Detection of traffic anomalies for a safety system of smart city

Rifkat Minnikhanov
"Road Safety" State Company
Kazan, Russia
its.center.kzn@gmail.com

Igor Anikin
Kazan National Research Technical University named after A.N. Tupolev–KAI
Kazan, Russia
anikinigor777@mail.ru

Alisa Makhmutova
Kazan National Research Technical University named after A.N. Tupolev–KAI
Kazan, Russia
phd.makhmutova@griat.kai.ru

Maria Dagaeva
"Road Safety" State Company
Kazan, Russia
dagaevam@rambler.ru

Tikhon Bolshakov
"Road Safety" State Company
Kazan, Russia
bear_333@mail.ru

Kamil Mingulov
Eastern Graphics GmbH
Ilmenau, Germany
kamil_mingulov@mail.ru

***Abstract*—Sustainable development of modern smart city depends on safety of its citizens and efficient management of the resources. Instant response to incidents and abnormal situations provides these objectives. Implementation of such response applications requires intelligent information processing and data analytics. This information can be collected by different ways: with using sensors, navigation systems, users and surveillance cameras. Video monitoring system can be a valuable source of such information. Cameras cover most of the city and could be efficiently used to find anomalies. Video monitoring system requires non-stop viewing, analysis of the current situation and anomalies identification. It could be provided without human intervention by modern applications based on machine learning and computer vision techniques. In this paper, we used both computer vision and machine learning methods for traffic anomalies detection and classification in real time. As a result of this work we suggest the approach for detection of vehicle/pedestrian violation of legal trajectory anomalies, which we tested on real-time video in Kazan city.**

Keywords—smart city, video monitoring systems, machine learning, object detection

I. Introduction

Smart city concept has been created as the answer to issues of modern large cities. This concept represents interconnected systems of information and communication technologies, which simplifies the management of internal urban processes and makes the lives of residents more comfortable and safer. The concept of smart city is based on the principles of optimization of the transport system, sustainable consumption of energy and other resources. It also involves simplification of everyday processes (paying bills online, quick search for a parking space, etc.), citizen's safety, comfort and active participation in urban life.

Optimization of the transport system is the first task for a smart city developing, because large cities have always suffered from the problems of developing safe road systems with efficient traffic flow [1][2]. Non-optimized traffic flow raises a whole list of economic, social and environmental issues: traffic delays, jams, and accidents on the road, fuel consumption and pollution. Therefore, one of the most important goal in modern smart city is to provide effective traffic management, which can be done with intelligent transport system (ITS) and its applications. It requires continues collection of the actual information about road situation as well as its constant monitoring. We can collect relevant data from different sources, which can be categorized into few groups:

- roadway data, which is collected by different IoT devises (sensors), active or passive in nature [4];
- vehicle-based data, collected by technologies, such as electronic toll tags and radio navigation-satellite services (global positioning systems (GPS), GALILEO, GLONASS, etc.), which combined with cell phone-based Bluetooth and Wi-Fi [5][6];
- traveler-based data, voluntarily provided by drivers, which use mobile communications and applications;
- wide area data, which obtained by system networks, space-based radars or Geographic Information Systems (GIS).
- indirect data from external systems. For example, emergency management information systems (EMIS) do not directly store information about the road environment, but can store data about incidents in the city. These incidents cause traffic jams, and impede traffic in the city. Data mining, forecasting and analysis of the EMIS data can reduce response time, which will scale down or even prevent traffic jams [7].

Usage of video surveillance cameras for collection of roadway information become popular in recent years due to their coverage of city and efficient assisting in solving the described above problems. Video processing offers opportunities to meet the challenges of smart city, despite on its shortcomings, like dependence on environmental factors (rain, fog, brightness, etc.) and accuracy loss. First, the installation of supported tools (cameras, wires etc) can be performed without any additional work on the roadway, for example, in comparison with the installation of sensors. The second advantage is the price of devices and the cost of their maintenance. Additional information received from the roads is a third and significant advantage. With growth of computational abilities in the past few years, the computer vision and deep learning methods can be applied to detect characteristics of traffic, which provides efficient automatic statistical monitoring of the roads [8].

