International Journal of Engineering Trends and Technology
ISSN: 2231 – 5381 / https://doi.org/10.14445/22315381/IJETT-V70I11P220

Volume 70 Issue 11, 187-196, November 2022
© 2022 Seventh Sense Research Group®

Original Article

YOLO Architecture-based Object Detection for Optimizing Performance in Video Streams

M. Maheswari[1], M. S. Josephine[2], V. Jeyabalaraja[3]

[1]*Department of Information Technology, Dr. MGR Educational and Research Institute, Deemed to be University, Chennai, Tamil Nadu, India.*
[2]*Department of Computer Applications, Dr. MGR Educational and Research Institute, Deemed to be University, Chennai, Tamil Nadu, India.*
[3]*Department of Computer Science & Engineering, Velammal Engineering College, Chennai, Tamil Nadu, India.*

[1]*Corresponding Author : m.mahe05@gmail.com*

Received: 26 July 2022 Revised: 31 October 2022 Accepted: 16 November 2022 Published: 26 November 2022

Abstract - *Nowadays, capturing images with greater quality has become so simple because of the rapid growth in the quality of devices capturing the same. Image capturing is now being accomplished less expensively with the use of modern technologies. Videos are a series of pictures with regular intervals of time. Video offers extra data about the object when the situations change with respect to time intervals. Handling objects in the videos manually is very difficult, requiring the process's automation. In recent years, many developed techniques and training deep neural networks have been used to improve accuracy in object detection, which is computationally intensive. In certain situations, most of the areas in a video frame are background, and the salient objects enclose a little part of the area in the video frame. There is a strong temporal correlation between consecutive frames in a video. Based on these examinations, this work proposes a Convolutional Neural Network (CNN), which reduces the computational needs for video object detection tasks. CNN uses an enhanced YOLO platform for classifying and detecting objects by creating new CNN architecture. The proposed model renders an accuracy of 96.7% in classifying the objects.*

Keywords - *Object Detection, Convolutional Neural Networks, Deep Learning, Videos, YoLo, Video Objects, Moving Cars detection.*

1. Introduction

Detecting moving objects has different applications in computer vision: vision-based control, video compression, human-computer interface, visual surveillance, medical image processing, and robotics[2]. Object detection helps to perceive and describe the behaviour of objects instead of supervising computers by human operators, and it also targets locating moving objects in a surveillance camera or video file[9]. Rosi et al.[23] has mentioned that object detection can be carried out through different methods, namely background subtraction, temporal differencing, region-based segmentation, etc. Chauhan et al.[4] have stated that the key to tracking objects is segmenting an interesting area from a scene in a video and tracking its motion, occlusion and positioning. According to Panchal et al.[19] states that the method of examining a video is the primary step in object identification. Object detection predicts the objects in a sequence of videos, and clustering these objects' pixels is done by different technologies, namely optical flow, frame differencing and background subtraction. Chauhan et al. have discussed about optical flow distribution features in moving objects. Das and Saharia (2014) have mentioned that background modeling is considered the hub of background subtraction. Rajkumar and Arunnehru [21] have mentioned that CNN is a strong learning model from neurons' biological concepts. Ren et al. [22] have mentioned that generalizing the CNN from two-dimensional image regions to three-dimensional video is difficult due to the irregularity between time and space. Video-based processing resolves the GPU memory cost[20]. Three-dimensional CNN and three-dimensional max-pooling are used in spatial and temporal dimensions [31]. Zia et al.[32] has mentioned that due to dedicated CUDA libraries and rapid GPUs for deep learning, three-dimensional CNN is becoming highly familiar. According to Schwarz et al. [24], the most similar strategy is to utilize an architecture of a deep convolutional neural network retrained on a huge dataset as an extractor of features or to fine-tune into a little set of data [7][29]. The best possible CDR (correct detection rate) is achieved in image recognition applications using CNN. This task has implemented the CNN in YOLO for classifying and detecting objects in video feeds. Buhler et al.[35] has stated that YOLO (You Only Look Once) is an individual feedforward neural network that finds class probabilities and bounding boxes of an image in an individual estimation. YOLO predicts image objects. Tsang[13] has stated that in the YOLO version, bounding boxes are carried out by anchor boxes.

The frequent occurrence of road accidents, traffic regulation violations, and traffic congestion is a common thing as far as India is concerned. A video object identification system capable of identifying automobile objects could serve as a solution for reducing traffic congestion and preventing mishaps on the roads. An enhanced YOLO architecture has been proposed in this research, trained and tested using videos of automobiles, and the performance has been evaluated.

2. Literature Review

Oneata et al. [17] have discussed spatio-temporal tubes, which are formed by the sequence of bounding boxes. In the study of Shahare and Shende [36], object detection is done by an object detector needing manual labelling or background subtraction. Mercanoglu et al. [16] discussed the optical flow-based algorithm for the videos captured by a moving camera.

Chen and Lu [5] have described in their research that object-level motion detection is a critical issue from moving cameras due to the dual motion established by the camera and object motion. This study predicts the motion of an object from a moving camera using two respective video frames. A descriptor of context-aware motion is framed based on the direction of moving objects moving and the speed associated with that of a moving camera. Kang et al. [11] proposed work on object identification from video tubelet with CNN. Deep CNN has shown performance improvement in tasks like the detection of objects, semantic segmentation and image classification.

Hou, Chen and Shah [10] split the video into similar-length scenes, and for every scene, tube proposals set are produced based on three-dimensional CNN network characteristics and detection is carried through linked proposals. Kang et al. [14] specify that Fast Region CNN and Faster Region CNN significantly impact object detection. A proposed deep learning structure involves contextual and temporal data from tubelet acquired in videos. It is referred to as a tubelet convolutional neural network that is tubelet with CNN.

