Dual-Mode Vehicle Motion Pattern Learning for High Performance Road Traffic Anomaly Detection

Yan Xu*[1] Xi Ouyang*[2] Yu Cheng*[3] Shining Yu*[3]
Lin Xiong[1] Choon-Ching Ng[1] Sugiri Pranata[1] Shengmei Shen[1] Junliang Xing[4]
[1] Panasonic R&D Center Singapore [2] Huazhong University of Science and Technology [3] Nanyang Technological University
[4] National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences
yan.xu@sg.panasonic.com ouyang@hust.edu.cn {chen0974, yush0012}@e.ntu.edu.sg
{linxiong, choonching.ng, sugiri.pranata, shengmei.shen}@sg.panasonic.com jlxing@nlpr.ia.ac.cn

Abstract

Anomaly detection on road traffic is an important task due to its great potential in urban traffic management and road safety. It is also a very challenging task since the abnormal event happens very rarely and exhibits different behaviors. In this work, we present a model to detect anomaly in road traffic by learning from the vehicle motion patterns in two distinctive yet correlated modes, i.e.*, the static mode and the dynamic mode, of the vehicles. The static mode analysis of the vehicles is learned from the background modeling followed by vehicle detection procedure to find the abnormal vehicles that keep still on the road. The dynamic mode analysis of the vehicles is learned from detected and tracked vehicle trajectories to find the abnormal trajectory which is aberrant from the dominant motion patterns. The results from the dual-mode analyses are finally fused together by driven a re-identification model to obtain the final anomaly. Experimental results on the Track 2 testing set of NVIDIA AI CITY CHALLENGE show the effectiveness of the proposed dual-mode learning model and its robustness in different real scenes. Our result ranks the first place on the final Leaderboard of the Track 2.*

1. Introduction

More and more families now have their own cars and traveling by car has become a very common and convenient way in the daily urban life. The road condition thus receives great attention from the public. Bad road conditions can cause massive loss to the social economy, and threaten personal safety of drivers on the road. With widely deployed traffic cameras that record the road conditions, it is feasible and important to develop a method to automatically find the anomalies on the roads using computer vision techniques. Traffic monitoring system equipped with these algorithms will bring about many benefits and conveniences. On one hand, when the anomalies take place, an automatic system can inform the traffic police immediately, to solve the anomalies on roads as soon as possible. On the other hand, when planning a trip, information about road condition can provide convenience for both drivers and passengers [26, 3].

However, it is a very challenging task to design a computer vision algorithm to detect anomaly in road traffic. One main reason is that the motion patterns of vehicles on roads are usually very complicated, and different abnormal events may exhibit very complex behaviors. At the same time, abnormal event happens very rarely as compared to normal events. Therefore, developing an efficient and effective intelligent algorithm for automatic video anomaly detection is a pressing need. Many works of anomaly detection in surveillance videos can only be applied to detect specific anomalous events. For instance, Mohammadi *et al*. develop a method to detect human violence in videos [12]. Also, the traffic detectors only works in very limited conditions [7, 25].

Facing with the above issues, we propose a dual-mode vehicle motion pattern learning model for anomaly detection in road traffic, which performs joint analyses of both the static and moving vehicles. Normally, the vehicles should keep moving on the roads except certain normal conditions (*e.g*., waiting for traffic lights). Therefore, the static vehicles have higher probability for being abnormal events. Generally, most of the abnormal events in road traffics will cause car stopping. For example, car stalling, traffic accident or car jamming. Meanwhile, the static vehicles can give us the accurate location of the anomalous events. To distinguish the static vehicles from the moving traffic, we introduce a static mode method to get the running average of

* indicates equal contribution. Xi Ouyang, Yu Cheng and Shining Yu are interns at Panasonic R&D Center Singapore during this work. Yan Xu is the corresponding author.

the frame sequence. Also, inspired by the great success of deep learning in computer vision field [9, 23, 6], we deploy the deep learning based method for the static vehicle detection. Moreover, we design a residual network to further distinguish the vehicle images from the background images after the vehicle detection.

