Video traffic estimation algorithm

To cite this article: E O Soroka et al 2020 IOP Conf. Ser.: Mater. Sci. Eng. 734 012156

View the article online for updates and enhancements.
Video traffic estimation algorithm

E O Soroka¹, A V Obedin¹, M V Karaseva¹,², V A Bartenev³ and A V Zeitler³

¹Siberian Federal University, 79, Svobodny Av., Krasnoyrsk, Russian Federation
²Reshetnev Siberian State Aerospace University 31, Krasnoyarsky Rabochy Av., Krasnoyarsk, Russian Federation
³JSC "Information Satellite Systems" Academician M.F. Reshetnev " 52, Lenin st., Zheleznogorsk, Krasnoyarsk Krai, Russian Federation

E-mail: alexandr.obedin@icloud.com

Abstract. The paper investigates the evaluating task of the traffic as the registered number of vehicles of one of the three classes (a bus, passenger car or truck) that passed in a certain direction per unit of time, according to data from CCTV cameras in real time. For simplification this problem solving was divided into three components. They are registration of traffic flow objects and their classification, finding trajectories of fixed objects and determining traffic lanes, as well as counting objects in each direction. A comparative analysis of the existing algorithms, software and hardware systems that solve problems similar to the problems was carried out; also, a proprietary approach was proposed. The analysis describes the feasibility for developing their own approach due to the lack of an algorithm that meets all the requirements, such as estimating the traffic on the basis of data from surveillance cameras. Besides, the existing hardware and software systems are of the high cost and complexity. The YOLO artificial neural network was used to recognize objects on the frame in the form of rectangles containing the image of a vehicle. To build tracks, objects are compared frame-by-frame based on a set of characteristics, including a simple (average) perceptual hash and the coordinates of the corresponding rectangle. The developed algorithm for constructing trajectories of moving traffic objects, highlighting lanes of automobile traffic and counting vehicles is proposed as a software module. The testing results concerning the quality of the algorithm are introduced. The main advantages and disadvantages of the proposed approach are highlighted.

1. Introduction
Today, a lot of cities introduce automated traffic control systems [1, 2]. In the process of modeling and optimizing a complex extensive transport system, some problems arise due to unexpected behavior of vehicle drivers on public roads. One of these problems is the question of the of data reliability that fix a traffic situation. The paper considers an algorithm for estimating traffic, which is able to collect data used to build a model of the vehicles movement.

2. Statement of the problem
In general, the problem is set as follows. On the basis of the video stream from the cameras installed in some road network sections, presented as a sequence of images I, it is necessary to estimate the traffic T. Define the traffic T as a number of vehicles of each of the three classes (a bus, passenger car or truck) that passed in a certain direction α per unit of time. All possible directions on the considered section of the road network sections should be determined automatically without any experts.
Formalize the statement of the problem as follows:

$$\hat{T}_{i,k} = F(I_k, I_{k-1}, \ldots, I_{k-p}, a_i), i = 1, m,$$

where $F$ is an operator, which means the sequence of actions, $I$ is a frame of the video stream, $k$ is an index of the current frame, $p$ is a number of previous frames needed to evaluate the traffic, $a$ is a detected direction, $m$ is a number of detected directions.

The input data for the problem solving are frames of the video stream; the output date is a table containing information about the movement of vehicles.

### 3. Overview of existing solutions

The methods for estimating the traffic value ensure the fulfillment of three subtasks to solve the problem set in the previous paragraph:

- Registration of traffic objects and their classification.
- Finding movement trajectories of fixed objects and determining lanes.
- Counting objects in each direction.

The main problem of the existing solutions is that they do not fulfill the task completely. Consider some of the existing methods for estimating the movement process of vehicles in more detail.

The concept of the algorithm proposed in [3] is to isolate moving objects located in a static space by comparing the sequence of frames from CCTV cameras. The current frame is compared with the background, i.e., an image of stationary objects; the contour method is used for select objects selection. Based on the obtained object trajectories, a model of the movement system is constructed.

This algorithm is the most suitable for solving the problem, but its main drawback is that it does not determine a class of recognizable vehicles.

One of the possible solutions is a method proposed in [4]. An intellectual analysis of edge maps and a modification pattern search are used to detect vehicles in the image. Objects recognition occurs using the method of patterns. The images selected by an operator are inputs of the proposed system. Then each image is filtered. After that, the data are collected in the recognition module, where the classification proceeds. The result is vehicle distribution according to classes. However, it is impossible to count vehicles with this algorithm on a specific section, since its functionality offers only the identification and classification of the vehicle.