In the work of Lekhak [15] region-based method for the detection of objects faster R-CNN was refined to predict objects in videos much more effectively. The system offers a mean average precision of 0.64 for object detection in the dataset, which has greater mean average precision of 0.56 for faster R-CNN without Long short-term memory (LSTM).

Tian et al. [28] method involve three perspectives: object recognition, video processing and target detection. Garg and Kotecha[8] have stated that modelling techniques suffer from memory costs in high-definition (HD)video, and greater computation might decrease accuracy.

Bertasius et al. [3]discussed about deformable convolutions across time for detecting objects. Yazdi and Bouwmans's study [30] discussed approaches for detecting moving objects captured by moving cameras. Several approaches in this sector can be categorized into four types of trajectory classification, background subtraction-based modeling, object tracking, sparse matrix and low-rank decomposition.

Tang et al. [27] in their study expanded previous approaches in the following perspectives: 1) a short tubelet network detection that predicts short tubelets in segments of short video; 2) a cuboid proposal network that retrieves cuboids of the spatiotemporal candidate which bound the movements of objects, and 3)an algorithm of short tubelet linking that relates temporarily overlapping short tubelet to form big tubelet.

Danyang Cao, Zhixin Chen & Lei Gao [33] discussed about detecting small, dense objects and objects with random geometric transformations by using deep CNN to obtain multi-scaled features.

Based on the literature survey on the articles mentioned above, some main research gaps are identified. Initially, the video data streams in existing works could not be rendered as per the threshold limits. Secondly, the overlapping segments were not efficiently addressed. Hence the proposed work focused on the YOLO framework for object detection in video streams.

3. Design of the System

3.1. Proposed System Design

YOLO is a new method for object detection, and YOLO learns general object representation. In other algorithms, namely CNN and fast-CNN, the algorithm will not view the image completely, but in YOLO, the algorithm views the image completely by finding the bounding boxes using CNN and the probabilities of class for these boxes and predicts the image faster when compared with other algorithms. This work has attempted to track the cars through the creation of new CNN architecture. The CNN is trained based on the object detection dataset. CNN combines convolutional layers, fully connected layers and max pooling layers. Before entering into CNN, the image passes through the stage of pre-processing first. Steps used in pre-processing are reorganising the input image, cropping or warping and resizing the image. CNN performs well on image classification tasks despite its complexity of computation and network. In this work, the CNN algorithm is presented based on open source object classification and detection platform complied under the project of YOLO, which denotes You Only Look Once. The figure 1 shows the working of YOLO architecture. From the Figure 1, it is identified that an input image is taken, and the YOLO algorithm is applied. The image is classified as 3*3 grid matrixes. Then the image can be classified into grid numbers relying on image complexity. Once the image is classified, every grid performs localization and classification of the object. The confidence or objectness score of every grid is predicted. If there is no appropriate object predicted in the grid, then the bounding box value and objectness of the grid will be 0, or if there predicts an object in the grid, then the objectness will be one, and the value of bounding box will be the respective values of bounding box of the predicted object. The YOLO views the input video or image as a single regression problem. Direct from pixels of the image to the bounding box is considered an individual issue. YOLO scans the image in one go and first considers an input video or image. YOLO classifies the image into grids of S * S. The bounding boxes are liable for finding five bounding boxes. The class is found along with the prediction score for every bounding box. The bounding box number which will be considered as 12 * 12 = 144, and every grid generates around five bounding boxes. Therefore a total of 144 * 5 = 720 bounding boxes have been created. Several bounding boxes will have a reduced confidence score; in the end, only the bounding boxes crossing some threshold value will be involved. For the threshold value of human detection of thirty sounds is better enough. In YOLO, every bounding finds five parameters: confidence and w, h, x, and y. The coordinates of x and y indicate the mid of the box associated with grid cell bounds. The height (h) and width (w)are found to be associated with to complete image. Lastly, the prediction of confidence produced nearly IOU between the ground truth box and the predicted box. This YOLO architecture proposed comprises of neural network with a 3×3×16 convolution layer and 2×2 max pooling layer. Below, figure 2 shows the modified implementation of YOLO architecture using CNN.

Figure 2 shows the design of the proposed architecture. In this model, first, the loss function is optimized. Once the loss function is optimized, the model is trained, which is used to detect objects and videos. In order to execute the code, apply the video's input and output path. It will automatically detect some objects in the video. This is the loss function which is optimized. From the above figure 1, first, the input image is given. Then the convolution layer is applied with the same padding, with filter 3 × 3× 16 and Batch Norm and Leaky ReLu. After this, the Max Pooling layer is applied with pool size, stride and same padding. The values of the filter in the convolution layer are changed at every time interval

3.2. Mathematical Modelling of Loss Function

The notion of YOLO is to make CNN find a (3×3×16) tensor. It employs a CNN to scale back the spatial dimension to 3×3 with 1024 output channels at each place. By using fully connected layers, linear regression creates a prediction of the bounding box. Lastly, a prediction is made regarding the high confidence score in a box. YOLO break through the maximum limit of the speed of CNN and realizes the actual balance of accuracy and speed. It is real-time and fast. YOLO reasons worldwide and encodes contextual data about the image. Thus it is less probable to find false positives in the background. YOLO studies general object representation outstripping other detection methods by a vast margin when generalizing from realistic images to other network domains. YOLO has limitations. It poses a powerful spatial constraint on predicts of bounding boxes such as identifying little objects. Still, it struggles to generalize objects in unusual or new aspect configurations or ratios. It is not perfect that the loss function of YOLO handles mistakes similarly in large versus small bounding boxes. The YOLO algorithm finds numerous boxes of bounding. The bounding box is used to calculate the loss function for the responsibility of the object. The loss function is based on the confidence, classification and localization losses [18]. Each loss function is explained with the following mathematical modeling.

