Deep Learning Technique for Object Detection from Panoramic Video Frames

Kashika P. H. and Rekha B. Venkatapur

Abstract—**The objective is to train a YOLOv3 algorithm with necessary enhancements to accurately detect the safety helmet from the video frames which can be used to find the people working in the construction site or riding bike without helmet in the traffic. During the recent past the dominance of deep learning algorithms increased in solving problems in the field of computer vision especially for image classification and object detection. The available algorithms can be divided in to two major categories, 2-stage detection (based on region proposal network) and 1- stage detection. For real time detection of objects from surveillance videos, YOLO based detection is considered to be more suitable approach due to its high speed detection. The loss function and other factors pose few challenges and limitations as the detection accuracy degrades especially when the training dataset is unbalanced. The loss function is modified to overcome the effect of different scale of the object of the same category. This paper utilizes the DarkNet-53 approach, a 53 layered deep convolutional neural network to extract features. The proposed YOLOv3 based safety helmet detector especially the feature extractor is trained on a custom built dataset. The detector achieves a higher detection speed and accuracy with higher generalization ability. The performance of the trained model is tested on panoramic images generated by stitching multiple video frames captured from the surveillance videos. The results demonstrate that the trained model can be utilized to detect the safety helmets from the video frames in real time. The presented approach will be an effective alternate solution for detecting the safety helmets and enhance the safety practices at construction site and road traffic.**

Index Terms—**Region proposal network, real time detection, one stage detection, Darknet-53, safety helmet detection, panoramic images.**

I. Introduction

Object detection is an interesting and more challenging problem in the field of computer vision, which is being explored by many researchers. The objective of object detection is identifying and localizing the object (single or multiple instances) in an image or sequence of images / video. In specific the object detection algorithm output bounding box and a corresponding class label for each of the objects detected within the given input image which is useful particularly in surveillance applications. Deep learning algorithms based object localization/ detection approaches are gaining more importance and they are more accurate when compared to the conventional machine learning based approach. The deep learning algorithms generated more abstract high level feature representation for the given input based on the low- level features [1].

In general the object detectors based on deep neural architectures can be classified as either one stage or two stage detectors. The two stage approach includes methods like R-CNN, Faster R-CNN [2], and Fast R-CNN [3] which uses a region proposal network in the first stage to generate candidate bounding boxes. The second stage contains a fully convolutional neural network which extracts features from all the selected candidate bounding boxes. Later these features are used for classification and bounding box regression. The other approach is termed as one stage detectors or single shot detectors which include algorithms like YOLO, SSD [5], and Squeeze Det [4]. The one stage detection methods are more suited for real time applications as their inference speed is higher. They use regression methods to identify target locations and utilize anchors with fixed-position. The anchors help to constrain the aspect ratio and prevent detection of irregular shapes. The detection accuracy will be affected by the number of anchors (which is a hyper parameter). The non max suppression used in the post processing stage to resolve the overlapped detection is prone to error in many situations [6].

The region based CNN are the preliminary approaches which utilize deep CNN to solve the object detection problem. Based on R-CNN, the initial method proposed for object detection, Fast R-CNN, and Faster R-CNN were proposed with an aim of reducing the training time and increasing the mean average precision. Even though these methods yield high detection accuracy they suffer due to complex architecture and more time consuming training process.

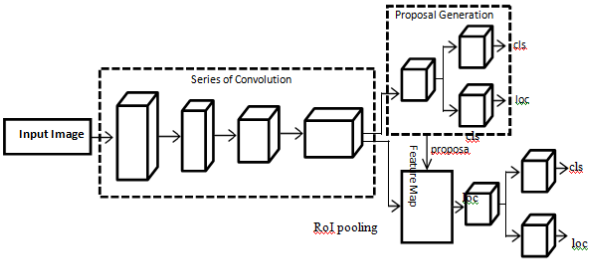


Fig. 1. Schematic view of two-stage detector.

The one shot detectors directly detect the location information of the objects and the respective class probabilities from the given input image with a full convolutional neural network. They don't require an initial region proposal network and a post classification network. The unified pipelined architecture is simple and capable of detecting objects quickly.

Manuscript received September 19, 2021; revised December 14, 2021.

Kashika P. H. and Rekha B. Venkatapur are with K. S. Institute of Technology, Bengaluru, India (e-mail: kashikaph23@gmail.com, rekhabvenkatapur@ksit.edu.in).