All described possibilities and advantages lead to the idea that video surveillance systems may be considered as solution of ITS tasks for smart city. Despite a strong connection with human during anomalies detection on the video, the task of objects' anomaly detection can be fully automated with the help of artificial intelligence (AI) like neural network and computer vision algorithms. The main goal of this paper is to give an overview of existing anomaly detection approaches in deep learning field in terms of their applicability for detection the anomalies on the video from surveillance cameras of the Kazan city.

II. Anomaly Detection Approaches in the Transport Environment

Incident control and anomalies management implies an understanding of the state of normal behavior. Any deviation from the norm will be considered as an anomaly. The definition depends on several factors: field of activity, type of processing data and its features, external conditions. It is important to identify parameters of normal behavior for different fields of activities (industry, road infrastructure, security) to understand what could be detected as anomalies.

In a transport environment normal behavior depends on various parameters (road condition, participants' behaviour, weather). Therefore, it is challenging task to identify what can be considered an anomaly and what is not.

Surveillance cameras provide visual control of a given area, which allows to constantly check zone of interest and identify the different events or changes: law enforcement, control over abandoned objects, crowd behavior [9, 10]. The zone of interest for surveillance cameras will be the road and the adjacent territory, and in our paper we consider following anomalies on the road, such as:

- vehicle/pedestrian violates legal trajectory;
- traffic congestions (including traffic flow deceleration/density increase on a road section).

A. Trajectory-based Anomaly Detection

Trajectory formation is complex and diverse task. However, this area is receiving attention of the computer vision community for last few years [11]. Besides transport area trajectories are used in suspicious activity detection [12], sports video analysis [13], video summarization [14], synopsis generation [15]. Object's trajectory it is captured motion changes of moving objects or, simple saying, the path. For many years researches face with problem of formation of accurate path. Objects are constantly in motion, and video capture many slightly different frames in each second. Processing of each frame of the video could significantly slow down further analysis. It is also important to understand that the processing systems on the cameras themselves have poor computing capabilities. Video monitoring system developers have a trade-off between processing speed and accuracy of the result.

There are two possible approaches for trajectory-based anomaly detection: first approach, often used in video analysis systems on the cameras themselves, is to define the normal behaviour or some rules and highlight areas of interest and track all deviations. The second approach is based on unsupervised learning, when we give to system the opportunity to learn on large amount of data, determine the norms automatically and then detect anomalies. Second approach is less reliable for critical systems, when the accuracy and confidence are extremely important, but it can reveal nontrivial patterns, which have not been described before. Another valuable advantage of the second approach is in its scalability and adaptation to constantly changing conditions (camera position, road conditions), which can significantly reduce manual work.

In this paper we focus on the second approach - on unsupervised learning algorithms for anomaly detection. Incident respond cannot be provided offline, all steps of algorithm should be processed in closely to/or in real-time. Therefore, some trade-off between accuracy and velocity of processing with privilege to the second parameter should be in mind.

First, video should be received and pre-processed. Usually video pre-processing applies to video frames. At this stage, noise reduction algorithms, color rendering, and contrast enhancements can be applied. After this step we can apply object detection and tracking techniques in order to extract information about entities in the frame and their movement. Depending on the field of activity, the video may contain objects that do not carry useful information (often they can be attributed to the background – buildings, trees, etc.). Not interesting for specific problem objects should be excluded from consideration on this step. On the next step we must highlight the trajectory of objects of interest. This step is important and must be done regardless of how the learning process will be organized (by the rules or with unsupervised learning). There are examples of visualization of trajectories and outlier detection on video frames on Fig. 1.

