JCBI
Journal of Computing & Biomedical Informatics
ISSN: 2710 - 1606
Volume 05 Issue 02
2023
Research Article
https://doi.org/10.56979/502/2023

Prohibitory Sign Detection Using Machine Learning

Shah Fahad[1], Muhammad Munwar Iqbal[1*], Mehmoon Anwar[1],and Anees Tariq[1]

[1]Department of Computer Science, University of Engineering and Technology Taxila, 47050, Pakistan
[*]Corresponding Author: Muhammad Munwar Iqbal. Email: mnwariq@gmail.com

Received: June 12, 2023 **Accepted:** August 19, 2023 **Published:** September 17, 2023.

Abstract: With advancement, technology is becoming increasingly creative in every field. It is atomized using machine learning and artificial intelligence approaches. The detection of traffic signs is a core technology used in automated vehicles. Real-world applications of traffic sign recognition include self-driving, traffic monitoring systems, and safe driving. Very limited work has been done on the Pakistani traffic signs after finding a huge gap and knowing the importance of growing technology and demand. In the results, we first created a dataset since Pakistan lacked a dataset of this kind, and then we used algorithms for machine learning to produce the desired outcomes. To process the gathered images and recognize traffic signs, we employed a variety of deep learning algorithms, including CNNs (convolutional neural networks) and SVMs (support vector machines). We have used YoloV4 and Darknet53 with pre-trained weights of conv.137 for detection purposes, and we have used Google's GPU for training due to limited resources. We got an accuracy of 83 on the new dataset (mAP=83). Although the accuracy of 83 was promising, it was challenging to keep this rate with our limited resources, as we had to terminate the training process after three weeks on Google's GPU after 16,600 iterations and at 83 Mean Average Precision. Then, I took a few real-world images of different classes to check the performance of the training model. The results show the images with boundary boxes and predicted class categories.

Keywords: You Only Look Once; Graphics Processing Unit; Convolutional Neural Network, Support Vector Machine.

1. Introduction

The integration of various technologies, such as image processing and information processing, has led to the development of intelligent transportation systems (ITS). These systems aim to improve road traffic safety through methods such as, traffic sign recognition [1] vehicle distinguishing proof, road marking identification [2], and driver interruption discovery[3]. Traffic Sign Recognition (T.S.R.) has been a vital area of examination to help driving upkeep the traffic signs and for vehicles which are automated [4]. At first, as an instrument for driving, T.S.R. was proposed as security help, helping drivers make decisions quickly and correctly during emergencies by using image processing technology to identify traffic signs and providing reminders to the driver through voice notifications or other means. T.S.R. is particularly useful in challenging situations and inclement weather, as it might lessen the demand for drivers to interpret signs correctly [5]. Variety-based methods utilize the shade of the traffic signs as the principal element to recognize their area on the picture [6] use variety division to identify red variety traffic signs. The pictures are changed to grayscale from RGB variety space. Although variety division utilizing the RGB space takes less computational power and time, it is hard to apply progressive conditions since it requires stable brightening conditions. Akinlar et al. use Hough change to recognize general shapes (circles, square shapes, and so forth) and is supplemented with edge identification to identify speed cutoff and cautioning signs[7].

Yang et al. [8] utilize a variety likelihood model to create areas of interest, close by with Histogram of Situated Slope (Hoard) highlights as an element descriptor, and a CNN plays out the order. Zabihi [9] presents a technique for discovering and acknowledging road markings in drivers' field-of-view attention. Hoard highlights with SVM are utilized for location, and Filter highlights with a variety of data for acknowledgment. Deep learning techniques have become more common in computer vision applications as a result of improvements in CPU capacity and the accessibility of relevant datasets. As was referenced in the past segment, the fundamental classifications for nonexclusive item locators in view of CNNs are two-stage finders and one-stage identifiers. Many traffic sign locators considering CNNs use variations of the conventional article identifiers.

Zhang et al. [10] utilize a changed form of YOLOv2 to identify traffic signs. The creators changed the channel sizes and the number of layers to track down the best harmony among speed and precision. Muller ¨ et al. [11] present one more intriguing use of S.S.D., it utilizes a more profound component extractor and changed default bounding boxes to increment precision for traffic signal's location. Jixiang Wan et al. have introduced the model to detect small traffic signs and used Yolov3 for this purpose [12]. Song, Y., et al. [13] planned a CNN engineering thinking that conventional item locators are prepared to recognize objects that possess a huge part of a picture. Interestingly, traffic signs, as a rule, involve a little part of a picture.

Our investigation has developed an approach for dissecting Pakistani Prohibitory Traffic Signs, which is to be expected for a variety of reasons. There is likewise an absence of the Pakistani traffic signs benchmark dataset. We created a Prohibitory Traffic Sign Dataset. We used YOLOv4 and got mAP 83 and a precision of 0.98 for our dataset.