DOI: 10.7763/IJCTE.2022.V14.1306

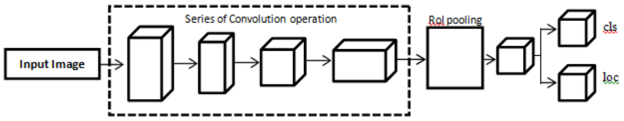


Fig. 2. Schematic view of one-stage detector.

Recently a new approach for object detection called Center Net has been proposed which predicts the centre of the object and their respective attributes instead of detecting objects by generating and classifying region proposals. The YOLO based object detection has numerous advantages when compared to the other parallel approaches. During the inference phase, it looks that the whole image and the predictions are made based on the global information available in the image. This approach makes them faster (~1000x faster when compared to R-CNN and ~100x faster when compared with Fast R-CNN). In the YOLOv3 architecture, the feature extractor network is pipelined in front with a residual block which deepens the network and helps to overcome the vanishing gradient issue. The ability of the multi scale object detection is achieved by including five down sampling blocks at the end. Due to the scale diversity the feature maps includes both general and semantic information.

The present research paper focuses on the utilization and enhancement of YOLO, an unified pipeline framework. For improving the convergence speed and reducing the over fitting issues YOLO algorithm uses batch normalization and predicts the bounding boxes using anchors to improve the sensitivity. In case of YOLO based object detection the bounding box regression is an important step and existing methods (detailed explanation given in literature survey) use the ℓn-norm loss. The loss function [7] and the model performance evaluation metric (IoU) doesn't have any correlation. This paper adopts IoU loss and generalized IoU to exploit the benefits of using IoU as the evaluation metric. The observation from the test results showed an improvement in the model efficiency to handle critical scenarios when the object of interest in the test image is stained, occluded, or found with low resolution.

II. Literature Survey

An object detection approach using Deep Learning Algorithms has four steps including pre-processing, feature extraction, instance classification & localization and post processing. Initially the raw images available in the dataset cannot be supplied to the model for training. Images has to be resized to match the input size of the pre trained network and enhance them by adjusting the brightness, and contrast or standardizing the color. Data Augmentation can be included to synthesis more samples by flipping, rotating, cropping or by adding noise to the original images. In addition using GAN [8] (generative adversarial networks) more synthetic images can be generated.

The YOLOv3 algorithm [9] has balance between the speed and accuracy by including the residual block [10], feature pyramid network [11], and improved loss function. Also, in feature pyramid networks the lateral connection method is adopted to combine the down-sampling and up-sampling feature maps. The final prediction is made from the merged layer.

These changes helped the model to detect objects even in the complex background and scenarios. To detect safety helmet in real time from the surveillance videos of the construction site an optimal approach is explored in [12] based on a deep learning networks especially using SSD-MobileNet algorithm. SSD-MobileNet relies on the convolutional neural network whose detection speed is faster as compared to the speed of YOLO algorithm, and hence suggested for real time utility. This network model is more suited for small input images. The R-CNN based networks are more accurate but their detection speed is slower and for this reason in majority of real time detection application either YOLO or SSD based detectors are modelled.

In general SSD algorithms suffer from poor accuracy in detection when the object of interest is small but the advantage is that when the model is trained on a feature pyramid instead of training on images the accuracy tend to be increased [13]. In other words the model has been trained using *n*dimensional feature map rather than from feature map extracted from single image. Feature maps are extracted at each scale of the image pyramid and the corresponding loss is backpropagated during the training time. The anchor boxes are estimated at each scale using a unique procedure. Similar to F-RCNN the SSD approach generates six varieties of anchor boxes by following a approach using the aspect ratios and scales. The accuracy of the detection can be improved as the network trained on an image pyramid (containing image at different scale).

A novel detection network based on the YOLOv3 algorithm is presented in [14] which uses the Darknet53 as the backbone network. The backbone network is improved by reducing the calculation cost and speed by using the cross stage partial network (CSPnet). Multi-scale object detection is combined with a top-down and bottom-up fusion of features for enhancing the features. Image processing based techniques are employed in [15] to detect the safety helmets in surveillance video frames. For each video frames the pedestrian classifier is used for detecting the presence of workers and from the detected head regions of the video frames, using transformation of color space and discrimination of color feature the presence of safety helmet is detected. Further to enhance the detection accuracy the HSV transformation and adaptive threshold selection [16] are adopted. The accurate detection of human in real time is carried out using C4 classifier which uses contour cues for its detection.

