Intelligent Traffic Light Control System

Surya Vishnuvardhan Joka[1], Taha Bhagat[1], Krishnaram Bhavesh[1], Prof. Mahesh A. Kamthe[2]

1Student, Dept. of Electronics and communication, MIT ADT University, Pune, Maharashtra, India
[2]Professor, Dept. of Electronics and communication, MIT ADT University, Pune, Maharashtra, India.

---***---

ABSTRACT

Traffic congestion remains a pervasive challenge in many metropolitan areas worldwide, with severe impacts on road users' safety and the economy.[1] Despite changes in traffic patterns, traditional traffic control systems, including traffic lights at intersections, have remained largely static for over 80 years. In response, we propose a new digital-logic based system, the Intelligent Traffic Light Control System (ITLCS), which promises to be more efficient and responsive to traffic conditions. The ITLCS system leverages a simple yet innovative principle, whereby the signal remains green until the present cars have passed. The algorithm captures a snapshot of the traffic and analyzes the number of vehicles present in each lane, enabling the system to adjust signal timings dynamically based on real-time traffic conditions. This approach reduces the average wait time for all vehicles and accounts for both macroscopic and microscopic changes in traffic, ensuring optimal traffic flow and safety for all road users. The ITLCS model requires object detection, including data acquisition and training a deep learning model to identify different classes of vehicles. By implementing the proposed ITLCS, we aim to address the pressing issue of traffic congestion and reduce the number of accidents, particularly those occurring at intersections. The ITLCS represents a significant improvement over traditional traffic control systems, and its adoption could have far-reaching benefits for metropolitan areas worldwide.

Keywords

Intersection; YOLOv5; Object detection; Traffic; Data.

1. INTRODUCTION

Traffic light systems are an essential part of urban transportation, and they play a crucial role in regulating the flow of vehicles and pedestrians on the roads. The current state of research in traffic light systems is focused on developing advanced technologies to optimize traffic flow, reduce congestion, and improve safety.

One of the most significant areas of research is the development of intelligent traffic management systems that use data from various sources, such as traffic sensors, cameras, and GPS devices, to dynamically adjust traffic signal timings in real-time. These systems can help reduce congestion by providing more efficient signal timing based on traffic patterns.

Research is also being done on using artificial intelligence and machine learning algorithms to optimize traffic light systems. These algorithms can analyze large amounts of data to identify patterns and make predictions about traffic flow, enabling more efficient signal timing and reducing congestion.

However, the current traffic signal system remains outdated, leading to inefficient time management at road intersections [2].

The traditional traffic signals operate by assigning a fixed fraction of time to each road, regardless of the flow density or the number of vehicles present. This results in inefficient traffic flow and does not distribute time based on traffic congestion. During certain periods of the day, some roads may have higher traffic volume than others, necessitating more time to alleviate congestion. Unfortunately, the traditional traffic signal system cannot cater to this need.

Therefore, there is an urgent need for a new traffic signal system that can detect the presence of vehicles at an intersection and directly close the signal once there are no more vehicles present, thereby opening the next road and reducing unnecessary waiting.

The proposed model will be utilizing multiple cameras based on the number of cross-roads at an intersection, the cameras capture information in real-time and the information is processed through and object detection algorithm which helps classify the vehicles on the road as per the individual time required by the vehicle for traversing the intersection.

The proposed model's central purpose is to reduce the time complexity at an intersection to increase efficiency of travel-time and also make travel through congestion at intersections.

2. OBJECTIVE

The objective of this proposed model is to formulate a solution for the heavy congestion caused by time-based traffic signals at crossroads. The proposed solution involves a machine learning-based traffic signal that can create a congestion-free environment. The current time-based switch system often leaves many roads empty during unnecessary times while causing traffic congestion on roads with heavy traffic. The ML-based traffic signal will dynamically adjust the signal timings by analyzing real-time traffic data and thus optimize traffic flow. This approach aims to improve traffic safety, reduce travel time, and minimize the environmental impact of traffic.