2. Literature Review

Most of the analysts offer unique consideration toward the recognition of traffic signs and prohibitory signs. There are numerous techniques that have been proposed for this work. They can be broadly divided into three categories: computations for traditional traffic sign identification, CNN-based (convolutional brain organizations) calculations for traffic sign location, and consideration instrument-based calculations for traffic sign discovery.

2.1 Traffic Sign Detection Using Convolutional Neural Network

In the meantime, the data pictures are separated into thick groups to truly distinguish traffic signs to create a superior part map. Lee and colleagues [14] structured traffic sign breaking point evaluation as a two-layered stance and shape class conjecture problem that could be handled entirely by a single Convolutional Neural Network. By extending the restriction of a contrasting format sign picture into the data picture plane, the goal sign was specifically evaluated. A Convolutional Neural Network-based stance and shape assumption task is used to solve the problem of breaking point evaluation.

Ibrahem et al. [15] introduced a multi-reason single convolutional neural network structure considering delicate oversight, including MobileNetv2 as the fundamental association. Zhang et al. [16] presented a streamed R.C.N.N. computation to obtain multi-scale features in the image pyramid. With the exception of the first layer, each subsequent layer of the streamed network merges as much of the previous layer's box as possible. Weighted multi-scale attributes are obtained using soft max and spot thing to complete joint planning, and traffic sign characteristics are included by finely tuned components to reduce traffic sign I.D. precision. By combining Faster R-CNN with a thought instrument and displaying a relationship between station features. [17] A deep learning technique based on convolutional neural network appropriate detection during driving was presented by Etiner et al. The suggested deep learning model's operational performance was evaluated according to the preprocessed and raw input types. The suggested model was used for training using the KFold 3 technique after the test and train sets of data were segregated.

2.2 Traffic Sign, Vehicle Movement and Road Marking Detection

The conventional traffic sign discovery calculations mostly extricate picture highlights considering the varieties and states of traffic signs, joined with relating picture handling procedures, and afterward perceive highway signage. Ellahyani et al. [18] presented a three-stage random forest-based live traffic sign identification and recognition system. The primary phase of the framework partitions the picture into areas of interest in light of a variety of data; the subsequent stage utilizes the invariant mathematical second to recognize triangles, square shapes, and circles; and To distinguish the identified shapes, the third stage employs an irregular timberland classifier to join Hoard (histograms of situated slopes) highlights with L.S.S. (neighborhood self-similitude) elements. Yıldız et al. [19] A traffic sign area estimation based on assortment and shape was proposed. During the component extraction stage, this method fully perceives the bundling relevant data and arithmetic attributes of road signs, employs RGB acquisition space and associated systems for shape verification, removes the plans without traffic signs from the image, and retains the accumulation traffic sign areas. Chen and colleagues [20] AdaBoost estimation was merged with S.V.R. (support vector backslide), and a saliency evaluation methodology was proposed that considered traffic sign variety, structure, and spatial information. Even though the area pace of the above-estimated numbers is adequate, it is not abstract because the part phonation is imbalanced and single. Because of a fantastic formation or various obstacle objects, however, if the authentic sign is acceded and hurt, the bogus dissemination and made it illegal rate may become extremely high. The deep learning-based traffic symbol discovery innovation has been widely adopted due to its outstanding presentation. A convolutional brain network-based traffic sign discovery strategy often outperforms current general objective location organization. Le et al. [21] introduced an additional protocol for amazing array reporting and division evaluating SVM, that is used to consistently recapture alternate solutions of traffic signs, use sensor blocks as that of the state vector of SVM for distribution portrait, and enter the Hough alter and structure area to deem polygons and circles to assert elective precincts.

Nguyen et al. [22] used a lightweight and successful association structure. Furthermore, to reduce a display of minor traffic sign distinguishing proof, a de-convolution module was applied to combine low-even out part maps to apparent level component maps, resulting in an improved incorporate aide, and the two better district suggestion networks were utilized to make alternate edges using the most elevated level component map as well as the superior part map. Considering the thought framework, the strategic approach can emphasize critical information while dismissing irrelevant information. Mnih et al. [23] suggested a technique in which information can be extracted from video and an image by flexibly selecting a movement of districts or positions and simply driving significant standards dealing with the chosen locales. Yuan et al. [24] suggested a multi-purpose feature mix system model that employs a thickly related de-convolution layer with bob affiliations, which is useful for extraction of minimal objective components and road signs recognition referred to as a space gathering task and operated as backslide task, as well as a vertical spatial progression thought module is suggested to obtain extra important information, thereby recognizing better I.D. Tian et al. [25] suggested yet another cyclic thought-based road signs revelation methodology for multi-scale assessment and use of adjacent settings through images. Kastner et al. [26] used a thought framework to swap potential elective road signs districts, and they used the number of Viola and Jones-style classification algorithms to determine the probability worth of every region. Zhang et al. [27] suggested an additional end-to-end design to promote minimal objective disclosure. This same thought instrument is used to refine part feedback. Furthermore, the thought instrument is employed to select as many more discriminative features as possible, such as box backslides and road sign requests. The preceding estimation uses a thought instrument to improve the area incorporate information, reduces the number of calculations performed by convolving the summarized information of the large picture, and promotes traffic sign distinguishing proof execution. It's doesn't, in any case, provide immediate convincing component filtering on the results.