For real time object detection YOLO algorithm has been used extensively in numerous research works and some of them have suggested few ways to improve the performance of YOLOv3 algorithm. In [17], the YOLOv3 algorithm has been improved by redefining the loss function using a Gaussian parameter.

In few literatures the feature fusion and detection at multi-layer level has been integrated to enhance the efficiency of the object detection at multiple scales.

In [18], a novel Intersection over Union (IoU) loss function for bounding box prediction is introduced, which regresses the four bounds of a predicted box as a whole unit. By taking the advantages of IoU loss and deep fully convolutional networks, the UnitBox is introduced, which performs accurate and efficient localization, and shows robust to objects of varied shapes and scales, and converges fast. An UnitBox on face detection task is applied and achieved the best performance on face detection masks.

In [19], the weaknesses of IoU by introducing a generalized version as both a new loss and a new metric is addressed. By incorporating this generalized IoU (GIoU) as a loss into the state-of-the art object detection frameworks, a consistent improvement on their performance using both the standard, IoU based, and new, GIoU based, performance measures on popular object detection benchmarks such as PASCAL VOC and MS COCO are introduced.

III. Object Detection Network Architecture

YOLOv3 is being developed by redefining and improving certain functionalities of YOLOv2. Independent logistic classifiers have been used in YOLOv3 for multi-label classification. YOLOv3 uses feature maps in three different scales to detect objects from complex images having objects labelled in overlapped fashion. The last convolution layer of the detector module generates a three dimensional tensor which contain the class predictions, confidence score, and the bounding boxes.

YOLO v3 has the ability to detect object at multiple scale and can extract optimal features required for object detection using strong feature extraction network. The loss function is modified when compared to the previous versions which help the model to detect objects at different scale. YOLO v3 can be adopted for real time detection tasks. The model has two main components Feature Extraction network and object detection network (both are tuned for multi-scale). At the first stage of detection the feature extractor module generated feature embeddings at three different scales and in the second stage these features are sent to the detector module for obtaining and bounding boxes and class information (presented in Fig. 3).

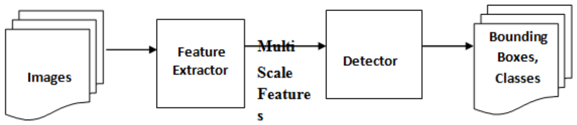
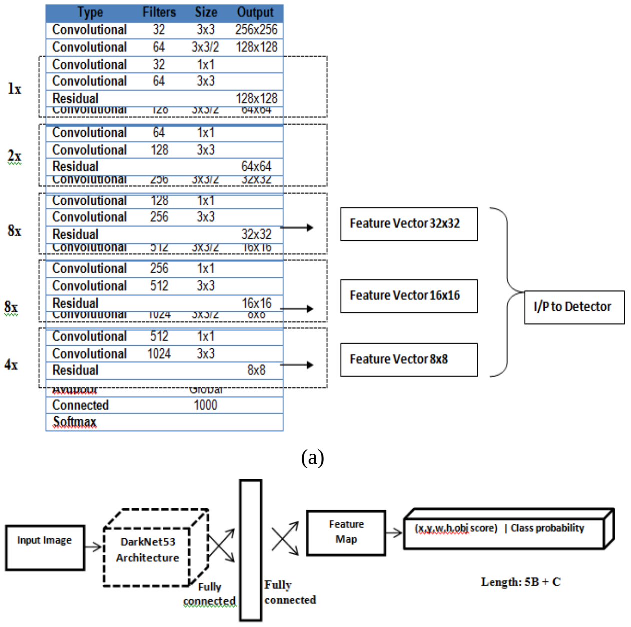


Fig. 3. Block diagram of detection process.

The feature extractor network of YOLO v3 is called Darknet-53, when compared to the early architecture of the Darknet network used in the previous versions of the YOLO, the image classification network has made a lot of progress instead of being more deeper alone. The idea of skip connections has been introduced in the ResNet model which helps to avoid vanishing gradient problem when propagating the activation through deeper layers (in total 53 layers details presented in Fig. 4a). When the darknet is used for multi-class classification, an average pooling layer and a soft max activation will be appended at the tail end. As the objective is to utilize this network to generate multi scale features for object detection a detection head is appended. For multi scale object detection the features maps from the last three residual blocks in the architecture were used.



