence and non-presence of face masks on the face area and further extended by developing face recognition systems that can recognize the leftover features of the face outside the area covered by the face mask. To address this challenge, as a first step towards developing face recognition systems that can recognize the identity of people behind face masks we propose a face masks detection dataset that is distinct and divergent from existing datasets. The proposed dataset consists of 52,635 images of people wearing face masks, people not wearing face masks, people wearing face masks incorrectly, and specifically, mask area in images where a face mask is present. The dataset is richly annotated for each class label with more than 50,000 tight bounding boxes. To determine the effectuality of the proposed dataset we employed four original variants of the YOLO algorithm and four variants of the tiny YOLO algorithm and trained and tested them on the proposed dataset. Furthermore, to propose improvements in tiny variants of the YOLO algorithm we selected all four tiny variants of the YOLO algorithm and embodied modifications in their feature extraction network. The proposed variants of tiny YOLO algorithms were trained and tested on the proposed and a benchmark dataset to determine their effectiveness and performance improvement. The proposed work mainly focuses on face mask detection that can be further extended to face recognition. We selected series of YOLO [2–5] object detection algorithm due to its fascinating results in areas such as generalized object detection [6], license plate detection for non-helmeted motorcyclist [7], six-digit car license plate recognition [8], real-time vehicle detection [9], fish detection and tracking in fish farms [10], pedestrian detection [11], molting detection in swimming crabs [12], road marking [13], flower detection [14] and vehicle detection [15], and YOLO v4 being a current state-of-the-art in object detection.

The major contributions of our approach are as follows:

(1) Proposal of a novel face masks detection dataset that is distinct and divergent from the existing dataset. The proposed dataset consists of 52,635 images that are richly annotated with 50,000 tight bounding boxes across class labels with masks, without masks, masks incorrectly and mask area. The proposed dataset is analysed on eight variants of the YOLO algorithm.
(2) New feature extraction networks for four variants of tiny YOLO algorithm are proposed to increase their performance.

The work is novel for the aspect of proposing and creating a dataset for face masks detection. The novelty of this work also lies in exhaustive experiments carried out to perform face masks detection on a large dataset containing images with varying contrast and classes. Furthermore, modifications are proposed in tiny variants of the YOLO algorithm that are capable of integrating with face recognition systems to perform real-time face masks detection. The proposed work has high implication for face masks detection in areas where complete visibility of the face area is a guideline.

This paper is composed of the following sections: Section 2 presents the related work in relevant fields; Section 3 describes the proposed dataset created for solving face masks detection problem with details and justification; Section 4 set-out the evaluation approach and propose new variants of tiny YOLO algorithm with detailed graphs and discussions. Section 5 presents the conclusion and future possibilities for the proposed work.

2. Related work

Detecting people wearing face masks in images and videos is a cornered problem and yet to be fully exploited to achieve benchmark results. Amid COVID-19 pandemic outbreak, in the last one year several research works have been published [16–18] for face masks classification and detection which has very much solved the problem of availability of datasets to carry out work in this cornered area. However, the bottleneck of scarcity of largely annotated datasets providing rich features for new and distinct object categories still exists.

Existing published work in this domain of face recognition explores possibilities in two areas: (1) face masks classification and (2) face masks detection. In Li et al. [19], the authors proposed a HGL method for head pose classification with mask using color texture analysis of pictures and line portraits. It achieved a frontal accuracy of 93.64 % and side accuracy of 87.17 % for recognition between wearing a face mask and not wearing a face mask. Khandelwal et al. [20] proposed a deep learning model that binarizes an image accordingly a mask is used or not used. They used MobileNet v2 model to train 380 images having a mask and 460 images having no mask. This model achieved AUROC of 97.6 %. This model finds it difficult to classify partial hidden faces. In Qin et al. [21], the authors proposed a face mask-wearing classification system by embedding image super-resolution using classification networks. The proposed model quantifies mask, no mask and incorrectly worn mask in 2D facial pictures. In Loey et al. [22], a hybrid face mask classification approach was presented combining ResNet-50 with machine learning classifiers. In the proposed work ResNet-50 was used as a feature extractor and classification of face mask was performed using decision trees, support vector machines and ensemble classifiers. The proposed hybrid classification approach achieved accuracy of 99.6 % with support vector machine classifier on RMFD dataset. In Ud Din et al. [23], the authors proposed a Generative Adversarial Network (GAN) based system for automatic removal of masks covering the face area. The proposed model has the ability to regenerate the image by building the missing hole and produce a realistic and natural image of a complete face. In the work of Ejaz et al. [24], a method using PCA was proposed to recognize the person under masked and unmasked conditions.

In Jiang et al. [16], the authors proposed a face mask detector with the name RetinaFaceMask. It presented a highly accurate and efficient face mask detector. The proposed algorithm was a one stage detector that uses a feature pyramid network to integrate high-level semantic information with multiple feature maps. The proposed algorithm also fuses a novel context attention module to focus on detecting face masks. In addition to this, in this algorithm a novel cross-class object removal algorithm was proposed that rejects prediction with low confidences and the high intersection of union. The results of this algorithm show that on a public face mask dataset it achieved precision higher than 2.3 % and 1.5 % as compared to baseline result and recall higher than 11.0 % and 5.9 % for

Fig. 1. Dataset classes with bounding boxes.

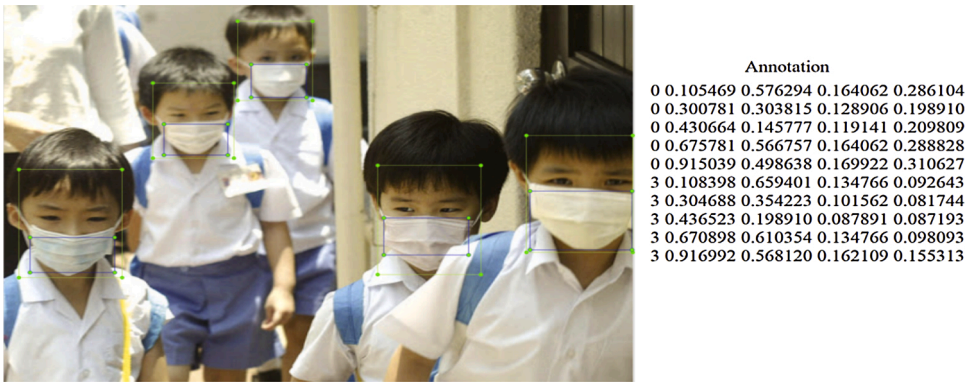


Fig. 2. Screenshot of the tight bounding boxes with annotations.

baseline result. In Inamdar et al. [17], a real time face mask detection method was proposed. This deep learning based network classifies three classes namely, a person wearing a mask, improperly worn mask and no mask detected. This method was trained and tested on a very small dataset consisting 10 images for wearing masks, 15 images for improperly worn masks and 10 images were used for no mask. The proposed approach achieved an accuracy of 98.6 %. In Roy et al. [18], the authors proposed a dataset with name MOXA that contains 3000 images for people with masks and people without masks and tested the dataset with YOLO v3, tiny YOLO v3, SSD and Faster R-CNN. On the proposed dataset, YOLO v3 achieved a mAP of 63.99 %, tiny YOLO v3 achieved a mAP of 41.06 %, SSD achieved a mAP of 46.52 % and Faster R-CNN achieved a mAP of 60.05 %.

