Object Detection in Images Using Deep Learning to Build Simulation Models

M. Shleymovich, A. Sytnik, N. Andreyanov(✉), and T. Evdokimova

Kazan National Research Technical University Named After A.N. Tupolev-KAI, 10, K. Marksa st., Kazan 420111, Russia
nvandreyanov@kai.ru

Abstract. This article addresses the issue of traffic congestion in urban areas specifically related to intersection regulation. The second chapter focuses on an analysis of applicable computer vision technologies for traffic flow analysis. The third chapter discusses the selection of a neural network for analyzing vehicles at intersections. The fourth chapter covers data preparation and neural network training. The conclusion summarizes the obtained results. The study explores the use of computer vision techniques to improve traffic management and proposes the application of neural networks for efficient analysis of vehicle behavior at intersections. By leveraging computer vision technologies, it becomes possible to accurately monitor and regulate traffic flow, leading to improved road safety and reduced congestion. The research analyzes various neural network models and evaluates their effectiveness in vehicle detection, tracking, and classification tasks. Experimental results demonstrate promising outcomes, indicating the potential of computer vision techniques in addressing urban traffic challenges. The findings highlight the importance of data preparation and the significant role of neural network training in achieving reliable and accurate results. Overall, the study contributes to the growing body of knowledge on computer vision-based approaches for traffic analysis and offers insights into their practical implementation for addressing traffic-related issues in cities.

Keywords: Computer vision · Image processing · Pattern recognition · Object detection · Image classification · Deep learning

1 Introduction

Transport is one of the key systems of urban infrastructure, which allows the city to fully perform the connecting, communication and support functions [1, 2].

To manage traffic on the transport network of cities, algorithms based on complex models of traffic flows are used. The complexity of such models is due to their high dimensionality. For example, at the simplest intersection, there may be 12 directions for the movement of vehicles. For a section of the road network with 10 such intersections, we are talking about 120 directions and it is necessary to minimize delays in each of these directions, provided that the traffic intensity is constantly changing in time and space.

A. A. Radionov and V. R. Gasiyarov (Eds.): RusAutoCon 2023, LNEE 1130, pp. 454–464, 2024.
https://doi.org/10.1007/978-3-031-51127-1_42

It should be noted that without transport modeling it is impossible to plan the construction of new and modernization of existing transport facilities, housing and business construction, traffic management schemes, emergency response, and the solution of a number of other practical problems.

One of the most significant applications of transport models is the design of intelligent transport systems, the need for which is due to fundamental changes in traffic conditions and traffic management tasks, caused, in turn, by an intensive, explosive growth in the level of motorization.

When roads are loaded at 80% or more, the task of traffic flow management becomes acute. Any overload of the road network in excess of capacity leads to fatal consequences. No coordination algorithms (and corresponding flow models) give a positive result under overload conditions. Effective traffic management in these conditions should ensure that the transport network is loaded to the limit of its capacity and maintain continuous uniform traffic. In other words, the problem of passing as many vehicles as possible changes to the problem of achieving a transport balance between the actual capacity of the road network and the demand for traffic volumes while maximizing the opportunities provided by the geometric parameters of the street network. Such a statement of the problem fundamentally changes the construction of the control system, control algorithms and, accordingly, the models underlying them. First of all, it is necessary to understand that throughput is a variable value, depending on weather conditions, accidents, repair work, etc. When loading up to 60–70%, there is a reserve that smooths out changes in throughput. With a load of 90%, there is no such reserve, and the intelligent transport system must evaluate the current throughput in real time and redistribute flows.

This problem cannot be solved without the use of simulation models based on the study of real traffic flows of the urban road network. To obtain data for building such models, the required physical characteristics are measured.

A promising approach to the construction of procedures for measuring the characteristics of traffic flows is the use of models, methods and computer vision tools that provide automatic information extraction through video image processing. Computer vision technologies make it possible to implement accurate and fast algorithms for solving the problems of detecting and recognizing objects in traffic images and calculating their movement parameters, such as direction of movement, lane, traffic intensity, etc.

2 Application of Computer Vision Technologies for the Analysis of Transport Streams

According to Chebykin, for the first time in Russia, transport and economic surveys were carried out using artificial intelligence as part of the design of the facility “Northern bypass of the city of Perm” (JSC “Institute Giprostroymost”) [3]. Accounting for traffic intensity here was carried out by means of video filming from an unmanned aerial vehicle and subsequent analysis of video materials using the TrafficData software package according to the methodology of transport research.