Type	Filters	Size	Output
Convolutional	32	3x3	256x256
Convolutional	64	3x3/2	128x128
Convolutional	32	1x1	
Convolutional	64	3x3	
Residual			128x128
Convolutional	128	3x3/2	64x64
Convolutional	64	1x1	
Convolutional	128	3x3	
Residual			64x64
Convolutional	256	3x3/2	32x32
Convolutional	128	1x1	
Convolutional	256	3x3	
Residual			32x32
Convolutional	512	3x3/2	16x16
Convolutional	256	1x1	
Convolutional	512	3x3	
Residual			16x16
Convolutional	1024	3x3/2	8x8
Convolutional	512	1x1	
Convolutional	1024	3x3	
Residual			8x8
Avgpool		Global	
Connected		1000	
Softmax			

(a)

(b)

Fig. 4. (a) Schematic View of multi-scale feature extraction. (b) High Level Architecture of YOLO v3.

The YOLO v3 model attempts to detect the bounding boxes containing the region of interested objects along with the probability and class of the objects. For detecting the objects, the network splits the image into ***SxS*** grid and from each grid, B number of bounding boxes along with the C class probabilities of object which fall inside the grid cells will be the output. The ***G*** number of bounding boxes ***B*** depends upon the number of anchors used and each bounding box helps to detect a particular type of object. For a given input image of size ***SxS,*** the model generates a 3D tensor output [***S***, ***S***, ***B****(5+***C***)]. The high level architecture of the YOLOv3 is presented in the Fig. 4b.

Earlier object detection methods followed a sliding window based detection process where a classifier is used to detect the presence of object at each window. In contrast, ConvNet based object detection approach followed a single shot method. As the object can be in any shape and can be marked using a rectangular bounding box, the anchor boxes have been used. For each scale of detection the YOLO uses 03 anchor boxes for total of 09 anchor boxes.

When multiple bounding boxes are generated for a particular object instance by the YOLO model then the Non-max suppression can be used to select the best bounding box out of a set of overlapping boxes. The output from the model includes many irrelevant or redundant bounding boxes which are to be filtered and removed. In the first step the bounding boxes with high probability is retained and with low are pruned. Even after initial pruning of the boxes based on the probability there may be multiple boxes for each object detected. By using the concept of Intersection over Union (IoU), highly overlapping bounding boxes are suppressed and one box per object instance is generated.

In convention, the deep learning algorithms when used for object detection have four major steps as follows.

i. pre-processing., ii. extracting features., iii. classification and localization., iv. post-processing. The raw images are pre-processed in the initial stage by performing resizing, and by enhancing brightness, color, and contrast. Data augmentation techniques are adopted to synthesize more images and enrich the diversity of the object detection process.

In the YOLO based object detection approach the given image is split in to SxS grid cells and the objectness score and the location of the bounding box for B objects in each grid cell were estimated. The objectness score can be expressed mathematically as follows:

$$C_j^i = P\ (object) * IoU(truth, pred) \qquad (1)$$

where *C* denotes the objectness score, *j* and *i* represents the bounding box number and the grid cell respectively. Objectness loss is estimated based on the binary cross entropy loss function which can be expressed as follows:

$$E_1 = \sum_{i=0}^{S^2}\sum_{j=0}^{B} W_{ij}{}^{obj}\ [C_j^i \log(C_j^i) - (1 - C_j^i)\log(1 - C_j^i)] \quad (2)$$

where S^2 and B denotes the total number of grid cells and bounding boxes respectively. $C\wedge\ j$ is the predicted objectness score. The object detector finds the position of each object instance and gives four predictions as output t_x, t_y, t_w, t_h assuming the c_x, and c_y as the offset of the grid cell from the top left corner of the input image. The centre of the predicted bounding box for the object is at b_x, by offset from the left top corner. The values of the above mentioned coordinates are computed as follows:

$$b_{x= \sigma(t_x) + c_x}$$
$$b_{y= \sigma(t_y) + c_y} \qquad (3)$$

The height and width of the bounding boxes can be estimated as:

$$b_{w=pw\,.\,e^{tw}}$$
$$b_{h=ph\,.\,e^{th}} \qquad (4)$$

where, pw and ph are the width and height of the prior bounding box which is calculated by dimensional clustering. The truth value ($t\wedge x$,$t\wedge y$, $t\wedge w$, $t\wedge h$) of the four associated parameters with the ground truth box namely (g_x , g_y , g_w , g_h) whose corresponding predicted parameters are (b_x , b_y , b_w , b_h) can be estimated using the following mathematical expression:

$$\sigma(\hat{t}_x) = g_{x - c_x}$$
$$\sigma(\hat{t}_y) = g_{y - c_y}$$
$$\hat{t}_w = \log(\frac{g_w}{p_w})$$
$$\hat{t}_h = \log(\frac{g_h}{p_h}) \qquad (5)$$