At the same time, we develop a dynamic mode method to analyze the motion patterns of the vehicles. We first extract the trajectories of all the moving vehicles on the roads. Then, we conduct the cluster analysis based on the trajectories on roads and find the mainstream moving pattern including moving directions and speeds. Therefore, the vehicles that diverge from the dominant pattern can be regarded as the abnormal events. Moreover, we design a re-identification driven multi-model fusion algorithm to combine the static and dynamic analysis to gain the confidence score and occurrence time of anomaly events.

The main contributions of this paper are summarized as follows in three-fold:

- We present a new Dual-Mode model for road traffic anomaly detection which jointly learns from both static and dynamic vehicle motion patterns to obtain more comprehensive abnormal events.

- We design an algorithm to detect the static vehicles for anomalies analyses. It can not only enhance the performance of anomaly detection, but also produce the accurate locations of anomaly events.

- We design a tracking-based method to detect the moving vehicles with unusual moving paths. It can distinguish the anomaly vehicles based on the moving directions and speeds. Also, a re-identification driven multi-model fusion algorithm is designed to combine the analyses of static and moving vehicles.

Based on the above technical contributions, we present a high-performance system in road traffic anomaly detection.We evaluate the proposed Dual-Mode model on Track 2 testing set of the NVIDIA AI CITY CHALLENGE[1]. The experimental result shows that our proposed method can perform well on the real scene data. We obtain the F1-score metric at 0.8649 and the RMSE metric at 3.6152, which ranks the first place on the Track 2 among 7 teams.

2. Related Works

Vehicle detection and tracking is the basic module in the road traffic condition analysis and plays an import role in many related applications such as driver assistance system. Due to the great success of deep learning technology, we have gained a huge improvement in those fields [17, 5, 1, 14]. In our paper, we also apply those deep learning methods for the vehicle detection and tracking.

Anomaly detection in both static images and surveillance videos have been studied in the past years due to the increasing interest in public security [13, 20]. The traditional methods usually learn the hand-crafted features to model the normal/abnormal event patterns [11, 8]. Recently, deep learning technology has been developed for the anomaly detection as its success in computer vision field [19, 21]. In [21], the authors introduce a generative adversarial network (GAN) [4] based method to detect the anomalies in images, using only normal data to train the models. For surveillance videos, there are several attempts to detect human violence or abnormal events in crowd scene [12, 16, 24]. In [24], a deep anomaly ranking model is proposed to predict high anomaly score in the testing videos.

Since road condition plays an important role in our daily life, detecting anomalies on roads has attracted attention from many researchers[22, 18]. For this task, the key is to find when and where the anomalies take place. Deublein et al. [2] propose a combination of techniques to predict the occurrence of road accidents. In [18], the authors present a visual analytics framework for the analysis of normal behavioral models and the detection of anomalous events. However, the recent works are designed to detect a specific anomalous event, and the real-world anomalies on roads are complicated and diverse. Therefore, we design a novel dual-model method to detect various road traffic anomalous events in real scenes, which can have a wide use in practice.

3. Dual-Mode Motion Pattern Learning Model

In this section, we first introduce the methods for dual model analysis of the vehicles, respectively. Then, the mechanism to fuse the two modules is explained in details.

3.1. Static Mode Analysis of the Vehicles

For the static analysis of static vehicles, we aim to find the static vehicles on roads, as the anomalies usually lead to car stopping. Normally, most of the anomalies on roads are abnormal car stalling or traffic accidents, which both make the vehicles stop in/beside the road. Based on this observation, we introduce a motion analysis method to find the static vehicles on roads and further recognize the abnormal events based on that.