2.3 Traffic Sign Detection Using Y.O.L.O.

Zhang et al. [28] presented a YOLOv2-based beginning-to-end convolutional network with multiple 1x1 convolutional layers in the association's central layer and a reduced top convolutional layer to reduce the computational multifaceted design. Zhang et al. [29] suggested M.S.A. YOlO-v3, presenting a multi-scale spatial pyramid pool block in the Darknet53 association, enabling the association to understand object incorporates significantly to a considerably greater extent, surrendered a base improvement method for building up the component pyramid in YOLOv3, and truly was using the fine-grained features of a base layer to obtain definite goal arranging.

3. Materials and Methods

The flow of the proposed methodology is shown in Figure 1

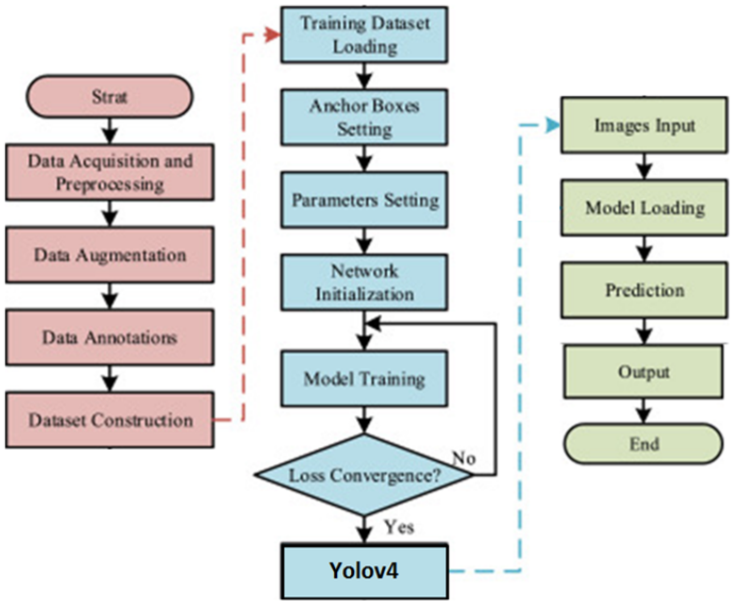


Figure 1. Prohibitory traffic sign detection flow

Detection of prohibitory traffic signs in Pakistan was challenging because we had no dataset and no already trained model existed to match our results. It is just starting and will be proved as a gateway toward automated technologies. Our proposed methodology and workflow are done in three major steps.

Dataset Construction, Model Training, and Object Detection. As we can see, the first step is to obtain the images. The dataset of prohibitory signs was then designed with essential features and information for detection. Dataset designing was the major job of research as there was no dataset of traffic signs in Pakistan. The dataset designed is Prohibitory Traffic Signs of Pakistan (PTSP-2D); a detailed overview is in table 1.

Table 1. Dataset Detail

Class/Object	Frequency	Class/Object	Frequency
No_Auto_Rickshaw	30	Fasten_the_Seat_Belt	30
No_Tractor	30	Slow	29
No_Cycle	46	Speed_Limit_20	11
Veichles_height_more_than_18_feet_prohibited	18	Speed_Limit_10	28
30_Speed_Limit_Termination	12	Truck_Stop_Sign	12
No_Veichles_width_more_than_3.5m	14	HTV_Speed limit__65	19
No_Pedistrian	35	HTV_Speed limit__50	31
No_Overspeed	16	LTV_Speed limit__70	36
No_Animal_Carts	32	LTV_Speed limit_80	40
No_Motor_Cycle	30	HTV_Speed Limit_60	29
Dead_Slow	29	Speed Limit 50	11
No_Overtake	11	Speed Limit_60	28
No_Parking	28	Speed__limit__80	12
Right_Turn_Prohibited	12	Speed__Limit_30	30
Use_of_Mobile_Prohibited	19	Speed_Limit_40	30
No_Entry	30	LTV_Speed limit_80	40
Reduce_your_speed	35	**Total**	803

3.1 Data Set Construction

We have different classes in our dataset, as shown above, but for a single class, there may be one or multiple traffic signs e.g., for No Entry, we come across three types of prohibitory signs. There are two varieties of stop traffic signs, but we have placed them under the same class I.D. After image collection, the preprocessing serves as the next step in our methodological approach, and it involves cleaning and removing noise from the dataset. When the dataset had been cleaned according to the criteria, we made annotations, for which we used the LabelImg tool to draw boundary boxes around the targeted classes (figure 2) and obtained a text file for each image.

