Vehicle recognition and its trajectory registration on the image sequence using deep convolutional neural network

Conference Paper · July 2017

DOI: 10.1109/DT.2017.8024333

2 authors:

Dmitry Yudin
Moscow Institute of Physics and Technology

Alexander Knysh
Belgorod State Technological University

Vehicle Recognition and Its Trajectory Registration on the Image Sequence Using Deep Convolutional Neural Network

Dmitry Yudin, Alexander Knysh
Department of Technical Cybernetics
Belgorod State Technological University named after V.G. Shukhov (BSTU n.a. V.G. Shukhov)
Belgorod, Russian Federation
yuddim@yandex.ru

Abstract – **The article shows the methods of vehicle recognition on the image sequence and its trajectory registration. As a recognition algorithm authors used Viola-Jones method with optical flow filter and the deep convolutional neural network in combination with sliding window technique for vehicle detection task. Also authors analyze approaches to registration of detected vehicle trajectories on image sequence based on its linear and angular velocities and Kalman filter. The efficiency of vehicle detection is shown in terms of the precision and recall of recognition. Quality of vehicle registration on the image sequence is estimated by the standard deviation of results from sample values. The article also shows usage prospects of proposed algorithms as a part of driver assistance system and unmanned vehicle control system.**

Keywords – recognition, detection, vehicle, image sequence, registration, trajectory, Viola-Jones method, deep learning, convolutional neural network, driver assistance system.

I. Introduction

Machine vision methods are actively researched in direction of development of driver assistance systems and unmanned vehicles to ensure trouble-free driving and road safety. Vehicle recognition and tracking its trajectory on image sequence is one of such methods. On the first hand, its implementation involves the solution of the vehicle classification and detection problem and, on another hand, registration and prediction of their trajectory on the basis of the video stream obtained from moving cameras. A number of approaches to solve both problems have been developed in the world.

Some methods of machine learning for vehicle detection are constructed on the principle of multi scale sliding window: the method reduces the detection problem to binary classification for each point of the image over a certain rectangular neighborhood. For each rectangular area of the image taken with all possible shifts and scales, the hypothesis of the desired object's presence in the region is checked using a pre-trained classifier.

Good results are shown using cascade schemes to recognize complex objects in images, when the detector is a sequence of cascades of the so-called strong classifiers. In a turn, a strong classifier is built using an algorithm of machine learning, for example, AdaBoost (or some other version of the booster), as a linear combination of weak classifiers. Cascading of strong classifiers allows to achieve high productivity due to fast (at the cascade's first-second level) failure for the overwhelming number of regions that do not contain the desired object. Amount of such "empty" windows is several times higher than windows containing the object. So processing time of the "empty" sub-window differs from processing time of sub-window with the object several times (in proportion to the cascade length). This approach is the basis of Viola-Jones method, which uses Haar features [1-3], but this approach is poorly understood in case of vehicle recognition in different angles and lighting conditions. Classifiers based on the support vector method (SVM), which use histograms of oriented gradients (HOG), are also effective in similar problems [4], but they require a large amount of computation and are not effective in real-time systems.

Modern computing devices (GPU, FPGA) made it possible to solve in real time the recognition objects task on stereo images according to the results of depth maps evaluation [5]. This approach allows you to separate partially overlapped objects, and also effectively limit the areas of interest in the image, for example, according to characteristic dimensions in the physical world. An additional advantage is the possibility of obtaining data for estimating the coarse three-dimensional profile of the road on which the movement is being made. But as a rule, special stereo camera and parallel computing devices are expensive. So it is important to solve this vehicle detection task using monocular camera. For these algorithms information about observable objects movement is obtained as a result of the optical flow analysis, which will allow us to identify areas of interest and / or supplement the results of the operation of the vehicle detectors [6, 7]. However, estimation quality of the optical flow depends on the nature of image textures, speed of movement, information about the motion, and therefore research in this area continues.

Practical works show a significant influence of lighting conditions (time of the year, time of day, sun position relative to the camera, etc.) on the recognition quality (7 basic types of lighting are allocated and marked in the iRoads dataset image base [8]). Compensation of this condition is made by

This article is sponsored by grant of the President of the Russian Federation for state support of young Russian scientists № MK-3130.2017.9 (contract №14.Z56.17.3130-MK) on the theme "Recognition of road conditions on images using deep learning".

configuration of the recognition algorithm parameters. In [9] it is proposed to make "samples" in the "sky" and "road" zones in order to clarify the illumination parameters.

Currently, there are a number of developments and publications related to usage of deep convolutional neural networks in vision systems for the vehicle recognition. The achievements and perspectives of convolutional and recurrent neural networks in the recognition of images, speech, texts and other multidimensional signals are briefly described in the article by Y. LeCun, I. Bengio and G. Hinton in "Nature" journal [10]. They note the effectiveness and significant versatility of deep learning and high interest in applying it in the field of driver assistance systems and unmanned vehicles.

