IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received January 10, 2022, accepted February 1, 2022, date of publication February 10, 2022, date of current version February 22, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3150882

Small Traffic Sign Detection in Big Images: Searching Needle in a Hay

YAWAR REHMAN[1], **HAFSA AMANULLAH**[2], **MUHAMMAD AYAZ SHIRAZI**[3,4], **AND MIN YOUNG KIM**[3,5], **(Member, IEEE)**

[1] Department of Electronic Engineering, NED University of Engineering and Technology, Karachi, Sindh 75270, Pakistan
[2] Dhanani School of Science and Engineering, Habib University, Karachi, Sindh 75290, Pakistan
[3] School of Electronics Engineering, IT College, Kyungpook National University, Buk-gu, Daegu 41566, South Korea
[4] Haptics, Human–Robotics and Condition Monitoring Laboratory (National Center of Robotics and Automation), NED University of Engineering and Technology, Karachi, Sindh 75270, Pakistan
[5] Research Center for Neurosurgical Robotic System, IT College, Kyungpook National University, Buk-gu, Daegu 41566, South Korea

Corresponding author: Min Young Kim (minykim@knu.ac.kr)

This work was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) by the Ministry of Education under Grant 2021R1A6A1A03043144.

ABSTRACT Traffic sign detection is an essential module of self-driving cars and driver assistance system. The major challenge being, traffic sign appear relatively smaller in road view images. It covers only 1%-2% of the total image area. Hence, its challenging to detect very small traffic sign in a larger image covering huge background of similar shape objects. Thus, we propose YOLOv3 network layers pruning and patch wise training strategy for small sized traffic sign detection. This aids in improving recall percentage and mean Average Precision. We also propose anchor box selection algorithm that uses bounding box dimension density to obtain optimal anchor set for the dataset. This reduces false positives and log-average miss rate. The proposed approach is evaluated on German traffic sign detection benchmark and Swedish traffic sign dataset and proves that it achieved a good balance between mAP and inference time.

INDEX TERMS Anchor box algorithm, network pruning, small object detection, YOLOv3.

I. INTRODUCTION

Among several fields on the canvas of artificial intelligence, the intelligent transportation system is the hot research area for the researchers and scientists. Today the automotive industry is developing vehicles that employ intelligence based technology that reduces the chances of accidents. An intelligent vehicle can delineate the road conditions on the basis of traffic signs that are fixed on either side of the road. Hence, an accurate traffic sign recognition system that can pick information from a traffic sign fixed several meters away with a smaller apparent size is necessary for such vehicles.

The traffic sign recognition system mainly consists of two steps; (1) detection and correct localization of the traffic sign from an image (2) classification of the detected traffic sign. Several different research articles claimed to have achieved near 100% detection accuracy on the German Traffic Sign Detection Benchmark (GTSDB) [1]. However, the authors in [2] noted that the accuracy of the traffic sign detection system decreases with the increase of distance between the camera and the traffic sign on the road. Furthermore, they also noted a void of annotation in GTSDB that can evaluate the detector in this perspective and proposed Korean Traffic Sign Detection (KTSD) dataset to fill the gap. The authors in [3] also noted the same and argued that the detector should be robust to detect the traffic signs of smaller size with reference to the image size. Hence, we may conclude that the robust traffic sign detection system should be able to detect the small size traffic signs with reference of the image size. This property of the detector will also leverage the driver assistance systems in warning the danger ahead of time.

In addition, the speed of the traffic sign detection system also plays an important role. These days the Convolutional Neural Network (CNN) provides promising feature set to detect and classify an object. CNN networks like Fast RCNN, Faster RCNN, and Mask RCNN [4]–[6] use region proposals to detect an object. The authors in [7] acquired the upper and lower human body part features from the fully connected layer using image patches. This helps the authors to use autoencoder for transforming the acquired features to

The associate editor coordinating the review of this manuscript and approving it for publication was Claudio Cusano.

discriminative lower dimensional values. These neural networks are highly accurate in object detection but lacks in per frame detection speed required for the real time traffic sign detection. On the other hand, Single Shot Detector and You Only Look Once detector [8], [9] are very fast in detection but lacks in the accurate detection of the small size objects such as traffic signs.

Hence, for intelligent transportation system, a reliable and fast traffic signs detection system is required. A reliable traffic sign detection system would guarantee the detection of smaller traffic signs that may lead to early drive assistance warning and may avoid fatal accidents. Moreover, fast traffic sign detection will ensure that every frame is sifted rapidly to provide reliable information to the driver assistance system in due time. In this work we propose few updates over YOLOv3 [9] network, which provides us the fast and reliable traffic sign detection system as compared with the state-of-the-art detection systems.

The main contributions of this paper are as follows:

- Tuned YOLOv3 network for small traffic sign detection: We propose to remove few of the DBL layers in the YOLOv3 network. For larger objects these DBL layers in the network extract fine features necessary for the detection and classification of an object. However, for the traffic signs that are small as compared to the image size, these DBL layer extract a lot of redundant features that leads to false detection with other objects. Hence, by removing these redundant DBL layers we may limit the detector to only learn basic core features of the objects such as basic shape of traffic signs. In addition, we also propose break an input image in patches before training and testing. This will cause in decreasing the relative size ratio of the traffic sign and image. We achieved 9% and 2.5% increase in mean Average Precision for German Traffic Sign Detection Benchmark and Swedish Traffic Sign (STS) dataset respectively.
- Sort and Scale Anchor Box Selection: We propose to divide the traffic signs into three groups of small, medium, and large traffic signs by analysing the bounding box distribution from the ground truth data. These three groups signify the pixel size of traffic signs. Each group is assigned an anchor box by calculating its median value. Once, the three anchor boxes are obtained, we find the remaining six anchors as the derivatives of the median anchors. Hence, total of nine anchors are selected for the dataset. The proposed technique gives 5.5% and 2.7% increase in the detection accuracy than the base k-means anchor box selection technique in YOLOv3 for GTSDB and STS datset respectively. The proposed method can be effective in cases where an object has high pixel size variations.
- Focal Loss: We propose to validate the effects of using focal loss as objectness score. We found the values of hyper-parameters alpha and gamma for the traffic signs as suggested by the authors of [10]. We concluded that our proposed method using Tuned YOLOv3 and Sort and Scale method achieves 1.37% higher detection accuracy than the detector tuned for the small size object detection using the focal loss as objectness score.

The remaining of the paper is organised as follows. In Section 2 related works are discussed. Section 3 deals with the methodology of the proposed traffic sign detection system. Results are presented in Section 4. And finally we conclude the paper in Section 5.

A. RELATED WORK

Rigorous research and developments have been achieved in the field of Traffic Sign Recognition (TSR) lately. It is because the TSR is an important block for the driver-less cars and Advance Driver Assistance Systems (ADAS). A plethora of research work containing different approaches for the traffic sign detection is available in the literature. The researchers have used the colour attributes as features and colour based detection models [11]–[13], shape features [14], [15], channel features [16], and the popular Maximally Stable Regions (MSERs) [17], [18] for the traffic sign detection. These methods have performed well in their time and achieved high detection accuracies on the well-known traffic sign datasets. Moreover, the researchers enhanced the performance of the aforementioned methods by intelligently incorporating the mathematical models in features selection, extraction, and classification phase.

The progress in the field of Convolutional Neural Network has shifted the interest of many researchers towards designing or tuning the CNN layers and parameters to achieve the desired detection accuracy. All the CNN networks can be classified into two categories i.e. single stage and multi-stage detectors. Single stage CNN detectors have built-in detection stage and accomplish the detection and classification of an object inside an image in a one go. On the other hand, multi-stage detectors first utilize some method to detect an object and then they classify that object. As per the literature information, generally, these CNN based object recognition techniques are better and efficient than recognition techniques based on the hand-crafted features. The authors in [19] used multi-stage CNN detector for the traffic sign recognition. In the first stage, the traffic signs were detected by a region proposal CNN network detector. Then in the second stage Hough transform was used to refine the localization of the traffic sign. And finally in the third stage CNN network based classifier was used for the classification. Similarly, Serna and Ruicheck in [20] proposed a multi-stage traffic sign recognition system. In the first stage the authors have utilized the region proposal CNN network for the traffic sign detection and in the next stage they proposed the CNN based classifier. Their network was able to achieve 96.16% detection accuracy at 3.3 frames per second or processing speed on German Traffic Sign Detection Benchmark (GTSDB). The multi-stage CNN detection systems are meticulously designed and provide good detection accuracy. However, these networks mostly lag behind in giving high FPS for real-time traffic sign detection. Now, the single stage

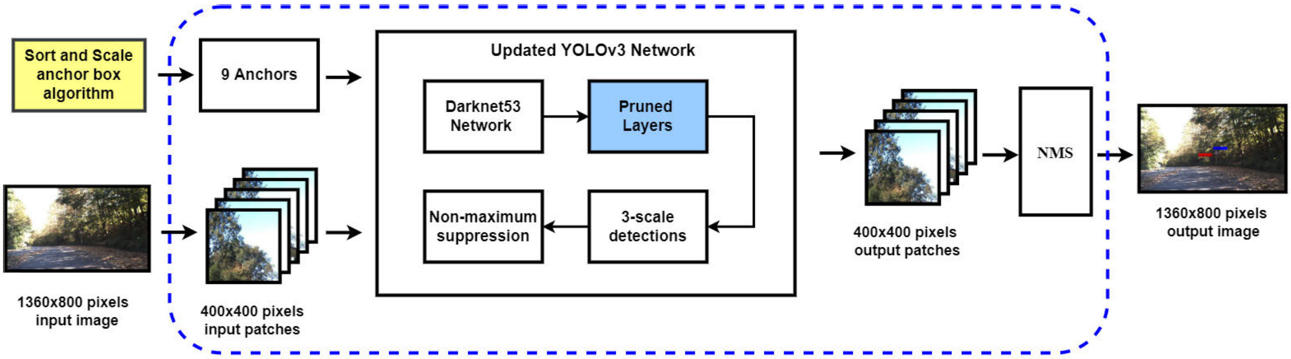

Sort and Scale anchor box algorithm
9 Anchors
Updated YOLOv3 Network
Darknet53 Network
Pruned Layers
Non-maximum suppression
3-scale detections
1360x800 pixels input image
400x400 pixels input patches
400x400 pixels output patches
NMS
1360x800 pixels output image

FIGURE 1. Proposed approach for small traffic sign detection. The yellow coloured box indicate proposed anchor box algorithm, where as blue coloured box represent improved section.