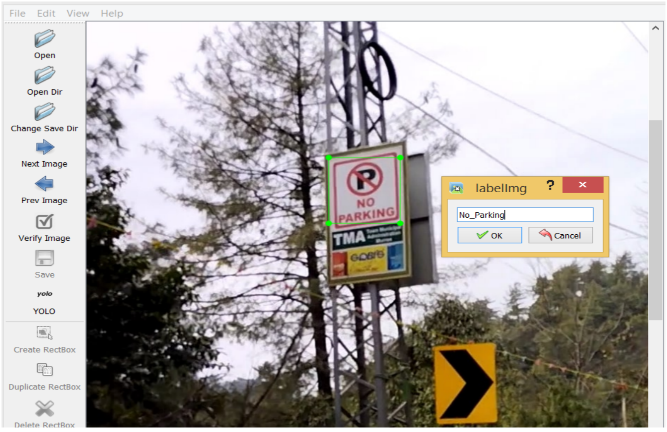


Figure 2. Labeling and Assigning Class Names

After that, the annotated data was divided into two sections for training and testing. We have used training data to train our model YoloV4 based on Darknet, while we test data have been used to evaluate the results. Thus, every image with the dataset would be linked to a .txt file with the same name, containing the object classes and coordinates as follows:

object-class> x centre> y centre> width> height>

3.2 Model Training

The model fed the number of images as per the setting we did in max batches. YOLOv4, the one-stage object detection algorithm, aims to improve accuracy and speed. The proposed approach is a detailed representation of our job YOLOv4:

• Darknet53 serves as the backbone

•SPP, P.A.N. for the neck •YOLOv3 as head

Darknet53 comprises convolution layers with measurements of 1 x 1 and 3 x 3, for a total of 53 layers (figure 3).

	Type	Filters	Size	Output
	Convolutional	32	3 × 3	256 × 256
	Convolutional	64	3 × 3 / 2	128 × 128
1×	Convolutional	32	1 × 1	
	Convolutional	64	3 × 3	
	Residual			128 × 128
	Convolutional	128	3 × 3 / 2	64 × 64
2×	Convolutional	64	1 × 1	
	Convolutional	128	3 × 3	
	Residual			64 × 64
	Convolutional	256	3 × 3 / 2	32 × 32
8×	Convolutional	128	1 × 1	
	Convolutional	256	3 × 3	
	Residual			32 × 32
	Convolutional	512	3 × 3 / 2	16 × 16
8×	Convolutional	256	1 × 1	
	Convolutional	512	3 × 3	
	Residual			16 × 16
	Convolutional	1024	3 × 3 / 2	8 × 8
4×	Convolutional	512	1 × 1	
	Convolutional	1024	3 × 3	
	Residual			8 × 8

Figure 3. Darknet 53 Convolutional Layers

3.2.1 YoloV4

You Only Look Once uses the following techniques:

Residual blocks, Bounding box regression, and Intersection Over Union (I.O.U.)

3.2.2 Residual blocks

Grids are initially applied to the image. S × S is the size of each grid. The figure below demonstrates how grids are created from such an input image.

3.2.3 Bounding box regression

A bounding box is a boundary that highlights an object in a picture. The image's bounding boxes each have the following features:

Width (bw, Height (bh, Class (for example, person, car, traffic light, etc. and Bounding box center (bx,by)

3.2.4 Intersection over Union (I.O.U.)

Convergence over association is a property in object recognition that shows how boxes intersect. Just go for I.O.U. is used to flawlessly provide a result box encompassing the articles. Each network cell is responsible for forecasting bounding boxes and their certainty scores. The I.O.U. is equal to 1 if the expected bounding box matches the real box exactly. This approach gets rid of bouncing boxes that aren't the same as the real thing. Knowing if a region contains an object or not depends on the IoU. By dividing the area of the intersection between the two boxes by the area of their Union, the IoU is determined using the following equation. The accuracy of the prediction increases with IoU.

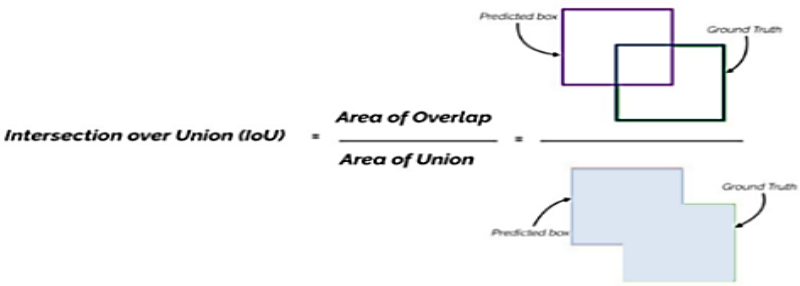


Figure 4. Intersection Over Union

We compute precision and recall using the IoU value for a given IoU threshold for tasks requiring object detection.