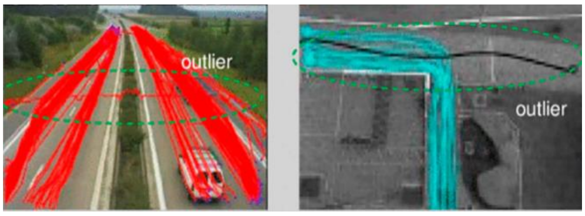


Fig. 1. Example of trajectories extraction and outlier detection.

On the next step represented outliers should be registered as anomalies, because its trajectory stands out from normal. Detection of anomalies requires a comparison of the current trajectory of a moving object with the legitimate one. Therefore, before we can compare trajectories, we must identify the rules or patterns, which could be received with the machine learning technique. It is hard process due to difficulties in defining boundaries between the normal and incorrect behaviour of vehicles, especially on different lanes of the same road. Finally, after rules modelling we can identify the anomalies.

B. Traffic Congestions Detection

Previously, we mentioned that the normal behaviour must be identified. At first glance, it might seem that it is easy to define a congestion. However, according to Downs there is no universally accepted definition of traffic congestion [16]. Numerous definitions can be categorized into 3 groups according to different measured parameters: demand capacity related, delay-travel time related, cost related [17]. Depending on a chosen definition of traffic congestion, different measurement metrics can be used:

- Speed. The average speed on any section of a road can be used to infer the state of traffic at the present point of time. This can be done by comparing present speed of traffic to off-peak period speed.
- Travel time and delay. Congestion is a travel time or delay in excess of the normally incurred under light or

free-flow travel conditions [18]. Unacceptable congestion is travel time or delay longer than accepted norm. This norm may vary depending on geographical location, time of the day etc.

- Level of service measures. The level of service (LOS) has been one of most popular measure of traffic congestion. It was adopted in 1985 Highway Capacity Manual [19]. LOS is subdivided into six classes ranging from A to F, which are categorized according vehicle-to-capacity ratio.

In the scope of this work 4 approaches to congestion detection were proposed:

- Measure time of presence of vehicles in frame and compare with historical data.
- Measure velocity of vehicles and compare with historical data.
- Count vehicles using virtual detecting line and compare number of passing vehicles with traffic capacity of a road calculated according to regulations of Russian Federation.
- Measure the movement in a frame and compare with calibration data.

In both tasks for anomalies detection - trajectory-based or traffic congestions - we need to process incoming video frames and provide image processing for object detection and tracking.

III. IMAGE PROCESSING

At the pre-processing step segmentation could be applied, to subdivide an image into nonoverlapping - for objects detection. When process of segmentation is complete, describing selected regions features have to be found. We can consider texture, color and shape as example of these features. Each of the regions, represented on image, can be described by properties of such features such as length, area or width of the shape. Segments are classified to meaningful classes by extracted features. For example, on a road image classes may be cars, trucks, buses, pedestrians and so on. The problems of scene segmentation and object classification can be solved by expert systems, semantic networks and neural network systems. Segmentation and classification together make up process, which is called *object detection*. During *object tracking process* we can develop object's trajectory.

A. Object Detection

Object detection imply identification of object's location in the frame. This is a difficult task, because objects can be classified in categories such as vehicles and people, and their appearance in the frame vary. Variations arise not only from changes in illumination and viewpoint, but also due to nonrigid deformations and intraclass variability in shape and other visual properties. For example, people wear different clothes and take a variety of poses, while cars come in various shapes and colors. Fig. 2 shows the taxonomy of object detection approaches in remote sensing image, but these approaches can be generalized to object detection as a whole.

The input of the classifier is a set of regions (sliding windows or object proposals) with their corresponding feature representations and the output is their corresponding predicted labels. Feature extraction, feature fusion, dimension reduction, and classifier training play the most important roles in the performance of object detection.