CNN detectors detect, localize, and classify the traffic sign in one go hence, they provide inherent advantage of speed over multi-stage detectors. These detectors are designed to avoid the processing time consumed in the generation of region proposals or sliding windows. Traffic sign detection system with Single Shot Detector (SSD) [21], feature pyramid network [22], SSD with feature pyramid network [23], and YOLOv3 [24], [25] suggests that a fast real-time traffic sign detection system can be made using the single stage detectors. The single stage networks provide a very fast way for object detection. However, these networks cannot achieve high accuracies on the benchmark datasets that contains smaller objects such as the traffic signs.

In addition, there is also some innovative work reported by the researchers in the field of traffic sign detection in both the categories. Kamal *et al.* [26] presented a traffic sign detection method by merging segmentation architectures SegNet and U-Net and named the network as SegU-Net. For training the presented network, authors have used the modified Tversky loss function. On GTSDB, their proposed network achieved the precision and recall of 95.29% and 89.01%, respectively. Liu *et al.* in [27] reported coarse to fine approach for traffic sign detection. On TT100K dataset, the authors reported the F1-score of 91.55 at 21 FPS. Tabernik and Skočaj in [28] presented a system for large scale traffic sign recognition. The authors presented end-to-end automatic learning technique using mask region proposal network on their proposed dataset. The dataset consisted of around thirteen thousand traffic annotations in different weather conditions and appearance variations. Only 3% of error rate is reported by using the proposed approach on the proposed dataset. Wang *et al.* in [29] presented a two-stage network model for the traffic sign detection. The authors have used the prior information such as locations and sizes of the traffic signs to make a probability distribution model. A light-weight superclass detector was then concatenated with the probability model for the detection purpose. The next stage includes design of a refinement classifier. The proposed system was tested on TT100K dataset and achieved 92.16% mAP at 7 FPS.

Almost all of the proposed detectors either excel in the detection accuracy or the speed in FPS. In the traffic sign detection, higher FPS can be achieved by the single shot detectors such as YOLOv3 detector. And for increasing the accuracy in YOLOv3, we noted two important factors (1) the detection of the small traffic signs; (2) accurate placement and localization of the anchors on the traffic signs. Hence, to achieve higher detection accuracy and FPS for traffic sign detection, detection of small traffic sign and a nice set of anchors are required.

B. PROPOSED METHOD

YOLOv3 is a single-stage detector. It processes complete image in a single run hence marking itself as fast object detector. It infers MS COCO dataset test image in 29ms, equivalent to 34.5 Frames Per Second (FPS), which is an impressing number. Despite being fast, it lags behind in terms of accuracy percentage. Author in [9] states that the YOLOv3 attains mAP of 31% for MS COCO dataset that is substantially low. For traffic sign detection, the detector must be accurate and fast to assist drivers to make well-timed appropriate decisions. Hence, there is a need to attain a balance between mAP and inference time. Moreover, the detectable objects in MS COCO dataset are much larger that covers at least 40% to 50% pixel area relative to complete image. While traffic signs appear quite smaller in road scenes and occupy only 1.5% to 2% pixel area relative to complete image. Thus, detection accuracy for small objects (for example, traffic signs) must be enhanced. In addition, anchor box sizes and scales play an important role in object detection. The anchor box appropriate size helps to reduce false detections and improve recall percentage, hence this aids in the improvement of mean Average Precision (mAP) eventually.

To address small traffic sign detection and to attain an optimal balance between mAP and inference time, we propose following improvements to YOLOv3 network: (1) network layers pruning, (2) patch-wise training strategy and (3) anchor box selection algorithm. We propose to reduce the network length to an appropriate size. This facilitates in saving fine features of traffic signs, which yields an improved recall percentage. Secondly, the proposed patch-wise training strategy aids to attain an optimal balance between detector accuracy and inference time. Thirdly, the proposed anchor box selection algorithm assists in determining best fit anchor boxes for the dataset, which helps in reducing False Positive (FP) and lowering log-average miss rate. Fig. 1 shows the proposed traffic sign detection approach, where at first; the input image is divided into 400 × 400 pixels patches and passed into proposed YOLOv3 network. The network outputs bounding box predictions for each patch. The output patches are then used to recreate original 1360 × 800 pixels image, and the redundant predictions of patches are removed using Non-Max Suppression (NMS). The proposed technique helped to lower log-average miss rate (lamr) and inference time, hence facilitated in obtaining an optimal balance between accuracy and processing speed.

C. ANCHOR BOX SELECTION

Anchor boxes play a major role in the detection accuracy of the single stage detectors. Their appropriate size and scale assists to localize objects faster and precisely. YOLOv3 default anchor box selection method uses k-means clustering to determine their size. It makes use of bounding box dimensions and their overlap in percentage with the selected anchors. It provides a good generalization of bounding box dimensions but does not considers the anchors requirement for a particular dataset. It means that for a typical traffic sign dataset, there is a higher chance of having many small size traffic signs and less amount of large size traffic signs with reference to the image size. The greater percentage of smaller traffic signs in a dataset demands higher percentage of smaller anchors than the larger ones to have faster and reliable detections. Therefore, we propose the anchor box selection algorithm based on bounding box dimension sorting and clustering.

Refering to Fig. 2, the algorithm estimates three basic anchor boxes based on the pixel size of traffic signs i.e. small, medium, and large. These anchors are then scaled to form the derivative anchors making the total up to nine different anchor sizes. The algorithm works in three steps: 1) Sorting and grouping the dataset according to object sizes in the training set, 2) Forming clusters based on condition specified in Equation (1), and 3) Extracting foundation anchors and its derivatives. Quantity of anchors in each group is estimated using Equation (2).

We validated the proposed approach on GTSDB dataset. The GTSDB dataset can be divided into three categories i.e. small, medium, and large based on traffic sign bounding box dimensions [30], [31]. The small traffic sign refers to traffic sign size smaller than 32 × 32 pixels, medium signs refers to size in the range of 32 × 32 & 96 × 96 pixels, and large signs refers to size greater than 96 × 96 pixels. According to given sizes, the dataset constitute of 41% small, 52% medium, and 7% large traffic signs as illustrated in Fig. 3.

In Fig. 3, it could be noted that GTSDB is a realistic dataset that can be used to train network models for real-time scenarios. Majority of the traffic sign are concentrated in small and medium range. While large range consists of lesser traffic signs. We may assume the similar distribution of traffic signs in real-scenario. Thus, detection must be fast to completely process the captured frames per second i.e. 30 FPS. For this purpose, there is a need of anchor boxes with appropriate size and scales. We propose the quantity of anchor boxes to be reserved for each group i.e. one anchor box for large sized traffic sign and rest for small and medium sized traffic signs.

The foundation anchors (represented in blue colour in Fig. 4)) are the median values of the clusters formed using condition specified in Equation (1).

$$\forall x \leq \frac{x_{max} - x_{min}}{3} \tag{1}$$

where x_{max} and x_{min} are the maximum and minimum data points respectively of stated groups (small and medium sized traffic signs). All bounding box dimensions fulfilling the provided condition forms a cluster. The median value of formed clusters are the foundation anchors. These anchors are then used to estimate derived anchors. In order to do so, we fix the total number of anchors to nine and calculate the number of anchors that can be assigned to each group by using Equation (2).

$$A_g = min(A_a, \lceil \frac{n_g}{n} \times A \rceil) \tag{2}$$

where A_g and A_a represents anchors in a group and allowed anchors per group respectively, n_g and n denotes bounding boxes in group range and total number of bounding boxes respectively, while A symbolize the total number of anchors. If a group contains more number of ground truth bounding boxes, more anchors will be assigned to the group and vice versa. The minimum number of anchors that can be assigned to each group is one (i.e. the foundation anchor) and maximum the four (i.e. foundation + derived anchors). Once the number of anchors that will be assigned to each group is determined, the foundation anchors of each group was scaled by the multiple of $2^{1/3}$, $2^{1/2}$ and $2^{2/3}$ to obtain the derived anchors. First 'm' derived anchors were selected per group as identified by Equation (2). E.g. A_g for small group is 4, then one foundation anchor plus three derived anchors were selected. It should be noted that from the large traffic sign group, only foundation anchor is the required anchor box. The anchor set obtained is illustrated in Fig. 4, where blue boxes are the foundation anchors and the white ones are derived anchors.