The total loss calculated includes the squared loss of the coordinate prediction which is expressed mathematically as follows and the classification defined in Eq. 7:

$$E_2 = \sum_{i=0}^{S^2}\sum_{j=0}^{B} W_{ij}{}^{obj}\left[\left(\sigma(t_x)_i^j - \sigma\,(\hat{t}_x)_i^j\right)^2 + \left(\sigma(t_y)_i^j - \sigma\,(\hat{t}_y)_i^j\right)^2\right] +$$
$$\sum_{i=0}^{S^2}\sum_{j=0}^{B} W_{ij}{}^{obj}\left[\left(\sigma(t_w)_i^j - \sigma\,(\hat{t}_w)_i^j\right)^2 + \left(\sigma(t_h)_i^j - \sigma\,(\hat{t}_h)_i^j\right)^2\right] \qquad (6)$$

$$E_3 = \sum_{i=0}^{S^2}\sum_{j=0}^{B} W_{ij}{}^{obj}\ [\hat{p}_i(c).\,log p_i(c) + (1 - \hat{p}_i(c)\,).\log(1 - \hat{p}_i(c))](7)$$

where $p\wedge i\ (c)$ represents the predicted conditional class probability of the class c type object in the grid cell.

During the training the objective is to optimize the following multi-part loss function:

$$Loss = E_1 + E_2 + E_3$$

IV. Experiments and Results

During the training process, the input images were resized to 416x416 pixels. The batch size was fixed as 4 due to the memory constraints of the GPU (Nvidia T4/ 16GB RAM/ 1.59GHz memory clock). The model is trained with different number of epochs and the evaluation metric and the model loss are estimated for each trial. The learning rate controls the change in weight after each step in the training process in response to the estimated error. Initial value of the learning rate and the rate of change of the learning rate will have a major impact on the training process. When the value of the learning rate is small the training time will be long and higher learning rate may yield sub-optimal weight values. Adam optimizer combines the goodness of AdaDelta and RMSprop optimizers. The learning rate is penalized when the weights are updated frequently and the learning rate will high when the weights are not updated frequently. The initial learning rate was fixed as 0.001. When the learning saturates learning rate reduction scheme was adopted to overcome the training plateau. To reduce overfitting the early stopping was configured. The value of different hyperparameters configured during the model training is tabulated in Table I.

TABLE I: List of Model Hyper Parameters

Parameter	**Value**
Initial Learning Rate (LR)	0.001
Batch size	4
Optimizer	Adam
Epochs	5 – 50
Learning Rate Scheduler	LR increased by 0.01 after every 10 epochs.
L2 regularization factor	0.0005
Penalty threshold	0.5

The object detection model is implemented using Keras software library which uses Tensorflow as the backend. The pre-trained weights of the YOLO v3 with the COCO dataset are used to initialize the weights and other parameter values. Finally the weights of the safety helmet detection model (YOLOv3's all layers) are trained during the training process. Out of total 5700 images available using random split 4400 images are used for training the model and remaining 1300 images were used for testing the model performance. The experiments does not use any specific validation set for adjusting the performance of the model and analyze the model performance initially. The generalizing capacity of the trained model is analyzed using the test set. During the training process the metrics including mean Average Precision (mAP), and the loss function for each epoch is recorded.

To understand the importance of mAP in case of object detection it is better to have an idea regarding the Intersection over Union (IoU). The IoU can be defined as the ratio between the area of intersection and area of union of the predicted and ground truth bounding box.

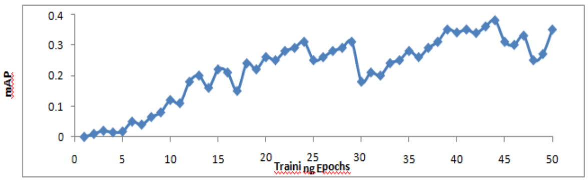


Fig. 5. Mean average precision.