The development of computer vision and machine learning technologies has made it possible to automate this task. Built on the basis of the latest architectures of convolutional networks, the developed TrafficData software package made it possible to bring the semi-automatic method described by M. R. Yakimov to a fully automated one.

The following functionality is implemented in the TrfficData software package:

1. Traffic intensity is determined by directions. The directions are set by the alignments indicated by the user on the video. Three types of gates are available: inlet, outlet and through.
2. A qualitative analysis of the traffic flow takes into account the requirements of the order of the Ministry of Transport of the Russian Federation [4] and standards for suburban and urban roads [5, 6]. 21 types of vehicles and pedestrians are recognized.
3. The video image determines the instantaneous speed at each point of the trajectory; average speed in the section between the alignments; speed with 85, 95% security.
4. The traffic density for the road section is determined from the video image.
5. The video image determines the length of the queue and the time spent by the car in the queue in the direction in question, as well as the parking time. The criterion for getting a car into the queue is the distance between cars in the clear no more than 5 m.
6. The movement delay time is determined. To do this, first determine the time to overcome the section of the road with free movement. The criterion of free movement is set by the user as the minimum interval between cars at a speed of movement above the threshold value. If there is no data on free movement, then they are determined through the maximum allowed speed on the road section. Further, based on the collected video data, the calculated motion parameters are determined: service level; congestion indicator; time index; buffer index.
7. Upon request, the results of the analysis are displayed in MS Excel.

It should be noted that various video analytical systems are currently used to analyze traffic flows, including those that provide real-time information processing. For example, article [7] states the following. In order to effectively manage traffic and ensure the safety of all its participants, it is necessary to monitor the traffic situation, evaluate the parameters of traffic flows, immediately detect dangerous situations and inform the relevant authorities about this. To solve the problem of analyzing traffic flows, video detectors of vehicles are becoming the most popular. They have the following benefits:

- detection of vehicles by one sensor in several lanes;
- collection of a wide variety of traffic data;
- Possibility of visual observation of vehicles.

If the source of information is a video camera, then two options for implementing data processing algorithms can be distinguished:

On the internal platform of video surveillance cameras—online processing (reducing the amount of transmitted data and, consequently, reducing the requirements for communication channels and a computing server).

On the central computing server—offline processing (significant reduction in requirements for the technical characteristics of video surveillance cameras).

This article presents algorithms for detecting markings, detecting and counting passing cars, detecting stopped cars, implemented using a fixed camera installed above the road. From the point of view of analyzing the intensity of the traffic flow, the algorithm for detecting and counting passing cars is of interest. The algorithm involves setting the area of interest in the image—the sensor. Each lane has its own sensor. Each sensor is divided into two zones (usually an entrance zone and an exit zone), which makes it possible to determine the direction of movement of a passing vehicle, as well as to estimate the speed of movement if the size of the zones in meters has been set. The counter of cars passing through the lane is incremented if the sensor registers the following sequence of events: an object is detected in the entry zone, an object is detected in the exit zone, the object has left the entry zone. The work of the algorithm begins with the estimation of the background in each zone. At this stage, it is important to identify the frames on which there is no movement, for which the number of “moving” points on each frame is estimated. If the number of "moving" points is greater than the threshold, which depends on the resolution of the image, then it is considered that motion has been detected in the zone. Otherwise, there is no movement.

If there is no motion in the area for a given short period of time, then a reference frame is selected to estimate the background. In addition, the operation of the zones is synchronized in order to determine the time when the vehicle left the entry zone and then left the exit zone. Such a check is necessary to exclude the selection of the reference frame when the car stops in the zone. After finding the reference frame for a specified time interval (usually several seconds), the background stability is checked, for which the analysis of the difference between the reference and current frame is already performed.

Once the background evaluation stage is completed, the zone enters the normal operation mode, which consists of the following main steps:

1. Calculation of the difference between the current frame and the background.
2. Threshold processing to determine the number of points belonging to different categories: object, background, shadow, highlight.