Fig. 5 show the plot between allowed anchors in a group (A_a) versus anchors selected in a group (A_g). This plot also depicts the working of Equation (2). All the parameters of small, medium, and large traffic signs were provided to the Equation (2) at a time. A_a was set to nine anchors as per requirement and was decreased in proportion to the increase in A_g. The A_g for small, medium, and large size traffic sign was calculated as four, six, and one respectively. However, we have selected four, four, and one for small, medium, and large size traffic signs to accommodate all anchors in total of nine possible anchor set. The response of Equation (2)

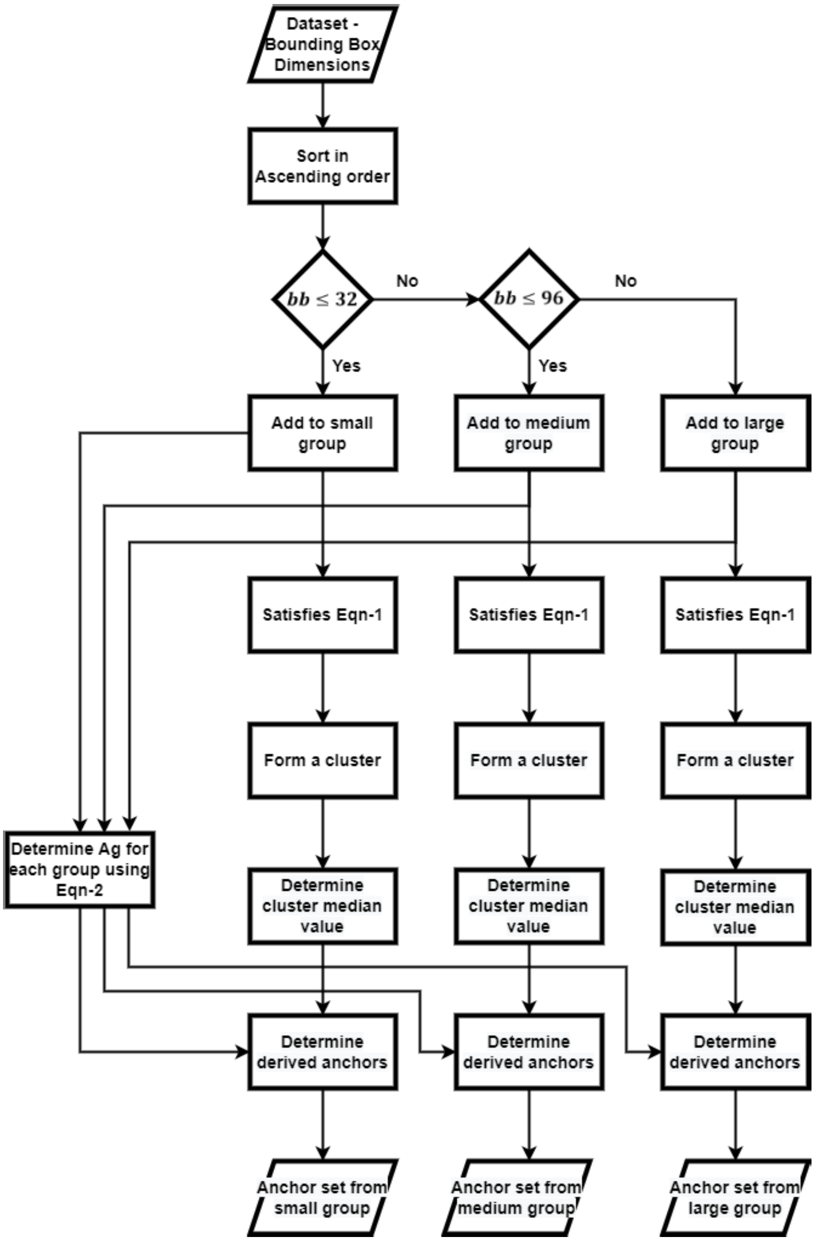

Dataset - Bounding Box Dimensions
Sort in Ascending order
$bb \leq 32$
No
$bb \leq 96$
No
Yes
Yes
Add to small group
Add to medium group
Add to large group
Satisfies Eqn-1
Satisfies Eqn-1
Satisfies Eqn-1
Form a cluster
Form a cluster
Form a cluster
Determine Ag for each group using Eqn-2
Determine cluster median value
Determine cluster median value
Determine cluster median value
Determine derived anchors
Determine derived anchors
Determine derived anchors
Anchor set from small group
Anchor set from medium group
Anchor set from large group

FIGURE 2. Flowchart for anchor box selection algorithm.

shows that more numbers of anchors were assigned to the areas where more number of traffic signs were present and vice versa.

Incorporating the obtained anchor boxes in the training process resulted in 100% recall percentage for GTSDB dataset, hence, detection accuracy has also been improved. Moreover, it aids in lowering number of false positives and log-average miss rate. The proposed method is a one go process, which determines optimal anchor boxes for the dataset. On the other hand, k-means clustering, which is the base method in YOLOv3 has to be executed multiple times along with network training and testing to determine the optimal anchor boxes.

D. YOLOV3 NETWORK PRUNING WITH PATCH-WISE TECHNIQUE

YOLOv3 is a single stage object detector. It processes complete input image at once and predicts bounding boxes at three different scales. It uses Darknet53 network with skip connections as feature extractor, which extracts features at multiple levels similar to feature pyramid network. It upsamples deeper feature map by stride of two and concatenates with a shallower feature map to detect objects of various sizes. This upsampling takes place twice in the network. The combined map is then processed by a stack of five convolutional layers, each followed by batch normalization and ReLU activation layer, represented as DBL block as shown

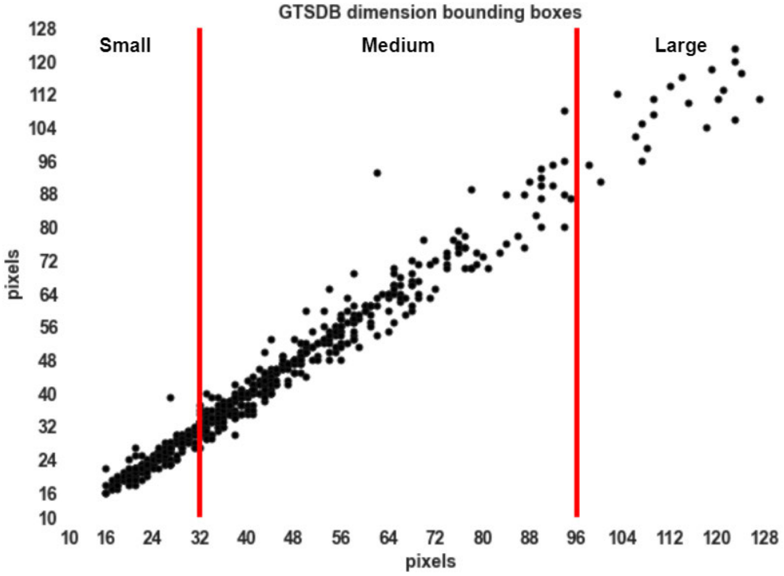


FIGURE 3. Graphical analysis of GTSDB dimension bounding boxes.

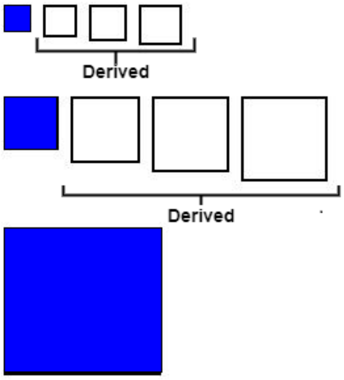


FIGURE 4. Anchor set for GTSDB dataset.

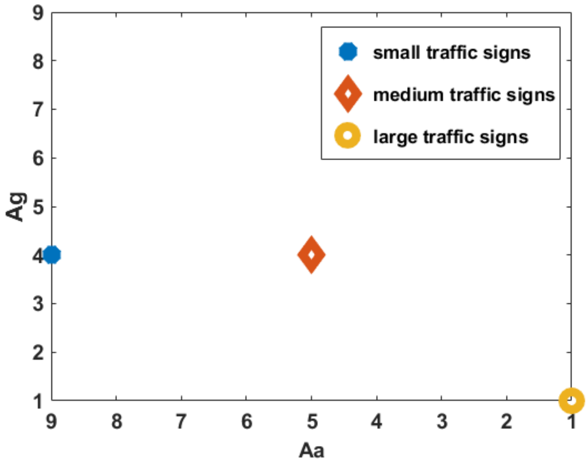


FIGURE 5. Plot of A_a vs A_g.

in Fig. 6a. Each level outputs bounding box predictions. The redundant bounding box predictions (least overlap with true outputs) are removed using NMS.

