23rd International Congress on Modelling and Simulation, Canberra, ACT, Australia, 1 to 6 December 2019
mssanz.org.au/modsim2019

Image recognition for vehicle traffic analysis at intersections

M. Namekawa [a] **, S. Kanagawa** [b] **and K. Shinkai** [c]

[a] Department of Management and Echonomics, Kaetsu Universit, Tokyo, Japan, [b] Department of Mathematics, Tokyo City University, Tokyo, Japan, [c] Department of Child Studies, Tokyo Kasei Gakuin University, Tokyo, Japan,
Email: Mitsuhiro@Namekawa.Com

Abstract: We have been conducting research on traffic simulation for traffic jam analysis. Before building a traffic simulation system, it is necessary to perform actual road observations. However, in the past, a great deal of labor has been required to acquire data on vehicles running on the road. In this study, we propose a method to easily obtain car driving data by using image recognition.

In recent years, cameras are becoming smaller and higher performance. Therefore, an image can be taken from a free position. In general vehicle traffic, traffic congestion often occurs due to the flow of vehicles at intersections. Therefore, in this research, we decided to take a bird's-eye view from the top of the building in contact with the intersection. We have taken many images from buildings in the city and made them sample images so that the vehicle can be recognized from the overhead position. And we built a system to track the position of the vehicle by using image recognition technology by machine learning. This system makes it easy to analyze the movement of vehicles at intersections. This vehicle tracking system in the intersection constructed makes it possible to measure the transit time of the intersection. Furthermore, by using this system, the traveling time of a group of vehicles can be automatically measured.

Our experimental system makes it easier to analyze the driving characteristics of vehicles at intersections. Then, we compared the YOLO (Redmon, J., et al. 2016) adopted for the image recognition method this time with the OpenCV cascade classifier used so far, and analyzed the characteristics of each method.

As a result of this research, we will be able to analyze vehicle traffic in detail, so that we can construct an effective traffic simulation, and will be able to perform more accurate traffic congestion analysis.

Keywords: *Traffic flow, image recognition, traffic analysis*

1. INTRODUCTION

Traffic congestion is a global social problem and an environmental problem. In order to solve this problem, we have conducted research on traffic congestion analysis by traffic simulation. In the conventional traffic simulation, first, an actual vehicle driving analysis is performed (Takahashi, M., Namekawa, M., et al. 1992). Then, driving models (microscopic models) of individual vehicles are constructed. However, the construction of this driving model is complicated and very difficult. This driving model also affects road weather and brightness. Furthermore, it differs on weekdays and holidays, daytime and morning and evening. In this way, the operating parameters change from moment to moment, and the optimal parameters are difficult, so the simulation accuracy could not be improved (Satoh, A., Namekawa, M., et al. 1995; Fujisawa, T., Abe, Y., et al. 1996).

For this reason, there have been many studies using neural network theory and fuzzy theory in the driving model in the vehicle traffic simulation constructed so far. For this reason, there have been many studies using neural network theory and fuzzy theory as the driving model in computer simulations (Itakura, N., Honda, N., et al. 1999; Yin, H., Wong, S., Xu, J., & Wong, C. K. 2002). In these studies, it was possible to predict the future by congestion analysis and simulation only in a certain place and time zone. However, as described above, the driving model changes from moment to moment. In these studies, it is difficult to cope with changes in the driving model, so it was difficult to perform wide-area and long-time simulations.

The most difficult point in the microscopic model traffic simulation is the behaviour of each vehicle as described above. Until now, it has not been easy to obtain data (real traffic data) for modelling the target road. Previously, we had a project to measure traffic data on actual roads. At that time, a lot of time and budget were spent, such as measuring the number of vehicles visually or attaching a measuring device to two rental vehicles and running an experiment (Takahashi, M., Namekawa, M., et al. 1992). Therefore, we could not collect actual driving data frequently. In the traffic simulation research so far, there are many problems in acquiring actual traffic data as described above, and the fitting with the actual road could not be sufficiently clarified.

Therefore, we decided to consider a method to easily collect and analyse real vehicle driving data. In recent years, cameras have become smaller, higher performance, and more energy efficient, so it has become possible to shoot roads for long periods of time from free locations. Such a new camera can be installed in a place that does not hinder traffic and can capture the entire movement of the vehicle. Apart from this, since the image recognition technology is improving, there is a high possibility that the movement of the vehicle can be grasped by using the image recognition technology by machine learning.