Fig. 5 presents the plot of the changes in the mean Average Precision which shows an upward trend and the plot is not raising steadily. After half of the training process the value of the mean average precision of the helmet detector is around 32.82%. In few of the real time object detection approaches, the generalized IoU loss has been adopted to exploit the benefits of the IoU metric. The ℓn-norm based loss estimation will not be suitable when the IoU is used as the evaluation metric [18]. Hence the IoU based bounding box regression loss is followed [19]. Still the IoU based loss function can be beneficial when the bounding boxes are overlapping and the model will not learn when the bounding boxes are non-overlapping. To overcome such issue Generalized IoU loss is used to improve the detection accuracy by adding penalty term to the loss. It is mathematically expressed as follows:

$$L_{GIoU} = 1 - IoU + \frac{|C - B \cup B^{gt}|}{|C|} \tag{8}$$

where C denotes the smallest box that contain predicted and ground truth box.

$$MSE = \frac{1}{n} \sum_{i=1}^{n} \left(Y_i - \hat{Y}_i\right)^2 \tag{9}$$

When compared to the standard loss based on the (MSE) Mean Square Error, the GIoU based loss estimation yields better results in terms of average precision calculated by taking average of mAP when the IoU threshold is varied between 0.5 to 0.95. The results presented in Table II. shows the improvement indetection when GIoU is used.

TABLE II: COMPARISON BETWEEN MSE AND GIOU LOSS

Loss	IoU
MSE	**0.361**
LGIoU	**0.424**
% improvement	**14.8**

Fig. 6 presents the total loss observed during the training phase where in the plot the loss value decreases slowly at the initial period of the training and saturates at the end of the training. The loss function helps to understand the training process, but it does not convey regarding the detection accuracy. The saturation of the loss function can be interpreted as the end of the training process. When the changes in the loss values are smaller the training is considered to be in progress.

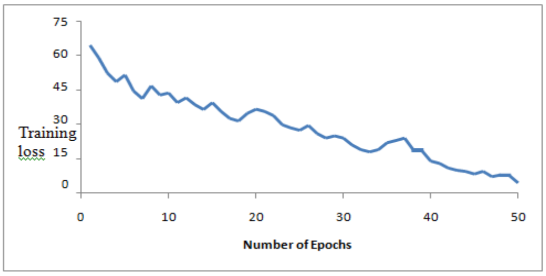
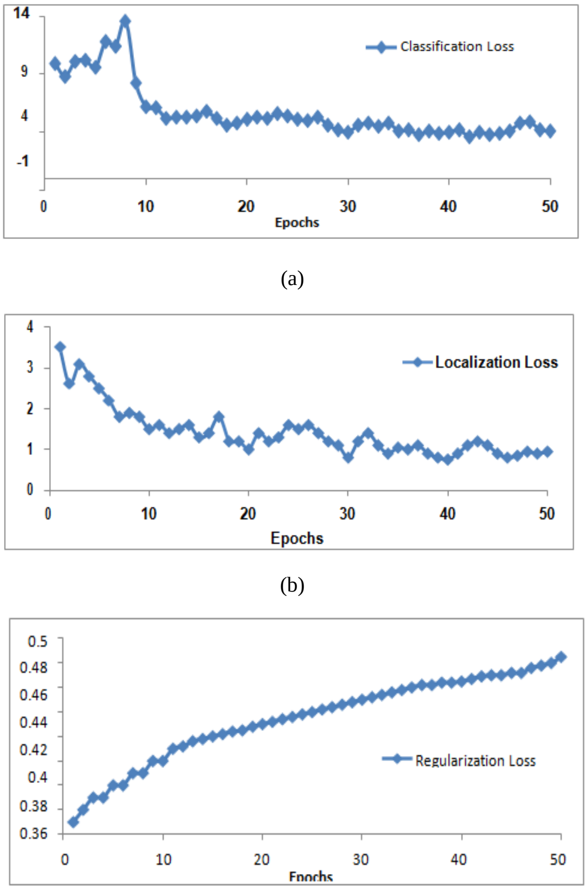


Fig. 6. Total training loss.

Fig. 7(a) to 7(c) present the variation of the classification, localization and regularization loss individually after each epoch. The classification loss initially decreases steadily and then decrease rapidly after 30% of the training process. The localization loss helps to ensure the minimization of the error in predicting the boundary box locations and its width and height. The absolute error estimated for a large and a small box should not be weighted equally and to address this issue the square root of the error in width and height are predicted in the loss estimation. Regularization loss helps to approximate the additional loss induced by the regularization term. Regularization term is inserted with an objective to aid the optimizer to generalize better and hence a regularization term is added to the loss function. This function modifies the overall loss to force the optimizer to converge towards desired directions.