Work [11] considers the usage of convolutional neural networks for the detection of vehicles and road markings on images. Results showed that with increasing distance, the method used by the authors shows a significant decrease in the recognition recall.

The approach to recognition and detection of vehicles on satellite image is proposed in [12]. Results of this study show high accuracy in the detection of vehicles, but they are obtained for a fairly limited training sample for one particular city. At the same time, this approach with some modifications can be used to detect vehicles not just the top view but in other view angles.

The construction of a multiclass classifier for detecting various objects on images is described in the article by authors from Google [13], which showed the need to improve the quality of object recognition.

A solution of image classification problem to 1,000 classes (including cars) of the ImageNet dataset was proposed in [14] on base of convolutional neural networks. It notes the need to further improve applied architectures and training methods of the network.

When an object (vehicle) is found, it is necessary to track its trajectory, on the first hand, to prevent a collision with the observer, and on another hand, not to lose this object in case of error of the second kind during the recognition algorithm execution.

For the vehicle trajectory restoration in the dissertation [15] a method is suggested that it involves the integration of object keypoints' trajectories to construct the whole object's trajectory.

Development of driver assistance system to evaluating and predicting the vehicle trajectory according to the information obtained from the three-dimensional machine vision system is presented in [16]. During estimates calculation authors have used the model of the car's movement on which the camera was installed and the detecting and tracking method of wheels centers of the vehicle hidden from the driver.

The application of an extended Kalman filter to prevent collisions of an unmanned aerial vehicle with moving objects is described in [17]. In this case, the optimal trajectory of these objects is calculated. Then it is used in the aircraft's motion model. Results and effectiveness of the method are confirmed with the simulation. In addition, it is possible to use that method for a ground robotic vehicle.

In [18] a probabilistic prediction model of road incidents is proposed using three-dimensional tracking of the vehicles trajectory according to their motion patterns. The fuzzy self-organizing neural network is trained to identify patterns of vehicle behavior on sections of its trajectories. Each section of the trajectory is associated with the vehicle behavior pattern, on the basis of which the probability of an accident (collision) on the road is calculated. Usage of motion patterns allowed the authors of [19] to predict the vehicle behavior for a few seconds in advance. The probability of a new vehicle position is calculated using the history of traffic patterns of this vehicle. However, it is noted that developed approach allows to predict not a new position itself but the probability distribution of possible positions of the observed vehicle. In a similar paper [20] aimed at the development of advanced driver assistance systems a method is proposed that combines vehicle trajectory prediction based on the existing motion model with constant yaw and acceleration and vehicle maneuver recognition. However, to consider all possible conditions and a lot of types of maneuvers, further researches in this direction required.

II. Task Formulation

In this article we consider an algorithm that provides vehicle detection and its trajectory registration. It involves the following stages:

1) reading an image from a sequence of images or capturing a new image from a monocular camera;

2) detection of the object (vehicle, participant of the road traffic) by one of methods or algorithms, using multi scale sliding window technique and providing acceptable recognition precision and recall, for example, by Viola-Jones method or deep convolutional neural network. Objects can be vehicles of various types, for example cars, trucks. Under the detection we mean finding coordinates of the object bounding box. Overlapping of the vehicle bounding box found by algorithm and real vehicle bounding box should exceed 50%;

3) finding estimates of the new position and size of the object bounding box by one of effective methods;

4) usage of the found estimates as the predicted position of the object on the next image and registration of refined coordinates and sizes of the object bounding box in the current frame. If on the next few images the object disappears and then appears again, the algorithm still remains functional, because for these images coordinates and sizes of the object bounding box are used as the measured coordinates and sizes until the disappearance.

If the object is not found on more than n following images, this indicates a loss of the object from the scope and algorithm termination.

III. Vehicle Recognition on Image Sequence

For the vehicle detection task, statistical classifiers based on Viola-Jones method and deep convolutional neural network are used in a combination with sliding window technique.

The quality of the vehicle model depends on the power of the training sample for statistical classifiers. The model is reliably trained only with a sufficient number of images corresponding to the possible vehicle appearance and lighting conditions. When the model hits the environment with vehicles, an appearance of which differs significantly from the training sample, the model needs to be retrained. When creating a vehicle model we were considered vehicles in an arbitrary perspective (the training sample contained 12,000 positive images (vehicles) and 20,000 negative images (non-vehicles), the testing sample contained 7,000 positive images (vehicles) and 10,000 negative images (non-vehicles)). A fragment of the training sample is shown in Fig. 1. Testing image dataset contains 7,000 vehicles and 7,000 negative images.

Viola-Jones method uses concepts of integral image, Haar features and their cascading classification using AdaBoost [1]. In this paper, classical Viola-Jones method of image classification in sliding window is supplemented by using the Lucas-Kanade-Tomashi method for optical flow calculation [21]. The construction of such filter for our task is described in [7]. Usage of an optical flow in this case increases the image recognition recall by detecting missing vehicle bounding rectangles (boxes) on image sequence. However, at the same time, it leads to false positives and lowers the recognition precision, and also increases the processing time per frame.