As a similar research example, there is a study that recognizes the front part of a vehicle and counts the number of vehicles (Billones et al. 2018; Teknomo 2016). There is also research on vehicle recognition centering on images taken from the sky of unmanned aerial vehicles (UAV) (Fujimoto et al. 2014; Ammour et al. 2017). In these studies, the recognition rate can be increased by recognizing the vehicle from the sky or recognizing the front of the vehicle from the side of the road. However, there are also problems such as cost and maintenance for image acquisition. Therefore, we decided to install the camera at a high place in the building that touches the intersection. The vehicle is recognized from the overhead angle.

By proceeding with these researches, it is possible to grasp the movement of the vehicle and to analyze the movement of the vehicle numerically by image recognition simply by pointing the camera at the site of the vehicle traffic.

In order to realize these, we used OpenCV image processing to recognize images obtained from cameras installed at intersections (Hori, T., Namekawa, M., Kanagawa, S., 2018). In order to create a cascade classifier in OpenCV, the object features obtained from the sample image (teacher image) are extracted. Then, it is necessary to learn the characteristics of the object to be recognized according to a specific algorithm. In our previous study, we used about 3,000 images of vehicles on the road as teacher images. Then, we performed image recognition experiments using machine learning algorithms.

In this experiment, we used a new image recognition algorithm called YOLO, and compared the recognition rate with the image recognition in the previous OpenCV cascade classifier.

2. IMAGE RECOGNITION

As mentioned in the previous section, we conducted an image recognition experiment in machine learning using OpenCV. (Hori, T., Namekawa, M., Kanagawa, S., 2018). In our research, we performed image recognition using a cascade classifier of Haar-Like features in OpenCV.

When recognizing an image using machine learning, the system learns using a large number of images to be recognized as samples (teacher images). The result data learned here is called a cascade classifier. By using this cascade classifier, it is possible to find a recognition object when a new image is given. In machine learning, the recognition accuracy of learning results varies depending on the characteristics and quantity of teacher images.

Thus, in recognition by machine learning, a cascade classifier is first generated. The OpenCV image processing library is a typical cascade classifier that supports this object detection function. In order to create a cascade classifier in OpenCV, the object features obtained from the teacher image are used. And it is made to learn so that the characteristic of the object to detect can be extracted by using a specific algorithm.

In OpenCV, there are three types of feature extraction methods. These are Haar-like feature, LBP feature, and HoG feature. We used Harr-Like feature in previous experiments. The reason for this is that we conducted a basic comparison experiment and found that the Haar-Like feature is the most suitable for recognizing vehicles. However, when we built and experimented with the system, the image recognition processing time was long and could not be significantly reduced. Therefore, in the experiment, we made various efforts to make the image recognition process in a shorter time. For example, in order to reduce the data amount of the target image, various measures have been required, such as reducing the image to such an extent that recognition does not deteriorate or using a higher-performance processor.

Therefore, in this study, we decided to adopt a recent new image recognition technique. In recent years, due to the rapid development of deep learning algorithms, high-speed image recognition methods called CNN (Convolutional Neural Network) and YOLO (You Only Look Once) (Redmon et al. 2016) have been developed. Using YOLO in this study, we can recognize images much faster than OpenCV.

3. ROAD PHOTOGRAPHY AND IMAGE RECOGNITION SYSTEM

In this study, we photographed an overhead view of the intersection from a building near the intersection using a micro camera. And the movement of a vehicle is recognized using image recognition technology. The purpose of this study is to grasp the characteristics of traffic flow using the statistical data obtained as a result.

The target location for image recognition was the Kichijoji area in Musashino City, where traffic jams often occur in Tokyo. Because it is an intersection of two roads through which many buses go to the residential area spreading north of Kichijoji Station, the traffic volume is relatively high at nearby intersections. In addition, there are many pedestrians because there is close to the station, and waiting for pedestrians to pass when turning right or left in a vehicle causes traffic congestion. We aimed the camera from the height of the building facing this intersection from about 14m.

In this study, as mentioned above, we use a bird's-eye view image taken from a building near the intersection to recognize and analyze the vehicle. Therefore, the recognition characteristics and the recognition rate vary greatly depending on the image recognition method. Moreover, since the image recognition method is machine learning, it greatly differs depending on the quality of the teacher image to be learned. This is because in machine learning, a computer learns feature quantities from a large number of samples, and classifies and recognizes data (Figure 1).