The pipeline of anomaly detection based on the static vehicle is presented in Figure 1. We take a two-step method to find anomalies: 1) the motion analysis. The aim is to extract the static vehicles in the moving traffic. 2) detection and recognition. We adopt the Faster R-CNN [17] to detect the static vehicle on roads and develop a recognition module to filter the false positives generated by Faster R-CNN. In the following sections, we will introduce these two steps in details.

[1] https://www.aicitychallenge.org/

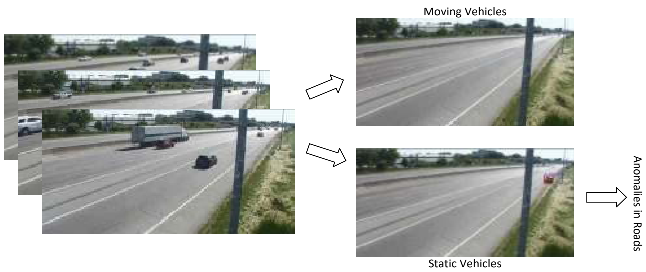


Figure 1. The pipeline to detect the anomalies in roads based on the analysis of static vehicles.

3.1.1 Background Modeling

In surveillance videos, the camera view is usually fixed, except seldom camera rotations. It means that the background is fixed in videos. To remove the moving vehicle, we continuously calculate the weighted sum of the input frame in the whole video, which can enhance the static parts but suppress the moving parts in objects. It can be formulated as:

$$Average = (1 - \alpha) * Average + \alpha * frame_i. \quad (1)$$

Here the $frame_i$ refer to the i_{th} frame in videos, and the $Average$ is the running average of the frame sequence from the 1_{st} to i_{th}.

As shown in Figure 1, the "Moving Vehicle" image and "Static Vehicle" image are two examples of the running average images. We can see that all the fixed parts including the background and static vehicles are kept in this image, but the moving cars are filtered. Meanwhile, α regulates the update speed (how fast the $Average$ image "forgets" about earlier images). In our experiments, we find $\alpha = 0.01$ is a good setting and use it for all video processing. After this processing in all videos, we can extract the static vehicles in all the frames, as the "Static Vehicle" image shown in Figure 1.

3.1.2 Detection and Recognition

After filtering the moving traffic in surveillance videos, we need to locate the positions of the static vehicles. Here we use the Faster R-CNN detector to detect cars in the $Average$ images as shown in Figure 1. However, not all the detection results can be regarded as real static vehicles. Since we calculate the running average of the frame sequence, there are many jitters in $Average$ images which can make the detector mistake those as vehicles. Also, the detection results contain some false positives from the background images. To further improve the precision, we build up a residual network [15] to distinguish the vehicles from the background patches, which is shown in Figure 2. For training, we collect tens of road surveillance videos from YouTube[2] and randomly crop the patches in the background images as the negative samples. For the positive samples, we use the car images cropped from the UA-DETRAC Benchmark [27, 10] as the positive samples. Thus, the network serves as a binary classifier and can learn to distinguish the true vehicle images from the background patches among the detection results of Faster R-CNN.

Moreover, not all the vehicles stops on roads are caused by abnormal events. On the roads, a normal situation is to wait for the traffic light which should be distinguished from abnormal events. To solve this issue, we calculate the stopping time of each static vehicles. If the stopping time does not last long, we will label those vehicles as normal ones.

3.2. Dynamic Mode Analysis of the Vehicles

The method of static mode analysis of vehicles is sensitive to the quality of averaged frames, which are influenced by camera shaking or video quality. To improve the robustness of our prediction, we propose to solve this problem in another perspective of analyzing the dynamic motion of vehicles.

3.2.1 Object-wise Segmentation and Tracking

With the aid of Mask-RCNN [5], we apply the object-wise segmentation on the input images as shown in Figure 3. The object-wise segmentation allows us to further develop tracking algorithm.

The tracking is implemented in pixel-wise optical flow and based on the assumption that the pixel intensities of the same object are similar in consequent frames, as in Equation 2.

$$I(x, y, t_0) \approx I(x + dx, y + dy, t_1), \quad (2)$$

where I and t annotates the pixel intensity and time stamp, respectively.