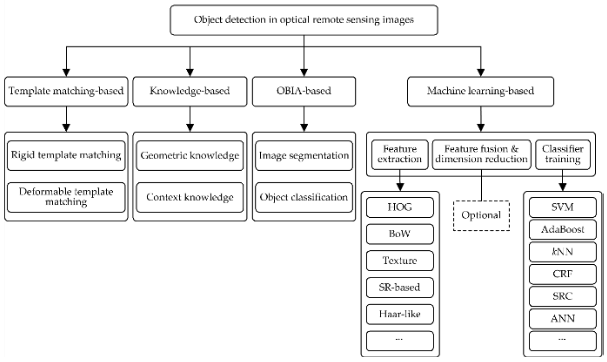


Fig. 2. Taxonomy of methods for object detection.

Traditional techniques from statistical pattern recognition like the Bayesian discriminant and the Parzen windows were popular until the beginning of the 1990s. Since then, neural networks have increasingly been used as an alternative to classic pattern classifiers and clustering techniques [20]. Since 2012 after work of Krizhevsky et al. [20] majority of state of the art object segmentation, and classification are performed by Convolutional Neural Networks (CNN). CNN is successfully used for complex objects detection [21]. These neural networks are outperforming other architectures due to data size reduction on convolution and pooling steps, which allow increasing complexity of neural network. There are four main operations in CNN: convolution, non-linearity (ReLU), pooling or sub sampling, and classification (provided be fully connected layer).

CNN for object detection are different in searching of extracted features on the image. There are search by regions on the image (R-CNN [22], Fast R-CNN, [23] Faster R-CNN [24]) Single Shot MultiBox Detector (SSD) [25] and You Only Look Once (YOLO) [26] method which predict bounding boxes and probabilities for each region. In YOLO network image is split into SxS grid. Each of resulting cells predicts B bounding boxes and confidences. Each cell also predicts class probabilities. Bounding boxes are combined with classes.

B. Object Tracking

Tracking is necessary step for extraction of object's trajectory through the video scene [27, 28]. Process of object tracking can be subdivided into three consequent steps - moving object detection, object classification and inter-frame tracking. On the first step moving objects are separated from static background. On second step - object classification – identified on the previous step moving objects are assigned to classes based on their features. Finally, on step of tracking classified objects are identified on subsequent frames. Neural networks are significantly simplified the tracking task, producing the first two steps. Therefore, we closely look on third step - the tracking algorithms.

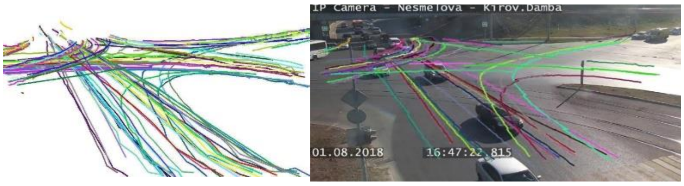
Modern trackers are not ideal. It is difficult for them to process frames with such environmental challenging factors like occlusion (when an object closer to the camera overlaps the object behind it), background clutters (road has similar to vehicle color), motion blur (target region is blurred due to the motion of vehicles or the camera) and etc. Modern algorithms have to be able partially cover these difficulties.

Boosting tracker [29] is based on an online AdaBoost algorithm. The initial bounding box of an object is considered as the positive example of the object, and the rest is treated as background. Algorithm is old and outperformed by many modern algorithms. Multiple Instance Learning (MIL) tracker algorithm [30] is based on approach similar to Boosting. The difference here is that this algorithm generates multiple hypotheses in neighborhood of a center of object. Together, all these hypotheses with original bounding box are put into labeled as ‘bag’, each containing many instances. In case of not well-centered prediction of main bounding box, there is a high probability that positive ‘bag’ will contain a better prediction. Tracker does not recover from full occlusion. Kernelized Correlation Filters (KCF) tracker [31] is based on ideas of MIL and Boosting. The fact that positive bag in MIL contain boxes with large overlap allows to reduce complexity of correlation calculation from $O(n^2)$ to $O(n \log n)$. This tracker is both faster and more accurate than MIL and reports tracking failure better. It does not recover from full occlusion. Minimum Output Sum of Squared Error (MOSSE) tracker [32] uses adaptive correlation for object tracking, which produces stable correlation filters when initialized using a single frame. MOSSE tracker is robust to variations in lighting, scale, pose, and non-rigid deformations. It also detects occlusion based upon the peak-to-sidelobe ratio, which enables the tracker to pause and resume where it left off when the object reappears. We compared such qualitative signs as the smoothness of the object tracking, work with different types of objects, the number of occlusions. After numerous tests we decided to make our choice on KCF tracking algorithm. This method not only tracks objects correctly, but also does not produce so many errors like other algorithms. Its advantage is that it can work with images of low quality, resolution and frame rate.