Figure 5 shows the intersection over union percentage, updated during the training process. Here, we have shown to clear its concept and to track our model performance. We overlook the model that is working well, is being trained, and is updating effective things like IoU, M.A.P. etc. Figure 5 shows the IoU percentage we got during training for one of our weights. The value of IoU decreases or increases with the training process. We got different IoU values for different weights during training i.e 10,000 weight, 20,000 weight, best weight, and last weight having different IoU during training, and the model keeps updating itself.

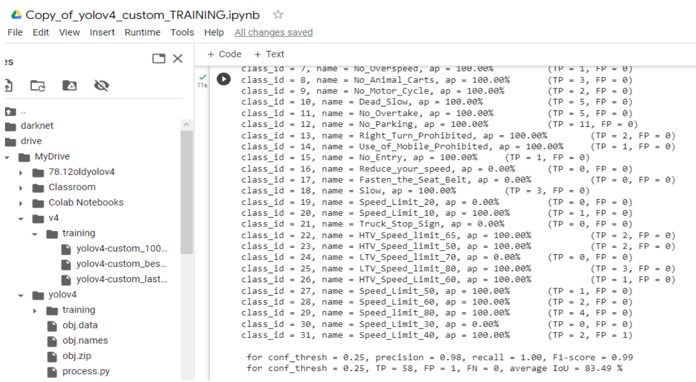


Figure 5. IoU Value at one of the training stages

3.3 Object Detection

Figure 6 is the object detection process. The suggested technique is a detailed visual representation of our work.

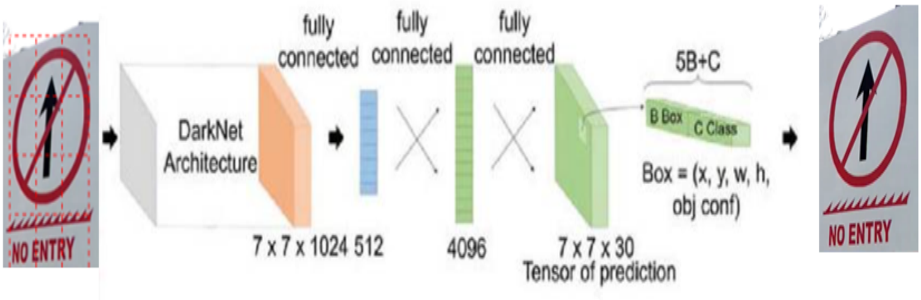


Figure 6. Object Detection Architeccture

4. Results

We have shown results in terms of mAP, performance metrics, and System predictions.

4.1 M.A.P. (Mean Average Precision)

It demonstrates how precise our model is and how effective our trained model is. The mean average precision (mAP) is used to evaluate models for object detection, such as Y.O.L.O. and R-CNN. The mAP calculates a score by comparing the detected box to the ground-truth bounding box. The greater the score, the more precise the model's detections. Precision measures the accuracy of our predictions.

$$Precision = \frac{TP}{TP+TN} \ldots\ldots\ldots \quad (1)$$

T.P. stands for true positives (predicted as positive and were found to be correct)

F.P. stands for false positives (predicted as positive but were found to be incorrect)

Mean Average Precision (mAP) is the average of all detected classes' A.P.

Mean Average Precision (mAP) = 1/n * sum(A.P.), in which n represents the total number of classes.

Hence, the Mean Average Precision Percentage of the model shows the effectiveness of the trained model.

Figure 7 is shown below with mAP%, Current Average Loss Max Batches etc.

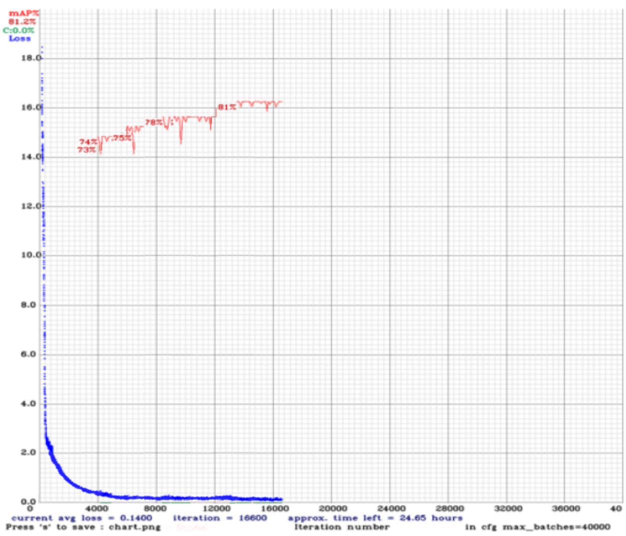


Figure 7. Mean Average Precision Chart

The loss/mAP chart can be used to track the progress of the model while it is being trained. The blue curve represents the training loss as well as the error on the training dataset (specifically Complete

Intersection-Over-Union or CIoU loss for YOLOv4). The 50% Intersection-over-Union threshold mean average precision is represented by the red line (mAP@0.5), which determines whether our model generalizes well on a previously unseen validation set or dataset (Allocated 40,000 max batches instead of 64,000 max batches to see quick results). We got an accuracy of 83 on the new dataset. Although the accuracy of 83 was promising, it was difficult to keep this rate with our limited resources, as we had to terminate the training process after three weeks on Google's GPU after 16,600 iterations and at 83 mAP.