3. LITERATURE REVIEW

3.1 Retinanet

RetinaNet[3] is an object detection model that utilizes a focal loss function to tackle class imbalance during training. The model is a one-stage detector and is composed of a single, unified network comprising a backbone network and two task-specific subnetworks. The backbone network computes a convolutional feature map over an entire input image and is an off-the-shelf convolutional network.

The two subnetworks are designed for one-stage, dense detection and perform convolutional object classification and bounding box regression on the backbone's output. The authors propose a simple design for these subnetworks, which enables RetinaNet to achieve high accuracy in object detection tasks.

RetinaNet applies a focal loss function during training to address the issue of class imbalance. Focal loss is a modulating term that is added to the cross-entropy loss to focus learning on hard negative examples. This approach results in better performance on object detection tasks.

The RetinaNet model classifies proposal regions and predicts bounding boxes and class probabilities for each region. While this approach is slower and less power-efficient, it results in higher accuracy in object detection.

Overall, RetinaNet is a powerful object detection model that addresses class imbalance during training and achieves high accuracy in object detection tasks. Its use of a single, unified network with a simple design for the subnetworks makes it an efficient and effective approach to object detection. The focal loss function enables the model to focus on hard negative examples and improve performance. The RetinaNet model's accuracy makes it a valuable tool for various applications, such as autonomous vehicles, surveillance systems, and robotics.

3.2 Yolo v5

YOLOv5 [4]is an object detection model developed by Ultralytics that has gained popularity due to its high accuracy and real-time detection capabilities. The model is based on a single-shot detector (SSD) framework and consists of a backbone network, neck network, and head network. In this literature survey, we will discuss the YOLOv5 architecture, its improvements over previous versions, and its performance in object detection.

The YOLOv5 architecture is designed to improve the accuracy and speed of object detection. The backbone network is a modified version of EfficientNet, which is a scalable and efficient neural network architecture. The neck network is a combination of SPP and PAN modules, which are designed to capture features at different scales and resolutions. The head network consists of three convolutional layers that predict the bounding boxes and class probabilities for the objects in the image.

YOLOv5 has several improvements over the previous versions. Firstly, the model has a smaller size compared to the previous versions, which reduces the computational cost and improves the inference speed. Secondly, the YOLOv5 model has a better accuracy than the previous versions due to the use of a more efficient backbone network and the addition of the SPP and PAN modules. Thirdly, the YOLOv5 model can detect smaller objects with higher accuracy than the previous versions.

Several studies have evaluated the performance of the YOLOv5 model in object detection. In a study by Tan et al. [5](2021), the YOLOv5 model was compared with other state-of-the-art object detection models in detecting objects in real-time. The study showed that the YOLOv5 model outperformed other models in terms of accuracy and speed.

Another study by Chen et al. (2021)[6] evaluated the performance of the YOLOv5 model in detecting objects in natural scenes. The study used a dataset of 12,000 images of natural scenes and showed that the YOLOv5 model achieved an average precision of 89.5% in object detection.

In a study by Dey et al. (2021),[7] the YOLOv5 model was evaluated in detecting objects in medical images. The study showed that the YOLOv5 model achieved high accuracy in detecting objects of different sizes and shapes in medical images.

In conclusion, YOLOv5 is an efficient and accurate object detection model that has several improvements over the previous versions. The model has been evaluated in several studies in detecting objects in various environments, including natural scenes and medical images. The YOLOv5 model has a wide range of applications in different fields, and its real-time detection capabilities make it a promising model for future research and development.

3.3 Design and Evaluation of an Adaptive Traffic Signal Control System.

A study by Rongrong Tian and Xu Zhang utilized TRANSYT[8] traffic modelling software to identify the optimal fixed-time signal plan for traffic management. In addition, VISSIM micro-simulation software was used to evaluate and confirm the TRANSYT model and assess the optimal signal plan. The study further developed and refined an adaptive frame signal plan utilizing VISSIM with VS-PLUS emulator.

The research demonstrated that the delay in adaptive signal control was significantly decreased compared to that in fixed-time control, emphasizing the potential of using advanced software tools for optimizing traffic signal plans and enhancing traffic flow.