Accordingly, separate thresholds are set to determine the points that belong to shadows or highlights. Filtering of such points is necessary to exclude false detections. If a sufficiently large area of the zone is occupied by points assigned to the object, then the zone goes into the “vehicle detected” state. The final decision on vehicle detection is made at the sensor level, as described above.

In [8], the authors say that there are ready-to-use technical tools on the market, but the quality of solving many typical tasks of transport analytics, not to mention promising ones, remains insufficiently high. A large number of false positives and omissions of objects of interest leads to the need to ensure a significant degree of human operator involvement in the operation of the video analytics system.

A deeper analysis allows us to conclude that the main reasons leading to unsatisfactory results of the transport analytics systems are as follows:

- many algorithms for analyzing video clips are not universal enough or are focused on solving problems in a narrow range of scenes and under special observation conditions;
- after the initial setup, the system is not always able to correctly adapt to changing observation conditions;
- many modern video and image analysis algorithms require an excessively large amount of calculations, therefore, in practice, they often prefer simpler and less efficient algorithms.

The construction of a transport analytics system is proposed to be carried out using intelligent cameras that are installed on the necessary sections of roads and communicate with the traffic control center (TCC). In this case, the MCC employees are assigned the following tasks: install and initial setup of cameras, monitor situations of emergency shutdown or breakdown of cameras, process the received statistical data, make decisions on the management and organization of traffic, and monitor the operation of the intelligent transport system.

The problems of processing the entire volume of video data are shifted to the platform of intelligent cameras. Thus, during development, the emphasis is on the computational efficiency of algorithms, which does not prevent their use in offline video sequence processing systems, since the quality of the algorithms is studied.

As an example of video analytics tasks that are quite relevant today, we can consider the detection of people and vehicles and tracking them. It should be noted that the scientific literature considers methods for constructing fundamentally new video analysis algorithms, puts forward interesting ideas related to the specifics of a particular problem. However, most works consider the problems of detection and tracking separately and solve one of them. The paper [9] solves the problem of counting the number of people in a video sequence based on a human head detector in a crowd. In [10], it is proposed to use an additional sensor that generates a scene depth map. Vehicle detection is the focus of such works as [11, 12]. At the same time, there are a number of works where tracking of abstract objects is considered in isolation from detection problems. Among them is an approach based on the use of the Hungarian algorithm [13]. There are works in which the problems of detection and tracking are inextricably linked [14, 15].

To detect and estimate the parameters of objects for a fixed camera, the authors of the article [8] propose to use the algorithm described in [6]. The number of vehicles passing through the controlled area is increased based on the decision of this algorithm. In this case, the speed and length of the vehicle are calculated and its classification is carried out:

(1) passenger vehicles—length up to 5 m;
(2) freight and passenger vehicles of medium dimensions—length from 5 to 7.5 m;
(3) cargo and passenger vehicles of large dimensions—length over 7.5 m.

The accuracy of parameter estimation and, accordingly, classification is degraded due to delays in obtaining images from the camera and segmentation errors. In order to improve accuracy, it is proposed to use tracking algorithms, because the inter-frame shift of points or object segments will be evaluated.

The tracking algorithm is based on the calculation of the optical flow. Tracking starts from the frame on which the basic algorithm detected the entry of an object into the controlled area. Tracking is performed in an area extended along the direction of movement, the length of which is 2–3 times greater than that of the controlled area. The first stage of the algorithm is initialization, which consists in the formation of a list of singular points of the object. The use of singular points is due to a significant reduction in computational costs when calculating the optical flow compared to using all points of the object. To highlight singular points, one of the known detectors can be used, and descriptor calculation is not required. The ORB detector was used in the research. The main stage of the tracking algorithm is to calculate the optical flow for the found set of object points. At this stage, the Lucas–Kanade pyramidal algorithm is applied. The algorithm sequentially refines the estimates of the optical flow velocity when moving from the highest to the lowest level of image representation in a multiscale pyramid. First consider the image with the coarsest resolution. The maximum movement at low resolution tends to occur within a few points. When searching at a more detailed level, the shift estimate obtained earlier is taken into account, the window position is corrected, and the optical flow rate is estimated on a more detailed image. This allows you to gradually get a more accurate estimate of the image shift within the window dimensions. The procedure is performed before moving to the most detailed representation of the image. If it is impossible to obtain a solution to the system of equations, the increase in image detail can be suspended. This situation occurs when the neighborhood of the point is no longer unique, or the effect of noise becomes noticeable. The Lucas–Kanade pyramidal algorithm makes it possible to obtain good estimates of point shifts with noticeable interframe shifts of the object. However, an additional point filtering procedure is required, since when the local illumination and the viewing angle of the object change, the tracking may be unstable, some of the points will move away from the true trajectory of movement.