C. Pattern Extraction

We need to extract frequent trajectories of road objects as the pattern, which is expressed in the form of curve. Since we had the goal to highlight the trajectory of only road related objects, we removed all objects that are not related to the class of vehicles and people, as well as all objects with a recognition probability less than 50%. For the remaining objects, we normalized their coordinates and pass them to the trackers. At the same time, we assign a color to each object for identification. Then, using the multitrack object, we visualize the trajectory of moving objects in the following way: the coordinates of the centre of the previously tracked object are remembered and connected to the centre of the next tracker position. After all frames have been processed, we combine them back into a video. Each object gets its own colour to draw a trajectory. After receiving all trajectories, we can divide them into categories by type of the objects and build groups of trajectories and track anomalies (represented on Fig. 3). During the long time of training, we get full collection of many similar approximated trajectories, which must be revised to identify the norm. Euclidian distance between object paths points was chosen to do that. If the Euclidian distance is less than the epsilon neighborhood, then we check whether the object deviates from this trajectory in further moving.

If final object’s path is not deviated from the trajectory, we remove this path from the list of reference trajectories. If the object’s starting place does not fit any reference path or final trajectory is too far from the existing ones, then the trajectory is added to the list for further analysis. The same method was chosen for anomalies identification.



a) b)

Fig. 3. Trajectories of objects: a) separately b) on the video frame.

The other unsupervised method for pattern extraction is based on DBSCAN algorithm [33], when we combine objects for the purpose of congestion identification, lane detection etc. If trajectory comparison method could be used in real time mode, this method is more suitable for off line checks. We can see there the formed cluster of vehicles by their first appearance on Fig. 4 The main advantage of the DBSCAN algorithm is that it allows to build clusters of arbitrary shape.

Fig. 4. Cluster shapes based on DBSCAN algorithm.

IV. Evaluation and Results

We used data from intersections of Kazan city for trajectory-based anomaly detection task and data from Moscow roadside cameras for congestion evaluation.

First, we provided evaluation of object detection algorithm. As we mentioned previously, our priority was focused on the speed and accuracy. After numerous test, represented in Fig. 5-6, we choose YOLO method, which provided best results by both parameters.

Fig. 7-8 shows the automatically detected anomaly of deviation from the reference trajectory in the form of road accident.

For traffic congestion we tried few approaches. To test performance of trackers a sequence of 600 frames was prepared. Trackers are initialized with bounding box around black car. Sequence is terminated if tracker fails to update. Resolution of the video is 1280x720 pixels. In Table 1 evaluation of the results of the tracker performance test, mark below 48-60 frames per second was labeled as Slow. KCF and MOSSE trackers were picked as most applicable for this task. KCF offers higher accuracy of tracking while MOSSE offers pure speed.

Approach, which is based on time of presence was found to be impractical. This approach requires ideal detection and tracking algorithms. Neither algorithms of YOLO family nor any other modern algorithm has detection rate that would make this approach work. On tracking failures time of presence cannot be calculated correctly. Another problem of this approach is connected to performance decrease with growing number of trackers. Approach, which is based on speed is also impractical. Furthermore, measurement of distance in pixels produces data incomparable to real distances. Without manual calibration of a camera it is impossible to get mapping from camera coordinates into real-world ones. Vehicle count based approach gives better result in comparison to previous two. In scenes with congestion due to high level of occlusion vehicle detection is unstable as well as tracking. Fig. 9 shows congestion detection. Unfortunately, this approach requires high level of user interaction for initialization.