From the literature, this can be concluded that there are very few works published in this domain and there is a dire necessity of large and reliable face masks detection datasets and exploration of more object detection methods with new proposal to scale up the research that can lead to the development of effective face masks detection and face recognition systems.

3. Proposed dataset

Exciting and challenging datasets are a motivation for advancement in the research of computer vision. The ImageNet, MS COCO, and PASCAL VOC datasets all provided a new dimension to the research of multi-class object detection and raised the overall performance evaluation of the object detection methods. Following a similar strategy, our aim in creating face masks detection dataset is to come up with a one-step-ahead benchmark and to help identify people hiding identity behind face masks and thus pivot efforts on this difficult research area. The created dataset has applications in facial identification, facial attendance, facial payment, face access control, and facial security checks.

3.1. Images and ground truths

To create the dataset we used Google [25] and Bing API [26] and crawled 11,000 images from the internet and resized all the images to the size of 416 × 416 pixels, which was also the input size of our evaluation approach employed to gauge the validity of the dataset. To extract distinct and rich information for each class label approximately 50,000 tight bounding boxes (BB) were drawn across 11,000 unique images of people in different conditions. For such a large labeling effort we used the Labelimg annotation tool [27]. The classes of the created dataset with bounding boxes are illustrated in Fig. 1. We followed the YOLO format to perform annotations and specify our evaluation approach. Fig. 2. shows the overlaid bounding boxes with annotations.

In order to enhance the size of the dataset, we employed the strategy of data augmentation. The operations of rotation, shearing, flipping, and HSV shift were applied to augment the originally collected image samples thus, providing more features to learn by object detectors. Applying data augmentation enhanced our dataset with 52,635 image samples from originally collected 11,000 images, increasing the dataset by approx. 20 %. Fig. 3. illustrates the image samples obtained after applying data augmentation.

Fig. 3. Augmented image samples.

Table 1
Comparison of proposed dataset with existing datasets.

Dataset	No. of images	Annotated	With masks	Without masks	Wearing masks incorrectly	Mask area	Suitable for face masks detection
MFDD [28]	With mask: 24,771	No	Yes	No	No	No	No
RMFRD [28]	With mask: 5,000 Without mask: 90,000	No	Yes	Yes	No	No	No
SMFRD [28]	With mask: 500,000	No	Yes	No	No	No	No
FMD [29]	For both the classes: 7,959	Yes	Yes	Yes	No	No	Limited class detection
MAFA [30]	With mask: 35,086	Yes	Yes	No	No	No	Limited class detection
MOXA [18]	For both the classes: 3,000	Yes	Yes	Yes	No	No	Limited class detection
Proposed dataset	For all four classes: 52,635	Yes	Yes	Yes	Yes	Yes	All possible face masks class detections

We split the dataset into training, testing, and validation data and specify our evaluation approach. This will allow different deep learning based state-of-the-art object detection algorithms to produce challenging and comparative results. Our dataset consists of four classes namely, with masks, without masks, masks incorrectly and mask area. To train and test, we split the dataset in a ratio of 80:10:10 respectively for the training set with 42,115 images, test set with 5,260 images, and validation set with 5,260 images. The dataset is available on request at Link

3.2. Comparison with existing datasets

Existing datasets have largely missed the bounding boxes and annotations for distinct class labels required for effective face masks detection. The datasets available are mostly useful for face masks classification where bounding boxes and annotations are not required. We effort to solve this bottleneck by creating a richly annotated dataset that can help to progress face masks detection research. This subsection presents an overview of existing datasets and a comparison with our dataset.

Masked Face Detection Dataset (MFDD) [28], one of the state-of-the-art dataset available in the literature is based on images crawled from the internet having a wide range of images for persons with masks. This dataset contains 24,771 masked face images. It misses class face without mask and face wearing mask incorrectly. Real-world Masked Face Recognition Dataset (RMFRD) [28] is based on images crawled from the internet covering classes face with mask with 5,000 images and face without mask with 90,000 images. This dataset misses class faces wearing masks incorrectly and mask area. Face Mask Dataset (FMD) [29] used in RetinaFaceMask is a collection of 7,959 images richly annotated with either face with a mask and face without a mask. This dataset does not contain the images for class face wearing mask incorrectly and mask area. MAskedFAces (MAFA) [30] dataset is based on images collected from the internet. This dataset is a collection of 35,806 images annotated for class label masked faces. This dataset misses class face without mask and face wearing mask incorrectly. MOXA [18] dataset has 3,000 images of people wearing medical face masks that were collected from the internet and publicly available datasets. It covers only two classes with mask and without mask.

As shown in Table 1, compared to existing datasets, the dataset created by us is far different that includes annotation for people wearing face masks incorrectly and specifically mask area in images having people wearing face masks correctly and incorrectly. This can help face mask detection and face recognition systems identifying the presence of face masks on any area of the face preventing recognition errors in the identification of a face due to hindrance of essential facial features.

4. Evaluation of proposed dataset

To evaluate the effectiveness of the proposed face masks detection dataset we employed eight variants of YOLO algorithm due to its advantage of generalization and fast detection, and YOLO v4 being the current state-of-the-art in the area of object detection. We re-implemented all eight variants of YOLO algorithm in-house using open-source deep learning libraries such as Tensorflow and Keras on an intel i5 based system with 8 GB of RAM and NVIDIA 1050i GPU.

To obtain anchor boxes on our face masks detection dataset we adopted the IoU mechanism [31,32] and used k-means++ clustering method to compare the IOU scores with different k values. We set the k value of 6 and 9 considering the complexity of the network. The evaluation results were obtained on the baseline of average precision for each class of the dataset and performance metrics such as precision, recall, f-1 score, and mAP achieved by the variant of YOLO algorithm on the overall dataset. The description of performance metrics employed to evaluate variants of YOLO and measure effectuality of the proposed dataset is shown in Eqs. 1–3.

Table 2
Performance of variants of YOLO on proposed dataset.