To eliminate false positives of such modified method of Viola-Jones, an attempt was made to apply an additional geometric filter for vehicles bounding rectangles, based on determining the horizon line and estimating the distance to objects [3]. It led to an increase in recognition accuracy on the training and test samples, but a significant decrease in recall, as shown in Table 2.

Also for image classification in sliding window we have used deep convolutional neural network which architecture is shown in Table 1. As loss function we have used binary cross entropy. Implementation of this network was made on python 3.5 using Keras [22] and Tensor Flow [23] libraries.

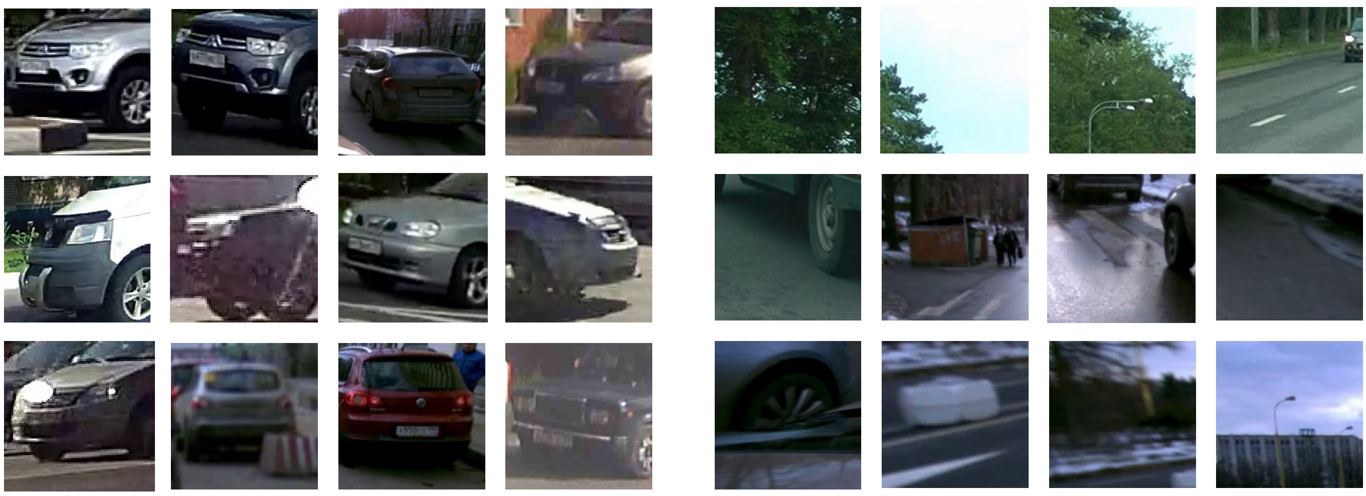
TABLE I. DEEP CONVOLUTIONAL NEURAL NETWORK ARCHITECTURE

No.	Type of Layer	Details
1.	Input Layer	24×24 neurons
2.	2D Convolutional Layer with ReLu Activation Function	3×3 core, 1×1 strides, 128 output maps with sizes 22×22
3.	2D Max Pooling	Pool size 2×2, stride 2×2, 128 output maps with sizes 11×11
4.	2D Convolutional Layer with ReLu Activation Function	3×3 core, 1×1 strides, 128 output maps with sizes 8×8
5.	2D Max Pooling	Pool size 2×2, stride 2×2, 128 output maps with sizes 4×4
6.	2D Convolutional Layer with ReLu Activation Function	3×3 core, 1×1 strides, 256 output maps with sizes 2×2
7.	2D Max Pooling	Pool size 2×2, stride 2×2, 256 output maps with sizes 2×2
8.	Dense (fully connected) Layer with ReLu Activation Function and Dropout	100 neurons, dropout with probability 0.5
9.	Dense (fully connected) Layer with Sigmoid Activation Function	1 neuron

Results of image classification for different methods are given in Table. 2.

Results of vehicle detection are estimated using three measures of evaluation: Precision, Recall, F-score [24].

By analyzing results of the vehicle classification method we can conclude that the recognition quality is at an acceptable level only for the deep convolutional neural network which gave precision of 90.16% and recall of 88.93% on testing set. So results of the deep convolutional neural network allow use it as inputs to algorithms of vehicle trajectory registration and prediction. But it also indicates the need for additional research for improving recognition quality.



a b

Figure 1. Fragment of training set with vehicles in an arbitrary perspective: a – positive images, b – negative images

TABLE II. VEHICLE CLASSIFICATION RESULTS

Type of classifier	Training set			Testing set		
	Precision	Recall	F-score	Precision	Recall	F-score
The usage of Viola-Jones method and optical flow filter without the filter of false positives	0,8005	0,6971	0,7453	0,7720	0,7309	0,7509
The usage of Viola-Jones method and optical flow filter with the filter of false positives	0,8570	0,6910	0,7650	0,8530	0,6560	0,7420
Proposed deep convolutional neural network	0,8971	0,8853	0,8912	0,9016	0,8893	0,8954

An example of vehicle detection using Viola-Jones method and optical flow is shown in Fig. 2.