4.2 Detailed Performance Metrics

Table 2 shows the performance metrics of our models.

Table 2. Performance Metrics

Evaluation Metrics	Average values
F1 Score	1.00
Precision	0.98
Recall	0.99
Average IOU	0.83

The graphical representation of performance metrics is shown in Figure 8.

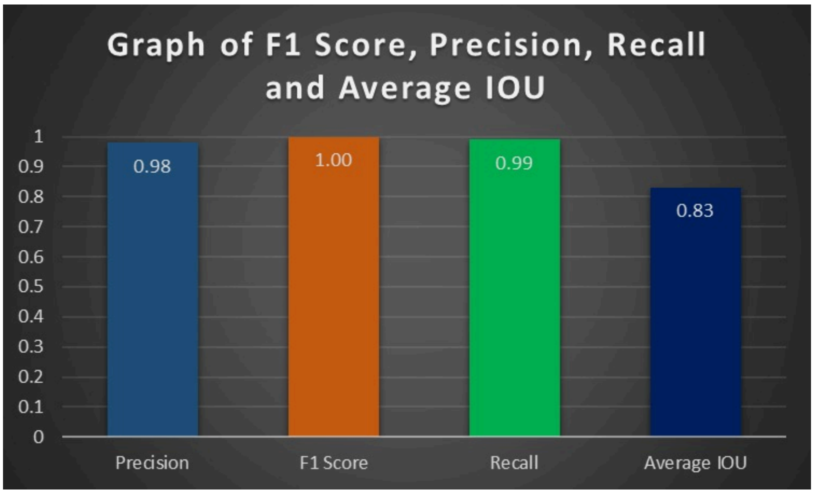


Figure 8. Graph of Performance Metrics

4.2.1 Precision:

Out of all the positives predicted, the percentage is truly positive in equation 1.

$$Precision = \frac{TP}{TP+TN} \tag{2}$$

The precision value lies between 0 and 1. We got a precision metric 0.98 for our model as shown in table 2.

4.2.2 Recall:

The proportion of the total positives are anticipated to be positive. The T.P.R. (true positive rate) is equivalent to this.

$$Recall = \frac{TP}{TP+FN} \tag{3}$$

We got a Recall value of 0.99 as shown in table 2.

4.2.3 F1 Score:

It refers to the recall and precision harmonic mean. It accounts for both false positives and false negatives. As a result, it works well with an unbalanced dataset. We got F1 score value 1 for our proposed model as shown in Table 4.1. F1 Score calculation is made through given equation.

$$F1\ score = \frac{2}{\frac{1}{Precision} + \frac{1}{Recall}}$$

$$F1\ score = \frac{2*(\text{Precision}*\text{Recall})}{(\text{Precision}+\text{Recall})} \tag{3}$$

Precision as well as recall are given the same weight in the F1 score.

4.2.4 I.O.U.:

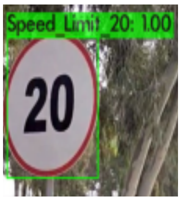
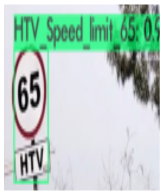
An assessment metric is an Intersection over Union. IoU may be employed to assess any method that outputs projected bounding boxes. The average IoU for our model is 0.83 as shown in table 2.

4.3 Prohibitory Traffic Signs Detection Using YoloV4 Model

We have shown the different traffic signs that our trained model predicted. The model successfully predicted the images. We have distributed our predicted table such that the same type of prohibitory, speed limit, and stop signs are placed in respective groups and discussed the same type of class objects collectively. **Table 3 shows the class object, the detected traffic sign, and the** predicted time of each prohibitory traffic sign.

Table 3. System Prediction Table

Class/Object	Sign	Predicted Time in milli-seconds	Class/Object	Sign	Predicted Time in milli-seconds
No_Entry		**32.163**	**Veichles_height_more_than_18_feet_prohibited**		**32.096**
No_Cycle		**32.141**	**No_Veichles_width_more_than_3.5m**		**32.233**

No_Over-speed		32.087	No_Over-take		32.24
No_Parking		32.127	No_Auto_Ri ckshaw		31.999
Right_Turn _Prohibited		32.064	Use_of_Mo-bile_Prohib-ited		32.053
Slow		32.027	Dead_Slow		32.076
Speed_Limi t_20		32.036	HTV_Speed _limit_65		32.017
Speed_limi t_80		32.086	Speed_Limit _40		32.195