In [16], the authors argue that adequate modeling of the complex dynamics of urban traffic flows requires the collection of large amounts of data to determine the nature of the relevant models and their calibration. At the same time, the equipment of specialized observation posts is a very expensive undertaking and is not always technically possible [17, 18]. The combination of these factors leads to insufficient factual support for both traffic flow management systems and transport planners, with obvious consequences for the quality of decisions made. As a way to ensure mass data collection, at least for a qualitative analysis of situations, survey video cameras have been used for a long time, broadcasting images to certain situational centers, where the corresponding operators control and manage processes. Quite a few of these survey cameras share their observations in the public domain, which makes them a valuable resource for transport research. At the same time, obtaining quantitative data from such cameras faces significant problems related to the theory and practice of video image processing, which is the subject of this work. The paper explores the practical application of some mainstream neural network technologies to determine the main characteristics of real traffic flows observed by public access cameras, classifies the problems that arise in this case and proposes their solutions. Variants of convolutional neural networks are used to track traffic objects, and ways of using them to determine the basic characteristics of traffic flows are being explored. Simple variants of the neural network are used to automate the generation of training examples for the deeper YOLOv4 neural network. The YOLOv4 network was used to evaluate traffic characteristics (speed, flow density) for different directions from CCTV camera records. The authors obtained a classification accuracy of 99.67% for cars and 98.9% for other objects on road images obtained for traffic flows in Vladivostok.

Thus, computer vision technologies for detecting and recognizing moving objects in the road environment make it possible to provide high-precision measurements based on video image processing. This can be used to obtain initial data when building traffic flow models and optimizing the urban road network [19, 20].

3 Selection of a Neural Network for Analysis of Cars at the Crossroads

Before choosing a neural network model, several models were tested:

1. SSD (Single Shot MultiBox Detector).
2. Faster-RCNN.
3. YOLO (You Only Look Once).

The SSD architecture is based on the VGG-16 high performance network architecture, but lacks fully connected layers. Instead of the original fully connected VGG layers, a set of auxiliary convolutional layers has been added, which allows you to extract features at several scales and gradually reduce the size of the input data for each subsequent layer. The network model [21]. Real-time processing: supported, does not work well with small objects.

In the Faster-RCNN architecture, an image is fed into a convolutional neural network to create a convolutional feature map. Instead of the selective search algorithm used in Fast-RCNN, a separate network is used to predict proposals by region. Real-time processing: supported at high computing power. The network model [22].

In the YOLO (You Only Look Once) architecture, the image is divided into a square grid. For each cell in the network, the CNN outputs the probabilities of the designated class. Cells having a class probability above a threshold value are selected and used to determine the location of the object in the image. Real-time processing: supported. The network model [23].

Preliminary testing was carried out on a trial training test set containing 500 images of intersections with cars, of which 350 were images for training and 150 were images for verification when training models. Class for training 2:

1. Cars.
2. Not cars.

The results of pre-testing after training are presented in Table 1.

Table 1. Pre-test results of the trained test model.

Crossroad name	Sample
Shcherbakovskiy Pereulok—Aydinova	14500
Ivanovskiy Spusk	8500
Universitetskaya—Profsoyuznaya	9000
Kol'tso	3000
Astronomicheskaya—Profsoyuznaya	3000
Profsoyuznaya—Musy Dzhaliya	12500

Accuracy is the ratio of correctly found classes relative to the total number of found classes.

Completeness is the ratio of correctly found classes relative to the total number of required classes.

From the table, you can see that the FasterR-CNN and YOLO models performed better, but the average accuracy of the FasterR-CNN algorithm is slightly higher.

FasterR-CNN has more deep learning, which makes it possible not to lose accuracy as much when obtaining an image with a new environment.

The accuracy of the FasterR-CNN algorithm is higher than other algorithms, but it will perform worse on the image stream.

Conclusion: The YOLO model will meet the requirements for streaming vehicle detection, while slightly inferior in accuracy to the Faster R-CNN model.