In [5], road sections in the image are first taken out from consideration. Then with the help of stereo cameras the image is segmented and areas that may be automobiles are detected. Then, features of the image are calculated, fed to the binary classifier in the form of a single-layer artificial neural network (ANN). It determines the presence or absence of the vehicle in the frame. In работе не рассматриваются разные классы автомобилей. After that, for each car on segment $t$, there is the most similar vehicle on segment $t+1$. This information is used to build tracks. The work does not consider different classes of vehicles.

In [6], changes in neighboring frames are studied. If a change level has exceeded the threshold, then it is a vehicle in this area. Then, a three-dimensional model of the vehicle is built for each area. According to this model, the rotation of the vehicle is estimated. Model patterns of the vehicle are predefined. The improved advanced Kalman filter is applied to track and predict traffic using a kinetic model [7]. This algorithm does not provide for traffic lanes and vehicles calculation.

In [8], an adaptive algorithm for distinguishing the background and moving objects based on changes between frames of a video is presented. The author does not consider methods for classifying vehicles with this approach. In addition, such a tracking algorithm will not perform well when the frame rate of the video is reduced to increase the speed of the information processing.

The paper [9] also combined three-dimensional modeling and a Kalman filter for tracking vehicles. The location of the vehicle is determined using a link of SSD [10] and YOLO9000 [11] detectors to increase the robustness of the object detection.
A comparative table of algorithms is given below (Table 1). For better understanding, the algorithms are called a number corresponding to a source from the list of references. In the table, the symbol “+” indicates the existence of a solution to the corresponding item from the first column, “-” indicates the absence of a solution.

Table 1. Comparison table of algorithms.

| Algorithm                   | Number | Algorithm Number |
|-----------------------------|--------|------------------|
| Vehicle recognition         | 3      | 4 5 6 8 9        |
| Vehicle classification      | -      | + - + + +        |
| Tracking the trajectory of each object | - | - + + + + |
| Automatic allocation of traffic lanes and calculation areas | - | - - - - |
| Objects calculation in a certain area | + | - + + - |
| Usage possibility without additional technical tools | + | + - + + |

In other words, there are no any algorithms for traffic estimation that satisfies all the requirements. There exist software and hardware systems in the market for detecting vehicles based on a video stream. The company FLIR proposes one of these solutions [1]. FLIR video, heat and radar sensors are highly accurate technologies developed for motion control. The problem with this solution is the high cost of FLIR technical devices, their complex installation and the need for road marking. The domestic analogue is a video detector of vehicles INFOPRO [2]. This system is easy to install and maintain; besides it is a device in the form of a camera with built-in sensors and software for operation. The disadvantages of the system are similar to the previous version.

4. Proposed Solution

It was decided to develop our algorithm due to the fact that there are no any algorithms satisfying all the requirements that allow estimating the traffic flow based on data from CCTV cameras.

The first step of the algorithm operating is the process of recognizing vehicles in a frame. The implementation of the YOLO (“You only look once”) neural network model from open sources was used for objects recognition in the image and their classification [12]. The operating principle of the artificial neural network is described in [13]. The main advantage of the YOLO architecture for this task is these objects are localized and classified in a single pass through the network. This allows performing frame-by-frame processing, and it makes it possible to process video in real time.

YOLO receives an image \( I_i \), on input, where \( i \) is an index of the current frame, and returns an array of objects \( L \) of some length \( m \) containing the following fields:

- coordinates of the upper left corner;
- length and width of the object in pixels;
- class of object.

At the next stage, an array of objects is developed. In the original image \( I_i \), rectangular areas \( Obj_{ij}, j = 1, m \) that correspond to objects from \( L \) are selected. For each received image, a simple (average) perceptual hash is determined [14]. Thus, each vehicle is described by the following characteristics:

- identifier (id) of the object;
- length and width of the object in pixels;
- coordinates of the center of the object;
• simple perceptual hash;
• number of frames where the object was absent, by default equal to 0.

Then, a comparison of the found objects with objects in subsequent frames takes place. The comparison takes place in two stages. First, the current frame $I_i$ is taken for the studied frame, and coordinates $s_{i-1}$ of the object on the previous frame are taken as the center of the desired object, then vice versa: the studied frame $I_{i-1}$, is a center of the object $s_i$.

For both cases, the search is the same. In the frame under study, $k$ of the nearest centers is searched for relative to $s$ by the Euclidean metric in a certain radius $r$, determined by the formula:

$$ r = (l + 1) \cdot \sqrt{w^2 + h^2}, $$

where $l$ is a number of frames where the object was absent.