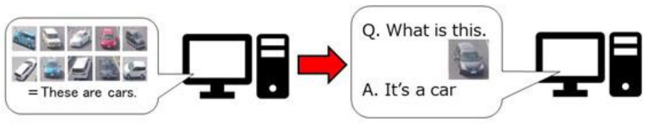

Figure 1. Mechanism of machine learning

3.1. Composition of correct images used in this study

In this research, as mentioned before, we used the image recognition method by YOLO in addition to the Haar-like feature of OpenCV

・CPU ・・・ Intel Core i7 8770k
・GPU ・・・ NVIDIA RTX 2070

Figure 2. Experiment environment

which is the conventional image recognition method. The execution environment of the computer we used is as shown at Figure 2.

3.2. Experimental system

In this vehicle recognition (detection) experiment, we trained a computer as a teacher image taken at the intersection mentioned above. At the intersection, alphabets A to D are attached to the approaching road, and the order is clockwise from the top of the image. image (Figure 3)

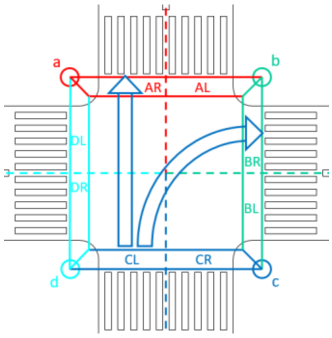

Figure 3. Identification within an intersection

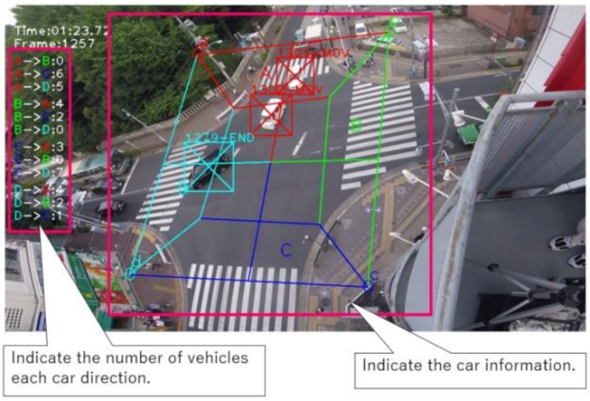

Figure 4. Image recognition system

The display screen of the constructed system is shown in Figure 4. The intersection to be recognized is indicated by a large red frame, and the zebra zone at the entrance of the intersection is indicated by alphabets according to Figure 3. And the statistical data classified by direction of the vehicle which passed each is shown on the screen left side.

4. COMPARISON OF RECOGNITION RESULTS

As described above, machine learning for image recognition has been adjusted so that many vehicles can be accurately recognized. This section describes the results of recognition of the same video by the Cascade classifier and YOLO.

The recognition rate was calculated from the number of recognized vehicles by visual recognition of the results of moving images at the target intersection. First, go straight in Figure 5.

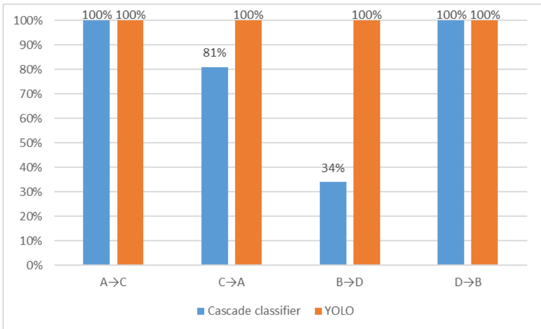

Figure 5. Recognition (straight)

In this rectilinear recognition result (Figure 5), the Cascade classifier maintained a recognition rate of 60-70%, but in the experiment at this intersection, B → D has a low recognition rate of 34%. On the other hand, with YOLO, the result was 100% in all directions. The reason for the poor recognition rate in the Cascade classifier is that the B → D location has a steep (deep)

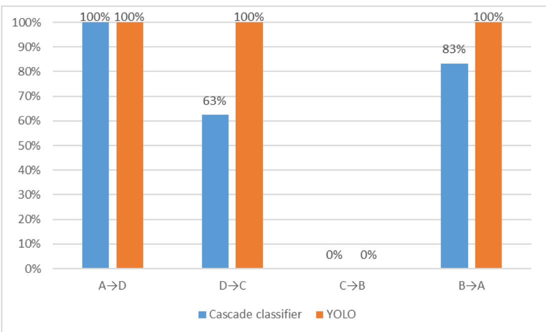

Figure 6. Recognition (turn right)