Generally, the traffic signs appear small in road view images and occupy smaller pixel area in the image. Thus it possesses lesser feature values. These features are lost with deeper networks since at deeper layers object texture and fine details are learnt while at shallow layers, network learns about object's basic shapes and strokes. Therefore, a natural idea is to use output feature map of shallow layers to detect small-sized objects with reference to the image size. We propose to prune YOLOv3 network layers to an optimum value for effectively detecting small traffic sign. Here, optimum value defines the number of layers that help to attain maximum mAP. Fig. 7 illustrates how the mAP first increases with reduction of layers and after a certain point it drops again following a Gaussian trajectory. The mAP is highest when DBL stack is reduced to 2. Therefore, we reduced the stack of 5 DBL layers to 2 DBL layers at each detection level as shown in Fig. 6b. This helped in improving mean log-average miss rate and mean Average Precision.

The default setting in YOLOv3 takes an input image of 1:1 aspect ratio. If the image doesn't qualify the aspect ratio check, it is resized to 416 × 416 pixel size. Thus for 1360 × 800 pixels image, it will be resized to 416 × 416 pixel size, which eventually lessens the pixel area of traffic signs. Hence, the features of traffic signs are lost. Therefore, we propose to train and test the network with patches of size 400 × 400 pixels. This helps to retain features of traffic sign and provide assistance in its detection process. Moreover, [21] suggest that a patch-wise input to the network takes lower inference time than complete image input. Thus, this enables to obtain a precise detection in minimum time.

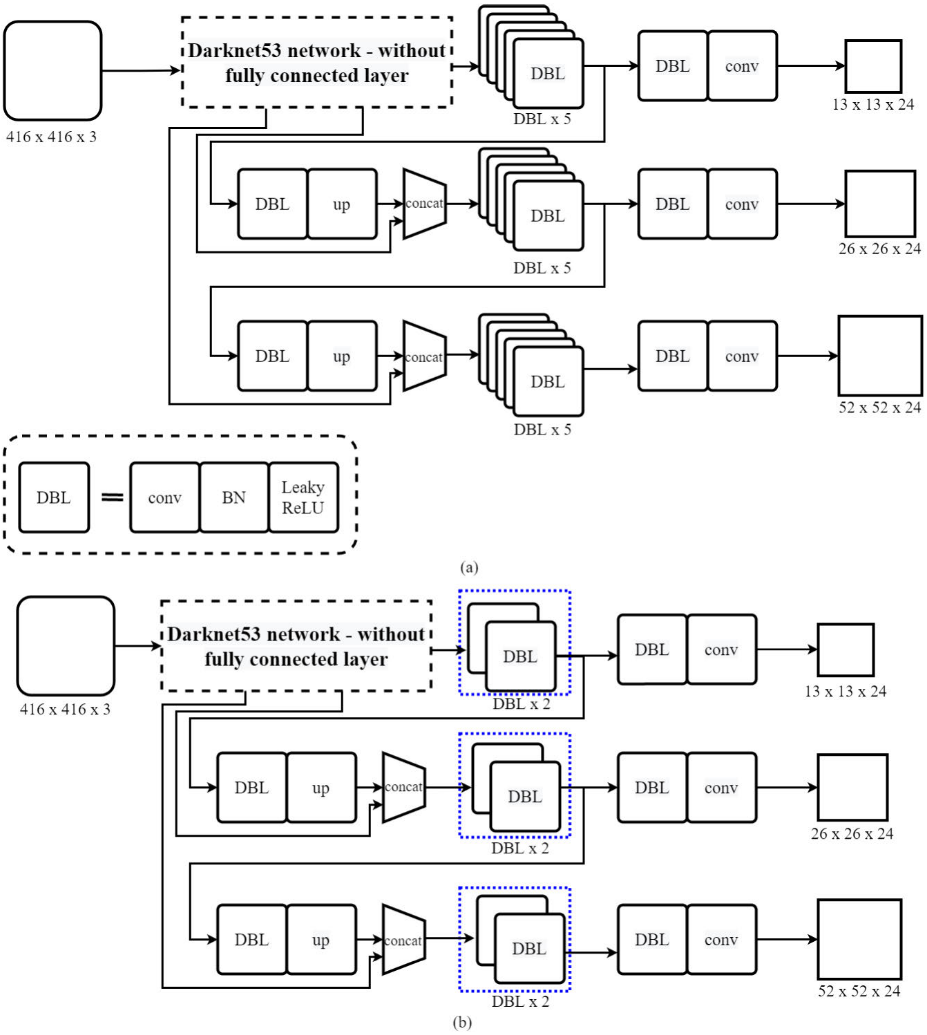


FIGURE 6. (a) Default YOLOv3 network (b) Proposed YOLOv3 network with pruned layers.

We trained the network with input image patches varying from size 400 × 400 to 800 × 800 pixels, depending upon the size and location of traffic sign. The annotations of traffic sign in an image were stretched 400 pixels further in each direction to have enough background along with traffic sign and also to ensure the essence of road scene does not vanishes. As an example, consider the image shown in Fig. 8a where bounding box co-ordinates for traffic sign are [825, 428, 862, 463] stretching the co-ordinates 400 pixels in each direction yields an image patch to train the network. In the case of limited area for stretching, limit constraint is applied. After patches extraction, the bounding box annotations are updated in a .csv file in accordance to the patch size.

To test the network, each test image is cropped and saved as patches of size 400 × 400 pixels. The patches are extracted using sliding window scheme, where 400 × 400 size window slides over the image and saves an image patch. The output image patches from the network are then consolidated to re-create the complete image, while the bounding box predictions by the detector are updated with reference to complete image in a .csv file. For example, consider the image shown in Fig. 8b, where red coloured box represents the sliding window. It is placed at the origin of the image and a patch is captured at this point as shown at the bottom of Fig. 8b. The window slides next in x-direction with a stride of 100 pixels as shown with blue coloured box in Fig. 8b. The window slides in x-direction again until it reaches the right bound of the image. Next the window slides in y-direction with the same stride as shown with green coloured box in Fig. 8b. The window slides along x-direction and then along y-direction until it covers the complete image. The overlap between image and sliding window is cropped and saved as patch, shown at the bottom of Fig. 8b.

Network pruning and patch-wise training and testing approach helped to learn and save fine features of traffic signs. This eventually aided to reduce the log-average miss rate and improved mean Average Precision. In addition, it assisted to lower test-image inference time and hence an optimal balance between mAP and inference time is achieved.

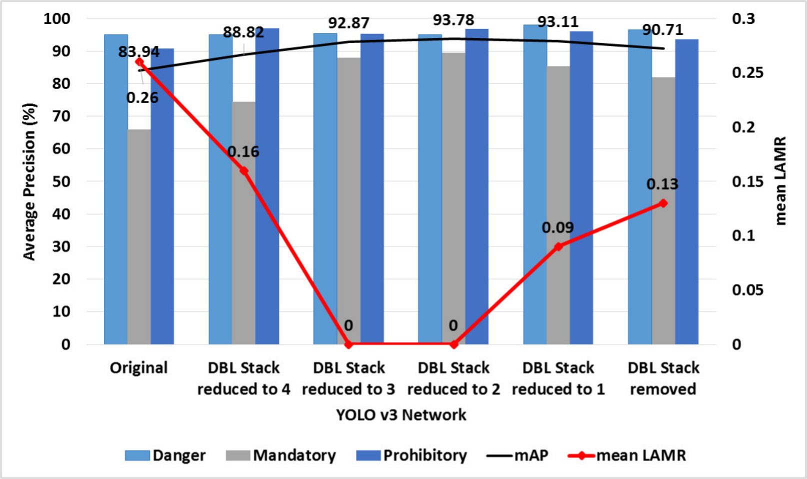


FIGURE 7. Effect of layers pruning in YOLOv3 network.