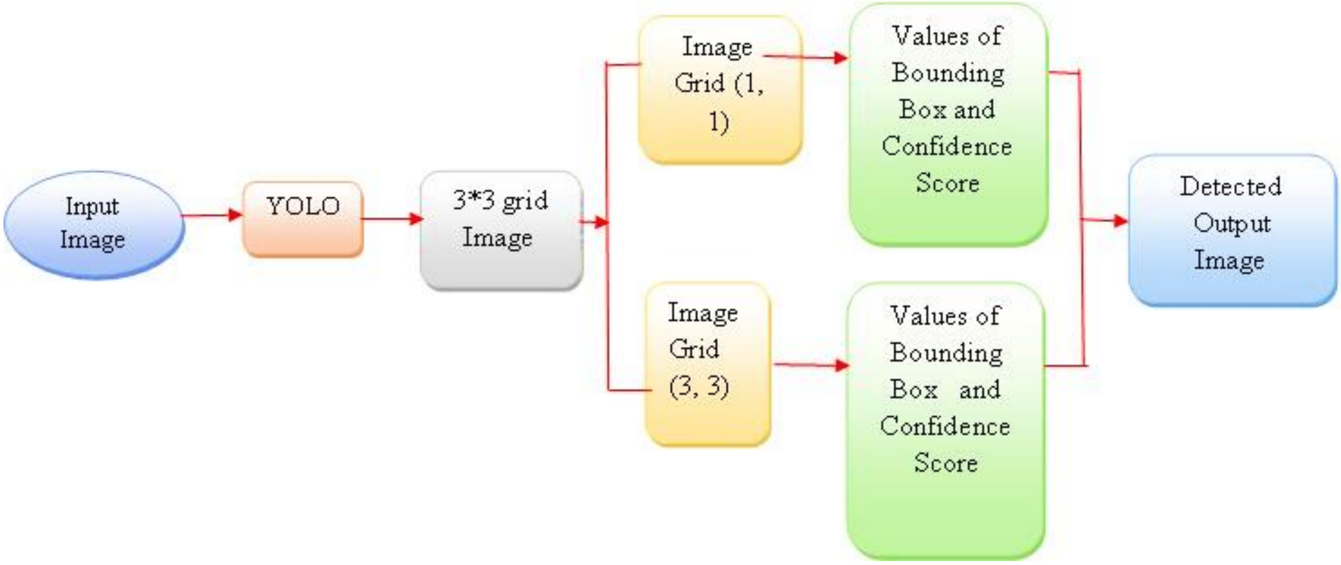

Input Image
YOLO
3*3 grid Image
Image Grid (1, 1)
Image Grid (3, 3)
Values of Bounding Box and Confidence Score
Values of Bounding Box and Confidence Score
Detected Output Image