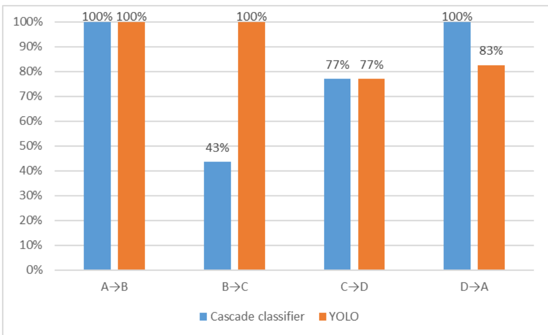

Figure 7. Recognition (turn left)

overhead angle because it is close to the camera, so it is difficult to grasp the side of the vehicle. Therefore, the recognition rate for D → B, which corresponds to the reverse lane, is 100% for both the Cascade classifier and YOLO.

Next, the recognition result in the right turn (Figure 6) is explained. First, C → B is 0% because traveling in this direction is prohibited and no vehicle has traveled. D → C and B → A are characterized in that there are few sides of the vehicle at the angle from this camera, and there are many front or rear and roof parts of the vehicle. Therefore, it can be seen that the Cascade classifier has a low recognition rate from this angle.

Finally, the recognition result of left turn (Figure 7) is explained. B → C is a case where the bird's-eye view angle is deep, as described in the recognition result in the straight line, so it is not easy to recognize with the Cascade classifier. Therefore, the result was quite low at 43%. The recognition rate for C → D is the same for both the Cascade classifier and YOLO, but it is not that the same vehicle cannot be recognized. In the future, we will conduct additional experiments and conduct deeper analysis. In addition, D → A has a low YOLO recognition rate.

5. ANALYSIS RESULT OF INTERSECTION FROM IMAGE RECOGNITION

In this section, we analyze the characteristics of the movement of the vehicle in the intersection from the image recognition results.

Direction 1	Direction 2	trip time
A -> B	Left	2.663
A -> C	Straight	2.721
A -> D	Right	3.063
B -> A	Right	3.104
B -> C	Left	2.153
B -> D	Straight	2.349
C -> A	Straight	2.361
C -> B	Right	2.517
C -> D	Left	2.161
D -> A	Left	2.452
D -> B	Straight	2.452
D -> C	Right	3.264

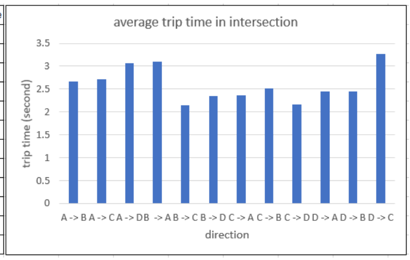

Figure 8. Vehicle travel time by direction

Figure 8 shows the driving characteristics of an vehicle in an intersection. In this graph, the average value of the travel time according to direction in the intersection is shown. A vehicle turning to the right checks the oncoming vehicle and pedestrians on the zebra zone, so it can be seen that it takes 1.1 to 1.3 times longer than straight ahead. In addition, it can be seen that there are many cases where the left turn is shorter than in travel time than straight ahead. Thus, by calculating an average value, a variance value, etc., it becomes possible to create a simulation model in an intersection.

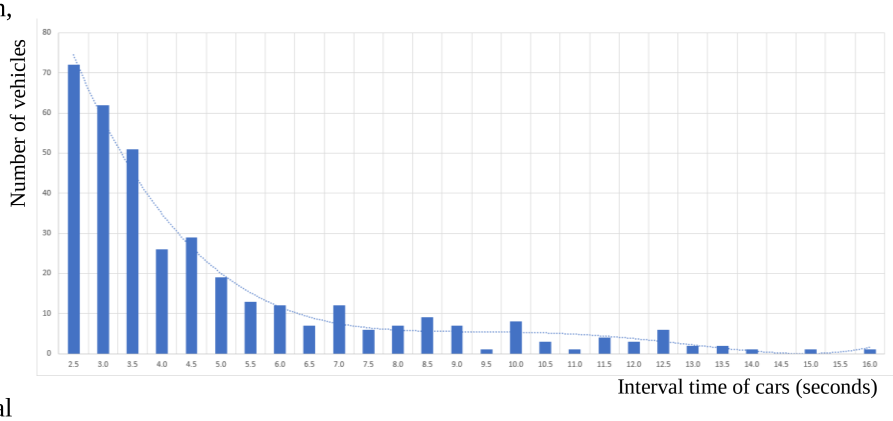

Figure 9. Vehicle entry time interval (seconds) frequency

Figure 9 shows the interval time of vehicles that has entered this intersection. The horizontal axis of this graph is the vehicle interval (seconds), and the vertical axis is the number of vehicles. As can be seen from this graph, the time from the arrival of the previous vehicle to the arrival of the next vehicle is often up to 6 seconds, but it decreases from there and then drops around 12-13 seconds.