After that, the difference in the hashes of the objects $Obj_s$ and $Obj_i$, $i = 1, k$ is calculated, and the most similar object is determined by the following rule:

$$ Obj_s^* = \min\{d_{s,i}, i = 1, k, d_{s,i} < 20\} $$

If a pair $(Obj_s^*, Obj_s)$ coincides at both stages of the search, then an object is marked as found and its center coordinates become $s^*$. Otherwise, this object is identified as a new one.

For objects from the previous frame that were not found in the current frame, the absence frame counter is incremented by 1. Due to the fact that some frames may overlap the object with a larger object in the foreground, the search for detected objects is carried out in several frames. The search for an object is terminated if it was absent for more than 3 frames in a row.

As the information is accumulated on the coordinates of the vehicle centers, clusters are allocated that correspond to the most similar trajectories. Each cluster is characterized by a start and end points of movement, defined as the average value along the X and Y axes for the start and end points of each cluster object, respectively. The detection region for each direction of motion is defined as a region within a certain radius from the end point.

The criterion for stopping the algorithm is the end of the video transmission. The output is a table in csv format containing information about the identified directions and intensity at the end point at a specific point in time.

5. Software implementation and results

YOLO is able to recognize 80 different classes of objects of the real world (people, household items, etc.). To improve the quality of the objects recognition of the three required classes, it was decided to retrain the ANN. The training of a neural network on a dataset with expertly corrected marking of vehicles took place in 2 stages: training of an ANN on a relatively small data set (3000 images) and on a larger one (25000 images). At the end of the training, recognition quality testing was conducted. Testing was carried out on a validation kit, the volume of which amounted to 10% of the total image volume.

Table 2 compares the recognition quality of the three versions: the YOLO implementation used without modification (standard version), trained in the first stage (version 1) and in the second stage (version 2). The quality metric is IoU (Intersection over Union). Data is expressed as a percentage.

The IoU quality metric is determined by the formula:

$$ IoU = \frac{A \cap B}{A \cup B}, $$

A is a true rectangular area with vehicles, B is a recognized area.
Table 2. Recognition comparison of vehicle classes.

|                | Standard version | Version 1 | Version 2 |
|----------------|------------------|-----------|-----------|
| Passenger cars | 17.66            | 17.74     | 86.8      |
| Buses          | 15.87            | 17.13     | 79.23     |
| Trucks         | 9.09             | 9.09      | 53.8      |

Thus, it was possible to improve significantly the recognition quality of objects of the required classes.

The software implementation of the described algorithm for estimating traffic provides the ability to view the visual rendering of markers of objects registered by the surveillance camera. Figure 1 shows a fully marked image, which is a frame from a surveillance camera at the crossroad.

![Figure 1. Demonstration of the recognition algorithm.](image)

To simplify tracking changes in the video, each vehicle is entered in a rectangle. The color of each vehicle corresponds to a specific identification number. The trajectories of vehicles on the basis of data on the centers of marked objects in each frame are shown in figure 2.

![Figure 2. Trajectories of vehicles.](image)
Having determined the lanes of automobile traffic based on the trajectories of vehicles movement, objects are counted at the end point of each lane.

The recognition quality estimation was carried out by experts. The criterion that determines the accuracy of recognition was defined as the ratio of the estimated traffic as a result of the algorithm to the value of the transport flow determined by a group of experts when viewing the video visually.

Five records from surveillance cameras in various sections of the road network were used as a video for the experiments. The first two records contain various disturbances expressed by atmospheric phenomena (snow, rain, etc.). Video parameters are as follows:

- frame rate is 30 frames per second;
- video resolution is 1280 x 720 pixels;
- video duration is 2 minutes.

Table 3 shows the accuracy of objects calculation on each video, averaged in all directions, for each of the three classes of vehicles. The sign “-” indicates the situation when there were no any objects of the corresponding class on the video.

| Class         | Video number | 1   | 2   | 3   | 4   | 5     | On the average |
|---------------|--------------|-----|-----|-----|-----|-------|----------------|
| Passenger cars| 0,848        | 0,903 |   | 0,925 | 0,91 | 0,925 | 0,902          |
| Truck         | 0,8          | 0,75  | 0,875 | 0,6  | 0,75 | 0,755 |                |
| Bus           | 0,714        | 1     | 0,75 | -    | 0,857| 0,830 |                |

According to table 3, the calculation algorithm has a fairly high accuracy. The main reasons of the calculation errors were as follows:

- interference on the video, distorting the object in the next few frames, caused the incorrect object’s mapping;
- incorrect classification of vehicles with trailers;
- incorrect classification of trucks from a front side;
- incorrect classification of buses as two vehicles.