YOLO Variant	Precision	Recall	F-1 Score	mAP
YOLO v1	63.2%	54.7 %	61.3 %	52.4 %
YOLO v2	68%	59 %	63 %	55.34 %
YOLO v3	81%	73 %	76 %	65.84 %
YOLO v4	78 %	79 %	78 %	71.69 %
tiny YOLO v1	29%	42 %	34 %	30.75 %
tiny YOLO v2	33%	49 %	39 %	33.78 %
tiny YOLO v3	74%	56 %	63 %	49.03 %
tiny YOLO v4	79 %	65 %	72 %	57.71 %

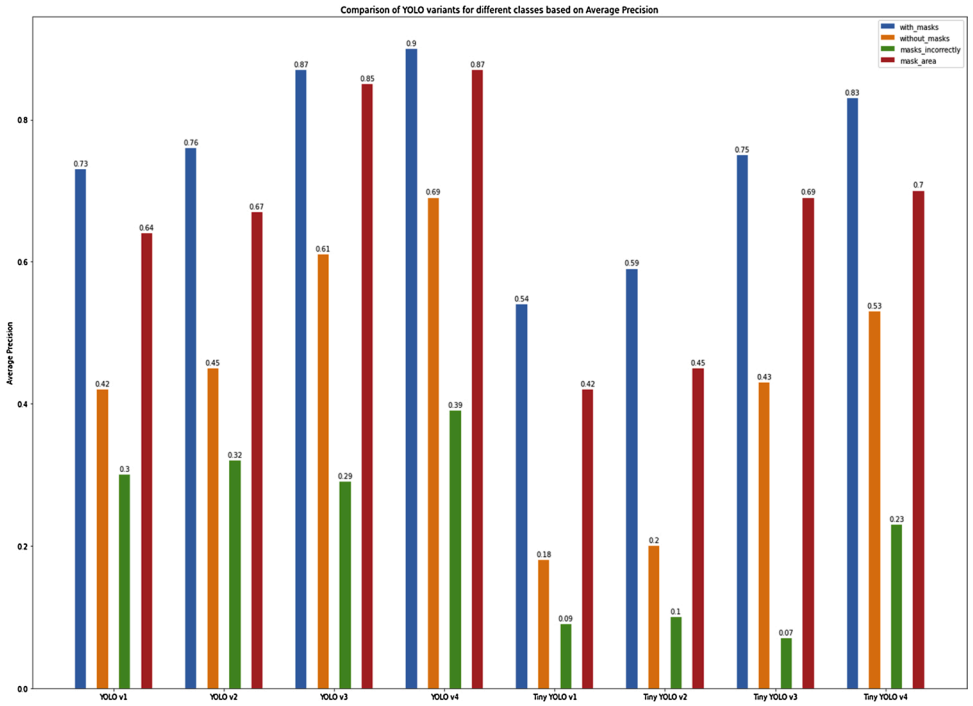


Fig. 4. Comparison of YOLO variants on the basis of average precision.

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

$$F - 1\ Score = 2 * \frac{(Precision * Recall)}{(Precision + Recall)} \tag{3}$$

In Eqs. (1–3), True positive (TP) is the number of positive instances that are correctly predicted; false negative (FN) is the number of positive instances that are incorrectly predicted. False positive (FP) is the number of negative instances that are incorrectly predicted. The metric used average precision (AP) is precision averaged across all values of recall between 0 and 1 whereas, mean average precision (mAP) is the mean value of average precisions which quantifies how good the model is performing a query.

4.1. Evaluation results

To specify our evaluation approach, for all employed variants of the YOLO algorithm we set the input to size 416 × 416 and the learning rate to 0.001, momentum to 0.9 with the decay of 0.0005. The algorithm was trained for 8000 iterations with a batch of size 64. The detailed evaluation results of each variant of YOLO algorithm on the basis of precision, recall, f-1 score, and mAP is presented in Table 2.

The detailed comparison of employed YOLO variants on the basis of average precision for each class of the proposed dataset is presented in Fig. 4.

The results obtained by employing variants of YOLO on the proposed dataset can be summarized as follows:

i YOLO v4 being the current state of the art achieved the highest mAP value of 71.69 % among all the employed variants of YOLO. The result indicates high detection accuracy and the proposed dataset suitable for face masks detection.
ii Among tiny YOLO variants, tiny YOLO v4 achieved a mAP value of 57.71 % which was highest among all the employed tiny YOLO variants.
iii Specifically for the face mask area which is a very small object to detect, YOLO v4 achieved the highest average precision with a value of 87.05 % thus making it suitable for the detection of small objects. Furthermore, based on the performance of YOLO v4 this can be deduced that the proposed dataset is suitable for the detection of the presence of face mask on the face area which is a very small object to detect and require correct annotation and further feature extraction by the object detector.
iv Among tiny YOLO variants, tiny YOLO v4 achieved the highest average precision for face masks with a value of 70.37 % that makes it capable of detecting face masks accurately.
v The performance of tiny YOLO v1, tiny YOLO v2, and tiny YOLO v3 was not up to the mark therefore, improvements are required to be proposed to enhance their performance.
vi The performance of YOLO v4 and tiny YOLO v4 on the proposed dataset has shown fascinating results therefore, this can be concluded that the proposed dataset is effective for face masks detection with these variants of YOLO.

The tiny variants of the YOLO have a smaller feature extraction network and an advantage of training in lesser time with fewer computation resources requirements as compared to their original counterparts. Considering the limitation of computation resources availability and unsatisfactory performance of tiny variants of YOLO on the proposed dataset, in the next subsection we effort to propose new variants of tiny YOLO that leads to scaling up their performance for face masks detection.

4.2. Proposed tiny YOLO variants

The entire working of the YOLO algorithm relies on a single-stage object detection mechanism. In the backend, it is constituted of a feature extraction network that extracts rich information from the input. The feature extraction network of the YOLO algorithm is based on convolutional neural networks. To enhance the performance of tiny YOLO variants and further improve the detection accuracy we tweaked and modified the feature extraction networks of the four tiny YOLO variants that improved the overall detection accuracy on the proposed dataset. The detailed modifications and proposed architectures of proposed variants of tiny YOLO are discussed below. The training parameters, system requirements, and evaluation criteria followed to implement and evaluate proposed tiny variants are the same as followed for implementation and evaluation of original YOLO and tiny YOLO as discussed under section heading 4.

4.2.1. M-T YOLO v1

To increase the performance of tiny YOLO v1 we embodied a few changes in the network architecture by adding more convolution layers to the original network. Original tiny YOLO v1 was made up of eight convolution layers and six maxpool layers. We added convolution layers of size 1×1 with 64 filters and 3×3 128 filters between the fourth and fifth convolution layer of the network and convolution layers of size 1×1 with 128 filters and 3×3 with 256 filters between the fifth and sixth convolution layer of the original network and layers of size 1×1 256 filters and 3×3 512 filters between the sixth and seventh convolution layer of the original network. This modification in the original network increased the mAP of tiny YOLO v1 by 1.25 %.