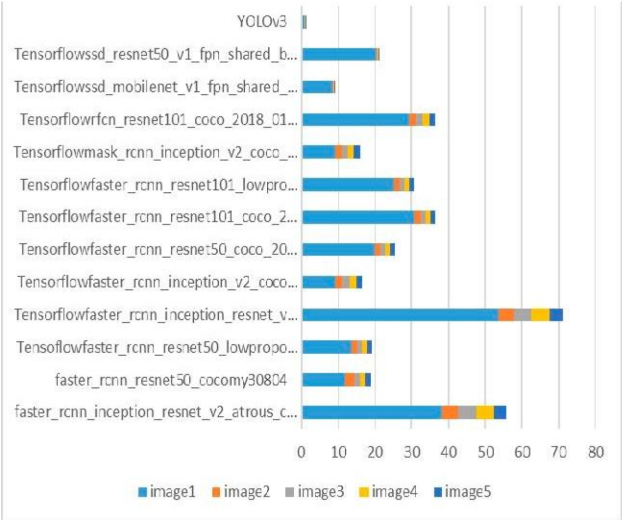


Fig. 5. Comparison of processing time (in seconds) of different CNN architectures.

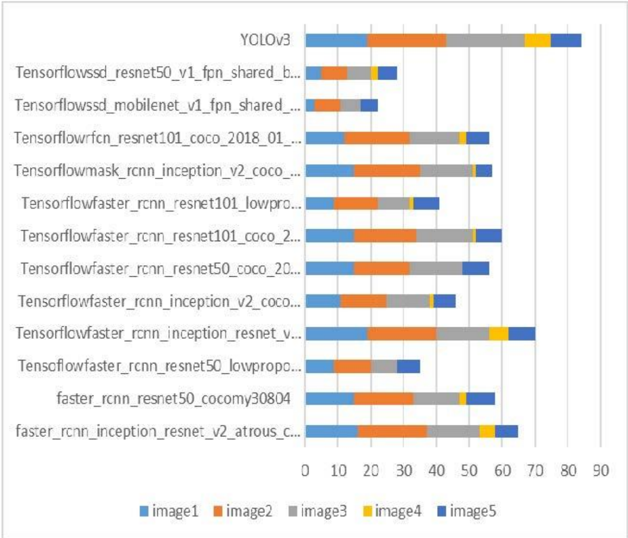


Fig. 6. Comparison of the accuracy (in number of correctly detected objects) of different CNN architectures.

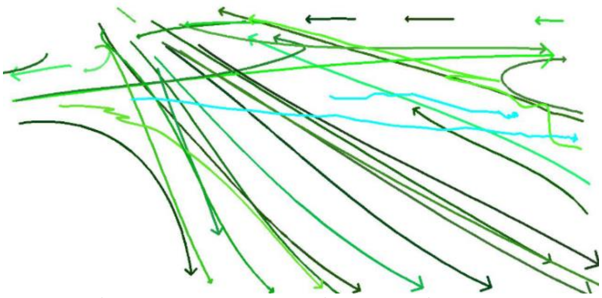


Fig. 7. Identified anomaly deviations from the reference trajectory: two trajectories that go beyond the normal.

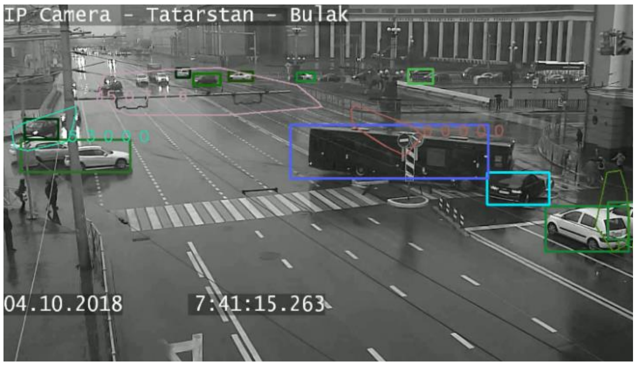


Fig. 8. Identified anomaly deviations from the reference trajectory: detected traffic accident.