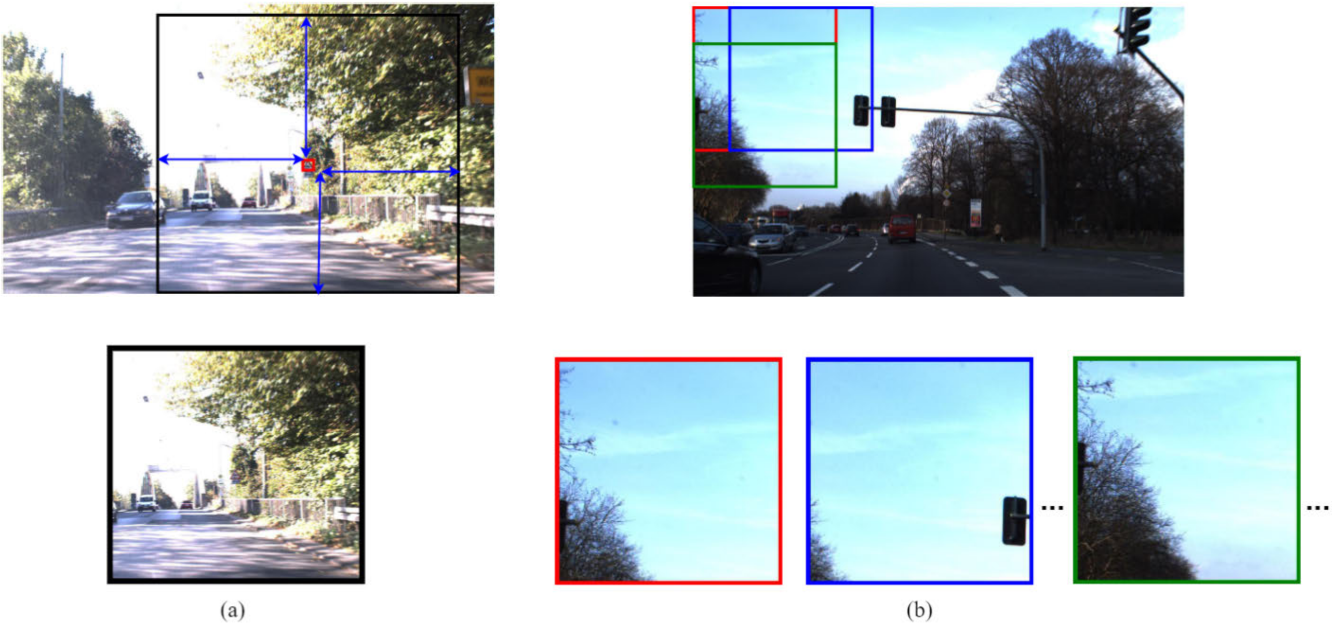


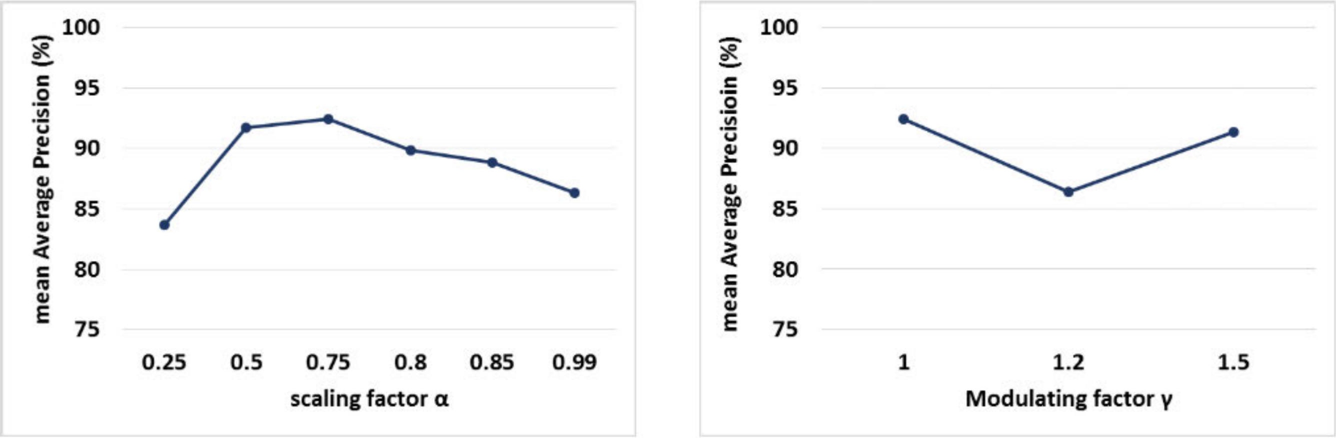
FIGURE 8. Patches creation-(a) for train images (b) for test images.

E. FOCAL LOSS

YOLOv3 computes losses for objectness score (probability percentage of object presence), bounding box predictions and classification loss. These losses are computed using different loss functions; bounding box width and height loss is computed using mean square error, while bounding box centroid loss, objectness score and classification loss are calculated using Binary Cross Entropy (BCE) loss. Each computed loss is averaged for the selected batch size. And in the end all the averaged losses are summed to determine total loss for the specific epoch.

Focal loss (FL) is an effective loss for small-size object detection in the single stage detectors or for class imbalance between foreground and background during training. In 1360 × 800 pixels image there is huge class imbalance between traffic sign and background. It is because the size of traffic sign is smaller as compared to the reference image size. Moreover, the traffic signs are small objects in a road scene image, therefore we found it effective to incorporate focal loss in YOLOv3 network.

Focal loss dynamically weights BCE loss with the help of modulating factor γ and scaling factor α. γ heavily penalizes hard negative examples and minimizes loss contribution by easy examples. For well classified examples, it decays the scaling factor against improvement in correct class probability. Equations (3) & (4) shows mathematical form



(a)

(b)

FIGURE 9. Plot of focal loss experimentation; (a) When scaling factor is varied from 0.25 to 0.99, while modulation factor is 1, (b) Plot of focal loss experimentation, when modulating factor is varied, while scaling factor is 0.75.

of BCE loss and FL respectively.

$$BCE\ loss = \begin{cases} -\log(p), & if\ y = 1 \\ -\log(1-p), & otherwise \end{cases} \tag{3}$$

$$FL = \begin{cases} \alpha(1-p)^{\gamma}.BCE\ loss, & if\ y = 1 \\ \alpha(p)^{\gamma}.BCE\ loss, & otherwise \end{cases} \tag{4}$$

where α and γ are two hyper-parameters and their values has to be determined experimentally. Tsung-Yi *et al.* in [10] proposes 0.25 and 2 as best values for α and γ respectively for MS COCO dataset. We found that the values of parameters α and γ are dependent on the size of the objects to be detected. Since MS COCO dataset comprises of large objects while traffic signs appear small. We followed the experimentation process provided in [10] and found 0.75 and 1 as optimal values for α and γ respectively. The graphs in Fig. 9 indicates how the mAP varies with varying α and γ respectively.

II. EXPERIMENTAL RESULTS AND DISCUSSION

We performed various experiments to validate our proposed approach. The proposed method stated in Section 3 is implemented by modifying the GitHub repository [32]. It uses Keras with Tensorflow at the backend. The experiments were performed on Google CoLab using Tesla T4 GPU having 16 GB memory and 12 GB RAM. The proposed approach is evaluated for six evaluation metrics, namely mean Average Precision (mAP), inference time, recall percentage, false positives and log-average miss rate.

The network was trained in two stages; for first 10 epochs, Darknet53 network is frozen and rest of the network is trained with the batch size of 32 image patches to obtain a stable loss, and finally complete network is trained for 50-60 epochs with the batch size of 8 image patches. The number of epochs was selected based on the trend observed from experimentation. The network training was halted upon reaching same validation loss for 10 consecutive epochs. It was observed that total of 70 epochs were sufficient for training the network. Default network values were used for batch size and learning rate. For loss function optimization adam optimizer is used with learning rate decay of 0.1 per three consecutive

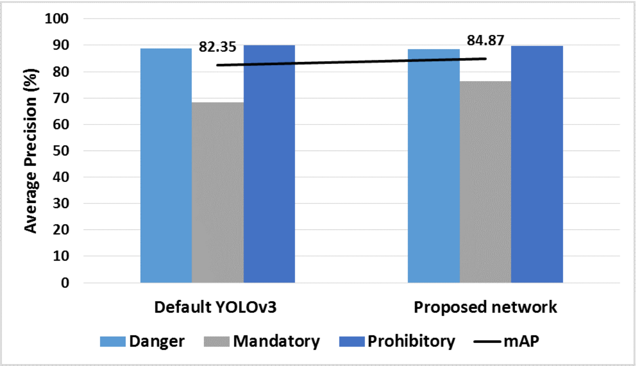


FIGURE 10. Comparison of average precision (for all three classes) and mean Average Precision results for default and tuned YOLOv3 network with STS dataset.

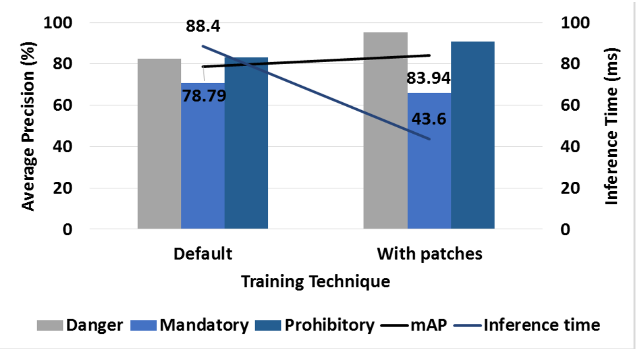


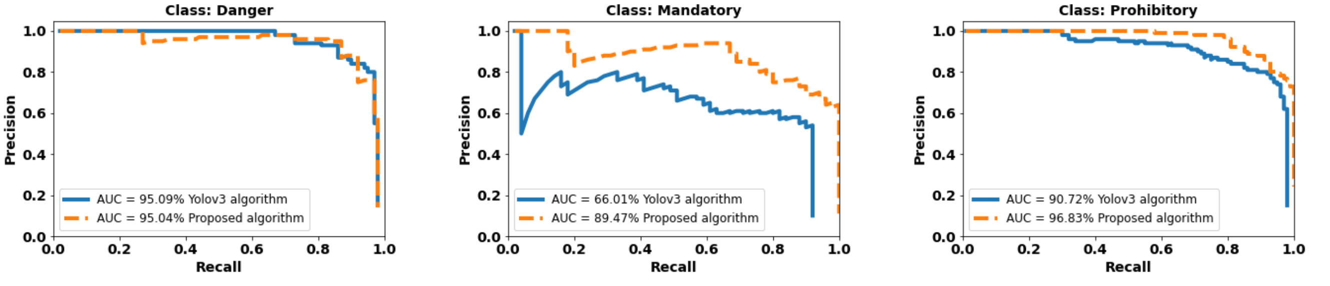
FIGURE 11. Effect of patch-wise training technique on mAP and inference time.

TABLE 1. Detail description of dataset.