4.2.1.1. Average precision. For class with masks it achieved an AP of 56 %, for class without masks it achieved an AP of 19 %, for class masks incorrectly the achieved AP was 10 % and for class mask area it achieved an AP of 43 %.

4.2.1.2. Performance. On the overall dataset modified tiny YOLO v1 achieved a precision of 30 %, recall of 43 %, f-1 score of 35 %, and mean average precision (mAP) of 32 %.

4.2.2. M-T YOLO v2

We modified the original network architecture of tiny YOLO v2 by adding a few more convolution layers between the convolution layers of the original network. The original architecture of tiny YOLO v2 was made up of nine convolutional layers and six maxpool layers. We added convolution layers of size 1×1 64 filters and 3×3 128 filters between the fourth and fifth convolution layer of the original network; 1×1 128 filters and 1×1 256 filters between the fifth and sixth convolution layer of the original network; 1×1 256 filters and 3×3 512 filters between the sixth and seventh convolution layer of the original network; and 1×1 512 filters and 3×3 1024 filters between the seventh and eighth convolution layer of the original network. The embodied layers improved the mAP of tiny YOLO v2 by 2%.

Average precision. For class with masks it achieved an AP of 61.53 %, for class without masks it achieved an AP of 23.19 %, for class masks incorrectly the achieved AP was 11.37 % and for class mask area it achieved an AP of 47.08 %.

Performance. On the overall dataset modified tiny YOLO v2 achieved a precision of 35 %, recall of 50 %, f-1 score of 41 %, and mean average precision (mAP) of 35.80 %.

4.2.3. M-T YOLO v3

To increase the performance of tiny YOLO v3 we altered its original network architecture by adding more convolution layers

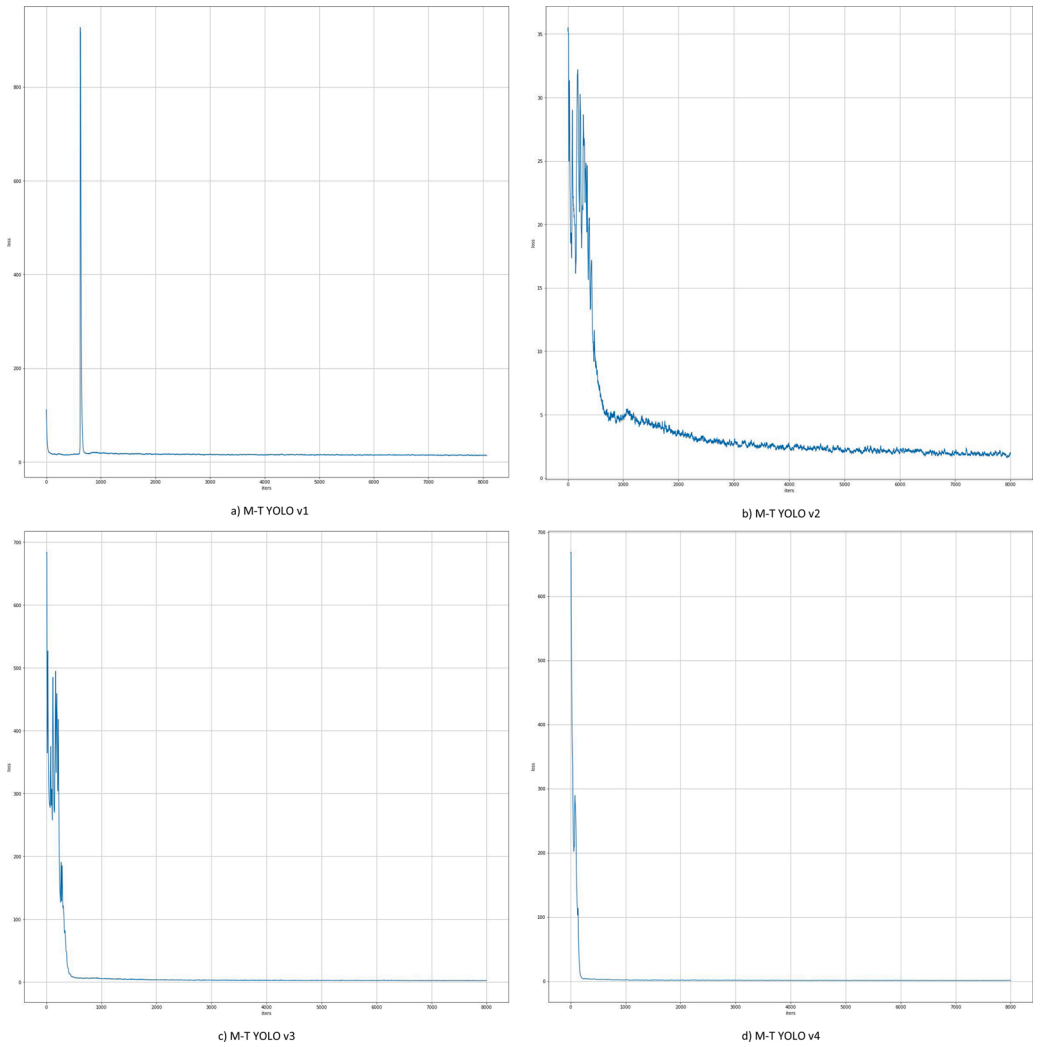


Fig. 5. Training loss plots for proposed tiny YOLO variants.

between the existing convolution layers. The original network of tiny YOLO v3 was made of thirteen convolution layers and six maxpool layers. We added convolution layers of size 1×1 64 filters and 3×3 128 filters between fourth and fifth convolution layer of the original network; 1×1 128 filters and 3×3 256 filters between fifth and sixth convolution layer of the original network; 1×1 256 filters and 3×3 512 filters between sixth and seventh convolution layer of the original network; and 1×1 512 filters and 3×3 1024 filters between seventh and eighth convolution layer of the original network. The proposed modification increased the mAP of tiny YOLO v3 by 4.12 %.

4.2.3.1. this sub numbering is not required Average precision. For class with masks it achieved an AP of 82.4 %, for class without masks it achieved an AP of 47.3 %, for class masks incorrectly the achieved AP was 17.6 % and for class mask area it achieved an AP of 76.5 %.

4.2.3.2. this sub numbering is not required Performance. On the overall dataset modified tiny YOLO v3 achieved a precision of 77 %, recall of 67 %, f-1 score of 69 %, and mean average precision (mAP) of 53.15 %.