Take the Taylor Expansion on the right hand side, then we get Equation 3.

$$I_x u_x + I_y u_y + I_t = 0, \quad (3)$$

where I_x represents the derivative of I respects to x.The target of object tracking is to find the velocity along x and y which is $u_x = \frac{dx}{dt}$ and $u_y = \frac{dy}{dt}$.

According to Equation 3, it will be impossible to find the velocity with one single pixel. Here, we take the nearby 9 pixels and fit the u_x and u_y by least square method. In tracking phase, we randomly sample n points (n should be large enough to ensure the tracking quality) in every object mask and apply the pixel-wise tracking on consequent i frames with the method above, as shown in Figure 4.

[2]https://www.youtube.com/

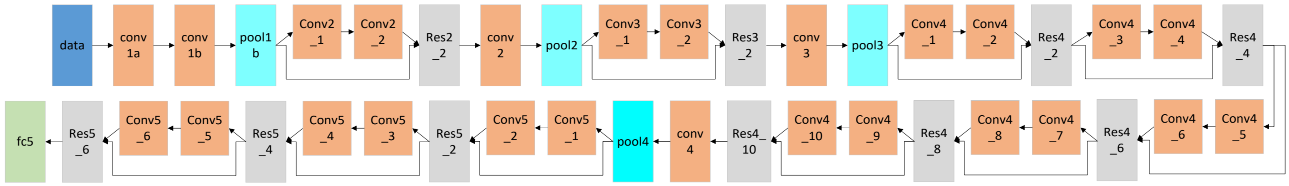


Figure 2. The structure of residual network to distinguish the vehicles from the background patches.

Figure 3. The object-wise segmentation result. The original image (first row) and the segmented masks (second row).

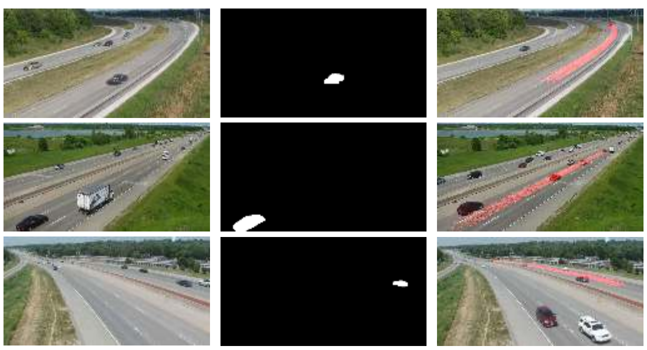


Figure 4. The pixel-wise tracking based on object-wise masks. The original image (first column), selected object (second column) and its trajectory (third column) is plotted in red.

3.2.2 Outliers Filtering and Velocity Measurement

Although the tracking algorithm works well on object without obstacle, the sample points will be misdirected by other moving objects due to our optical flow based approach. Therefore, an outlier filter should be applied on each frame stamp to extract the exact object location information.

We propose a simple method of measuring points' neighborhood distance density. For each point we track, the average of neighborhood distance could be found as in Equation 4.

$$D_x = \frac{k}{\sum_{i=0}^{k} d_i}, \tag{4}$$

where k is the customized parameter to indicate the k-nearest neighbor that count into computation. The value of k usually indicates the minimum size of one cluster.

The distance density is then computed by the mean logarithm value as indicated in Equation 5.

$$\rho_x = -\log(D_x) - \frac{1}{k}\sum_{i=0}^{k}\log(D_i). \tag{5}$$

After setting a threshold to filter the outliers, it becomes more reasonable and robust for calculating the object speed. We take the average of sample points as the object center and thus find the velocity in the unit of pixels/frame as in Equation 6.

$$u = \sqrt{d_x^2 + d_y^2}. \tag{6}$$

In addition, merely measure the speed and test the anomaly is unstable and unreliable because noise from images such as shaking and video compressing. Statistically, if the noise is assumed to have a Gaussian distribution, an **E**xponential **M**oving **A**verage can significantly reduce the noise from environment such as camera shaking and bad video quality, thus it should be applied on the velocity measurements, which can be formulated as Equation 7:

$$v_1^r = (1-\alpha)v_0^m + \alpha v_1^m, \tag{7}$$

where α denotes the smoothing factor and v^r and v^m stands for the estimated real velocity and measured velocity.