Fig. 1 System Flow Diagram of Enhanced YOLO Architecture

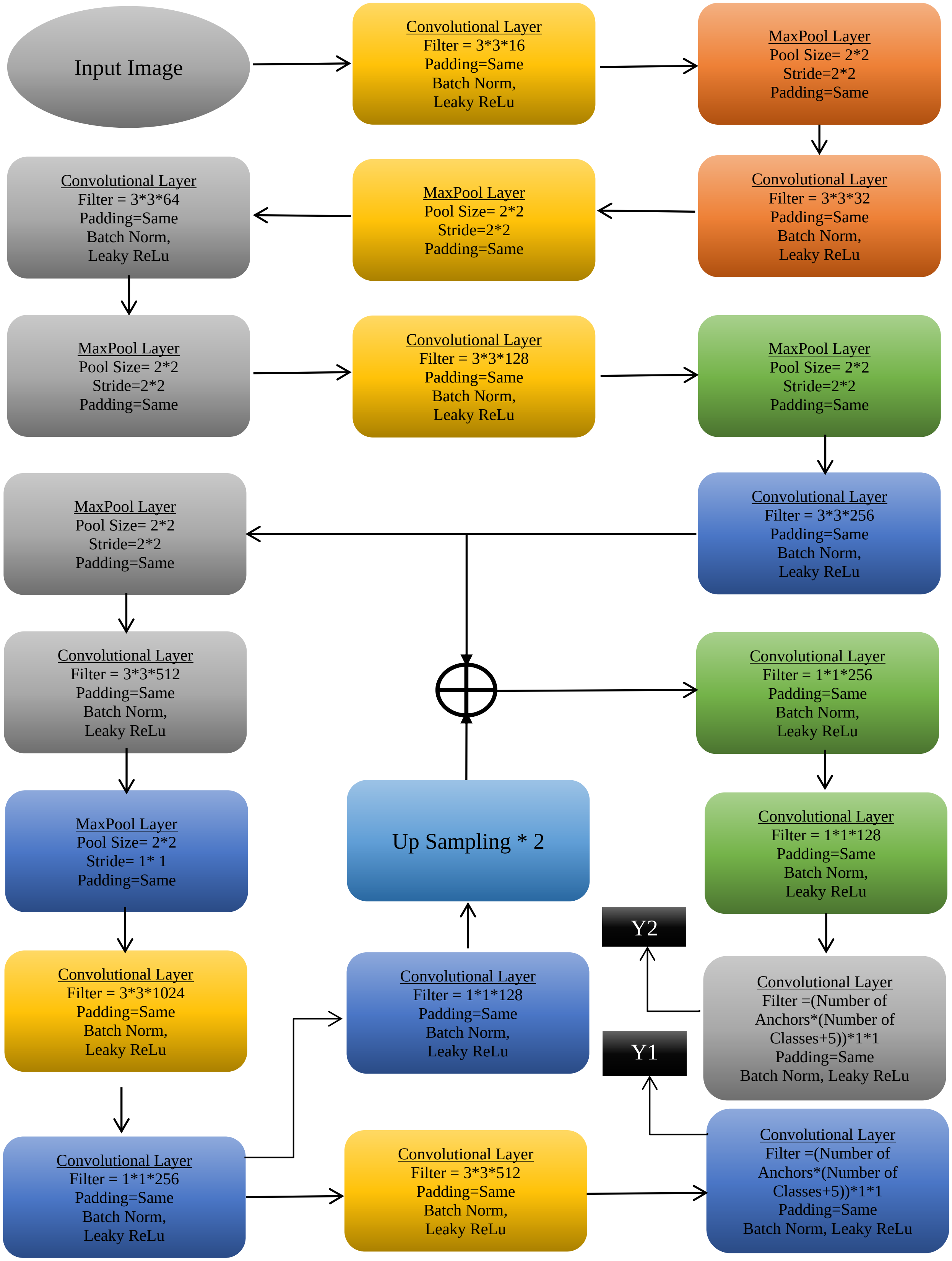


Fig. 2 Design of Proposed system

3.2.1. Confidence Loss

The confidence loss is measured by detecting an object in the box. The equation of confidence loss when the object is detected in the box is shown in Eqn 1.

$$\sum_{m=0}^{S^2} \sum_{n=0}^{b} 1_{mn}^{object} (CB_m - CB_n)^2 \quad (1)$$

Where

CB$_m$ is the score of confidence of Box n in cell m

1_{mn}^{object} = 1 if the n, the box of boundary in cell m is liable for finding the object; otherwise, it is 0. b denotes the boundary box

s^2 denotes the number of grids based on which the object was divided. Similarly, the confidence loss function if an object is not detected in the box is mentioned below Eqn 2.

$$\lambda_{noobject} \sum_{m=0}^{S^2} \sum_{n=0}^{b} 1_{mn}^{object} (CB_m - CB_m)^2 \quad (2)$$

Where

$1_{mn}^{noobject}$is 1_{mn}^{object} compliment

CB$_m$ is the score of confidence of Box n in cell m

$\lambda_{nooobject}$ measures the loss when detecting the background

3.2.2. Classification Loss

At every cell, if an object is found, then the classification loss is the squared error of the conditional probabilities for every class, as shown in Eqn 3.

$$\sum_{m=0}^{s^2} 1_{m}^{object} (P_m(cc) - \hat{P}_m(c))^2 \quad (3)$$

Where

1_{m}^{object}= 1 is an object seems in m cell; otherwise, it is zero

$\hat{P}_m(c)$ indicates the probability of conditional class for c class in m cell

$\hat{P}_m(cc)$indicates the probability of conditional class for c class in m cell

s^2indicates the probability of conditional class for c class in m cell

3.2.3. Localization Loss

The localization loss estimates the mistakes in the found boundary box sizes and locations. The box is only counted for predicting the object. The equation for localization loss is given below in Eqn 4.

$$\lambda_{coordinate} \sum_{m=0}^{S^2} \sum_{n=0}^{b} 1_{mn}^{object} [(X_m - \hat{X}_m)^2 + (Y_m - \hat{Y}_m)^2] + \sum_{m=0}^{S^2} \sum_{n=0}^{b} 1_{mn}^{object} [(\sqrt{W}_m - \widehat{\sqrt{W}}_m)^2 + (\sqrt{H}_m - \hat{H}_m)^2] \quad (4)$$

Where

1_{m}^{object}= 1 if the nth box of boundary in m cell is liable for predicting the object; otherwise, it is zero

$\lambda_{coordinate}$ enhances the loss of weight in the coordinates of the boundary box

3.2.4. Loss Function

The final loss function integrates confidence loss, localization loss and classification losses together. The equation for the loss function is mentioned below in Eqn 5.

$$\lambda_{coordinate} \sum_{m=0}^{S^2} \sum_{n=0}^{b} 1_{mn}^{object} [(X_m - \hat{X}_m)^2 + (Y_m - \hat{Y}_m)^2] \sum_{m=0}^{S^2} \sum_{n=0}^{b} 1_{mn}^{object} [(\sqrt{W}_m - \widehat{\sqrt{W}}_m)^2 + (\sqrt{H}_m - \hat{H}_m)^2] + \sum_{m=0}^{S^2} \sum_{n=0}^{b} 1_{mn}^{object} (C_m - C_m)^2 + \lambda_{noobject} \sum_{m=0}^{S^2} \sum_{n=0}^{b} 1_{mn}^{noobject} (CB_m - CB_m)^2 + \sum_{m=0}^{s^2} 1_{m}^{object} (P_m(cc) - \hat{P}_m(c))^2 \quad (5)$$

3.3. Training of Network

The convolutional neural network must be trained, and proper parameters must be determined, while YOLO offers a platform for object classification and detection. The momentum, batch size, rate of learning, number of iterations, decay and thresholds of detection are all specific task parameters that must be given as input to the YOLO algorithm. The epochs number that the network required to be trained empirically decided. Epoch defines as an individual presentation of the whole set of data to a neural network. For batch training, the entire training samples pass through the learning algorithm [33] in one epoch simultaneously before weights are updated. The learning rate is an element used to handle the learning rate. The batch size defines the number of training instances in one backwards or forward pass.