The first error arose because the analysis used video transmitted through the network. The problem can be leveled by improving the quality of video transmission or installing a computing device directly at the crossroad. Error classification is related to the implementation of YOLO. According to Table 2, it was possible to improve significantly the quality of recognition.

If we do not take into account a class of vehicles, the accuracy of the calculation of moving objects in directions is 0.98 on average.

6. Conclusion
The described solution has several advantages.

- The manual marking is not obligatory. The algorithm recognizes vehicles and lanes only on the basis of the video stream.
- Speed of work is higher. Thanks to a simple image comparison algorithm, calculations can be organized in real time directly on equipment installed in the physical location of CCTV cameras. Thus, the problems associated with interference in the video transmission over the network are solved.
- Solutions are flexible. The use of the ANN for object recognition provides an opportunity for further training and improvement of its work. At the same time, it is possible to improve the
image comparison algorithm depending on the computing resources. It will reduce the error classification and error comparison of objects on adjacent frames.

Thus, it is possible to estimate traffic in real time based only on data from CCTV cameras. It will improve traffic lights and increase traffic capacity at crossroads. Moreover, such a solution can significantly save financial resources in implementing this technology in the streets of cities, since the proposed software product can operate at crossroads already equipped with video cameras and does not require re-equipment.

If we consider the current situation in the market of devices and systems for recognizing, tracking and counting the number of vehicles, it can be noted that on one hand there is no shortage of solutions in the market, and on the other hand, they all have certain disadvantages.

The hardware and software systems described in the overview of the existing solutions require special technical equipment. To perform the claimed functions, this equipment uses many sensors and scanners, which reduces its operational reliability, increases the complexity of installation, as well as the recovery time in the event of a failure. In addition, a big disadvantage is high cost of equipment and conversion of intersections with such devices.

Thus, in this paper, new software for determining the traffic of vehicles is presented. It is not necessary to purchase additional equipment in the form of special cameras, sensors and scanners for the operation of the algorithm proposed in the article unlike the offers available in the market.

References
[1] Thermal Imaging, Night Vision and Infrared Camera Systems Retrieved from: https://www.flir.com
[2] Infoprocess Retrieved from: http://www.infoprocess.ru/products.html#infopro
[3] Potekhin A S and Strelnikov A V 2017 The methodology for estimating traffic flow at the intersection according to video surveillance Information Technologies 23(2) 129-34
[4] Subbotin S A 2010 Automatic system for detecting and recognizing vehicles in an image Software Products and Systems 44 114-6
[5] Li X, Yao X, Murphey Yi L, Karlsen R, Gerhart G 2004 Real-time vehicle detection and tracking system in outdoor traffic scenes Proceedings of the 17th International Conference on Pattern Recognition pp 1214-33
[6] Lou J, Tan T, Hu W, Yang H 2005 3-D Model-Based Vehicle Tracking IEEE Transactions on Image Processing 14(10) 1561-9
[7] Tang Z., Wang G, Xiao H, Zheng A 2018 Single-camera and inter-camera vehicle tracking and 3D speed estimation based on fusion of visual and semantic features IEEE Conference on Computer Vision and Pattern Recognition pp 108–15
[8] Chen S, Shyu M, Peeta S, Zhang C 2002 Learning-Based Spatio-Temporal Vehicle Tracking and Indexing for Transportation Multimedia Database Systems Journal of latex class files 1(11) 1–12
[9] Tang Z, Wang G, Liu T, Lee Y 2017 Multiple-Kernel Based Vehicle Tracking Using 3D Deformable Model and Camera Self-Calibration IEEE Smart World NVIDIA AI City Challenge pp 1–6
[10] SSD: Single Shot MultiBox Detector Retrieved from: https://arxiv.org/pdf/1512.02325v5.pdf
[11] Redmon J 2017 YOLO9000: Better, Faster, Stronger / A. Farhadi The IEEE Conference on Computer Vision and Pattern Recognition pp 1–9
[12] YOLO: Real-TimeObjectDetection Retrieved from: https://pjreddie.com/darknet/yolo/
[13] Redmon J, Divvala R, Girshick A, Farhadi 2016 You only look once: Unified, real-time object detection IEEE Conference on Computer Vision and Pattern Recognition pp 779–788
[14] GitHub - JohannesBuchner/imagehash: A Python Perceptual Image Hashing Module Retrieved from: https://github.com/JohannesBuchner/imagehash