The system prediction table 3 is shown for a few classes. Our trained model also detected prohibitory signs like no use of mobile, no entry, no parking, no right or left turn, no overspeeding, and no overtaking prohibitory traffic signs. As we have included the Stop Signs and a few speed limit signs in our dataset, we have also trained on them, and the last portion of our table shows all signs related to speed limits. Increasing the speed from the mentioned speed limit is also prohibited; these are essential to these types of traffic signs; therefore, we have included them as well. Moreover, it can be seen in the table how effectively YoloV4 detected all the signs with high-speed detection. Our training model detected the objects with high accuracy. The results can be utilized for the development of real-world applications for Pakistan, such as self-sufficient driving, traffic surveillance, and driver safety that fall under the purview of Deep Learning and Machine Learning to perform:

- Traffic Regulatory Authorities
- Self-Driven Auto-mobiles manufacturers
- For Researcher's working in the same area

5. Discussion

We have compared our proposed model's results with the other models, as shown in Comparison Table 4. Our model achieved the best precision on the new dataset and showed the fast detections required, as detecting signs in real time is quite challenging. For this purpose, our proposed technique uses YoloV4.

Table 4: Comparison Table

Ref	Technique	Accuracy
[30]	Multiscale ResNeSt101	78.9
[31]	YoloV4	79
[32]	MobileNet V2 (CNN)	76.7
[33]	YoloV5+CNN	66
Proposed	YoloV4	83

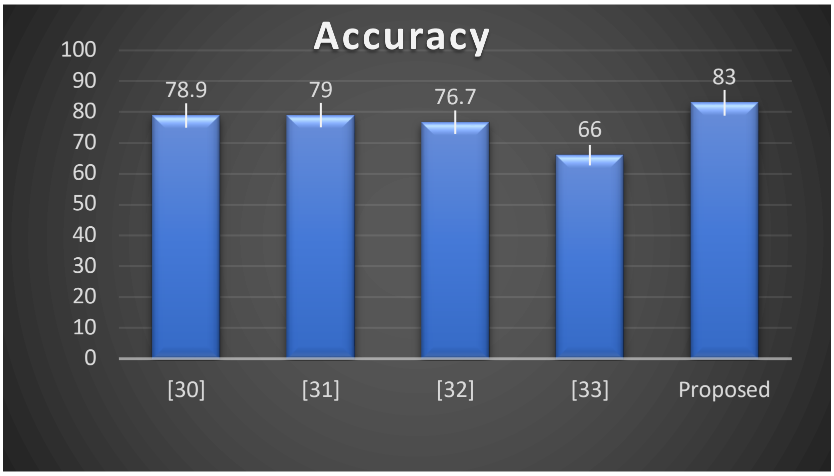


Figure 9. Comparison Chart

As 83% Accuracy, we got on the newly created dataset of Prohibitory Traffic Signs of Pakistan and the proposed model, which is much faster, more accurate, and more efficient. Our model shows a precision of 0.98 and an accuracy of 83 in terms of mean average precision for our prohibitory traffic signs dataset.

Finally, based on all the experiment results, the best weight will provide the highest accuracy. Hence, the final weight and last weight will produce the same accuracy. The fastest time detection is the last weight. In future work, our plan is to extend the dataset focus not only on Pakistan's prohibitory signs but also on all of Pakistan's traffic signs, including Traffic Signals. Other advanced versions of Yolo are also available, like Yolo5 and Yolo7 etc., and we will use them in the future to increase the model's mean average precision easily. Instead of only prohibitory traffic signs, all traffic signs of Pakistan during daytime and nighttime can be included in the future. This will be proved as one step towards growing technology and opening a new era.

6. Conclusions

This research includes machine learning techniques to detect the prohibitory traffic signs in Pakistan. Detection of these signs is useful for advancing technology, can be used for drivers' assistance, or can be used in unmanned vehicles / self-driven vehicles. The optimal weight will deliver maximum accuracy depending on the experiment findings. The accuracy will be the same whether we use the final or last weight. The final weight has the quickest time detection. In subsequent work, we intend to expand the dataset focus to include Pakistan's prohibited signs and all of Pakistan's traffic signs. Instead of only prohibitory traffic signs, all traffic signs of Pakistan during daytime and nighttime can be included in the future. It improves the model and makes it more effective in the future. All traffic signs can be included, and updated versions of YoloV5 and YoloV7 can be used. Moreover, Traffic signals and road markings can be included as well.