The decay defines the ratio between the epoch and the learning rate, while momentum is a factor that handles the improvement of the learning rate. The network was designed to have a 3×3 structure of the grid and was verified on only one class of object. This architecture of the network provides a tensor output with 3×3×16 dimensions. It is essential to mention that a highly cited and used database of images, namely PASCAL VOC 2007 and 2012, has not been employed for the purpose of training. The preliminary outputs displayed that images taken by developers varied essentially from the images available in PASCAL VOC databases in terms of the composition of the scene and the angle at which the images were clicked. For instance, several images from the database of PASCAL VOC were taken in the front view, while the images taken by developers comprised mainly from the top view angle. Therefore it is not a coincidence that the trained networks on PASCAL VOC database images alone, when tested, the developer-obtained video feeds proved unstable with low confidence recognition. However, a confidence of recognition of 84 percent was met when a database comprising car images was utilized for the purpose of training. The schedule of learning rate also relies on the dataset of learning. In contrast, it has been recommended that the rate of learning emerges gradually for the first epochs, and this may not be original for their training network. It is known that initiating the learning rate at greater levels causes the models to be unstable.

The input and output videos are taken with the same duration so that the objects can clearly be detected in the output video. In the videos, the objects are detected, and the video is classified into image frames. This work has used CNN to enhance the contrast, features and distinction of the objects in the videos.

4. Results

4.1. Validation of Neural Network

CNN validation was undertaken by testing the accuracy of classification on objects class labelled as "cars". The object class "cars" were trained on the Pascal Visual Object Class (VOC) dataset link (http://www.videvo.net/stock-video-footage/car/). In this link, the images comprise different types of cars and many image resolutions, compositions and scales. Out of the 1676 videos available, only 257 video clips were available to access free of cost. So 257 video clips were taken into consideration for this research. Out of the 257, only 235 videos had the actual car object inside them, while the other videos had car components like a speedometer, tyre, etc. The category of car objects was made using the images for network training. There were a total of 257 images comprising 50 car objects in this training dataset. The below figure 3(a) and figure 3(b) show the training set and testing set of object class "cars".

The Bounding Box label, an open-source component, was utilized to mention real car examples in this data set, creating bounding boxes with ground truth. The best accuracy was obtained as the result of the training phase when the batch size used was 32, the learning rate was 10^ (-3), the decay was 0.00002, and the number of epochs was 406. The threshold level for detection was set as 0.5 for the training (ignore threshold = 0.5).

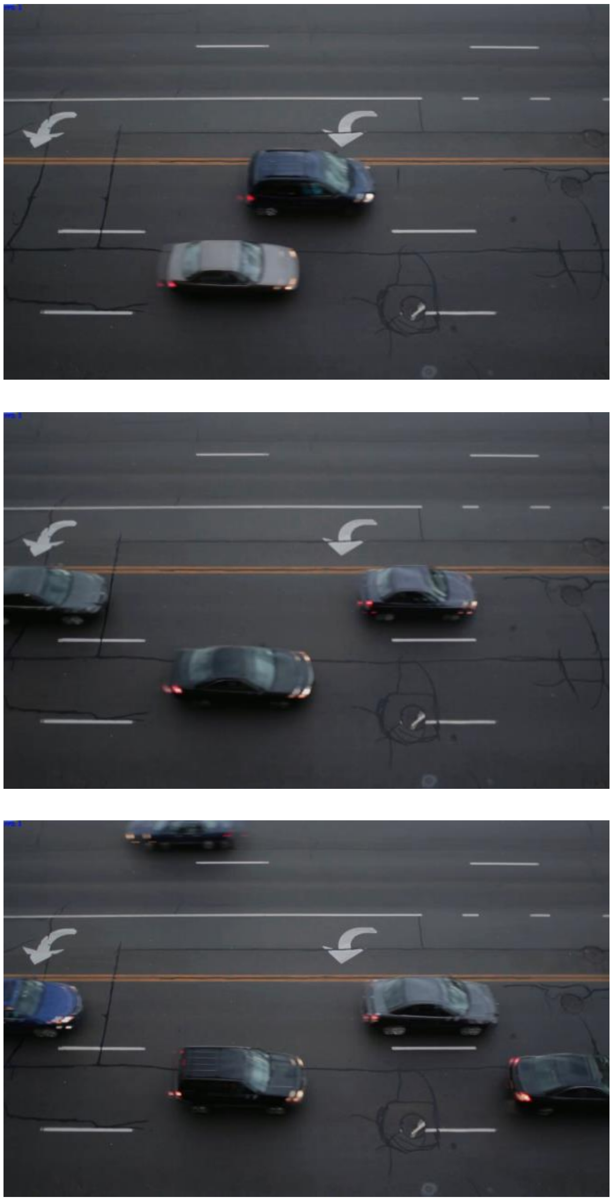
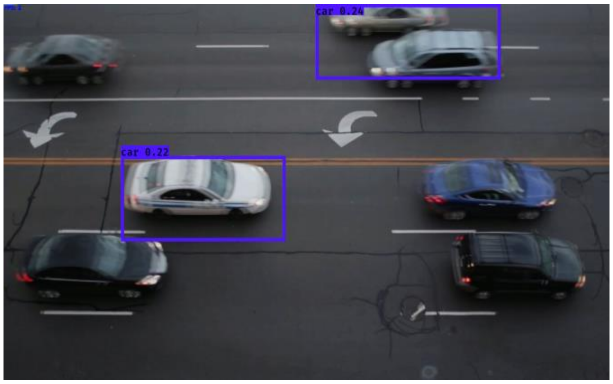