3.3. Re-identification Driven Multi-Mode Fusion

To combine those two methods, we firstly introduce a Car Re-Identification (Car-Reid) module. It is designed for identify the vehicles in both two methods. Since a vehicle will slow down when suffering the anomaly event, our method for static vehicles works when it stops completely but the analysis of moving vehicles runs before the stopping. It requires the Car-Reid module to identify whether it is the same vehicle that suffer the anomaly event on our two analysis. In this paper, we use the binary classifier in Section 3.1.2 which is used for recognize the car and background. We pass the car images to the residual network and get the feature from the last Fully-Connected layer as the feature representation. If the cosine similarity of two car representations is no more that 0.45, we consider they should be the same car.

Figure 5. The examples of the Track 2 testing set.

After identify the vehicle in our two method, we need to fuse the two methods to gain the confidence scores and occurrence time of the anomaly events, which can be formulated as:

$$\mathrm{S} = \lambda * S_1 + (1-\lambda)S_2, \tag{8}$$

$$T = T_1 + (1-\lambda) * (T_2 - T_1). \tag{9}$$

Here S_1, S_2 refer to the confidence scores from the method of static and moving vehicles, and S is the final score of our whole system. Similarly, T_1, T_2, T refer to the occurrence time in both two methods and the final result. We get the final score S as a weighted sum of two methods in Equation 8, where λ is the weight of the score S_1 of static vehicles. If the score S is more than 0.6, we regard it as the anomaly event in our experiments. Moreover, the occurrence time of the anomaly events is also an important evaluation criterion. There may exist some differences between two methods, so we select an intermediate value between T_1 and T_2. Also, λ controls the distance from the occurrence time T_1 in the method of static vehicles.

4. Experiments

In this section, we first introduce the evaluation dataset. Then, we show the experimental results of the static and dynamic mode, respectively. Finally, we present the performance of our method in the Challenge dataset.

We test and evaluate our method on the Track 2 testing set of NVIDIA AI CITY CHALLENGE. It aims to detect anomalies caused by crashes, stalled vehicles, etc, which requires even human to pay very close attention to extract meaningful visual information. The Track 2 testing set contains 100 videos, each approximately 15 minutes in length, recorded at 30 fps and 800×410 resolution. The anomaly can be due to car crashes or stalled vehicles. We present some examples on the Track 2 testing set in Figure 5, which contains the real scene videos with diverse backgrounds, light condition, weather. Therefor, it is a quite challenging dataset.

4.1. Results of the Static Mode

As discussed on Section 3.1, we calculate the running average of the frame sequence and use Faster R-CNN to detect the static vehicles in $Average$ images, which is shown in Figure 6. The moving traffic can be erased in $Average$ images. In this paper, we use a ResNet101 [6] based Faster R-CNN, and fine-tune it on a car detection dataset (UA-DETRAC Benchmark [27, 10]) to further improve the performance. This dataset consists of 10 hours of videos captured with a Cannon EOS 550D camera at 24 different locations at Beijing and Tianjin in China. There are more than 140 thousand frames in the UA-DETRAC dataset and 8250 vehicles that are manually annotated, leading to a total of 1.21 million labeled bounding boxes of objects. As shown in Figure 6, the Faster R-CNN can produce the bounding boxes of the static vehicles in $Average$ images after fine-tuning on such huge dataset.