4.2.4. M-T YOLO v4

To increase the performance of tiny YOLO v4 we altered its original network architecture by adding more convolution layers between the existing convolution layers. The original network of tiny YOLO v4 was made of twenty-two convolution layers and three maxpool layers. We added convolution layers of size 3×3 128 filters between sixth and seventh convolution layer of the original network; 3×3 128 filters between seventh and eighth convolution layer of the original network; 1×1 32 filters between ninth and tenth convolution layer of the original network; 3×3 256 filters convolution layers before and after the eleventh layer; 1×1 64 filters convolution layer after the thirteenth layer; and 3×3 512 filters layer before and after fifteenth convolution layer of the original network. The proposed modification increased the mAP of tiny YOLO v4 by 2.54 %.

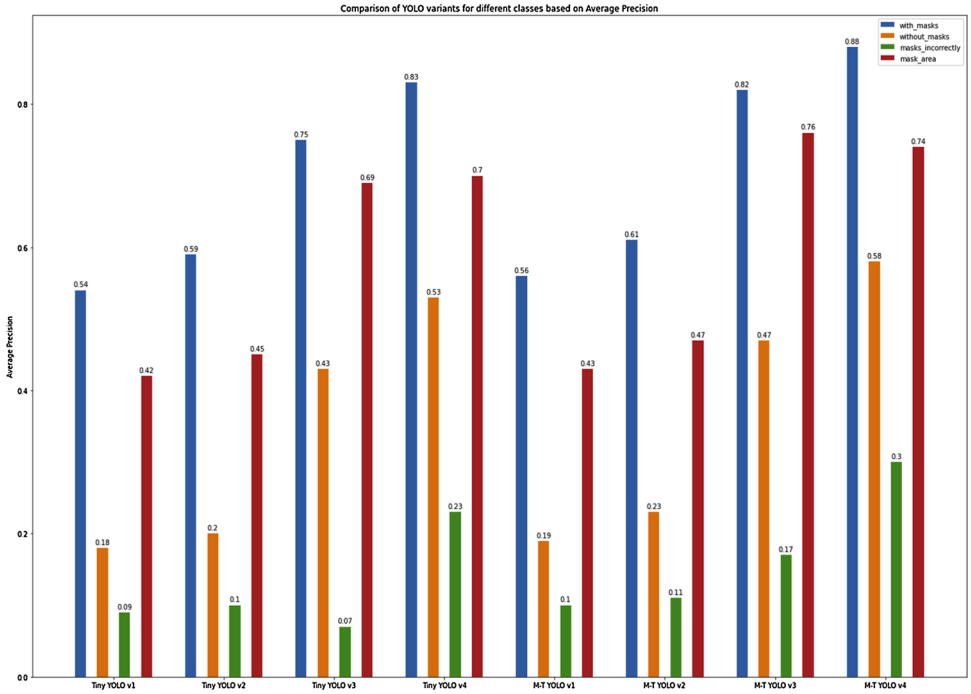


Fig. 6. Comparison of proposed tiny variants with original variants on the basis of AP.

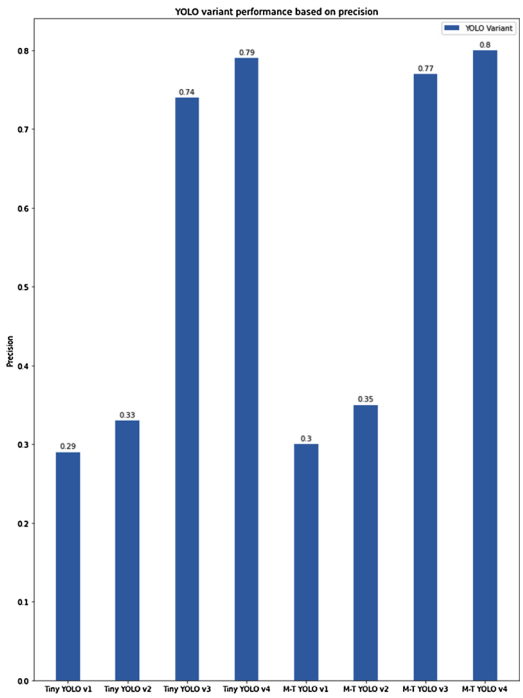


Fig. 7. Comparison based on precision.

4.2.4.1. this sub numbering is not required Average precision. For class with masks it achieved an AP of 88.15 %, for class without masks it achieved an AP of 58.17 %, for class masks incorrectly the achieved AP was 30.18 % and for class mask area it achieved an AP of 74.68 %.

4.2.4.2. Performance. On the overall dataset modified tiny YOLO v4 achieved a precision of 80 %, recall of 69 %, f-1 score of 76 %, and

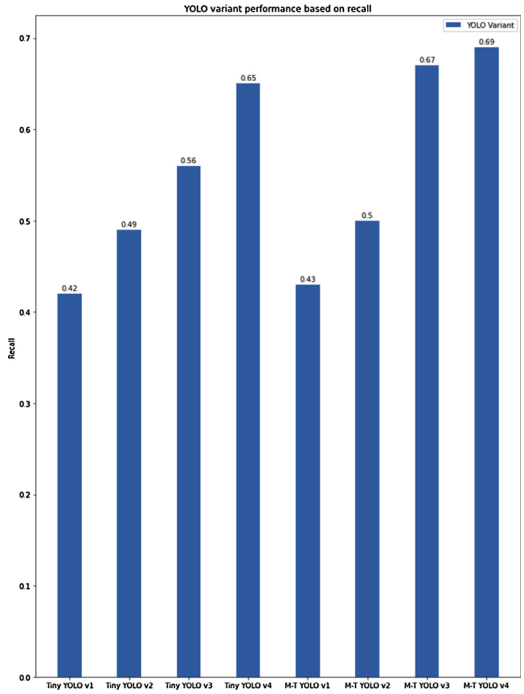


Fig. 8. Comparison based on recall.

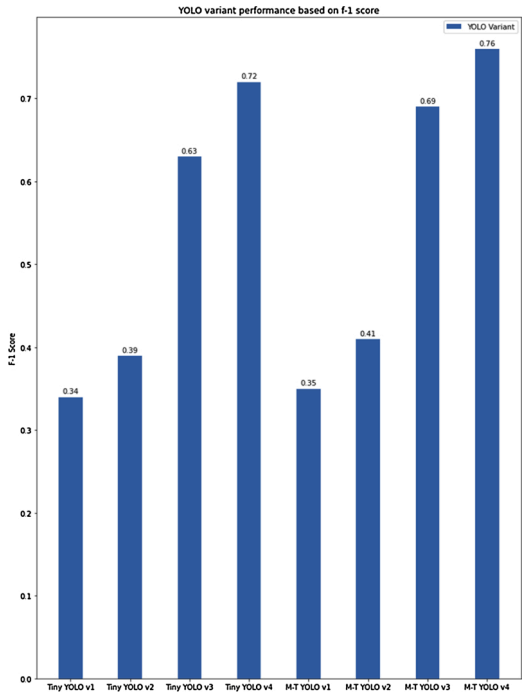


Fig. 9. Comparison based on f-1 score.

mean average precision (mAP) of 60.25 %.