Fig. 3 (a) Training set of "Cars" object class



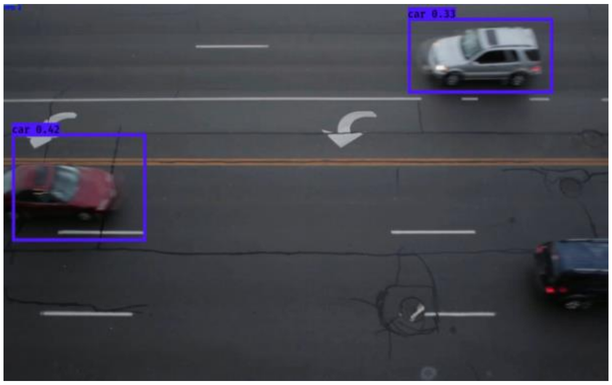


Fig. 3 (b) Testing set of "Cars" object class

In order to test the proposed approach and identify the recognition accuracy of the model, the research has used 235 video clips with actual car images as the input dataset. There were 485 car images inside the videos. Out of the 485 car objects, the proposed model identified 469 objects accurately, thereby rendering an accuracy value of 96.7%. The number of instances that were not correctly categorized was only 16. An incorrect classification is said to have happened when the image comprises a car but is not identified by the proposed model or if the model identifies only one image despite several cars in the frame. Of the 16 misclassifications, 13 car objects remained unidentified, and 3 were wrongly recognised as cars.

The following Table 1 presents the confusion matrix for the model proposed.

Table 1. Confusion Matrix for "Cars" object class

Classification	**Class**	**Detected**	
		Car	**No Car**
Actual	Car	469	13
	No Car	3	NA

Table 1 indicates that the true positive rate rendered by the model is 96.7%. The positive prediction value rendered by the proposed model is 99.4%, and the false discovery rate rendered is 0.6%. Further, the false negative rate rendered by the model is 2.7%.

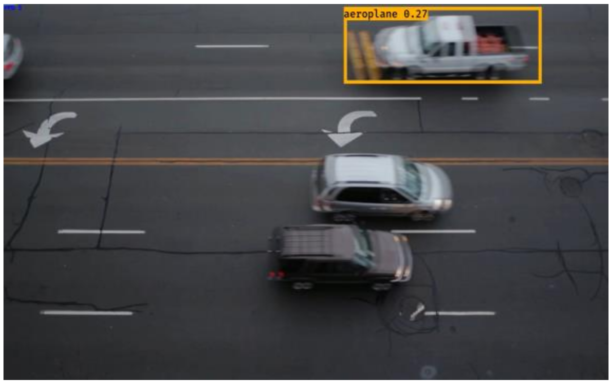


Fig. 4 (a) Orange box points at an instance where the "cars" object is existing but is not detected by the convolutional neural network

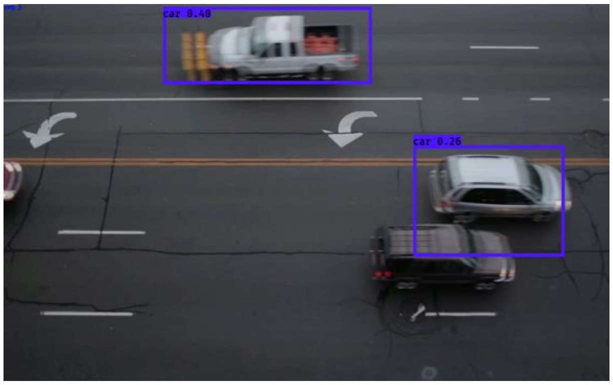


Fig. 4 (b) Blue box points at the instance where the "cars" object is identified correctly

4.2. Real-Time Recognition of Object from Car Video Feed

After training and validation, the accuracy of the network was verified in the actual time car video feed. Additionally, the identification and detection of the multi-object situation were also estimated. Figure 4 (a) represents the orange box points at an instance where the "cars" object exists but is not detected by the convolutional neural network, and figure 4(b) the Blue box points at an instance where the "cars" object is identified correctly. In this research, the accuracy can be predicted by counting the number of cars in the video divided by the number of cars detected. The equation of accuracy can be depicted in Eqn.6

$$Accuracy = \frac{Number\ of\ cars\ in\ the\ video}{Number\ of\ cars\ which\ is\ detected} \quad (6)$$

Table 2. Recognition rate in different timeframes of the video

Time	**Number of cars in the video**	**Number of cars detected**	**Recognition rate**
10 seconds	9	7	77%
20 seconds	16	10	62.5%
30 seconds	13	9	69.2%
40 seconds	6	4	66.6%
50 seconds	6	5	83.3%

It can be inferred from Table 2 that with the increase in time, the recognition rate decreases at the initial stages. However, as time goes on, there is a remarkable and steady increase in the recognition rate. This shows that the proposed model can recognise objects correctly, irrespective of the increase in time.

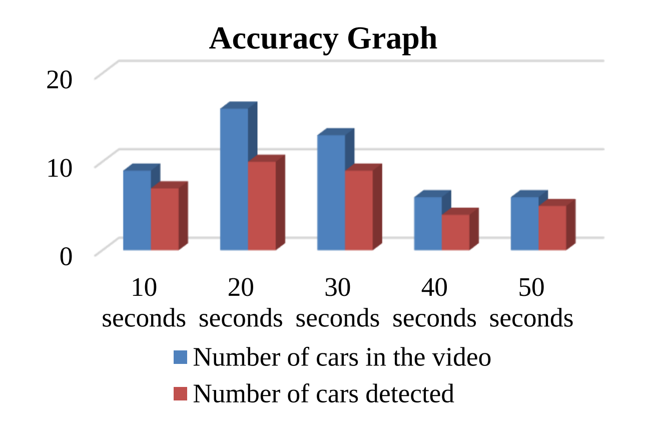


Fig. 5 Recognition rate

4.3. Comparison with Existing Approaches

Table 3 presents a comparative view of the proposed approach with the existing approaches concerning the object detection accuracy that the model could do.