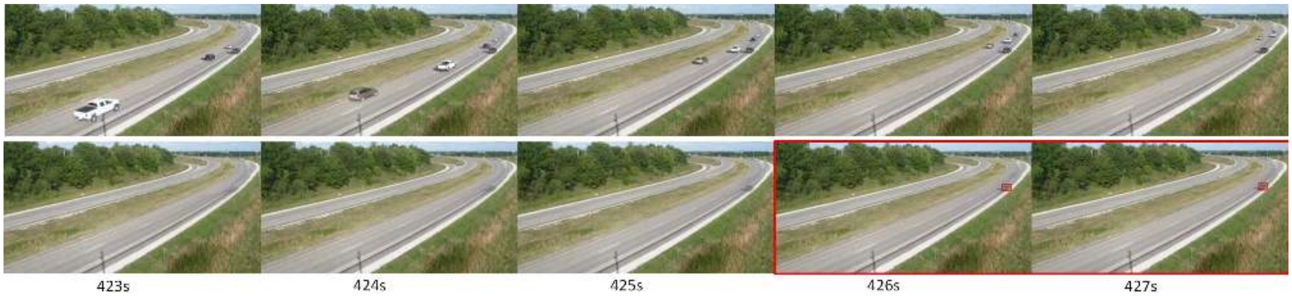
However, even the Faster R-CNN has been fine-tuned for this task, it will still give some false positives due to the poor video quality as shown in Figure 7. To solve this, we design some manual rules to reduce the easy false positives, including removing over-sized detection results and selecting the stable detection results in a continuous frame sequence. However there are some hard false positives, which can not be easily removed by the designed rules. As we discuss in section 3.1.2, we build up a residual network as a binary network which can gain the performance of 98% accuracy for recognizing the vehicle images in our self-design test set. We crop the patches on the $Average$ images based the bounding boxes from Faster R-CNN, and pass through this binary classifier. In this way, most of the false positives like the examples in Figure 7 can be removed.

Moreover, in the Track 2 testing set, there might exist the camera rotating frames which make our method mistakes a new anomaly event for the same vehicle. To reduce such false positives, we also use the Car-Reid module to verify whether it is the same vehicle.

4.2. Results of the Dynamic Mode

We implement detection on moving vehicles as discussed in Section 3.2. The interval between two detections are chosen to be 200 frames in consideration about trade-off between running speed and result reliability. A wrong tracking due to moving obstacles could be corrected by previous or following detections.

A Mask-RCNN based on ResNet101 [6] is used in our object detection and segmentation. Then we randomly take 300 sample points and track the trajectories of every point. After outliers filtering, we are able to retrieve the velocities of detected vehicles as shown in Figure 8. The abnormal vehicles are always static thus have a significant difference between normal ones. Then, we find the 25 percentage (Q_1) and 75 percentage (Q_2) and calculate the **Inter**-**q**uartile **R**ange (IQR) on logarithm velocity values, which is the difference between Q_1 and Q_2. The lower bound of normal velocities are shown as Equation 10.



(a) Video 49 on the Track 2 testing set

(b) Video 51 on the Track 2 testing set

(c) Video 73 on the Track 2 testing set

Figure 6. The result in video 49 (a), video 51 (b) and video (c) on the Track 2 testing set of static mode analysis. In (a), (b) and (c), The first row shows the occurrence of an anomaly event in the original video, while the second row shows the corresponding running average images and the detection results. We also plot the corresponding time of each frame in original videos.

Figure 7. The false positives of Faster R-CNN. Those false positives can be removed by our binary classifier.

$$B_l = Q_1 - 1.5IQR. \tag{10}$$

Therefore, a list of abnormal frame number and mask index can be retrieved.

However, there might exist multiple anomaly events in one video, so merely choosing the earliest frame of anomaly is insufficient. In addition, we match the suspicious masks of two continuous detections, and measure their **I**ntersection **O**ver **U**nion (IOU). The pairs with IOU more than 0.9 are regarded as the same vehicle. Then, we do a backward check from the last frame and the abnormal frame is proposed as the first appearance of abnormal object.