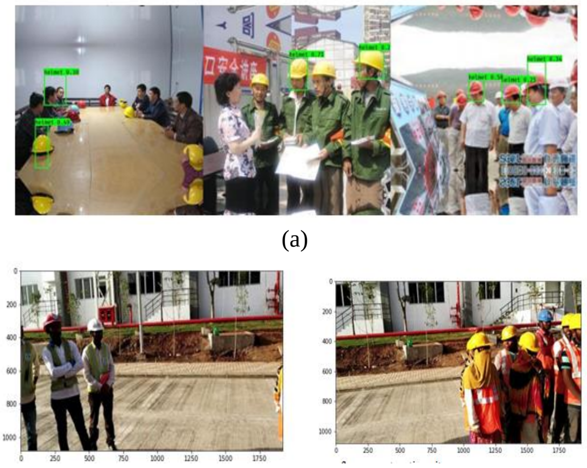


(a)

(b)

(c)

Fig. 7. (a) Classification loss. (b) Localization loss. (c) Regularization loss.



(a)

(b) (c)

Fig. 8. (a) Results of safety helmet detection in test video frames. (b)&(c). Video frames from construction site.

The results of the safety helmet detection in the surveillance video frames are shown in Fig. 8(a), 8(b) and 8(c) and it is to be noted from the figures that when the object of interest lies completely out of focus or overlapping with other objects, the model struggles to identify it.

V. Panoramic Image Generation

The panoramic images are obtained by stitching multiple images extracted from the surveillance video frames. Initially matching points between the images to be stitched together were detected. The SIFT features were used for finding the matching feature points where SIFT gives a robust descriptions of the feature points that can yield lower error when compared to Harris corner points based matching points detection approach. But the SIFT based method also introduces few erroneous points and RANSAC algorithm was employed to remove those erroneous points based on the high dimensional feature representation generated by the SIFT algorithm. The result of the safety helmet detection in panoramic image is presented in Fig. 10, and it is observed that detection in case of blurred and overlapped helmets were not successful. Future attempts will focus on generating synthetic image samples for robust training of the model so that these issues are resolved. The result of the feature point matching is shown in Fig. 9 and the erroneous points were corrected using RANSAC algorithm. The result of the safety helmet detection in panoramic image is demonstrated in Fig. 10.and it is observed from the figure that detection in the case of blurred and overlapped helmets not successful. Future attempts will focus on generating synthtic image samples for robust training of the model so that these issues are resolved.

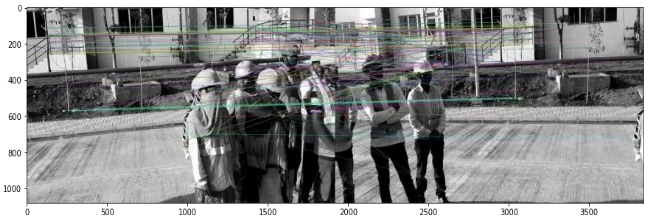


Fig. 9. Result of Feature points matching using SIFT.

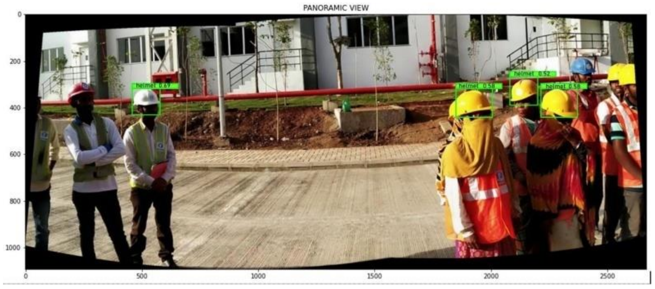


Fig. 10. Panoramic stitched image (borders padded with zero).

VI. Conclusions

Thus, in the present paper it is attempted to develop an efficient safety helmet detector based on the YOLO v3 algorithm which uses Darknet-53 network for feature extraction. A dataset of 5700 images containing several safety helmets is considered and split into two parts for training and testing the model. After training the model for several number of epochs the mean average precision (mAP) of the detector are observed to be stable and the helmet detection was accurate. The regular MSE or binary cross entropy based loss is replaced with the generalized IOU loss and the average precision for different threshold values of the IoU was measured. The results demonstrate that the trained model can be utilized to detect the safety helmets from the video frames in real time. The presented approach will be an effective alternate solution for detecting the safety helmets and enhance the safety practices at construction site, and road traffic.