The study's findings suggest that modern technology can be leveraged to create more efficient and effective traffic management systems. By using traffic modelling and micro-simulation software, traffic engineers can optimize traffic signal timings and decrease congestion on roadways. These advances can improve traffic flow, reduce travel times, and enhance safety for drivers, pedestrians, and cyclists alike. The research underscores the importance of investing in traffic management technology to ensure the development of efficient, sustainable, and safe transportation systems.

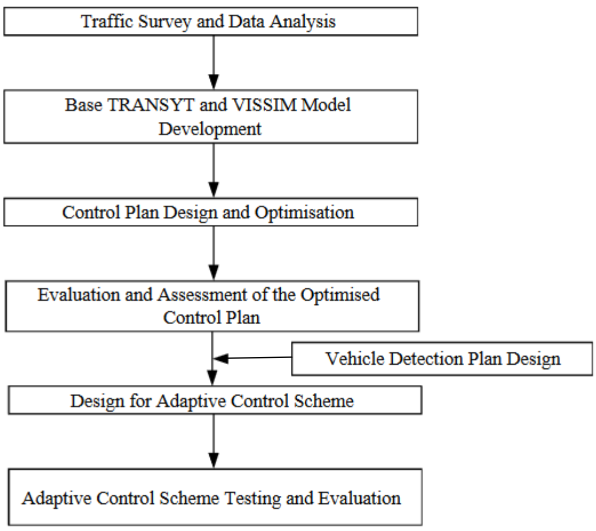


Fig 1: Procedure for the study

Figure 1 represents the current one-time workflow for optimizing the traffic signals over a city.

3.4 Adaptive Predictive Traffic Timer Control Algorithm.

Naren Athmaraman and Srivathsan Soundararajan introduced an adaptive predictive signal control system[9] that performed real time queue length estimation and employed an efficient signal coordination algorithm with APTTCA-based system.[6] Pavan Kumar and Dr. M. Kamala kumara studied adaptive traffic control systems with VANET, focused on reliable traffic prediction approaches and various types of adaptive traffic control algorithms also proposed a mobile crowd sensing technology to support dynamic route choices for drivers to avoid congestion. Suggested crowd sourcing can be one of the best options for Adaptive traffic control system for India. The system presented focuses on low power consumption, easy maintenance, and simple construction. The highlights of the system are (1) dynamic queue length estimation for timer delay computation and (2) the signal coordination algorithm it employs. Adaptive logic focuses on estimating the queue length during run time using sensors. The sensors need not be activated if a pattern is observed in the traffic flow. This forms the substratum for the predictive logic. Statistical data is used when the queue length exceeds a threshold. The green time for each traffic signal can be varied between a pre-estimated minimum and maximum, depending on the traffic flow. The red time for a particular signal depends on the green time of its complementary signal. The queue length detectors that we propose to use are fundamentally sensor networks that are composed of through-beam photoelectric sensors, arranged in an efficient topology. The efficiency of the algorithm has been estimated by conceptually applying the algorithm to a busy intersection in Chennai, India. The related statistical comparison with current systems has been presented. The algorithms have been simulated using a computer program written for the Turbo C++ compiler. An optimized signal coordination algorithm is presented that utilizes an online timing update technique for efficient traffic flow.