The results obtained with modified tiny variants of YOLO were comparatively higher than their original tiny variant counterparts with improved mean average precision (mAP), precision, recall, and f-1 score for tiny YOLO v1, v2, v3, and v4 thus indicating improved and deeper feature extraction network aid in extracting more features and enabling proposed tiny YOLO variants to achieve a better detection accuracy. Fig. 5 presents the training loss plots for proposed tiny YOLO variants.

The detailed comparison of proposed tiny YOLO variants with their original counterparts on the basis of average precision,

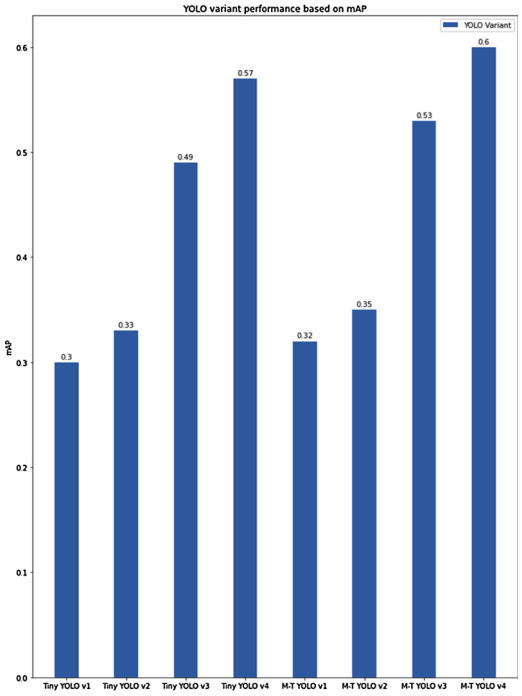


Fig. 10. Comparison based on mAP.

precision, recall, f-1 score and mAP are presented in Figs. 6–10. The detection behaviour of proposed modified YOLO variants is shown in Fig. 11. The detailed feature extraction networks of proposed tiny YOLO variants corresponding to their original counterparts are appended in Fig. A1 in the appendix.

As shown in Fig. 11 all proposed modified tiny YOLO variants detected with masks, mask area, and without masks class correctly but modified tiny YOLO v1 and v2 struggled to detect images where masks were worn incorrectly. They detected masks incorrectly as with masks and mask area as both were present in the images. This indicates modified tiny YOLO v1 and v2 as ineffective detectors for accurate and reliable detections in real-time applications. However, with modified tiny YOLO v3 and v4 detections results were accurate across all the classes of the dataset. The variance in detection results is due to the network architectures of the tiny YOLO variants thus, it can be concluded that with each successor of tiny YOLO a better feature extraction network was used.

The evaluation results of proposed tiny YOLO variants have shown improved mAP and precision for tiny YOLO v1, v2, v3 and v4 by 1–4 % and other performance metrics such as average precision, precision, recall, and f-1 score by 1–11 %. Embodied changes improved performance metrics indicate better detection accuracy with the proposed modified tiny YOLO variants on the proposed face masks detection dataset containing multi-class face masks categories.

4.3. Comparison with similar work

To validate the proposed modified tiny YOLO variants and original YOLO variants employed in this work, we tested our baseline performance and compared it with performance of YOLO variants employed by authors in [18] on MOXA dataset. We tested on test images of the MOXA dataset consisting of 200 images after re-annotating with our dataset class labels namely, with masks, without masks, masks incorrectly and mask area to validate the employed YOLO variants. For testing we used weights trained on face masks detection dataset proposed in this work. The results indicate YOLO v3 and v4, tiny YOLO v3 and v4 and modified tiny YOLO v3 and v4 achieved a higher mAP on MOXA dataset promising these to be suitable for real-time face masks detection.

The comparison results of our baseline approach with existing similar work is presented in Table 3. The values indicate that training on the proposed dataset improved the results of almost all variants of YOLO thereby justifying the importance of the dataset. The modified architectures further deliberated an improvement in results justifying the architectures.

5. Conclusion

This paper has undertaken to solve the face mask detection problem by proposing a new state-of-the-art dataset that is significantly feature rich than available datasets in the problem domain. The proposed dataset consists of 52,635 images with more than 50,000 tight bounding boxes across classes with masks, without masks, masks incorrectly, and mask area that aid in achieving accurate and reliable results outperforming existing state-of-the-art. The dataset has been tested and validated on original and tiny variants of YOLO to perform face masks detection. The results obtained show original YOLO v4 as the accurate object detection algorithm achieving mAP of 71.69 %.

Fig. 11. Detection behaviour of proposed modified tiny YOLO variants.

Table 3
Comparison with existing similar work.

Method	Performance based on mAP on MOXA dataset					
YOLO variants	YOLO v3	YOLO v4	Tiny YOLO v3	Tiny YOLO v4	M-T YOLO v3	M-T YOLO v4
Roy et al. [18]	63.99 %	–	56.27 %	–	–	–
Proposed	**64.**50 %	65.13 %	**57.**78 %	61.70 %	**59.**78 %	63.17 %

Furthermore, we also proposed changes to different tiny variants of YOLO and the new proposed architectures resulted in improved mAP and precision of 1–4 %. With proposed modified tiny YOLO variants highest mAP was achieved with tiny YOLO v3 with value of 53.15 % and tiny YOLO v4 with value of 60.25 % thus, promising these to be capable of performing real-time face masks detection when integrated with handheld devices as these variants required less computation and hardware resources and lesser training time to train on custom dataset. However, the proposed tiny YOLO v1 and v2 struggled in detecting small changes in masks wearing conditions in masks incorrectly scenarios leading to false detection. We recommend proposed modified tiny YOLO v4 as an effective and improved face masks detector in real-world applications due to its optimized and improved feature extraction network. The future work can be extended by combining proposed algorithms with face recognition systems for identification of people behind face masks.

Funding

The research work has been funded by All India Council of Technical Education, India under Research Promotion Scheme wide file no. 8-108/FDC/RPS(POLICY-1/2019-20). The authors therefore thank All India Council of Technical Education, India for the support.

Availability of data and material

The face masks detection dataset proposed and used in this work is self-created and available on request at Link.

Declaration of Competing Interest

The authors report no declarations of interest.