Dataset	Country	Traffic Sign scenes	Traffic Sign images	Image size (px)	Sign size (px)
GTSDB	Germany	900	939	1360×800	16×16 to 128×128
STS	Sweden	2909 (Visible signs)	1963	1280×960	9×9 to 264×249

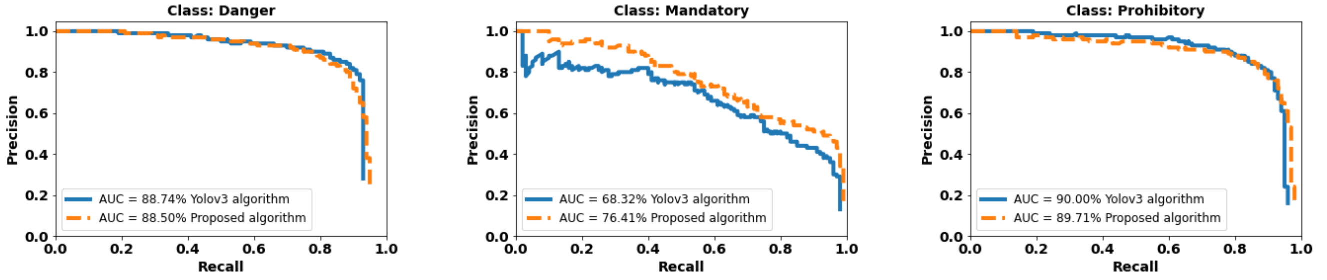
TABLE 2. Comparison of anchor box algorithms and YOLOv3 network.

Dataset	Anchor box algorithm	Evaluation Metric	Default YOLOv3 Network	Tuned YOLOv3 Network
GTSDB	Default-Kmeans	Recall percentage	90.86%	100.00%
		AUC	78.29%	93.93%
	Sort and Scale (Ours)	Recall percentage	95.90%	100%
		AUC	83.94%	93.78%
STS	Default-Kmeans	Recall percentage	96.24%	95.33%
		AUC	79.65%	81.28%
	Sort and Scale (Ours)	Recall percentage	95.80%	96.97%
		AUC	82.35%	84.87%



(a) (b) (c)

FIGURE 12. Precision-recall curves of GTSDB dataset for Original and proposed YOLOv3 network (a) Danger class (b) Mandatory class (c) Prohibitory class.



(a) (b) (c)

FIGURE 13. Precision-recall curves of STS dataset for Original and proposed YOLOv3 network (a) Danger class (b) Mandatory class (c) Prohibitory class.

epochs for consistent validation set loss, while initial learning rate used is 1e-3.

A. DATASETS

In order to validate our proposed approach, we experimented with two datasets namely German Traffic Sign Dataset Benchmark (GTSDB) [1] and Swedish Traffic Sign (STS) dataset [33]. GTSDB is a widely used dataset. It comprises of 600 training and 300 test images, of resolution 1360 × 800 pixels. The number of traffic signs in an image varies from 0 to 6, of various sizes, ranging from 16 × 16 to 128 × 128 pixels. The dataset is categorized into three super classes: danger, mandatory and prohibitory.

TABLE 3. Number of parameters in default and tuned YOLOv3 network.

Network	Parameters
Original - Stack of 5 DBL layers	65M
DBL Stack reduced to 4	63M
DBL Stack reduced to 3	49M
DBL Stack reduced to 2	47M
DBL Stack reduced to 1	33M
DBL Stack removed	32M

In comparison to GTSDB dataset, STS dataset is a large dataset with more than 20000 images of 1280 × 960 pixels resolution, but among them only 20% are annotated. We used Set1 Part0 as train set and Set2 Part0 as test set, using only

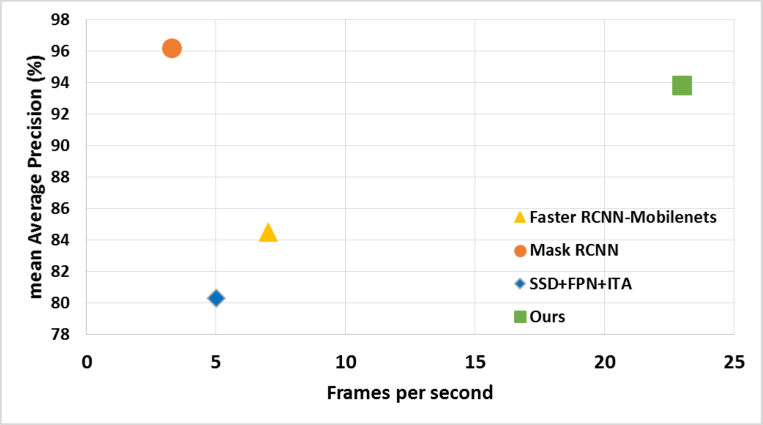


FIGURE 14. mAP and inference time comparison for state-of-the-art methods and proposed method for GTSDB dataset.

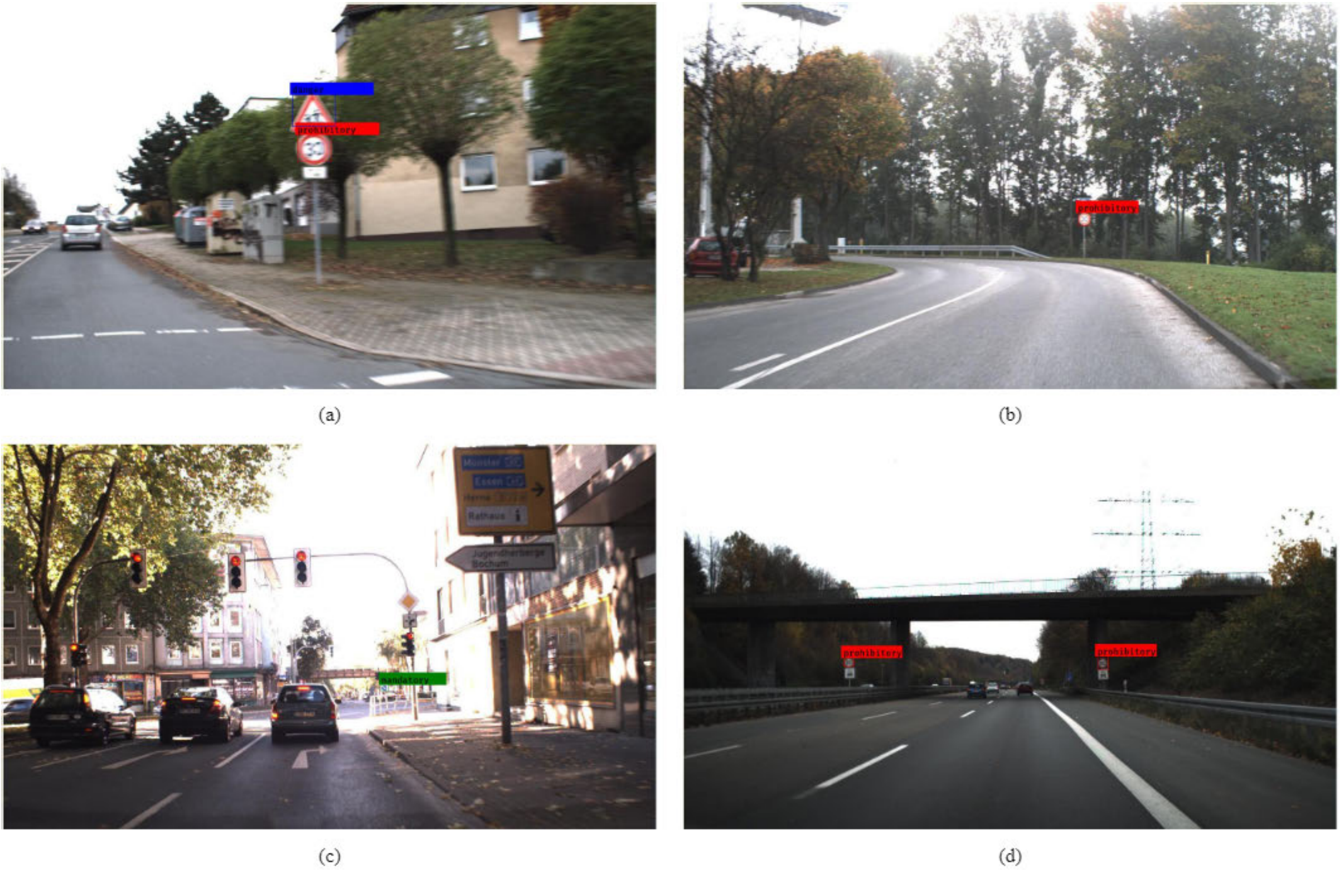


FIGURE 15. Qualitative results of proposed approach for GTSDB dataset (a) blurred traffic sign (b), (c), (d) small traffic signs.

visible signs for experimentation. The size of traffic sign varies from 9 × 9 to 264 × 249 pixels. The dataset can be categorized into three super classes: danger, mandatory and prohibitory. The detail description of both the dataset is provided in Table 1.