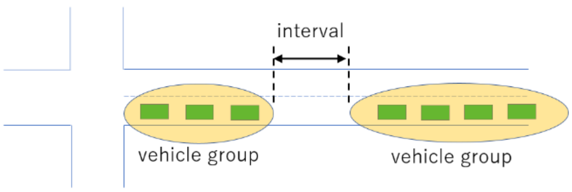

Figure 10. Conceptual diagram of vehicle group

Although it is hard to understand from this graph, when we observe in the video, the vehicle continues to run like a flock, it breaks, and after a while, the next vehicle group arrives. This situation is represented as a vehicle group and its interval as shown in Figure 10.

In this way, as an important index for expressing the characteristics of traffic flow, it was found that the driving characteristics of the vehicle in the intersection can be expressed from the recognition result of the interval between the vehicle groups. In the experimental system in this research, since vehicles are individually recognized and analyzed, a vehicle group can be captured unlike the method of measuring a single passing number. A vehicle group is a group of continuous vehicles moving at regular intervals as shown in Figure 10. By capturing the time distance between vehicle groups and the number of vehicle groups, it was possible to obtain statistical data that was difficult until now.

For example, when an vehicle group is observed at an intersection, if the number of vehicles included in the vehicle group is large and the number of vehicle groups is small, it can be predicted that the number of vehicles stopped at the intersection will increase. In other words, it can be determined that there is a constant traffic jam. On the other hand, when there are many vehicle groups and the number of vehicles included in the vehicle groups is small, it can be predicted that the intersection is relatively free of traffic. Further, when a plurality of intersections are observed at the same time, an appropriate signal timing can be calculated by predicting the timing at which a vehicle group passing at a certain intersection passes another intersection.

Vehicles in urban areas naturally group vehicles. Therefore, it is possible to reduce traffic congestion by grasping and controlling the movement of the grouped vehicles. In actual traffic, system signal control is carried out on the assumption that the vehicle is traveling at a legal speed, but this is not sufficient. It is possible to reduce traffic congestion by capturing vehicle traffic flow with image recognition and performing signal control from the data.

In addition, it was found that it is reasonable to separate the vehicle group at the intersection that we observed from when the time interval of the traveling vehicle passed 12 seconds. This can be seen from the graph in Figure 9.

Figure 11 shows the measured travel interval (seconds) between vehicle groups. As you can see from this graph, there are many groups up to 25 seconds. Further, Figure 12 shows the number and number of vehicles constituting the vehicle group. The horizontal axis of the graph is the number of vehicles constituting the vehicle group, and the vertical axis is the number of groups.

In this graph, it can be seen that there are the largest number of vehicles from one to two. This was the result of the video we analyzed this time because it was daytime when there was no traffic jam.

In addition, at this intersection, after the previous vehicle enters the intersection, the vehicle cannot continue to enter. This is because there are many situations where a vehicle turning right waits for a pedestrian crossing pedestrian. On the other hand, there are many vehicles with 3 to 8 vehicles. These are straight vehicles, and there are no particular obstacles. Therefore, in order to continue to enter the intersection, a large number of vehicles are formed. In the future, we would like to be able to automatically identify these vehicles using image recognition.