4 Data Preparation and Neural Network Training

The selected model was trained on data from the following intersections:

1. Shcherbakovsky lane—Aydinova.
2. Ivanovsky descent.
3. University—Trade Union.
4. Ring.
5. Astronomical—Trade Union.
6. Trade union—Musa Jalil.

The markup was done using the markup tool at supervis.ly. An example markup is shown in Fig. 1.

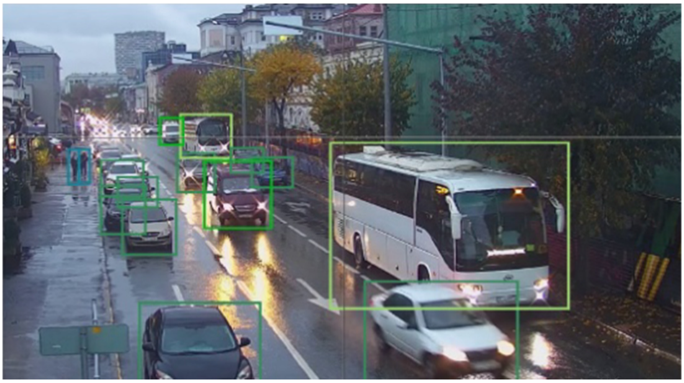

Fig. 1. An example of marking up an image for training.

The number of cars in the sample is presented in Table 2.

At the same time, the classes were divided by number as follows: the number of cars—243381, the number of trucks—2343, the number of minibuses—7088, the number of people—94029, the number of motorcycles—401, the number of taxis—5593, number of buses—12833.

Table 2. Sample for training.

Neural network architecture	Accuracy, %	completeness, %	Speed, FPS
SSD	87.68	76.2	46
FasterR-CNN	90.30	82.8	7
YOLO	90.23	80.05	125

As a result, the model was trained by classes:

1. Cars.
2. Buses.
3. Trucks.
4. Motorcyclists.

The accuracy of the model is 92%, the recall is 83%, the number of frames produced is 20 fps when using an RTX 3060 GPU and 2.5 fps when using an Intel Core i5-11400 CPU.

5 Conclusion

In this article, a study was presented on training a neural network model for a module for collecting data from intersections in order to build simulation models. The results of the study indicate the potential of this model for use in the development of a module for collecting data from intersections for building simulation models.

The study collected data using video cameras at intersections and applied deep neural networks to classify and analyze this data. The training of the Yolov8 neural network allowed us to successfully classify various types of vehicles such as cars, buses, trucks, and motorcyclists. The resulting accuracy of the model was 92%, and the recall was 83%.

These results indicate that the developed neural network model is promising for use in a future module for collecting data from intersections. The use of such a model will make it possible to effectively collect traffic information necessary for building simulation models of transport systems.

For further development and improvement of the model, it may be necessary to conduct additional research and work on the development of a data collection module. It is important to take into account the features of a particular system and adapt the model to it, as well as continue to improve the learning and optimization algorithms of the model to improve its accuracy.

The development of a module for collecting data from intersections using a trained neural network is an important step in the field of simulation of transport systems. This will allow the creation of accurate and realistic simulation models, which will improve traffic safety and efficiency.

Thus, based on the results obtained, it can be concluded that the developed neural network model has the potential for successful application in the future module for collecting data from intersections for building simulation models. Further work on the development of the module will unlock the full potential of this model and its applicability in practical scenarios.

Acknowledgements. The research was supported by the grant from the Russian Science Foundation 22-29-20092, https://rscf.ru/project/22-29-20092/ and the Cabinet of Ministers of the Republic of Tatarstan.