B. ANCHOR BOX ALGORITHM

We performed experiments with GTSDB and STS dataset to evaluate the proposed anchor box algorithm on default and tuned YOLOv3 network with patch-wise training and testing approach. The experimentation results are presented

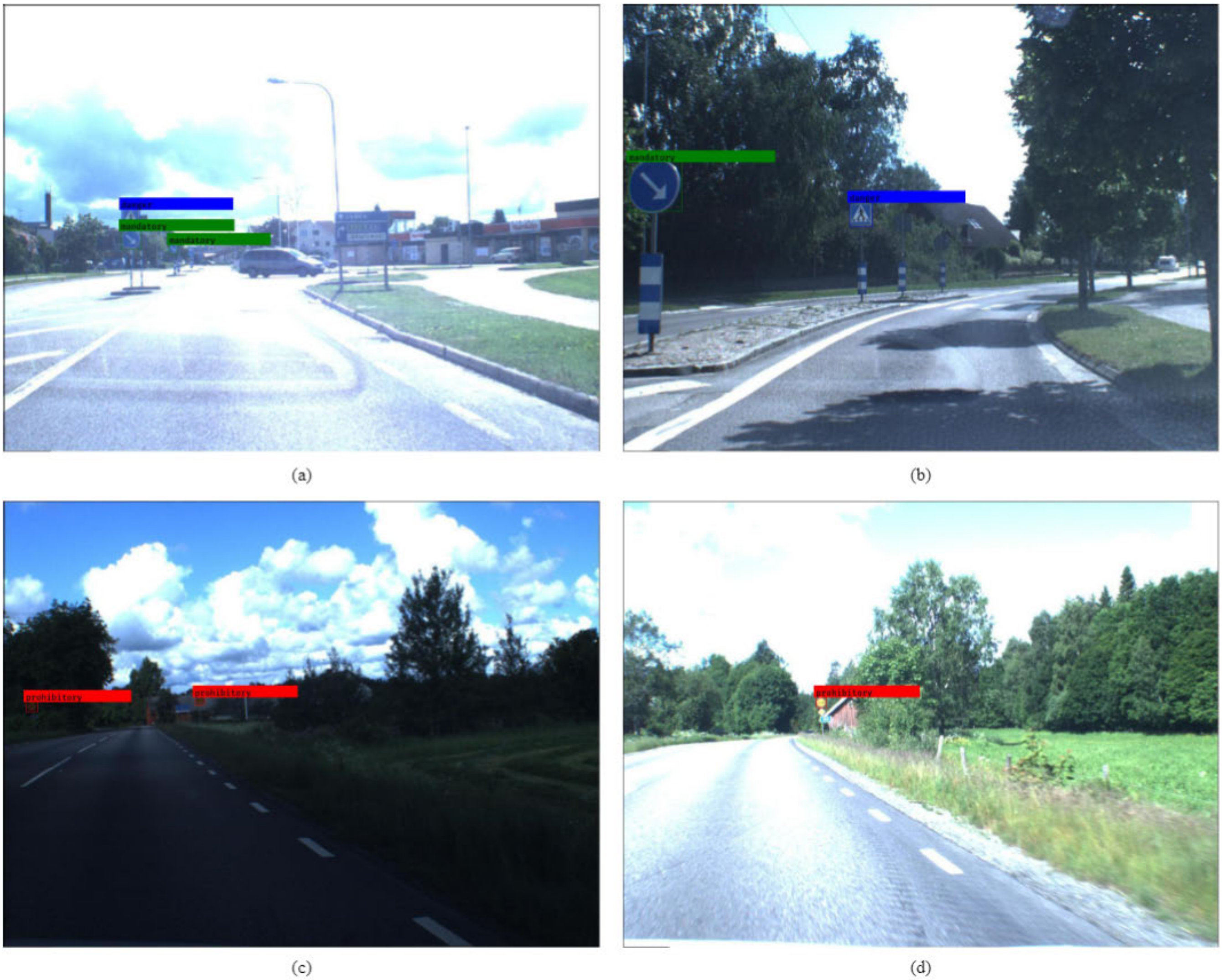


FIGURE 16. Qualitative results of STS dataset for proposed method (a) variable small size traffic signs, (b) size variation of small and large sign, (c) signs detection in dim light (d) small sign detection in bright sunshine.

in Table 2. The evaluation metrics considered here are recall percentage and AUC.

For GTSDB dataset, the recall percentage for sort and scale algorithm is 5% higher than the default k-means method which further improves to 100% for tuned YOLOv3 network. Similarly, AUC gets a rise of 5.5% with sort and scale algorithm anchor set with a further boost of 9% with tuned YOLOv3 network.

For STS dataset, the recall percentages are nearly equal for both the anchor sets but there is an increment of 1.6% with tuned YOLOv3 network coupled with sort and scale anchor set. On the other hand, AUC improves by 2.7% with sort and scale anchor set and it further improves with tuned network. Hence, it proves that proposed anchor box selection algorithm is more effective than default kmeans algorithm in terms of detection accuracy and recall percentage.

C. YOLOV3 NETWORK PRUNING WITH PATCH-WISE TECHNIQUE

We experimented with pruning YOLOv3 network layers for small traffic signs detection on GTSDB dataset. We tested with removing the DBL layers from the stack of 5 to 0 for all three levels of detection. The mAP for two DBL blocks in the stack is the maximum among all as shown in Fig. 7. The graph in Fig. 7 indicates that pruning the layers to certain limit helps the network, while additional pruning can drop the mAP. Just like Gaussian distribution, it is symmetric around pruned YOLOv3 network with a stack of two DBL layers. In addition, the mean log-average miss rate is zero for pruned YOLOv3 network with a stack of 2 and 3 DBL layers and increases as the layers are increased or decreased. Thus the pruned network with 2 DBL layers is more accurate than the original YOLOv3 network.

TABLE 4. Experimental results for focal loss as object-ness score with GTSDB dataset. The maximum among all are highlighted in bold.

Modulating Factor γ	Scaling Factor α	mAP	Recall %
0	0.25, 0.5, 0.75	-	-
1	0.25	83.68%	99.33%
1	0.50	91.70%	99.80%
1	0.75	**92.41%**	**100.00%**
1	0.80	89.83%	98.93%
1	0.85	88.84%	97.80%
1	0.99	86.31%	98.70%
1.2	0.75	86.38%	99.80%
1.5	0.75	91.37%	99.58%
Ours		93.78%	100%

The network parameters for each network model is presented in Table 3. The number of parameters for network model was reduced from 65M to 47M. The proposed network pruning was also evaluated on STS dataset, yielding 2.5% rise in mAP than the default network as shown in Fig. 10.

We experimented to train default YOLOv3 network with default resizing method and patch-wise technique, discussed in section 3.2. Training the network with patches helped in obtaining an optimal balance between mAP and inference time. As the plot in Fig. 11 indicates, with patch-wise technique mAP has improved by 5% with reduction of inference time to half of the original value.

The proposed approach is tested on two traffic sign datasets namely: GTSDB and Swedish dataset. Fig. 12 and 13 shows precision-recall curves for GTSDB and STS dataset respectively, where proposed approach has better AUC than the default network. Our approach out-performs among other state-of-the-art methods. Fig. 14 illustrates mAP and inference time comparison for state-of-the-art methods and proposed method. Among all, our method excels in terms of achieving a balance between mAP and inference time. The qualitative results of proposed approach for GTSDB and STS dataset are shown in Fig. 15 and 16 respectively. It qualifies that the proposed method is very effective in the detection of small and blurred traffic signs in a dense and cluttered environment.

D. FOCAL LOSS

Focal loss is effective for small objects detection. We used focal loss for objectness score loss, to help network differentiate between traffic signs and the background. Tsung-Yi *et al.* in [10] states 2 and 0.25 as best values for scaling and modulating factor respectively. We followed the experimental procedure stated in [10] to determine optimal values for α and γ for small objects such as traffic signs. Following the procedure we equated γ to zero to find optimal value for α. For modulating factor γ set to 0, training quickly failed and the network diverged, it continued the same for range 0 to 1. Therefore, γ was initiated to 1 to find optimal value for α. For α ranging from 0.5 to 0.75, we achieved relatively good mAP with nearly 100% recall percentage for GTSDB dataset. For finding optimal value for γ, we varied the range for γ while keeping α constant to 0.75. The experimental results for various combinations of α and γ are stated in Table 4. Maximum detection accuracy has been achieved for α =0.75 and $\gamma = 1$, as shown in Table 4.

The usage of focal loss does not seem effective here, there is a drop in mAP than the proposed approach. Hence, it can be concluded that tuned network and patch-wise training strategy is more effective for small traffic signs and it results 1.37% increase in mAP than focal loss implementation.

III. CONCLUSION

In this paper we propose methods that assist in small traffic signs detection. We propose sort and scale based anchor box selection algorithm that uses bounding box dimension density to extract optimal anchor boxes. This aids in lowering false positives and log-average miss rate. Moreover, we propose YOLOv3 network layers pruning and patch wise training and testing strategy for small traffic signs. The proposed approaches helped to improve recall percentage and hence mAP is improved. It helps to achieve an optimal balance between detection accuracy and inference time.

To further assist in small traffic sign detection. We tested the network with focal loss incorporated as objectness score. This actually reduced the detection accuracy by 1.37% thus proving that our approach is more effective for small traffic signs detection than focal loss. Experiments prove that the proposed approach is effective and competent to other state-of-the-art methods for small traffic sign detection. We noticed that the number of anchors would be increased if there is high number of large size traffic signs in a dataset. This will result in a possible false positive generation, which can be counted as a limitation of the proposed technique. The fidelity of the proposed technique can be validated for other objects and detectors in future.