(a) Normal Cases

(b) Abnormal Cases

Figure 8. The velocity chart of normal (a) and abnormal cases (b). The images in the left side show single-object trajectory of frame 5,20,40,80 and 100. The curves in the right side show the all objects' velocity including one stalled vehicle.

Table 1. Our result on the Track 2 testing set.

	F1	RMSE	Local S2
Our result	0.8649	3.6152	0.8628

4.3. Evaluation on Track 2 testing set

Evaluation for the Track 2 testing set will be based on model anomaly detection performance, measured by the F1-score, and detection time error, measured by RMSE. Specifically, the Track 2 score will be computed as:

$$\mathrm{S2} = \mathrm{F1} * (1 - \mathrm{NRMSE}). \tag{11}$$

Here, the detection time error is the RMSE between the ground truth anomaly time and predicted anomaly time for all TP predictions. NRMSE is the normalized RMSE score across all teams, obtained via min-max normalization given all team submissions.

We evaluate our method on the Track 2 testing data and obtain the best result as shown in Table 1. As you can see, we achieve 0.8649 F1-score while detection time error is only 3.6152 seconds, which demonstrates our proposed method's superiority and robustness. Local S2 score is obtained to 0.8621 by Equation 11. The final Learderboard results among all the teams are shown in Figure 9, we achieve 0.8649 S2 score and rank the first place among all the participant teams.

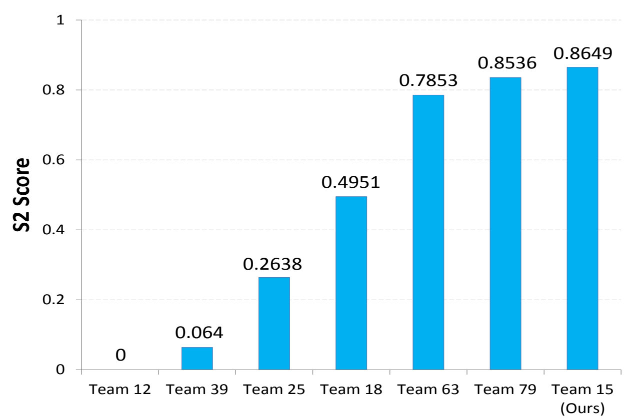


Figure 9. Compared results on the Track 2 testing set from the Leaderboard.

5. Conclusions

In this paper, we present a dual-mode motion pattern learning model for anomaly detection in urban road traffic, which jointly analyzes the static and dynamic properties of the vehicles. We evaluate our method in real scenes: Track 2 of NVIDIA AI CITY CHALLENGE and obtain 0.8641 F1-score with only 3.6152 Seconds detection time error. Our results rank the first place on the Track 2 testing set among all the participant teams, which demonstrates that the superiority of our method. In the future work, we plan to improve our method to be more robust on the traffic videos with very low qualities.

References

[1] L. Bertinetto, J. Valmadre, J. F. Henriques, A. Vedaldi, and P. H. Torr. Fully-convolutional siamese networks for object tracking. In *European Conference on Computer Vision*, pages 850–865, 2016.

[2] M. Deublein, M. Schubert, B. T. Adey, J. Köhler, and M. H. Faber. Prediction of road accidents: A bayesian hierarchical approach. *Accident Analysis & Prevention*, 51:274–291, 2013.

[3] V. Douangphachanh and H. Oneyama. Exploring the use of smartphone accelerometer and gyroscope to study on the estimation of road surface roughness condition. In *International Conference on Informatics in Control, Automation and Robotics*, volume 1, pages 783–787, 2014.

[4] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, pages 2672–2680, 2014.

[5] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *IEEE International Conference on Computer Vision*, pages 2980–2988, 2017.

[6] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.

[7] S. Kamijo, Y. Matsushita, K. Ikeuchi, and M. Sakauchi. Traffic monitoring and accident detection at intersections. *IEEE Transactions on Intelligent Transportation Systems*, 1(2):108–118, 2000.