TABLE I. TRACKER PERFORMANCE

Tracker	Frames per second	Seconds per image	Negative side
Boosting	20.23251	0.0494	Slow
CSRT	19.85937	0.0504	Slow
KCF	68.4742	0.0146	-
MedianFlow	109.1616	0.0092	Early failure
MIL	10.75797	0.0930	Slow
MOSSE	1159.608	0.0009	-
TLD	5.929194	0.1687	Tracker shifted to different object

Fig. 9. Congested detected, even image contains ghost tracks and untracked vehicles.

V. CONCLUSION

As a result of this work, the approach for detection congestions and trajectories-based anomalies was developed.

The developed approach is able to recognize independent reference trajectories for certain classes of objects with unsupervised learning algorithm, and identify anomalies if the spatial trajectory of the object violates to them. In the future, the expansion of the type of identified anomalies, as well as testing the system in real time is planned. There are various road rules for road lanes in the city of Kazan. For example, there are special lane for public transport and large sized vehicles. Drivers of vehicles often violate the travel ban on these lines and drive along it. We can also think about creating rules for each lane of the road, which is a very challenging task from both unsupervised learning and video processing parts.

References

[1] Y. Wiseman, "Real-time monitoring of traffic congestions," IEEE International Conferenceon Electro Information Technology, pp. 501-505, 2017.

[2] K. Zavitsas, I. Kaparias and M.G.H. Bell, "Transport problems in cities" [Online]. URL: https://trimis.ec.europa.eu/sites/default/files/project/documents/20120402_173932_45110_D%201.1%20-%20 Transport%20problems%20in%20cities%20-%20v3.pdf.

[3] F.J. Kelly and T. Zhu, "Transport solutions for cleaner air," Science, vol. 352, no. 6288, pp. 934-936, 2016.

[4] New Traffic Data Sources - An Overview [Online]. URL: https://www.bitre.gov.au/sites/default/files/documents/National_Infrastructure_Data_Collection_and_Dissemination_Plan-draft.pdf.

[5] E. D'Andrea and F. Marcelloni, "Detection of traffic congestion and incidents from GPS trace analysis," Expert Syst. Appl., vol.73, pp. 43-56, 2017.

[6] M.D. Fontaine and B.L. Smith, "Investigation of the Performance of Wireless Location Technology-Based Traffic Monitoring Systems," J. Transp. Eng., vol.133, no. 3, pp. 157-165, 2007.

[7] M. Dagaeva, A. Garaeva, I. Anikin, A. Makhmutova and R. Minnikhanov, "Big spatio-temporal data mining for emergency management information systems," IET Intelligent Transport Systems, vol. 13, no. 11, pp. 1649-1657, 2019.

[8] J. Chung and K. Sohn, "Image-Based Learning to Measure Traffic Density Using a Deep Convolutional Neural Network," IEEE Transactions On Intelligent Transportation Systems, vol. 19, no. 5, pp. 1670-1675, 2018

[9] H. Chen, O. Nedzvedz, S. Ye and S. Ablameyko, "Crowd Abnormal Behaviour Identification Based on Integral Optical Flow in Video Surveillance Systems," Informatica, Lith. Acad. Sci., vol. 29, no. 2, pp. 211-232, 2018.

[10] H. Chen, S. Ye, A. Nedzvedz, O. Nedzvedz, H. Lv and S. Ablameyko, "Traffic extreme situations detection in video sequences based on integral optical flow," Computer Optics, vol. 43, no. 4, pp. 647-652, 2019. DOI: 10.18287/2412-6179-2019-43-4-647-652.