References

1. Zhu, Y., W.Q.J.M.T. Yan, and Applications, Traffic sign recognition based on deep learning. 2022. 81(13): p. 17779-17791.
2. Lee, S., et al. Vpgnet: Vanishing point guided network for lane and road marking detection and recognition. in Proceedings of the IEEE international conference on computer vision. 2017.
3. Chhabra, R., S. Verma, and C.R. Krishna. A survey on driver behavior detection techniques for intelligent transportation systems. in 2017 7th International Conference on Cloud Computing, Data Science & Engineering-Confluence. 2017. IEEE.
4. Mihalj, T., et al., Road Infrastructure Challenges Faced by Automated Driving: A Review. 2022. 12(7): p. 3477.
5. Zhang, J., et al., A shallow network with combined pooling for fast traffic sign recognition. 2017. 8(2): p. 45.
6. Narvaez, J.L.M., Adaptation of a Deep Learning Algorithm for Traffic Sign Detection. 2019, The University of Western Ontario (Canada).
7. Akinlar, C. and C.J.P.R. Topal, EDCircles: A real-time circle detector with a false detection control. 2013. 46(3): p. 725-740.
8. Yang, Y., et al., Towards real-time traffic sign detection and classification. 2015. 17(7): p. 2022-2031.
9. Zabihi, S., et al. Detection and recognition of traffic signs inside the attentional visual field of drivers. in 2017 IEEE Intelligent Vehicles Symposium (IV). 2017. IEEE.
10. Zhang, J., et al., A real-time Chinese traffic sign detection algorithm based on modified YOLOv2. 2017. 10(4): p. 127.
11. Müller, J. and K. Dietmayer. Detecting traffic lights by single shot detection. in 2018 21st International Conference on Intelligent Transportation Systems (I.T.S.C.). 2018. IEEE.
12. Wan, J., et al., An efficient small traffic sign detection method based on YOLOv3. 2021. 93: p. 899-911.
13. Song, Y., et al., A three-stage real-time detector for traffic signs in large panoramas. 2019. 5: p. 403-416.
14. Lee, H.S. and K.J.I.T.o.I.T.S. Kim, Simultaneous traffic sign detection and boundary estimation using convolutional neural network. 2018. 19(5): p. 1652-1663.
15. Ibrahem, H., A. Salem, and H.S. Kang. Weakly supervised traffic sign detection in real time using single CNN architecture for multiple purposes. in 2020 IEEE International Conference on Consumer Electronics (I.C.C.E.). 2020. IEEE.
16. Zhang, J., et al., A cascaded R-CNN with multi-scale attention and imbalanced samples for traffic sign detection. 2020. 8: p. 29742-29754.
17. Çetiner, H. Classification of Traffic Signs with Convolutional Neural Network. in International Conference on Frontiers in Academic Research. 2023.
18. Ellahyani, A., M. El Ansari, and I.J.A.S.C. El Jaafari, Traffic sign detection and recognition based on random forests. 2016. 46: p. 805-815.
19. Yıldız, G. and B. Dizdaroğlu. Traffic sign detection via color and shape-based approach. in 2019 1st International Informatics and Software Engineering Conference (U.B.M.Y.K.). 2019. IEEE.
20. Chen, T. and S.J.I.T.o.V.T. Lu, Accurate and efficient traffic sign detection using discriminative adaboost and support vector regression. 2015. 65(6): p. 4006-4015.
21. Le, T.T., et al. Real Time Traffic Sign Detection Using Color and Shape-Based Features. in A.C.I.I.D.S. (2). 2010.
22. Nguyen, H.J.J.o.S., Fast traffic sign detection approach based on lightweight network and multilayer proposal network. 2020. 2020.
23. Mnih, V., N. Heess, and A.J.A.i.n.i.p.s. Graves, Recurrent models of visual attention. 2014. 27.
24. Yuan, Y., Z. Xiong, and Q.J.I.t.o.i.p. Wang, VSSA-NET: Vertical spatial sequence attention network for traffic sign detection. 2019. 28(7): p. 3423-3434.
25. Tian, Y., et al., Traffic sign detection using a multi-scale recurrent attention network. 2019. 20(12): p. 4466-4475.

26. Kastner, R., et al. Attention-based traffic sign recognition with an array of weak classifiers. in 2010 IEEE Intelligent Vehicles Symposium. 2010. IEEE.
27. You, S., et al., Traffic sign detection method based on improved S.S.D. 2020. 11(10): p. 475.
28. Yang, W., et al. A vehicle real-time detection algorithm based on YOLOv2 framework. in Real-Time Image and Video Prcessing 2018. 2018. S.P.I.E.
29. Zhang, H., et al., Real-time detection method for small traffic signs based on Yolov3. 2020. 8: p. 64145-64156.
30. Liang, T., et al., Traffic sign detection via improved sparse R-CNN for autonomous vehicles. 2022. 2022: p. 1-16.
31. Li, Y., et al., Attention-YOLOV4: a real-time and high-accurate traffic sign detection algorithm. 2023. 82(5): p. 7567-7582.
32. Naimi, H., T. Akilan, and M.A. Khalid. Fast traffic sign and light detection using deep learning for automotive applications. in 2021 IEEE Western New York Image and Signal Processing Workshop (W.N.Y.I.S.P.W.). 2021. IEEE.
33. Wang, J., et al., Improved YOLOv5 network for real-time multi-scale traffic sign detection. 2023. 35(10): p. 7853-7865.