References

1. Gasnikov, A., et al.: Introduction to Mathematical Modeling of Traffic Flows, 2nd edn. MCNMO, Moscow (2013)
2. Gasnikov, A., et al.: Introduction to Mathematical Modeling of Traffic Flows, 1st edn. MIPT, Moscow (2010)
3. Chebykin, I.: Automation of traffic monitoring using computer vision. World Transport (2020). https://doi.org/10.30932/1992-3252-2020-18-6-74-87
4. On Approval of Methodological Recommendations for the Development and Implementation of Measures for Organizing Road Traffic in Calculating the Values of Basic Parameters of Road Traffic: Order of the Ministry of Transport of the Russian Federation (2018). http://docs.cntd.ru/document/552196818. Accessed 21 Feb 2023
5. SP 34.13330.2012: Automobile roads. Streets and roads of settlements. Urban planning design rules (2012). http://docs.cntd.ru/document/1200095524. Accessed 21 Feb 2023
6. SP 396.1325800.2018: (2018). http://docs.cntd.ru/document/552304870. Accessed 21 Feb 2023
7. Ershov, M., Shubin, N.: Image processing algorithms for road traffic analysis tasks. Dig. Sig. Proc. **3**, 63–67 (2017)
8. Alpatov, B., Babayan, P., Ershov, M.: Approaches to detection and estimation of parameters of moving objects in video sequences applied to transport analytics. Comp. Vis. **5**, 746–756 (2020)
9. Subburaman, V., Descamps, A., Carincotte, C.: Counting people in the crowd using a generic head detector. In: IEEE Ninth International Conference on Advanced Video and Signal-Based Surveillance (2012). https://doi.org/10.1109/AVSS.2012.87
10. Kirchner, N., Alempijevic, A., Virgona, A., Dai, X., Ploger, P., Venkat, R.: A Robust People Detection, Tracking, and Counting System. Proc. Australas. Conf. Rob. Autom. **1**, 1–8 (2014)
11. Han, S., Han, Y., Hahn, H.: Vehicle detection method using Haar-like feature on real-time system. Int. J. Electr. Comp. Eng. **11**, 1957–1961 (2009)
12. Wang, H., Zhang, H.: A hybrid method of vehicle detection based on computer vision for intelligent transportation system. Int. J. Multimedia Ubiquitous Eng. **6**, 105–118 (2014)
13. Nandashri, D., Smitha, P.: An efficient tracking of multi-object visual motion using hungarian method. Int. J. Eng. Res. Technol. **4**, 1307–1310 (2015)
14. Andriluka, M., Roth, S., Schiele, B.: People-Tracking-by-detection and people-detection-by-tracking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2008). https://doi.org/10.1109/CVPR.2008.4587583
15. Dehghan, A., Idrees, H., Zamir, A., Shah, M.: Automatic Detection and Tracking of Pedestrians in Videos with Various Crowd Densities. Springer, Cham (2012). https://doi.org/10.1007/978-3-319-02447-9_1
16. Zatserkovny, A., Nurminsky, E.: Neural network analysis of traffic flows in urban agglomerations based on public video camera data. Comp. Res. Model. (2021). https://doi.org/10.20537/2076-7633-2021-13-2-305-318

17. Rakhmatullin, A., Gibadullin, R.: Synthesis and analysis of elementary algorithms for a differential neural computer. Lobachevskii J. Math. **43**, 473–483 (2022). https://doi.org/10.1134/S1995080222050225
18. Cherny, S., Gibadullin, R.: The recognition of handwritten digits using neural network technology. In: International Conference on Industrial Engineering, Applications and Manufacturing (ICIEAM) (2022). https://doi.org/10.1109/ICIEAM54945.2022.9787104
19. Andreyanov, N., Shleymovich, M., Sytnik, A.: Driver assistance system for agricultural machinery for obstacles detection based on deep neural networks. In: International Conference on Industrial Engineering, Applications and Manufacturing (ICIEAM) (2022). https://doi.org/10.1109/ICIEAM54945.2022.9787218
20. Bikmullina, I., Andreyanov, N., Medvedev, M.: Stand for development of tasks of detection and recognition of objects on image. In: International Russian Automation Conference (RusAutoCon) (2019). https://doi.org/10.1109/RUSAUTOCON.2019.8867608
21. Wjddyd66: TensorFlow 2.0 (2020). Website Blog. https://wjddyd66.github.io/tnesorflow2.0/Tensorflow2.0(17)/. Accessed 10 Feb 2023
22. Sefidian: R-CNN, Fast R-CNN, and Faster R-CNN for object detection explained (2020). Website Blog. https://sefidian.com/2020/01/13/rcnn-fast-rcnn-and-faster-rcnn-for-object-detection-explained/. Accessed 15 Feb 2023
23. Mavink: Yolov5 architecture (2020). Website Blog. https://mavink.com/explore/Yolov5-Architecture. Accessed 17 Feb 2023