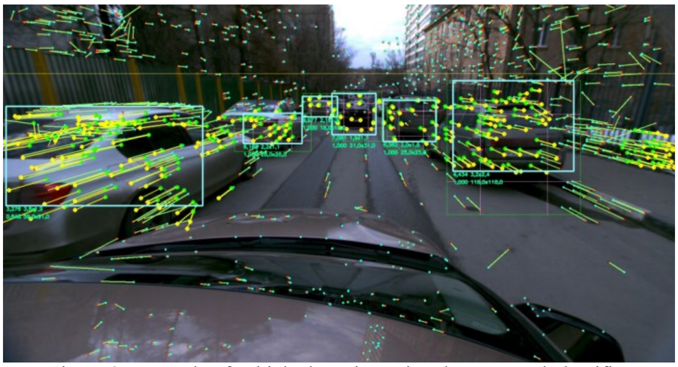


Figure 2. Example of vehicle detection using the proposed classifier

IV. TRAJECTORY REGISTRATION OF DETECTED VEHICLES

The vehicle movement trajectory is defined as a set of their positions at times t_i, i = 1, 2,…, n, n – number of frames available in the video sequence. An object position detected on the image is determined by the bounding rectangle (box) $r = (x, y, w, h)$, where x and y – coordinates of the rectangle center, w and h – its width and height in pixels.

The registration and prediction model of the vehicle position should provide a return of complex information about the current (x_{ip}, y_{ip}, w_{ip}, h_{ip}) position of vehicle found in t_i-th moment of time and thereby ensure its trajectory registration.

To predict the trajectory of an object recognized on the image, an algorithm based on an estimation of its linear and angular velocities can be used. A simplified diagram of the observed object motion is shown in Fig. 3.

Here we use the following notations: (x_i, y_i) – coordinates of the center of the object found on i-th frame, v_i – linear object velocity on i-th frame, ω_i – angular object velocity on i-th frame, h_i u w_i – sizes (width and height) of the vehicle bounding box.

A distance between centers of the object on the current i-th frame and previous (i-1)-th one is determined on the basis of Euclidean distance on the basis of formula

$$L_i = \sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2} \quad (1)$$

An angle of the object movement direction on the i-th frame is calculated as

$$\varphi_i = \arccos\left(\frac{x_i - x_{i-1}}{\sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2}}\right) \cdot sign(y_i - y_{i-1}) \quad (2)$$

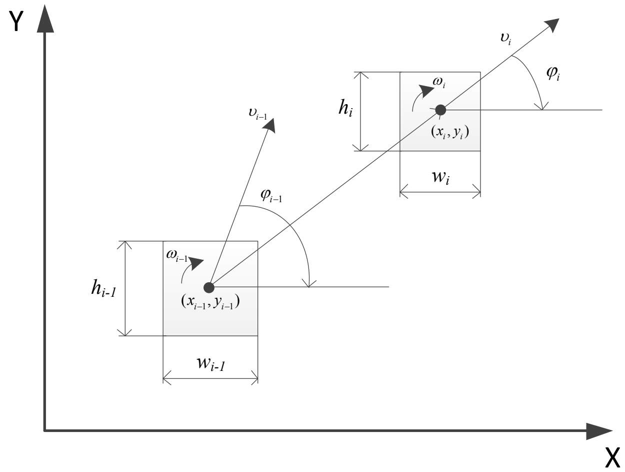


Figure 3. Simplified diagram of the observed object motion taking into account it's linear and angular velocities

To register object's position we propose the model a simplified diagram of which is shown in Fig. 4. It uses as input data the object position, found on the basis of the detection (recognition) algorithm, on the current frame (x_i, y_i, w_i, h_i) and previous three frames (x_{i-1}, y_{i-1}, w_{i-1}, h_{i-1}), (x_{i-2}, y_{i-2}, w_{i-2}, h_{i-2}), (x_{i-3}, y_{i-3}, w_{i-3}, h_{i-3}) forming an input vector V_1.

At the output of the model the registered (x_{ip}, y_{ip}, w_{ip}, h_{ip}) vehicle position in t_i–th moment is formed. This position is the output vector O_{1i}.

Thus the mathematical model is a functional relationship expressed by the formula $O_{1i} = f_1(V_{1i})$. An evaluation of coordinates of the object is calculated from formulas

$$\begin{aligned} \hat{x}_i &= x_{i-1} + L_{i-1} \cdot \sin(2 \cdot \varphi_{i-1} - sign(x_{i-1} - x_{i-2}) \cdot \varphi_{i-2}), \\ \hat{y}_i &= y_{i-1} + L_{i-1} \cdot \sin(2 \cdot \varphi_{i-1} - sign(y_{i-1} - y_{i-2}) \cdot \varphi_{i-2}), \end{aligned} \quad (3)$$

where $(\hat{x}_i, \hat{y}_i)$ – estimation (prediction result) of coordinates of the object at the i-th step (frame), computed using three previous points (i–1), (i–2) and (i–3), L_{i-1} – Euclidean distance between points (i–1) and (i–2), calculated by the formula (1), φ_{i-1} and φ_{i-2} – angles of the object movement direction respectively at (i–1)-th and (i–2)-th frames, calculated by the formula (2).