Table 3. Comparison of the proposed approach with the existing approaches

Author and Year	Accuracy
Song (2019) [26]	95.3%
Ahmed et al. (2020) [1]	68.9%
Proposed approach	96.7%

Song et al. (2019) developed a vehicle detection system with deep learning, and it was found to render an accuracy of 95.3% average correct rate in identifying Car objects. Likewise, Ahmed et al. (2020) proposed a modified YoLo approach for object detection and tested the same on multiple objects. The accuracy of their approach with respect to the detection of the object, "Car," alone is found to be 68.9%. The proposed approach reveals a higher recognition rate and accuracy when compared with that of the already existing approaches.

5. Conclusion and Future Work

The preliminary tests displayed that the convolutional neural network could recognise and detect various classes of the object in multi-object situations in real-time from a video feed offered by developers with accuracy. The Convolutional neural network could classify and detect an object in the image even if other objects obscured the complete object of interest contours. The CNN can also detect and classify objects not shown in the video. Based on the greater level of classification and detection accuracy met, there are fewer chances in both military and commercial applications. The approach can be applied successfully in several transportation-related works with easy changes.

Video object detection is a major challenge in computer vision technology. Many approaches have been used, namely temporal differencing, background subtraction and optical flow. This work proposes CNN for detecting objects in videos. Since the establishment of CNN, is used for image classification and detection and has raised state of the art in object classification and detection. CNN provides state-of-the-art object detection performance in the video dataset. YOLO is a unified model for detecting objects, and it is easy to build and can be trained entirely on whole images. YOLO is trained on the function of loss that directly corresponds to the detection performance, and the whole model is trained jointly. YOLO generalizes correctly to new domains, making it ideal for applications that depend on robust and fast detection of objects. YOLO is an accurate and fast detector of objects, making it ideal for computer vision applications.

The CNN architecture created based on the modification of YOLO renders better accuracy in terms of detecting objects. Real-time surveillance systems could be constructed on a wide scale in this research. The architecture can be improved further in the future and trained with a large number of datasets, and a comparison would be made on how effective the improvisation of the architecture has resulted in the enhancement of the accuracy of boundary objects as well as overall performance. Another limitation of this research is that the researcher has tested the proposed architecture only concerning videos of cars. However, in real-time traffic, there will be numerous other light vehicles, such as two-wheelers and three-wheelers, as well as heavy vehicles like load trucks, tankers, cement mixers etc., that cause actual traffic during peak hours. Future researchers can simultaneously test the proposed architecture with numerous other vehicle objects and predict the model's performance.

References

[1] Ahmad T., Ma, Y. Yahya, M.Ahmad, B., Nazir, S and Haq, A.U, "Object Detection through Modified YOLO Neural Network, An Intelligent Decision Support System," *Scientific Programming,* Article ID. 8403262, pp 1-10, 2020. Crossref**,** https://doi.org/10.1155/2020/8403262

[2] Balaji S. R and Karthikeyan S, "A Survey on Moving Object Tracking Using Image Processing," *In 2017 11th International Conference on Intelligent Systems and Control (ISCO)*, pp. 469-474, 2017. Crossref, https://doi.org/10.1109/ISCO.2017.7856037

[3] Bertasius G, Torresani L and Shi J, “Object Detection in Video with Spatiotemporal Sampling Networks,” In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 331-346, 2018. Crossref, https://doi.org/10.48550/arXiv.1803.05549

[4] Chauhan A. K and Krishan P, "Moving Object Tracking using Gaussian Mixture Model and Optical Flow," *International Journal of Advanced Research in Computer Science and Software Engineering,* vol. 3, no. 4, pp. 5212-5215, 2014.

[5] Chen T and Lu S, "Object-Level Motion Detection from Moving Cameras," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 11, pp. 2333-2343, 2016. Crossref**,** https://doi.org/10.1109/TCSVT.2016.2587387

[6] Das D and Saharia S, "Implementation and Performance Evaluation of Background Subtraction Algorithms," *International Journal on Computational Sciences & Applications (IJCSA)*, vol. 4, no. 2, pp. 49-54, 2014. Crossref, https://doi.org/10.48550/arXiv.1405.1815

[7] Eitel A, Springenberg J. T, Spinello L, Riedmiller M and Burgard W, "Multimodal Deep Learning for Robust RGB-D Object Recognition," In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 681-687, 2015. Crossref**,** https://doi.org/10.1109/IROS.2015.7353446.

[8] Garg D and Kotecha K, "Object Detection from Video Sequences Using Deep Learning: An Overview," *Advanced Computing and Communication Technologies*,vol. 562, pp. 137-148, 2018. Crossref, https://doi.org/10.1007/978-981-10-4603-2_14

[9] Gupta R. K, “*Object Detection and Tracking in Video Image*,” Doctoral Dissertation, 2014.

[10] Hou R, Chen C and Shah M, "Tube Convolutional Neural Network (T-CNN) for Action Detection in Videos," In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 5822-5831, 2017.

[11] Kang K, Li H, Yan J, Zeng X, Yang B, Xiao T and Ouyang W, "T-CNN: Tubelets with Convolutional Neural Networks for Object Detection from Videos," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 10, pp. 2896-2907, 2018. Crossref, https://doi.org/10.1109/TCSVT.2017.2736553

[12] Sujata Chaudhari, Nisha Malkan, Ayesha Momin and Mohan Bonde, "Yolo Real-Time Object Detection," *International Journal of Computer Trends and Technology*, vol. 68, no. 6, pp. 70-76, 2020. Crossref, https://doi.org/10.14445/22312803/IJCTT-V68I6P112

[13] Tsang S H, “Review: R-CNN (Object Detection), Coinmonks,” 2019. [Online]. Available: https://medium.com/coinmonks/review-r-cnn-object-detection-b476aba290d1

[14] Kang K, Ouyang W, Li H, and Wang X, "Object Detection from Video Tubelets with Convolutional Neural Networks," In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 817-825, 2016. Crossref, https://doi.org/10.1109/CVPR.2016.95

[15] Lekhak D, “*Object Detection in Videos using Region Based Convolutional Neural Network*,” Thesis, 2017.