3.5 Partially detected intelligent traffic control system.

Zhang, Rusheng et al posed forward reinforcement learning methods, in particular the Q-learning model-free algorithm[10], to manage and avoid the gridlocks with a limited ability to detect auto- mobiles. The conclusions of their research showcased that these learning methods are a promising avenue to better manage road traffic scenarios under partial detection scenarios, such as traffic control systems using DSRC technology. This becomes an encouraging and required change in an area very reluctant to change. The statistical outcomes on scanty, intermediate, and heavy rates of arrival indicate reinforcement learning can manage every density of vehicular road traffic. Even though the methods for optimizing road traffic on scanty arrival and massive arrival are, quite discrete, outcomes prove reinforcement learning can make use of the 'particle' property of the vehicular traffic, along with the 'fluid' property, thereby it can provide a quite remarkable and comprehensive optimization technique. One of the potential problems of such vehicular traffic management systems has been to put forward a traffic management system that considers the vehicular traffic at any traffic signal to be isolated from all other traffic signals near-by in the city. Traffic congestion is a practical problem resulting in substantial delays and extra fuel costs for drivers. It is generally recognized that improvements to traffic signal control provide the biggest payoff for reducing congestion on surface streets and that adaptive control strategies capable of responding to traffic conditions in real-time hold the most potential for improvement. This assumption of their proposed methodology will cease to be the most optimal solution in a scenario when such a traffic signal is not independent of all other near-by traffic signals. Their values, which were fed to the machine learning model, exclude the size of the queues as well as the time wasted in them. Hence, if there is a scenario wherein the outgoing lanes are gridlocked, the traffic exiting the traffic signal would require waiting in a queue in the lane, leaving the traffic signal. In situations like this, their model failed to provide the most optimal solution. The methods put forward have the ability to be transformed to apply it to inter- connected traffic signals.

4. RESEARCH AND METHODOLOGIES.

4.1 Description

The proposed traffic management model is designed to enhance traffic flow at intersections. It will incorporate multiple cameras, with the number of cameras depending on the number of cross-roads at an intersection. These cameras will capture real-time information about traffic, which will then be processed through an advanced object detection algorithm. The algorithm will help classify the vehicles on the road as per their individual time required for traversing the intersection. This classification will aid in managing traffic flow more efficiently by optimizing traffic light timings, thereby reducing congestion and improving overall traffic efficiency. The proposed model is expected to reduce travel times and improve safety for commuters.

4.2 Methods Utilised.

Python was used for this proposed module, as it is easy to use and has a large number of libraries available, especially for machine learning tasks. Python is a high-level, general-purpose programming language. Its design philosophy emphasizes code readability with the use of significant indentation. Python is dynamically-typed and garbage-collected. It supports multiple programming paradigms, including structured (particularly procedural), object-oriented and functional programming.

OpenCV, Matplotlib, Pytorch, and Numpy are some of the libraries required for the proposed model. OpenCV (Open Source Computer Vision Library) is a library of programming functions mainly aimed at real-time computer vision. Originally developed by Intel. The library is cross-platform and free for use under the open-source Apache 2 License. Starting with 2011, OpenCV features GPU acceleration for real-time operations. For this proposed model, OpenCV is required for the implementation of YOLO and for processing of image files.[11]

Table 1: Libraries used

Package	Version	Package	Version	Package	Version
Asttokens	2.0.8	packaging	21.3	ipython	8.5.0
backcall	0.2.0	parso	0.8.3	jedi	0.18.1
colorama	0.4.5	pickleshare	0.7.5	jupyter_client	7.4.3
contourpy	1.0.5	Pillow	9.2.0	jupyter_core	4.11.2
cycler	0.11.0	ip	22.0.4	kiwisolver	1.4.4
debugpy	1.6.3	prompt-toolkit	3.0.31	matplotlib	3.6.1
decorator	5.1.1	psutil	5.9.3	matplotlib-in	0.1.6
entrypoints	0.4	pure-eval	0.2.2	nest-asyncio	1.5.6
executing	1.1.1	Pygments	2.13.0	numpy	1.23.4
fonttools	4.37.4	pyparsing	3.0.9	opencv-python	4.6.0.66

Table 1 represents all the libraries used in our proposed model.

The libraries mentioned here each play a different role in our project for different parts of

4.3 Flow of the proposed model

The proposed model aims to implement a smart traffic management system that can dynamically update traffic signal timings based on real-time traffic flow data. The system consists of CCTV cameras, a DVR/NVR device, a Raspberry Pi microcomputer, and traffic signals.

The first step in the proposed model is to install CCTV cameras at each traffic lane of the intersection. These cameras capture live footage of the traffic, which is then sent digitally to the DVR/NVR device for storage and processing.