Registered coordinate values (x_{ip}, y_{ip}) are calculated as the average value between the forecast and the value found on the i-th frame

$$x_{ip} = \frac{(\hat{x}_i + x_i)}{2}, \; y_{ip} = \frac{(\hat{y}_i + y_i)}{2}. \quad (4)$$

978-1-5090-5689-7/17/$31.00 ©2017 IEEE

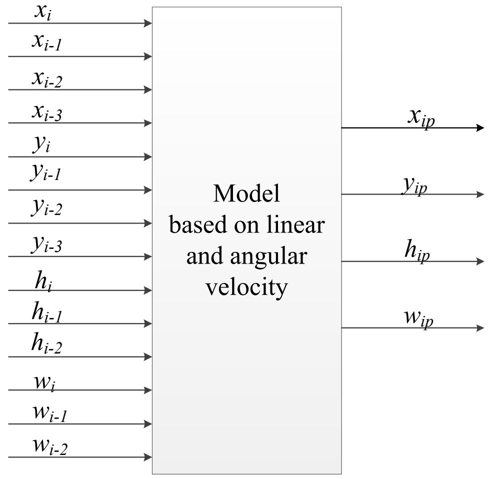


Figure 4. The scheme in the black box form of a mathematical model for object position registration on the basis of its linear and angular velocities

Estimation (forecasting) of the size of the object ($\hat{w}$, $\hat{h}_i$) is carried out as its linear extrapolation:

$$\hat{w}_i = 2 \cdot w_{i-1} - w_{i-2}\,,\ \hat{h}_i = 2 \cdot h_{i-1} - h_{i-2}. \tag{5}$$

Object size values (w_{ip}, h_{ip}) which are subject to registration are calculated as average values between the forecast and the value found on the i-th frame

$$w_{ip} = \frac{(\hat{w}_i + w_i)}{2},\ h_{ip} = \frac{(\hat{h}_i + h_i)}{2}. \tag{6}$$

Found position (x_{ip}, y_{ip}, w_{ip}, h_{ip}) clarifies vehicle coordinates and dimensions obtained as a result of the image recognition algorithm. So it reduces noise and improves the quality of object detection.

To register the detected vehicles position we can also use the model using advanced Kalman filter [25], which is schematically shown in Fig. 5.

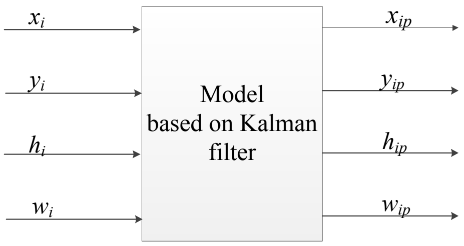


Figure 5. The scheme in the black box form of a mathematical model for object position registration on the basis of the advanced Kalman filter

As the input data the object position (x_i, y_i, w_i, h_i) found by the detection algorithm on the current i-th frame is used. It forms an input vector V_{2i} =[x_i, y_i, w_i, h_i]^T. At the model output the registered vehicle position (x_{ip}, y_{ip}, w_{ip}, h_{ip}) is formed in the t_i–th moment. It forms the output vector O_{2i} = [x_{ip}, y_{ip}, w_{ip}, h_{ip}]^T.

Thus the mathematical model is a functional relationship expressed by the formula $O_{2i} = f_2(V_{2i})$. At the next i-th step before the arrival of measurement results Y_i (Y_i = (x_i, y_i, h_i, w_i)^T) in Kalman filter, new object position is evaluated (prediction of the state vector) in accordance with an expression

$$\hat{X}_{i|i-1} = \hat{X}_{i-1|i-1}. \tag{7}$$

This kind of estimation is due to the fact that the motion law of found object is unknown beforehand, therefore transition matrix F and control matrix B are accepted as single ones, and control vector U is zero [25]. Here, the matrix of a priori estimation $\hat{X}_{i|i-1}$ is composed of predicted values of the object coordinates and sizes $\hat{X}_{i|i-1} = (\hat{x}_i, \hat{y}_i, \hat{h}_i, \hat{w}_i)^T$. Matrix $\hat{X}_{i-1|i-1}$ is composed of estimates of the object coordinates and sizes of in the previous step $\hat{X}_{i-1|i-1} = (\hat{x}_{i-1}, \hat{y}_{i-1}, \hat{h}_{i-1}, \hat{w}_{i-1})^T$.