Acknowledgements

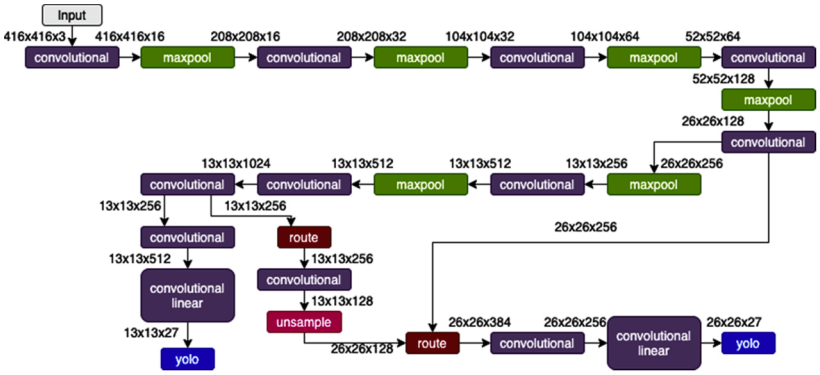
Not applicable.

Appendix 1 Architectures of original tiny and proposed modified tiny YOLO variants

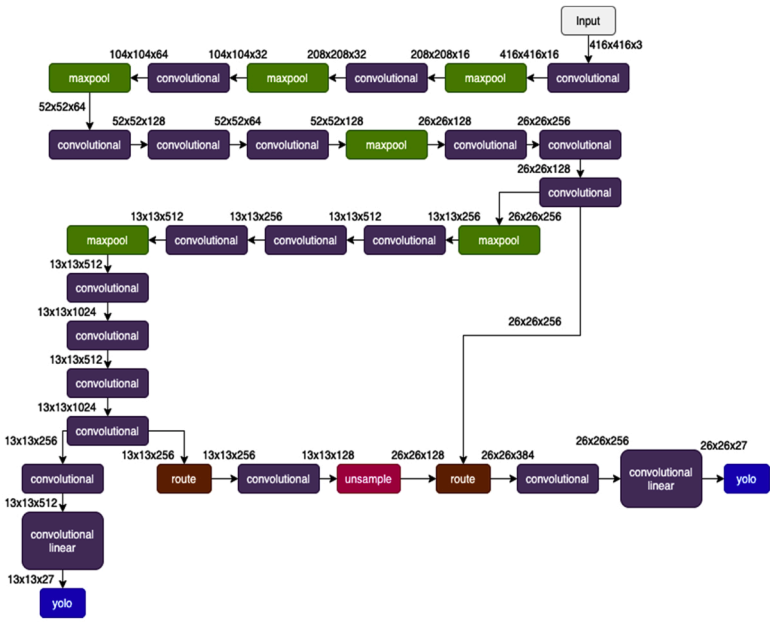
Below are the feature extraction network architectures used for original tiny YOLO and proposed modified tiny YOLO variants (M-T YOLO).

.

Fig. A1. Architectures of original tiny YOLO and proposed modified tiny YOLO variants.

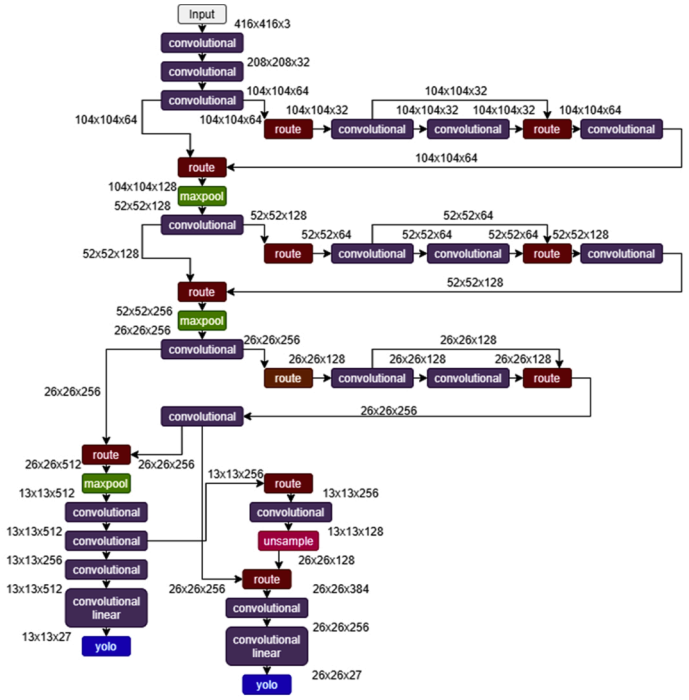


e) Tiny YOLO v3

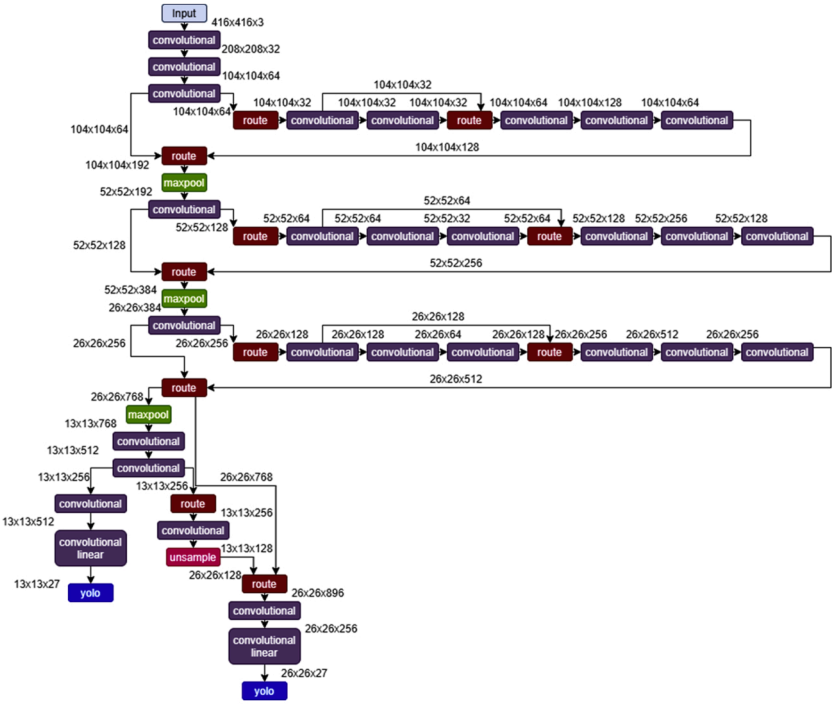


f) M-T YOLO v3

Fig. A1. (*continued*).



g) Tiny YOLO v4



h) M-T YOLO v4

Fig. A1. (*continued*).

References

[1] J.G. Sanders, Y. Ueda, K. Minemoto, E. Noyes, S. Yoshikawa, R. Jenkins, Hyper-realistic face masks: a new challenge in person identification, Cogn. Res. Princ. Implic. 2 (2020), https://doi.org/10.1186/s41235-017-0079-y.