[11] A. Ahmed, D.P. Dogra, S. Kar and P.P. Roy, "Trajectory-Based Surveillance Analysis: A Survey," IEEE Trans. Circuits Syst. Video Techn., vol. 29, no. 7, pp. 1985-1997, 2019.

[12] T. Xiang and S. Gong, "Video behavior profiling for anomaly detection," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 30, no. 5, pp. 893-908, 2008.

[13] A. Rehman and T. Saba, "Features extraction for soccer video semantic analysis: current achievements and remaining issues," Artificial Intelligence Review, pp. 1-11, 2014.

[14] T. Zhang, D. Wen and X. Ding, "Person-based video summarization by tracking and clustering temporal face sequences," uS Patent 9,514,353, 2016.

[15] Y. Pritch, A. Rav-Acha and S. Peleg, "Nonchronological video synopsis and indexing", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 30, no. 11, pp. 1971-1984, 2008.

[16] A. Downs, "Still stuck in traffic: coping with peak-hour traffic congestion," Washington, D.C: Brookings Institution Press, 2004.

[17] M. Aftabuzzaman, "Measuring traffic congestion: a critical review," Australas. Transp. Res. FORUM, vol. 30, 2007.

[18] H.S. Levinson, T.J. Lomax, "Developing a Travel Time Congestion Index," Transp. Res. Rec. J. Transp. Res. Board, vol. 1564, no. 1, pp. 1-10, 1996.

[19] M. Egmont-Petersen, D. Ridder and H. Handels, "Image processing with neural networks – a review," Pattern Recognit., vol. 35, no. 10, pp. 2279-2301, 2002.

[20] A. Krizhevsky, I. Sutskever and G.E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," Advances in Neural Information Processing Systems, vol. 25, pp. 1097-1105, 2012.

[21] R.P Bohush, I.Y Zakharava, "Person tracking algorithm based on convolutional neural network for indoor video surveillance," Computer Optics, vol. 44, no. 1, pp. 109-116, 2020. DOI: 10.18287/2412-6179-CO-565.

[22] R. Girshick, J. Donahue, T. Darrell and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," Proc. of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 580-587, 2014.

[23] R. Girshick, "Fast R-CNN," Proc. of the IEEE International Conference on Computer Vision, pp. 1440-1448, 2015.

[24] S. Ren, K. He, R. Girshick and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, no. 6, pp. 1137-1149, 2017.

[25] W. Liu, D. Anguelov, D. Erhan, C. Szegedy and S. Reed, "SSD: Single shot multibox detector," European Conf. on Computer Vision. Lecture Notes in Computer Science, pp. 21-37, 2016.

[26] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 779-788, 2016.

[27] K. Kale, S. Pawar and P. Dhulekar, "Moving object tracking using optical flow and motion vector estimation," 4th International Conference on Reliability, Infocom Appendix (Trends and Future Directions), pp.1-6, 2015.

[28] R. O. Chavez-Garcia and O. Aycard, "Multiple Sensor Fusion and Classification for Moving Object Detection and Tracking," IEEE Trans. Intell. Transp. Syst., vol. 17, no. 2, pp. 525-534, 2016.

[29] H. Grabner, M. Grabner and H. Bischof, "Real-time tracking via on-line boosting," Proc. BMVC, vol. 1, pp. 47-56, 2016.

[30] B. Babenko, M.-H. Yang and S. Belongie, "Visual Tracking with Online Multiple Instance Learning," IEEE Conference on Computer Vision and Pattern Recognition, 2009.

[31] J.F. Henriques, R. Caseiro, P. Martins and J. Batista, "High-Speed Tracking with Kernelized Correlation Filters," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 37, no. 3, 2014.

[32] D. Bolme, J.R. Beveridge, B.A. Draper and Y.M. Lui, "Visual object tracking using adaptive correlation filters," IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 2544-2550, 2010.

[33] M. Ester, H.P. Kriegel, J. Sander and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, AAAI Press, pp. 226-231, 1996.