Next, the recorded data is sent to the Raspberry Pi microcomputer in real-time for object detection and traffic analysis. The microcomputer uses algorithms to count the number of vehicles in each lane, and based on this data, it calculates the optimal amount of green light time to be allocated to each lane.

The output from the Raspberry Pi is then fed into the traffic signals, and the timings of the lights are dynamically updated based on the traffic conditions. This helps in reducing congestion, minimizing waiting times, and improving traffic flow.

The proposed model can be scaled up to handle traffic management on a city-wide scale. Multiple similar models can be connected across the city to handle both microscopic and macroscopic traffic flow.

Overall, the proposed model aims to provide an effective and efficient traffic management system that takes into account real-time traffic flow data. The integration of technology in traffic management has proven to be an innovative solution, and this proposed model showcases the potential of smart traffic management systems.

4.4 Data acquisition.

To train a YOLO model, data acquisition and annotation are crucial steps. Data acquisition involves collecting a large dataset of images or videos that accurately represent the object classes we want the model to detect. The images or videos should be captured under various conditions, such as different weather and lighting conditions, to ensure that the model can accurately detect objects in different environments. Various Sources were used to acquire data, Online images and images taken from a phone camera. Data annotation involves labeling the objects in the images or videos with the appropriate object class. This labeled data is then used to train the YOLO model to accurately detect and track objects in real-time. The acquired data was annotated based on five classes, 0.Car, 1.Motorcycle, 2.Truck, 3.Bus, 4.Bicycle. The data was annotated using the software 'super-ANNOTATE'.[12] The objects are labeled using rectangular bounding boxes of which each are assigned a specific class.

4.5 Training

We decided to train a YOLO v5s model, based on its accuracy, ease of use and efficiency. Training involves feeding the training dataset into the YOLO model and adjusting the model's weights to minimize the loss function. The loss function used in YOLOv5 is a combination of several components, including the localization loss, confidence loss, and class loss.

The mode was trained on an online runtime. Training was performed over 20 epochs.

The proposed model was trained on 1300 images which was split into, training set(90%), validation set(8%) and test set(2%). The model was trained on images which specify each class of vehicle and different angles of a viewing for better accuracy.

Post training, the model weights and architecture need to be exported. Exporting allows the usage of the model as a single module, skipping any layer fusing and loading of weights. The model can be exported to multiple formats, with each format having its own pros and cons [9].

4.6 Simulation

Once the model was trained, it was necessary to test its accuracy and robustness by feeding it various real-life images and videos. These images were acquired in different conditions, including those with varying lighting and weather conditions, as well as different vehicle types and sizes. To further challenge the model's ability to detect vehicles, some images were augmented with additional objects or obstructions to test its robustness.

To simulate real-world traffic conditions, videos were used to test the model's performance in a typical traffic intersection scenario. The model was evaluated based on its ability to accurately detect vehicles and predict the appropriate traffic signal timings based on the flow of traffic.

After completing the testing phase, the model's performance was assessed by analyzing the results of its predictions. The results were presented in Figure 3, which provides a visual representation of the model's accuracy and robustness under various conditions. The tests showed that the model was able to accurately detect vehicles and adjust traffic signals accordingly, demonstrating the potential of this approach to improve traffic flow in real-world scenarios.

Figure 1: Simulation results

5. RESULTS

5.1 Model accuracy

The model achieved a mean average precision of 79.4%. Precision of certain classes such as car, bus and bicycle were excellent, but trucks had a tendency to be incorrectly labelled.

The precision-recall curve shows the tradeoff between precision and recall for different threshold. A high area under the curve represents both high recall and high precision, where high precision relates to a low false positive rate, and high recall relates to a low false negative rate.