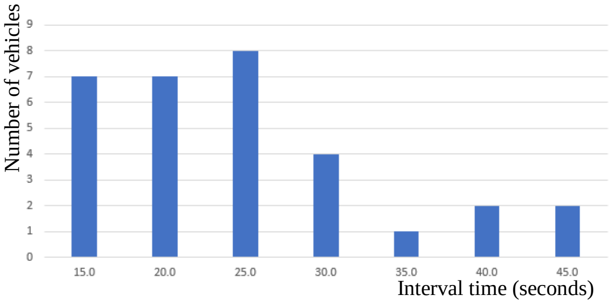

Figure 11. Vehicle group entry time interval (seconds)

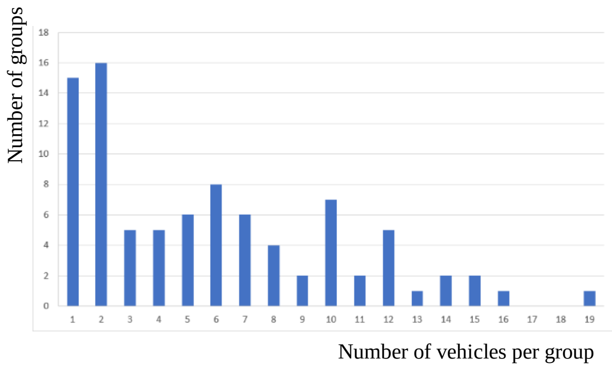

Figure 12. Number of vehicles constituting the vehicle group

6. CONCLUSION

In this study, we performed an image recognition experiment using the OpenCV Cascade classifier and YOLO. As a result, using YOLO, we were able to obtain better recognition results than the Cascade classifier used in the experiments so far.

In addition, by comparing YOLO image recognition results, the weaknesses of vehicle recognition from the overhead angle using Cascade classifiers became clear. Even in YOLO, where the image recognition rate is high, the recognition rate is lower than the Cascade classifier depending on the recognition conditions. In the future, we would like to raise awareness by analyzing this and preparing teacher images that compensate for the weaknesses.

In addition, the analysis of our experimental results revealed that this system can be used to analyze the situation of vehicles at intersections and build a model of vehicle traffic. Then, by automating all of these analyses, it is possible to flexibly reflect the state of vehicle driving that changes from moment to moment depending on the day of the week, the time zone, etc., in the driving model.

ACKNOWLEDGMENTS

Thanks to “Kichijoji Parking Plaza” and “Shiraishi” for providing the shooting location when we collected the traffic data. And, we also thank the students of Kaetsu University Namekawa Laboratory for their help in observation and analysis.

REFERENCES

Takahashi, M., Namekawa, M., et al. (1992). An analysis of road traffic data in a congestion condition for MITRAM. Proceedings of the 11th Simulation Technology Conference, JSST, pp.185-188

Satoh, A., Namekawa, M., et al. (1995). A system verifying the MITRAM model. Proceedings of the International Congress on Modelling and Simulation (MODSIM95), Vol.4, pp.331-336

Fujisawa, T., Abe, Y., et al. (1996). A trial system for verifying the MITRAM models. Proceedings of the 15th Simulation Technology Conference, JSST, pp.153-156

Itakura, N., Honda, N., et al. (1999). Reconstruction of Hierarchical Fuzzy Driving Logic for Road Transportaion Simulator with Using Measurement Data Reference Simulation Method, Proceedings of the International Congress on Modelling and Simulation (MODSIM07), pp.971-976

Itakura, N., Honda, N., et al. (1999). Realization of driving logic of succeeding vehicle with using fuzzy neural network model, Proceedings of the 18th Simulation Technology Conference, JSST, pp.273-281

Yin, H., Wong, S., Xu, J., & Wong, C. K. (2002). Urban traffic flow prediction using a fuzzy-neural approach. Transportation Research Part C: Emerging Technologies, 10(2), pp.85-98.

Hori, T., Namekawa, M., Kanagawa, S.,(2018). Analysis of Overhead View Images at Intersection Using Machine Learning, Intelligent Systems Design and Applications 2018 "Advanced Theory and Applications", pp.783-791

Billones, Robert Kerwin C., et al. (2018). Microscopic road traffic scene analysis using computer vision and traffic flow modelling. Journal of Advanced Computational Intelligence and Intelligent In-formatics 22.5, pp.704-710.

Teknomo, K., (2016). Microscopic pedestrian flow characteristics: Development of an image pro-cessing data collection and simulation model. arXiv preprint arXiv:1610.00029.

Fujimoto, R., et al. (2014). A dynamic data driven application system for car tracking. Procedia Computer Science 29, pp.1203-1215.

Ammour, N., et al. (2017). Deep learning approach for car detection in UAV imagery. Remote Sensing 9.4: p.312.

Redmon, J., et al. (2016). You only look once: Unified, real-time object detection. Proceedings of the IEEE conference on computer vision and pattern recognition. 2016. pp.779-788