[2] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: unified, real-time object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, https://doi.org/10.1109/CVPR.2016.91.
[3] J. Redmon, A. Farhadi, YOLO9000: better, faster, stronger, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 2017, https://doi.org/10.1109/CVPR.2017.690.
[4] J. Redmon, A. Farhadi, YOLOv3: An Incremental Improvement, arXiv, 2018, https://arxiv.org/abs/1804.02767.
[5] A. Bochkovskiy, C.Y. Wang, H.Y. Lio, YOLOv4: Optimal Speed and Accuracy of Object Detection, arXiv, 2020, https://arxiv.org/abs/2004.10934.
[6] Y. Yin, H. Li, W. Fu, Faster-YOLO: An accurate and faster object detection method, Digit. Signal Process. 102 (2020), https://doi.org/10.1016/j.dsp.2020.102756.
[7] Y. Jamtsho, P. Riyamongkol, R. Waranusast, Real-time license plate detection for non-helmeted motorcyclist using YOLO, Ict Express (2020), https://doi.org/10.1016/j.icte.2020.07.008.
[8] Hendry, R.C. Chen, Automatic license plate recognition via sliding-window darknet-YOLO deep learning, Image Vis. Comput. 87 (2019) 47–56, https://doi.org/10.1016/j.imavis.2019.04.007.
[9] S. Jamiya, E. Rani, LittleYOLO-SPP: A delicate real-time vehicle detection algorithm, Optik 255 (2021), https://doi.org/10.1016/j.ijleo.2020.165818.
[10] H.E. Mohamed, A. Fadl, O. Anas, Y. Wageeh, N. ElMasry, A. Nabil, A. Atia, MSR-YOLO: method to enhance fish detection and tracking in fish farms, Procedia Comput. Sci. 170 (2020) 539–546, https://doi.org/10.1016/j.procs.2020.03.123.
[11] Z. Yi, S. Yongliang, Z. Jun, An improved tiny-yolov3 pedestrian detection algorithm, Optik 183 (2019) 17–23, https://doi.org/10.1016/j.ijleo.2019.02.038.
[12] C. Tang, G. Zhang, H. Hu, P. Wei, Z. Duan, Y. Qian, An improved YOLOv3 algorithm to detect molting in swimming crabs against a complex background, Aquac. Eng. 91 (2020), https://doi.org/10.1016/j.aquaeng.2020.102115.
[13] X.Y. Ye, D.S. Hong, H.H. Chen, P.Y. Hsiao, L.C. Fu, A two-stage real-time YOLOv2-based road marking detector with lightweight spatial transformation invariant classification, Image Vis. Comput. 102 (2020), https://doi.org/10.1016/j.imavis.2020.103978.
[14] Z. Cheng, F. Zhang, Flower end-to-end detection based on YOLOv4 using a mobile device, Wirel. Commun. Mob. Comput. (2020), https://doi.org/10.1155/2020/887064.
[15] P. Mahto, P. Garg, P. Seth, J. Panda, Refining Yolov4 for vehicle detection, Int. J. Adv. Res. Eng. Technol. 11 (5) (2020) 409–419, https://doi.org/10.34218/IJARET.11.5.2020.043.
[16] M. Jiang, X. Fan, Retinamask: a Face Mask Detector, arXiv, 2020, https://arxiv.org/abs/2005.03950.
[17] M. Inamdar, N. Mehendale, Real-Time face mask identification using face masknet deep learning network, Ssrn Electron. J. (2020), https://doi.org/10.2139/ssrn.3663305.
[18] B. Roy, S. Nandy, D. Ghosh, MOXA: A deep learning based unmanned approach for real-time monitoring of people wearing medical masks, Trans. Indian Nat. Acad. Eng. 5 (2020) 509–518, https://doi.org/10.1007/s41403-020-00157-z.
[19] S. Li, X. Ning, L. Yu, L. Zhang, X. Dong, Y. Shi, W. He, Multi-angle head pose classification when wearing the mask for face recognition under the COVID-19 coronavirus epidemic, in: 2020 International Conference on High Performance Big Data and Intelligent Systems (HPBD&IS), Shenzhen, China, 2020, https://doi.org/10.1109/HPBDIS49115.2020.9130585.
[20] P. Khandelwal, A. Khandelwal, S. Agarwal, Using Computer Vision to Enhance Safety of Workforce in Manufacturing in a Post Covid World, arXiv, 2020, https://arxiv.org/abs/2005.05287.
[21] B. Qin, D. Li, Identifying facemask-wearing condition using image super-resolution with classification network to prevent COVID-19, Sensors 20 (18) (2020), https://doi.org/10.3390/s20185236.
[22] M. Loey, G. Manogaran, M.H. N.Taha, N.E. M.Khalifa, A hybrid deep transfer learning model with machine learning methods for face mask detection in the era of the COVID-19 pandemic, Measurement 167 (1) (2021), https://doi.org/10.1016/j.measurement.2020.108288.
[23] N. Ud Din, K. Javed, S. Bae, J. Yi, A novel GAN-based network for unmasking of masked face, IEEE Access 8 (2020) 44276–44287, https://doi.org/10.1109/ACCESS.2020.2977386.
[24] M.S. Ejaz, M.R. Islam, M. Sifatullah, A. Sarker, Implementation of principal component analysis on masked and non-masked face recognition, 2019 1st International Conference on Advances in Science, Engineering and Robotics Technology (ICASERT) (2019), https://doi.org/10.1109/ICASERT.2019.8934543.
[25] Google API, 2020. Accessed on 25-May-, https://pypi.org/project/google_images_download/.
[26] Bing API, 2020. Accessed on 27-May-, https://pypi.org/project/bing-image-downloader/.
[27] Labelimg Annotation Tool, 2020. Accessed on 01-June-, https://tzutalin.github.io/labelImg/.
[28] Z. Wang, G. Wang, B. Huang, Z. Xiong, Q. Hong, H. Wu, P. Yi, K. Jiang, N. Wang, Y. Pei, H. Chen, Y. Miao, Z. Huang, J. Liang, Masked Face Recognition Dataset and Application, arXiv, 2020, https://arxiv.org/abs/2003.09093.
[29] D. Chiang, Detect faces and determine whether people are wearing mask. FaceMaskDetection, 2020. Accessed on 22-May-2020, https://github.com/AIZOOTech/.
[30] S. Ge, J. Li, Q. Ye, Z. Luo, Detecting masked faces in the wild with LLE-CNNs, in: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, 2017, https://doi.org/10.1109/CVPR.2017.53.
[31] S. Nowozin, Optimal decisions from probabilistic models: the intersection-over-union case, in: 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, 2014, https://doi.org/10.1109/CVPR.2014.77.
[32] F. Ahmed, D. Tarlow, D. Batra, Optimizing expected intersection-over-union with candidate-constrained CRFs, in: 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, 2015, https://doi.org/10.1109/ICCV.2015.215.