New covariance matrix (a priori error estimation) is calculated as

$$P_{i|i-1} = F \cdot P_{i-1|i-1} \cdot F^T + Q. \tag{8}$$

Since during object detection the probability of imposing random systematic errors is small, then elements of the covariance matrix-column Q are assigned a value equal to one pixel.

By an a priori estimate of the state $\hat{X}_{i|i-1}$ from (7) we can calculate the measurement forecast:

$$\hat{Y}_i = H \cdot \hat{X}_{i|i-1}. \tag{9}$$

The measurement matrix H is selected as a unit.

After the next measurement is received Y_i = (x_i, y_i, h_i, w_i)^T a forecast error of the i-th measurement is calculated by the formula

$$E_i = Y_i - H \cdot \hat{X}_{i|i-1}. \tag{10}$$

Then the state evaluation is corrected by selecting a point lying somewhere between the initial estimation $\hat{X}_{i|i-1}$ and a point corresponding to a new dimension Y_i:

$$\hat{X}_{i|i} = \hat{X}_{i|i-1} + G_i \cdot E_i, \tag{11}$$

where G_i – matrix of filter coefficients. This adjusted estimation is the registered position of the object $\hat{X}_{i|i} = (x_{ip}, y_{ip}, h_{ip}, w_{ip})^T$.

Finally the estimation of the covariance matrix of the state estimation error is corrected:

$$P_{i|i} = (I - G_i \cdot H) \cdot P_{i|i-1}, \tag{12}$$

where I – unit matrix. Covariance matrix of measurement forecast error E_i is calculated by the formula:

$$S_i = H \cdot P_{i|i-1} \cdot H^T + R, \tag{13}$$

and the matrix of filter coefficients at which the minimum error of the state estimation is reached is calculated as

$$G_i = P_{i|i-1} \cdot H^T S_i^{-1}. \tag{14}$$

Elements of the covariance matrix-column of measurements R are also set equal to one pixel because of the low probability of measurement errors.

Actions are repeated for each new value of the vector Y_i. At the initial time $\hat{X}_{0|0}$ = (x_0, y_0, h_0, w_0)^T, $P_{1|0}$ = 0.

During Kalman filter operation the center coordinates and sizes of the vehicle bounding box on the new frame are calculated on the basis of all previous information about the object's movement (by analogy with the integration operation over the entire time interval). It increases the registration accuracy in comparison with the method based on linear and angular velocity estimates.

Both models of object's trajectory registration have limitations on the case when the movement direction of observed vehicle changes sharply on the new frame. It is rarely observed in the case of high frame processing speed, for example, more than 20 frames per second.

The proposed mathematical models based on linear and angular object velocities and Kalman filter were implemented in Matlab environment. These implementations were tested in six cases of vehicle behavior (width and height for each vehicle bounding box was assumed to be the same, so we considers square boxes):

1) overtaking the observer on the left by the vehicle,

2) appearance and approach on the right of a slowly moving vehicle in the transverse traffic direction,

3) overtaking the observer on the right by the vehicle,

4) following the observed vehicle,

5) beginning of the vehicle overtaking on the left by the observer,

6) overtaking the vehicle on the right by the observer (shown in Figure 6).

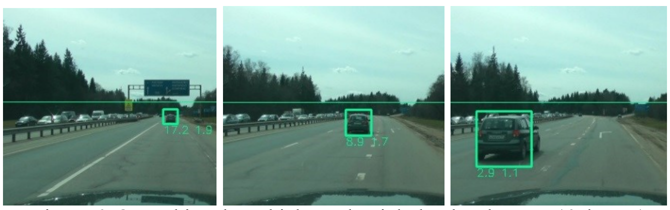


Figure 6. Overtaking the vehicle on the right by the observer (6-th case)

Registration results of the center coordinates and sizes of the vehicle bounding box for case 6 are shown in Fig. 7-8. The solid line shows coordinates or sizes of the region obtained by the vehicle detection algorithm on image sequence. The dashed line is registered coordinates or sizes obtained by the algorithm on the basis of the linear and angular object velocity. Bar-dashed line – registered coordinates or sizes obtained using an algorithm based on Kalman filter.

An application of both proposed models yielded adequate results, and the model in which Kalman filter is applied provides smaller “emissions” on graphs. This feature is required in real systems of trajectory registration and prediction of the observed objects, when a sharp change in position is usually the result of inaccurate vehicle detection.

As a measure of comparison of developed models, the following are applied:

- standard deviation (σ_1) of registered object coordinates (x, y) from the coordinates obtained as a result of image recognition, and

- standard deviation (σ_2) of the registered object sizes (w, h) from sizes obtained as a result of image recognition.