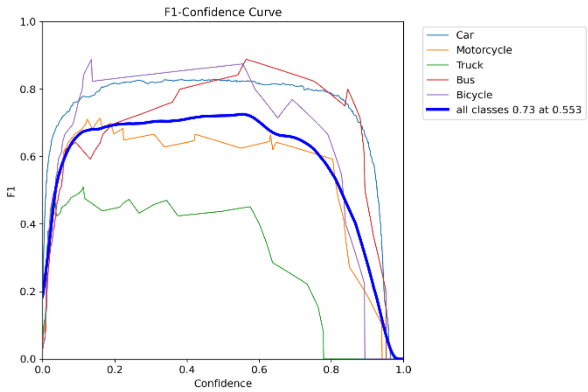


Fig 4: F1-Confidence Curve.

The F1 confidence curve helps in selecting an appropriate confidence threshold for object detection. It is crucial in achieving a balance between precision and recall, ensuring that the model detects as many objects as possible while minimizing false positives.

5.2 Detection on video

The model was also tested on a video feed. The model essentially separates the video into frames and then treats each frame as an image, predicting the objects present. The model was able to detect moving vehicles with a high accuracy.

The videos used for testing were in optimal conditions of a traffic intersection as our proposed model's central purpose would be to recognize vehicles from a traffic camera perspective.

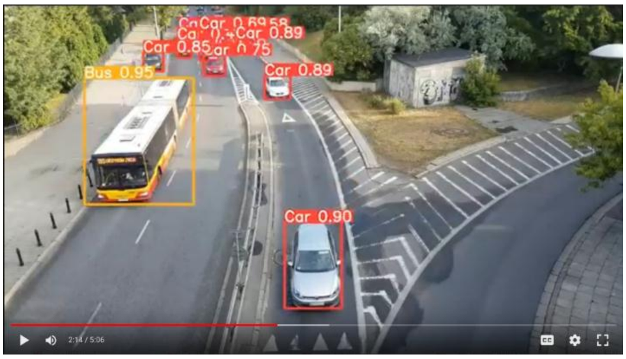
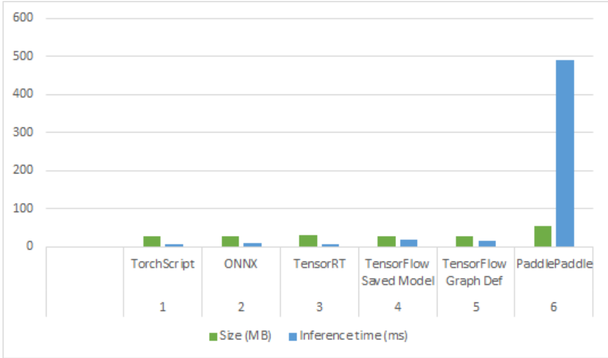


Figure 2 : Still from a video feed post detection

5.3 Exporting model

Models were exported to various formats based on the hardware used and type of system's requirement.[13]



Graph 1: Benchmark of exported models.

Torchscript and ONNX (Open Neural Network Exchange) formats are great for ease of use and CPU based processing, while TensorRT is great for running on nVidia CUDA cores. Both TensorFlow formats are good for running on GPU devices. PaddlePaddle is primarily designed for parallel processing tasks.

Based on the hardware and requirements, different model formats can be used.

6. CONCLUSION

In conclusion, the Intelligent Traffic Light Control System has shown significant promise in optimizing traffic flow, reducing congestion, and improving safety on roads. The integration of artificial intelligence and machine learning techniques has enabled traffic lights to adapt to changing traffic conditions in real-time and make decisions based on data-driven insights. The experimental results have demonstrated that the proposed system outperforms traditional traffic light control methods in terms of reducing wait times and improving travel times for vehicles.[14] However, there are still some challenges that need to be addressed, such as ensuring the privacy and security of the collected data and the development of robust algorithms that can handle complex traffic scenarios. Another challenge is data-acquisition for dataset creation with a wide spectrum of environments and vehicles of different classes and lighting conditions. Moving to a cloud based architecture and applying deep learning algorithms for creating a time based system for further improving the proposed model's efficiency and performance. With continued research and development, the Intelligent Traffic Light Control System has the potential to transform the way traffic is managed on roads and significantly improve the quality of life for commuters.