REFERENCES

[1] J. Stallkamp, M. Schlipsing, J. Salmen, and C. Igel, "The German traffic sign recognition benchmark: A multi-class classification competition," in *Proc. Int. Joint Conf. Neural Netw.*, Jul. 2011, pp. 1453–1460, doi: 10.1109/IJCNN.2011.6033395.

[2] Y. Rehman, I. Riaz, X. Fan, and H. Shin, "D-patches: Effective traffic sign detection with occlusion handling," *IET Comput. Vis.*, vol. 11, no. 5, pp. 368–377, Aug. 2017, doi: 10.1049/IET-CVI.2016.0303.

[3] Z. Zhu, D. Liang, S. Zhang, X. Huang, B. Li, and S. Hu, "Traffic-sign detection and classification in the wild," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 2110–2118, doi: 10.1109/CVPR.2016.232.

[4] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1440–1448.

[5] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

[6] K. He, G. Gkioxari, and P. Dollar, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Mar. 2017, pp. 2961–2969.

[7] S. U. Khan, T. Hussain, A. Ullah, and S. W. Baik, "Deep-ReID: Deep features and autoencoder assisted image patching strategy for person re-identification in smart cities surveillance," *Multimedia Tools Appl.*, 2021, doi: 10.1007/s11042-020-10145-8.

[8] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Y. Fu, and A. C. Berg, "SSD: Single shot multibox detector," *Proc. Eur. Conf. Comput. Vis.*, Amsterdam, The Netherlands, Oct. 2016, pp. 21–37.

[9] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.

[10] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, "Focal loss for dense object detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.

[11] A. Gupta and A. Choudhary, "A framework for real-time traffic sign detection and recognition using Grassmann manifolds," in *Proc. 21st Int. Conf. Intell. Transp. Syst. (ITSC)*, Nov. 2018, pp. 274–279.

[12] S. Khalid, N. Muhammad, and M. Sharif, "Automatic measurement of the traffic sign with digital segmentation and recognition," *IET Intell. Transp. Syst.*, vol. 13, no. 2, pp. 269–279, Feb. 2019.

[13] Y. Yang, H. Luo, H. Xu, and F. Wu, "Towards real-time traffic sign detection and classification," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 7, pp. 2022–2031, Jul. 2016.

[14] Á. Gonzalez, M. Á. Garrido, D. F. Llorca, M. Gavilan, J. P. Fernandez, P. F. Alcantarilla, I. Parra, F. Herranz, L. M. Bergasa, M. Á. Sotelo, and P. R. de Toro, "Automatic traffic signs and panels inspection system using computer vision," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 2, pp. 485–499, Jun. 2011.

[15] N. Barnes, A. Zelinsky, and L. S. Fletcher, "Real-time speed sign detection using the radial symmetry detector," *IEEE Trans. Intell. Transp. Syst.*, vol. 9, no. 2, pp. 322–332, Jun. 2008.

[16] Y. Yuan, Z. Xiong, and Q. Wang, "An incremental framework for video-based traffic sign detection, tracking, and recognition," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 7, pp. 1918–1929, Jul. 2017.

[17] H. Luo, Y. Yang, B. Tong, F. Wu, and B. Fan, "Traffic sign recognition using a multi-task convolutional neural network," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 4, pp. 1100–1111, Apr. 2018.

[18] M. Donoser and H. Bischof, "Efficient maximally stable extremal region (MSER) tracking," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 1, Jun. 2006, pp. 553–560.

[19] J. Li and Z. Wang, "Real-time traffic sign recognition based on efficient CNNs in the wild," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 3, pp. 975–984, Mar. 2019.

[20] C. G. Serna and Y. Ruichek, "Traffic signs detection and classification for European urban environments," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 10, pp. 4388–4399, Oct. 2020.

[21] H. S. Lee and K. Kim, "Simultaneous traffic sign detection and boundary estimation using convolutional neural network," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 5, pp. 1652–1663, May 2018.

[22] T. Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, Jul. 2017, pp. 2117–2125.

[23] E. H. Chen, P. Rothig, J. Zeisler, and D. Burschka, "Investigating low level features in CNN for traffic sign detection and recognition," in *Proc. IEEE Intell. Transp. Syst. Conf. (ITSC)*, Oct. 2019, pp. 325–332.

[24] J. Choi, D. Chun, H. Kim, and H.-J. Lee, "Gaussian YOLOv3: An accurate and fast object detector using localization uncertainty for autonomous driving," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 502–511.

[25] G. N. Doval, A. Al-Kaff, J. Beltran, F. G. Fernandez, and G. F. Lopez, "Traffic sign detection and 3D localization via deep convolutional neural networks and stereo vision," in *Proc. IEEE Intell. Transp. Syst. Conf. (ITSC)*, Oct. 2019, pp. 1411–1416.

[26] U. Kamal, T. I. Tonmoy, S. Das, and M. K. Hasan, "Automatic traffic sign detection and recognition using SegU-Net and a modified Tversky loss function with L1-constraint," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 4, pp. 1467–1479, Apr. 2020.

[27] L. Liu, Y. Wang, K. Li, and J. Li, "Focus first: Coarse-to-fine traffic sign detection with stepwise learning," *IEEE Access*, vol. 8, pp. 171170–171183, 2020.

[28] D. Tabernik and D. Skocaj, "Deep learning for large-scale traffic-sign detection and recognition," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 4, pp. 1427–1440, Apr. 2020.

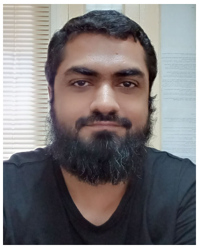
[29] Z. Wang, J. Wang, Y. Li, and S. Wang, "Traffic sign recognition with lightweight two-stage model in complex scenes," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 2, pp. 1121–1131, Feb. 2022, doi: 10.1109/TITS.2020.3020556.

[30] H. Zhang, L. Qin, J. Li, Y. Guo, Y. Zhou, J. Zhang, and Z. Xu, "Real-time detection method for small traffic signs based on Yolov3," *IEEE Access*, vol. 8, pp. 64145–64156, 2020, doi: 10.1109/ACCESS.2020.2984554.

[31] J. Wan, W. Ding, H. Zhu, M. Xia, Z. Huang, L. Tian, Y. Zhu, and H. Wang, "An efficient small traffic sign detection method based on YOLOv3," *J. Signal Process. Syst.*, vol. 93, no. 8, pp. 899–911, Aug. 2021, doi: 10.1007/s11265-020-01614-2.

[32] (Oct. 2020). *Github Repository for Train Your Own YOLO*. Accessed: Oct. 17, 2020, doi: 10.5281/zenodo.5112375. [Online]. Available: https://github.com/AntonMu/TrainYourOwnYOLO

[33] *Swedish Traffic Signs Dataset*. Accessed: Dec. 27, 2020. [Online]. Available: https://www.cvl.isy.liu.se/research/datas ets/traffic-signs-dataset/



YAWAR REHMAN received the bachelor's degree in electronics engineering from the Mehran University of Engineering and Technology, Pakistan, in 2008, and the master's and Ph.D. degrees in electronics and communication engineering from Hanyang University, South Korea, in 2017. He is currently an Assistant Professor with the NED University of Engineering and Technology, Pakistan. His current research interests include computer vision, image processing, feature extraction, and deep learning.

HAFSA AMANULLAH received the B.E. degree in electronic engineering and the M.Eng. degree in industrial electronics from the NED University of Engineering and Technology, Pakistan, in 2017 and 2021, respectively.

Currently, she is working as a Senior Research Assistant with Habib University. Her research interests include computer vision, deep learning, and artificial intelligence.

MUHAMMAD AYAZ SHIRAZI received the B.E. degree in electronics engineering from the NED University of Engineering and Technology (NEDUET), Pakistan, in 2012, and the Ph.D. degree from the Optomechatronics and Multi-Scale Robotics Laboratory, Kyungpook National University, South Korea, in 2018. He was the Former Postdoctoral Researcher at KAIST, South Korea. Currently, he is working as a Senior Researcher with NCRA, NEDUET. His research interests include camera calibration and 3D reconstruction, digital image processing, augmented reality, deep learning, and structured light imaging.

MIN YOUNG KIM (Member, IEEE) received the B.S., M.S., and Ph.D. degrees with the Korea Advanced Institute of Science and Technology, South Korea, in 1996, 1998, and 2004, respectively. From 2004 to 2005, he worked as a Senior Researcher at Mirae Corporation. From 2005 to 2009, he worked as a Chief Research Engineer at Kohyoung Corporation, in the research field of artificial vision systems for intelligent machines and robots. He was a Visiting Associate Professor with the Department of Electrical and Computer Engineering and the School of Medicine, Johns Hopkins University, from 2014 to 2016. Since 2009, he has been an Assistant Professor with the School of Electrical Engineering and Computer Science, Kyungpook National University. Currently, he is a Full Professor with the School of Electronics Engineering, Kyungpook National University, and the Deputy Director of the Research Center for Neurosurgical Robotic Systems and the KNU-LG Convergence Research Center. His research interests include visual sensor system for robotic perception and recognition, human augmentation devices, control system for microrobotic systems, and surgical robotic systems.