In future, some advanced deep learning techniques can be applied for improving the overall accuracy of the system. Further, a 3-axis accelerometer sensor can be attached to safety helmet to develop and identify whether the helmet is being properly worn or not worn.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

Kashika P. H. has carried out research under the guidance of Dr. Rekha B. Venkatapur using the open source tools and data set keras, tensorflow and Darknet-53 along with additional customized videos from the construction site. The efficacy and efficiency of the outcome of this work is verified through standard formulas provided in the paper.

Acknowledgment

The authors sincerely acknowledge the Head of Institute, K. S. Institute of Technology, Bengaluru, India, to carry out this research work.

References

[1] L. C. Yann, Y. Bengio, and G. Hinton, “Deep learning,” *Nature*, vol. 521, no. 7553, pp. 436-444, 2015.

[2] S. Ren, K. He, R. Girshick, and J. Sun, “Faster R-CNN: Towards real-time object detection with region proposal networks,” arXiv preprint arXiv:1506.01497, 2015.

[3] R. Girshick, “Fast R-CNN,” in *Proc. the IEEE International Conference on Computer Vision*, pp. 1440-1448, 2015.

[4] Iandola, N. Forrest *et al*., “SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and < 0.5MB model size,” arXiv preprint arXiv:1602.07360, 2016.

[5] L. Wei, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, “SSD: Single shot multibox detector,” in *Proc. European Conference on Computer Vision*, pp. 21-37, 2016.

[6] H. Jan, R. Benenson, and B. Schiele, “Learning non-maximum suppression,” in *Proc. the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4507-4515, 2017.

[7] D. Zhou, J. Fang, X. Song, C. Guan, J. Yin, Y. Dai, and R. Yang, “IoU loss for 2d/3d object detection,” in *Proc. International Conference on 3D Vision (3DV)*, pp. 85-94, 2019.

[8] J. Ian *et al*., “Generative adversarial networks,” arXiv preprint arXiv:1406.2661, 2014.

[9] J. Redmon and A. Farhadi, “YOLOv3: An incremental improvement,” arXiv preprint arXiv:180402767, 2018.

[10] K. M. He, X. Y. Zhang, S. Q. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proc. the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770-778, 2016.

[11] T. Y. Lin, P. Dollar, R. Girshick, K. M. He, B. Hariharan, and S. Belongie, “Feature pyramid networks for object detection,” in *Proc. the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp 936-944, 2017.

[12] Y. Li, H. Wei, Z. Han, J. Huang, and W. Wang, “Deep learning-based safety helmet detection in engineering management based on convolutional neural networks,” *Advances in Civil Engineering*, vol. 20, 2020.

[13] X. Long, W. Cui, and Z. Zheng, “Safety helmet wearing detection based on deep learning,” in *Proc. IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC)*, pp. 2495-2499, 2019, Chengdu, China.

[14] H. Wang, Z. Hu, Y. Guo, Z. Yang, F. Zhou, and P. Xu, “A real-time safety helmet wearing detection approach based on CSYOLOv3,” *Applied Sciences*, vol. 10, no. 19, p. 6732, 2020.

[15] Z. Li, X. J. Bian, and M. Tan, *Automatic Safety Helmet Wearing Detection*, 2018.

[16] N. Otsu, “A threshold selection method from gray-level histograms,” *Automatica*, vol. 11, no. 285-296, pp. 23- 27, 1975.

[17] C. Jiwoong, D. Chun, H. Kim, and H.-J. Lee, “Gaussian yolov3: An accurate and fast object detector using localization uncertainty for autonomous driving,” in *Proc. the IEEE/CVF International Conference on Computer Vision*, pp. 502-511, 2019.

[18] J. Yu, Y. Jiang, Z. Wang, and T. Huang, “Unitbox: An advanced object detection network,” in *Proc. the ACM International Conference on Multimedia*, 2016.

[19] H. Rezatofighi, N. Tsoi, J. Gwak, A. Sadeghian, I. Reid, and S. Savarese, “Generalized intersection over union: A metric and a loss for bounding box regression,” in *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[20] Hyeok-June *et al*., *Image Preprocessing for Efficient Training of YOLO Deep Learning Networks*, 2018.

Kashika P. H. is a research scholar in Computer Science Department, K. S. Institute of Technology, Bengaluru. She has completed her MTech in computer science. Her area of interest is in image processing.

Rekha B Venkatapur is a professor and HOD in Computer Science Department, K. S. Institute of Technology, Bengaluru. She has completed her PhD from Jawaharlal Nehru Technological University, Hyderabad. Her area of interest is in image processing.