[8] L. Kratz and K. Nishino. Anomaly detection in extremely crowded scenes using spatio-temporal motion pattern models. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1446–1453, 2009.

[9] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, pages 1097–1105, 2012.

[10] S. Lyu, M.-C. Chang, D. Du, L. Wen, H. Qi, Y. Li, Y. Wei, L. Ke, T. Hu, M. Del Coco, et al. Ua-detrac 2017: Report of avss2017 & iwt4s challenge on advanced traffic monitoring. In *IEEE International Conference on Advanced Video and Signal Based Surveillance*, pages 1–7, 2017.

[11] V. Mahadevan, W. Li, V. Bhalodia, and N. Vasconcelos. Anomaly detection in crowded scenes. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1975–1981, 2010.

[12] S. Mohammadi, A. Perina, H. Kiani, and V. Murino. Angry crowds: detecting violent events in videos. In *European Conference on Computer Vision*, pages 3–18, 2016.

[13] H. Mousavi, S. Mohammadi, A. Perina, R. Chellali, and V. Murino. Analyzing tracklets for the detection of abnormal crowd behavior. In *IEEE Winter Conference on Applications of Computer Vision*, pages 148–155, 2015.

[14] H. Nam and B. Han. Learning multi-domain convolutional neural networks for visual tracking. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 4293–4302, 2016.

[15] X. Ouyang, S. Kawaai, E. G. H. Goh, S. Shen, W. Ding, H. Ming, and D.-Y. Huang. Audio-visual emotion recognition using deep transfer learning and multiple temporal models. In *ACM International Conference on Multimodal Interaction*, pages 577–582, 2017.

[16] M. Ravanbakhsh, E. Sangineto, M. Nabi, and N. Sebe. Training adversarial discriminators for cross-channel abnormal event detection in crowds. *arXiv preprint arXiv:1706.07680*, 2017.

[17] 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*, 39(6):1137–1149, 2017.

[18] M. Riveiro, M. Lebram, and M. Elmer. Anomaly detection for road traffic: A visual analytics framework. *IEEE Transactions on Intelligent Transportation Systems*, 18(8):2260–2270, 2017.

[19] M. Sabokrou, M. Fayyaz, M. Fathy, and R. Klette. Deep-cascade: cascading 3d deep neural networks for fast anomaly detection and localization in crowded scenes. *IEEE Transactions on Image Processing*, 26(4):1992–2004, 2017.

[20] V. Saligrama and Z. Chen. Video anomaly detection based on local statistical aggregates. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2112–2119, 2012.

[21] T. Schlegl, P. Seeböck, S. M. Waldstein, U. Schmidt-Erfurth, and G. Langs. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In *International Conference on Information Processing in Medical Imaging*, pages 146–157, 2017.

[22] N. Silva, J. Soares, V. Shah, M. Y. Santos, and H. Rodrigues. Anomaly detection in roads with a data mining approach. *Procedia Computer Science*, 121:415–422, 2017.

[23] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[24] W. Sultani, C. Chen, and M. Shah. Real-world anomaly detection in surveillance videos. *arXiv preprint arXiv:1801.04264*, 2018.

[25] W. Sultani and J. Y. Choi. Abnormal traffic detection using intelligent driver model. In *Pattern Recognition (ICPR), 2010 20th International Conference on*, pages 324–327, 2010.

[26] A. Vittorio, V. Rosolino, I. Teresa, C. M. Vittoria, P. G. Vincenzo, et al. Automated sensing system for monitoring of road surface quality by mobile devices. *Procedia-Social and Behavioral Sciences*, 111:242–251, 2014.

[27] L. Wen, D. Du, Z. Cai, Z. Lei, M. Chang, H. Qi, J. Lim, M. Yang, and S. Lyu. UA-DETRAC: A new benchmark and protocol for multi-object detection and tracking. *arXiv CoRR*, abs/1511.04136, 2015.