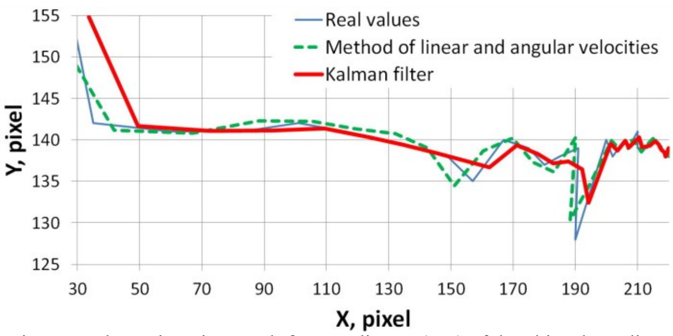


Figure 7. The registration result for coordinates (x, y) of the object bounding box center for the 6th case

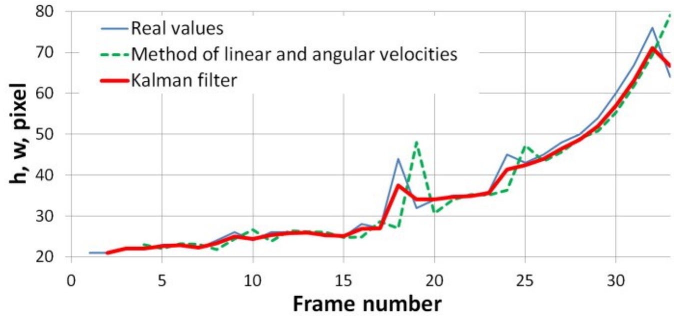


Figure 8. The registration result of the object bounding box sizes (w, h) for the 6th case

Results of calculation measures σ_1 and σ_2 for both models for all 6 cases are presented in Table 3.

TABLE 3. RESULTS OF CALCULATION OF STANDARD DEVIATIONS σ_1 AND σ_2 FOR BOTH MODELS FOR ALL 6 CASES

Algorithm of trajectory registration	Registered value	Standard deviation (σ_1 or σ_2), pixels.						
		Case 1	Case 2	Case 3	Case 4	Case 5	Case 6	Average value
Algorithm for object trajectory registration based on its linear and angular velocities	The coordinates of the center (x, y) of the object bounding box	4,79	32,04	4,20	12,37	10,96	4,33	10,74
	Sizes (w, h) of the object bounding box	5,79	20,71	2,24	5,8	4,24	4,13	6,71
Algorithm for object trajectory registration based on the Kalman filter	The coordinates of the center (x, y) of the object bounding box	7,91	20,04	4,82	8,26	9,32	5,84	8,75
	Sizes (w, h) of the object bounding box	3,39	7,38	1,92	2,28	3,14	2,10	3,85

On the basis of this table we can conclude that both standard deviations σ_1 and σ_2 are lower on average for the algorithm of object trajectory registration based on Kalman filter and are at an acceptable level for use in driver assistance systems.

V. CONCLUSION

In this research two algorithms for the vehicle detection (recognition) on image sequence are proposed. The first one represents a combination of Viola-Jones method and Lucas-Kanade-Tomashi method for optical flow calculation. Second one is a deep convolutional neural network with multi scale sliding window technique. An implementation of the first method gave precision of 85.30% and recall of 65.6% on testing set. Second algorithm gave precision of 90.16% and

 978-1-5090-5689-7/17/$31.00 ©2017 IEEE

recall of 88.93% on testing set. So results of deep convolutional neural network are better and allow us use it as inputs to algorithms of the vehicle trajectory registration.

We have analyzed effectiveness of the proposed algorithms for the vehicle trajectory registration based on vehicles linear and angular velocities and Kalman filter. We have implemented it in Matlab environment. The algorithm using Kalman filter, on average, showed the best performance in the sense of the standard deviation from the sample values. If the object disappears on the next few images and then appears again, the method still works, because for these images coordinates and sizes of the vehicle bounding boxes are used as measured coordinates and sizes until the disappearance.

Received test results note to prospects of the developed algorithms for usage as a part of information support for the driver assistance system, navigation and control system for a robotic vehicle [26] or an unmanned vehicle.

ACKNOWLEDGMENT

This article is written in a course of the grant of the President of the Russian Federation for state support of young Russian scientists № MK-3130.2017.9 (contract № 14.Z56.17.3130-MK) on the theme “Recognition of road conditions on images using deep learning”.

REFERENCES

[1] P. Viola and M. J. Jones, “Rapid object detection using a boosted cascade of simple features”, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR’01), vol. 1, pp. I-511–I-518, 2001.

[2] P. Viola, M. J. Jones, and D. Snow, “Detecting pedestrians using patterns of motion and appearance”, Proceedings of the 9 th International Conference on Computer Vision (ICCV’03), vol. 1, pp. 734–741, 2003.

[3] D. A. Yudin, A. S. Knysh, and E. O. Kapustina, “Detection of cars on static images using the Viola-Jones method”, Collected materials of the III International Scientific and Practical Conference “Innovative development of automation, information and energy-saving technologies, metallurgy and metal science. Current state, problems and prospects”, Moscow, MISA, pp. 280–287, 2015.

[4] N. Dalal, B. Triggs, and D. Europe, “Histograms of Oriented Gradients for Human Detection”, CVPR 2005, vol. 1, pp. 886 – 893, 2005.