[16] Mercanoglu O, Ajabshir V B, Keles H and Tosun S, “Moving Object Detection by a Mounted Moving Camera,” *International Conference on Computer as a Tool*, Spain, 2015. Crossref, https://doi.org/10.1109/EUROCON.2015.7313714

[17] Oneata D, Revaud J, Verbeek J and Schmid C, "Spatio-Temporal Object Detection Proposals," In *European Conference on Computer Vision,* Springer, Cham, pp. 737-752, 2014. Crossref, https://doi.org/10.1007/978-3-319-10578-9_48

[18] Ordania S, “*Detecting Cars in a Parking Lot using Deep learning*,” Masters Dissertation in Computer Science, San Jose State University, pp. 62, 2019. Crossref, https://doi.org/10.31979/etd.m6as-epyd

[19] Panchal P, Prajapati G, Patel S, Shah H and Nasriwala J, "A Review on Object Detection and Tracking Methods," *International Journal for Research in Emerging Science and Technology,* vol. 2, no. 1, pp. 7-12, 2015.

[20] Peng X and Schmid C, "Multi-Region Two-Stream R-CNN for Action Detection," In *European Conference on Computer Vision, Springer, Cham*, pp. 744-759, 2016.

[21] Rajkumar, R and Arunnehru J, "A Study on Convolutional Neural Networks with Active Video Tubelets for Object Detection and Classification," In *Soft Computing and Signal Processing, Springer,* Singapore, pp. 107-115, 2019. Crossref, https://doi.org/10.1007/978-981-13-3393-4_12

[22] Ren S, He K, Girshick R and Sun J, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," In *Advances in Neural Information Processing Systems (NIPS),* vol. 1, pp. 91–99, 2015. Crossref, https://doi.org/10.5555/2969239.2969250

[23] Rosi S, Meshach W T and Prakash J S, "A Survey on Object Detection and Object tracking in Videos," *International Journal of Scientific and Research Publications,* vol. 4, no. 11, pp. 1-4, 2014.

[24] Schwarz M, Schulz H and Behnke S, "RGB-D Object Recognition and Pose Estimation Based on Pre-Trained Convolutional Neural Network Features," In *Robotics and Automation (ICRA), IEEE International Conference*, pp. 1329–1335, 2015. Crossref, https://doi.org/10.1109/ICRA.2015.7139363

[25] Ms.S.Supraja and P.Ranjith Kumar, "An Intelligent Traffic Signal Detection System Using Deep Learning,*" SSRG International Journal of VLSI & Signal Processing,* vol. 8, no. 1, pp. 5-9, 2021. Crossref, https://doi.org/10.14445/23942584/IJVSP-V8I1P102

[26] Song H, Liang H, Li H, Dai Z and Yun X, "Vision-Based Vehicle Detection and Counting System using Deeplearning in Highway Scenes," *Journal of Cardiovascular Electrophysiology*, vol. 11, pp. 1-11, 2019.

[27] Tang P, Wang C, Wang X, Liu W, Zeng W and Wang J, "Object Detection in Videos by High Quality Object Linking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 5, pp. 1-7, 2019. Crossref, https://doi.org/10.1109/TPAMI.2019.2910529

[28] Tian B, Li L, Qu Y and Yan L, "Video Object Detection for Tractability with Deep Learning Method," In *2017 Fifth International Conference on Advanced Cloud and Big Data (CBD)*, pp. 397-401, 2017. Crossref, https://doi.org/10.1109/CBD.2017.75

[29] Wang A, Lu J, Cai J, Cham T. J and Wang G, "Large-Margin Multimodal Deep Learning for RGB-D Object Recognition," *IEEE Transactions on Multimedia,* vol. 17, no. 11, pp. 1887-1898, 2015. Crossref, https://doi.org/10.1109/TMM.2015.2476655

[30] Yazdi M and Bouwmans T, "New Trends on Moving Object Detection in Video Images Captured by a Moving Camera: A Survey," *Computer Science Review*, vol. 28, no. 2, pp. 157-177, 2018. Crossref, https://doi.org/10.1016/j.cosrev.2018.03.001

[31] Zeng X, Ouyang W, Yang B, Yan J and Wang X, "Gated Bi-Directional CNN for Object Detection," In *European Conference on Computer Vision Springer, Cham*, vol. 9911, pp. 354-369, 2016. Crossref, https://doi.org/10.1007/978-3-319-46478-7_22

[32] Zia S, Yuksel B, Yuret D and Yemez Y,"RGB-D Object Recognition using Deep Convolutional Neural Networks," In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 896-903, 2017.

[33] Danyang Cao, Zhixin Chen and Lei Gao, "An Improved Object Detection Algorithm Based on Multi-Scaled and Deformable Convolutional Neural Networks," *Human-centric Computing and Information Sciences*, vol. 10, pp. 1-22,2020. Crossref, https://doi.org/10.1186/s13673-020-00219-9

[34] Manish Suyal and Parul Goyal, "A Review on Analysis of K-Nearest Neighbor Classification Machine Learning Algorithms based on Supervised Learning," *International Journal of Engineering Trends and Technology*, vol. 70, no. 7, pp. 43-48, 2022. Crossref, https://doi.org/10.14445/22315381/IJETT-V70I7P205

[35] Buhler K, Lambert J and Vilim M, “Real-time Object Tracking in Video CS 229 Course Project,” 2016.

[36] Shahare D and Shende R, "Moving Object Detection with Fixed Camera and Moving Camera for Automated Video Analysis," *International Journal of Computer Applications Technology and Research*, vol. 3, no. 5, pp. 277-283, 2014.