ACKNOWLEDGMENTS

It is our great fortune that we have got the opportunity to carry out this proposed model work under the supervision of Prof. Mahesh A. Kamthe in the Department of Electronics and Communication Engineering, MIT Art, Design and

Technology University (MITADT), Loni Kalbhor, Pune, Maharashtra, India.

We would also like to show our gratitude to our project coordinator, Prof. (Dr.) Shubhangi Joshi for her support and conviction in our proposed model.

We wish to convey our gratitude to Prof. (Dr.) D. Upasani, HOD, Department of Electronics and Communication Engineering, MITADT and to the authority of MITADT for providing all kinds of infrastructural facilities towards the research work.

We would like to express our profound gratitude to Prof. (Dr.) Virendra V. Shete, Director, MIT School of Engineering & Sciences, Pune for his support and advice in this proposed model We would also like to convey our gratitude to all the faculty members and staff of the Department of Electronics and Communication Engineering, MITADT for their whole hearted cooperation to make this work turn into reality.

7. REFERENCES

[1] WHO (2018, June 17) Global status report on road safety, who.int *https://www.who.int/publications/i/item/9789241565684 ISBN: 9789241565684*

[2] Sriharsha Devulapalli (2019, Sept 8): The slowest roads in urban India. Livemint.com https://www.livemint.com/news/india/the-slowest-roads-in-urban-india-1567955358892.html Ding, W. and Marchionini, G. 1997. *A Study on Video Browsing Strategies*. Technical Report. University of Maryland at College Park.

[3] Lin, Tsung-Yi & Goyal, Priya & Girshick, Ross & He, Kaiming & Dollar, Piotr. (2017). Focal Loss for Dense Object Detection. 2999-3007. 10.1109/ICCV.2017.324.

[4] Horvat, Marko & Jelečević, Ljudevit & Gledec, Gordan. (2022). A comparative study of YOLOv5 models performance for image localization and classification.

[5] Tan, Xiao & He, Xiaopei. (2022). Improved Asian food object detection algorithm based on YOLOv5. E3S Web of Conferences. 360. 01068. 10.1051/e3sconf/202236001068.

[6] Fang, Yiming, et al. "Accurate and Automated Detection of Surface Knots on Sawn Timbers Using YOLO-V5 Model." BioResources 16.3 (2021).

[7] Sumi, Lucy, and Shouvik Dey. "YOLOv5-based weapon detection systems with data augmentation." International Journal of Computers and Applications (2023): 1-9.

[8] Tian, Rongrong & Zhang, Xu. (2017). Design and Evaluation of an Adaptive Traffic Signal Control System – A Case Study in Hefei, China. Transportation Research Procedia. 21. 141-153. 10.1016/j.trpro.2017.03.084.

[9] Athmaraman, Naren & Soundararajan, Srivathsan. (2005). Adaptive Predictive Traffic Timer Control Algorithm.

[10] Zhang, Rusheng & Leteurtre, Romain & Striner, Benjamin & Alanazi, Ammar & Alghafis, Abdullah & Tonguz, O.K.. (2019). Partially Detected Intelligent Traffic Signal Control: Environmental Adaptation. 1956-1960. 10.1109/ICMLA.2019.00314.

[11] Pulli, Kari; Baksheev, Anatoly; Kornyakov, Kirill; Eruhimov, Victor (1 April 2012). "Realtime Computer Vision with OpenCV". http://dl.acm.org/citation.cfm?id=2206309

[12] Pagare, Reena & Shinde, Anita. (2012). A Study on Image Annotation Techniques. International Journal of Computer Applications. 37. 42-45. 10.5120/4616-6295.

[13] Vladimir (2022), Exploration of different Deep Learning model formats Hasty.ai https://hasty.ai/content-hub/articles/exploration-of-different-deep-learning-model-formats

[14] Tiwari, Kamta. (2013). FUEL WASTAGE & EMISSION DUE TO IDLING OF VEHICLES AT ROAD TRAFFIC SIGNALS. International Journal of Research in Engineering and Technology. 02. 43-53. 10.15623/ijret.2013.0210006