[5] C. Rabe, T. Muller, A. Wedel, and U. Franke, “Dense, Robust, and Accurate Motion Field Estimation from Stereo Image Sequences in Real-Time”, Computer Vision – ECCV 2010, pp. 582–595, 2010.

[6] A. Ramirez, E. Ohn-Bar, and M. M. Trivedi, “Go with the flow: Improving Multi-View vehicle detection with motion cues”, IEEE International Conference on Pattern Recognition (ICPR), 2014.

[7] K. A. Rudakov, A. S. Pikalov, and D. A. Yudin. “Analysis of methods of tracking objects on a sequence of images”, International conference “Actual problems of robotics and automation”, Belgorod, pp. 154–158, 2015.

[8] M. Rezaei and M. Terauchi, “Vehicle Detection Based on Multi-feature Clues and Dempster-Shafer Fusion Theory”, Image and Video Technology, Springer, vol. 8333, pp. 60–72, 2014

[9] M. Rezaei, M. Terauchi, R. Klette, N. Zealand, and N. Zealand, “Robust Vehicle Detection and Distance Estimation Under Challenging Lighting Conditions”, IEEE Transactions on Intelligent Transportation Systems (T-ITS), pp. 2723–2743, 2015.

[10] Y. LeCun, Y. Bengio, and G.Hinton, “Deep learning”, Nature, vol. 521, pp. 436–444, 2015.

[11] B. Huval, T. Wang, S. Tandon, J. Kiske, W. Song, J. Pazhayampallil, M. Andriluka, P. Rajpurkar, T. Migimatsu, R. Cheng-Yue, F. Mujica, A. Coates, A.Y. Ng, “An Empirical Evaluation of Deep Learning on Highway Driving”, ArXiv: 1504.01716 [cs.RO], 2015.

[12] X. Chen, S. Xiang, C.-L. Liu, and C.-H. Pan, “Vehicle Detection in Satellite Images by Hybrid Deep Convolutional Neural Networks”, IEEE geoscience and remote sensing letters, vol. 11, no. 10, pp. 1797–1801, 2014.

[13] C. Szegedy, A. Toshev, and D. Erhan, “Deep Neural Networks for Object Detection”, Advances in Neural Information Processing Systems 26 (NIPS 2013), pp. 2553–2561, 2013.

[14] A. Krizhevsky, I. Sutskever, G. Hinton, “ImageNet classification with deep convolutional neural networks”, Advances in Neural Information Processing Systems 25 (NIPS 2012), 2012.

[15] V. D. Kustikova, “Methods and algorithms for analysis of motion trajectories in solving the problem of video detection of vehicles: dis. ... cand. Tech. Sciences”, The Nizhny Novgorod state. University of. N.I. Lobachevsky, Nizhny Novgorod, 145 p., 2015.

[16] W. Hu, X. Xiao, D. Xie, and T. Tan, “Traffic accident prediction using 3-D model-based vehicle tracking”, IEEE Transactions on Vehicular Technology, vol. 53, no. 3, pp. 677–694, 2004.

[17] C.G. Prevost, A. Desbiens, and E. Gagnon, “Extended Kalman Filter for State Estimation and Trajectory Prediction of a Moving Object Detected by an Unmanned Aerial Vehicle”, American Control Conference. ACC '07, pp. 1805–1810, 2007.

[18] J. Wiest, M. Hoffken, U. Kresel, and K. Dietmayer, “Probabilistic trajectory prediction with Gaussian mixture models”, IEEE Intelligent Vehicles Symposium (IV), pp. 141–146, 2012.

[19] C. Oh and T. Kim, “Estimation of rear-end crash potential using vehicle trajectory data”, Accident Analysis & Prevention, vol. 42, no. 6, pp. 1888–1893, 2010.

[20] A. Houenou, P. Bonnifait, V. Cherfaoui, and W. Yao. “Vehicle trajectory prediction based on motion model and maneuver recognition”. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 4363–4369, 2013.

[21] C. Tomasi and T. Kanade, “Detection and Tracking of Point Features”, Carnegie Mellon University Technical Report CMU-CS-91-132, 1991.

[22] “Keras: Deep Learning library for Theano and TensorFlow”, URL: https://keras.io/.

[23] “TensorFlow. An open-source software library for Machine Intelligence”, URL: https://www.tensorflow.org/.

[24] D. L. Olson and D. Delen, “Advanced Data Mining Techniques”, Springer, 1st edition, p. 138, 2008.

[25] I. N. Sinitsyn, “Filters of Kalman and Pugachev”, Moscow, University Book, Logos, 640 p., 2006.

[26] D. A. Yudin, G. G. Postolsky, A. S. Kizhuk, and V. Z. Magergut, “Mobile Robot Navigation Based on Artificial Landmarks with Machine Vision System”, World Applied Sciences Journal, 24 (11), pp. 1467–1472, 2013.