Hybrid Modules for Traffic Anomalies Detection

Huy N.M. Nguyen
Advance Program in Computer Science
Faculty of Information Technology
VNUHCM University of Science
nnmhuy@apcs.vn

Duc H. Trinh
Advance Program in Computer Science
Faculty of Information Technology
VNUHCM University of Science
thduc@apcs.vn

Thuc Q. Nguyen
Advance Program in Computer Science
Faculty of Information Technology
VNUHCM University of Science
nqthuc@apcs.vn

Dang-Tam T. Le
Advance Program in Computer Science
Faculty of Information Technology
VNUHCM University of Science
ltdtam@apcs.vn

***Abstract*—Detecting anomalous vehicles through surveillance videos is a demanding traffic problem due to its potential to reduce severity of accidents. However, low resolution of traffic surveillance videos, wide variety of weather conditions in which videos were recorded, and complex behavior of anomalous vehicles make it difficult for any computer vision to handle all anomalous cases in reality. The authors propose a hybrid anomalies detection module, which based on background extraction combined with motion analysis, to detect traffic anomalies given conditions that camera angles are fixed and videos are recorded in daylight. Examining our method on the subset of Track 3 training set of NVIDIA AI CITY CHALLENGE 2019, we obtain 0.6677 F1-Score and 0.565 S2-score.**

I. Introduction

Video surveillance technologies are used widely in traffic management systems due to their valuable information about traffic congestion, accidents, and illegal driving behaviors. Surveillance videos also give us ability to have immediate responses to anomalous event on roads, especially ones that cause massive crashes, However, watching all recorded videos and detecting abnormal events from those is a labour-intensive task. Therefore, there is an increasing demand for an automatic anomalies detection algorithm that have small latency but still give precise result from surveillance traffic cameras.

There are many works on abnormal detection on specific context. For example, crowd [1], [2], street, avenue or subway [3]–[7], and road intersections [8]. Abnormal events, which often have complicate combination of behaviors, do not have a clear definition. Moreover, an anomalous event is also defined in terms of its context and surrounding objects. For instance, a car that stop on a highway could be considered as abnormal event, but is normal if it stop nearby a parking site. A general approach for all kinds of anomaly is not feasible, yet often does not have high accuracy. Therefore, the authors decided to focus on detecting anomalies in highway traffic via surveillance videos.

We propose a hybrid anomalies detection method which based on static and dynamic motion pattern of vehicle on highways. We observed that vehicles would normally keep moving except some special situations such as waiting for red light or congestion. From this observation, vehicles which are not moving, especially over a long period of time, are more likely to be part of anomalous events. Therefore, we get rid of moving vehicles from videos by calculate running average of contiguous video frames, then deploy a deep learning model to detect static vehicles from other static objects (roads, trees, house, etc...) in background images.

However, the approach based on static vehicles is not feasible for cases that anomalies occur in a far distance from camera. In those cases, the image of stalled vehicle is very small that is difficult to detect even with human eyes. With the observation that abnormal vehicles (i.e running into emergency lane, having a car crash or a rollover) have separated trajectory from normal ones. We combine the static vehicles detection with a motion tracking method. We use PWC-Net flow to calculate motion of all pixels in video frames to have motion vectors. Each motion vector of a pixel is compared with the cumulative motion vectors of the region neighbouring to that pixel to distinguish unusual motions of abnormal vehicles.

Compiling results of the two approaches gives out a confidence score to conclude anomaly events.

Eventually, We have experimented our method with NVIDIA AI CITY CHALLENGE 2019 track-3 dataset. With the method based on car detection, we achieve 0.667 in F1-score and 0.565 in S2-score. That also means we can reach the top 3 of NVIDIA AI CITY 2018 CHALLENGE. Nevertheless, the result doesn't satisfy our ambition, and for real world problems, we must do more than that.

II. Related Works

Previous works on the field of anomalies detection can be divided into three main approaches:

A. Hand-crafted features based method

Hand-crafted features based approaches handcraft features or extract them from training datasets for models to learn characteristics or pattern of normal scenario. Events whose features are too different from ones learned by those models would be identified as anomalies. Early work tends to use low-level

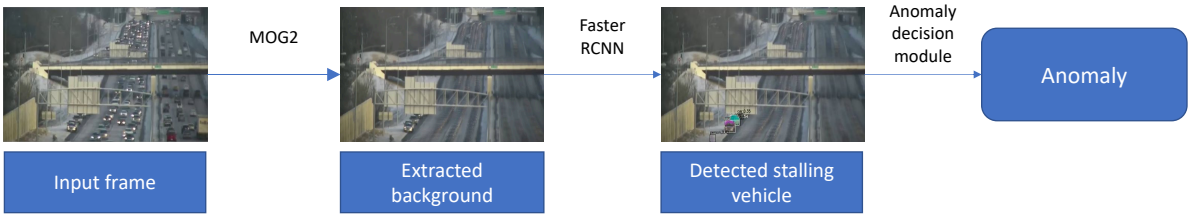

Fig. 1. **Background Modeling Based Method**

trajectory, which is the sequence of image coordinate frame by frame, to represent patterns of normal events [9], [10]. However, these methods are easy to fail in case of occlusion and shadows appearing in videos because trajectory features are based on object tracking and most tracking methods do not work well in this situation. Taking the instability of trajectory features into consideration, low-level spatial-temporal features, such as histogram of oriented gradients (HOG) [11] and histogram of oriented flows (HOF) are used. [12]. Sparse coding and dictionary learning is also used to encode regularity [5], [13], [14]

B. Deep Learning based method

Deep Learning methods have shown their success in computer vision tasks, especially detecting and tracking object [15]–[17]. Convolutional Neural Networks (CNN) has strong capability to learn spatial features and some Recurrent Neural Networks (RNN), such as Long Short Term Memory (LSTM) are widely used for sequential data modeling. In [3], Hasan *et al.* propose a 3D convolutional auto-encoder (ConvAE) to model regular frames. Futhermore, taking advantages of CNN and LSTM networks, [7] use a Convolutional LSTMs Auto-Encoder (ConvLSTM-AE) to learn normal appearance and motions at the same time, which improve the performance of ConvAE approaches. W.Lou *et al.* in [18] combine sparse coding based approach with a stacked RNN framework. Most of these works based on an assumption that reconstruction cost to discriminate abnormal and normal event. However, in road scenes, the reconstruction cost between frames are usually high given that there are no anomalies appearing in video. Therefore, these models have low performances when testing on data like NVIDIA AI CITY CHALLENGE.

C. Frame prediction method

Prediction learning attracts many researchers because of its potential in video features learning. In [4], W.Liu *et al.* use GAN (Generative Adversarial Network) to generate future frames from contiguous past frame, then based on difference in low-level features between predicted and actual frame to differentiate normal and abnormal events. However, in road and vehicles context, predicted frames generated by the model are usually have large difference compared to actual ones, which result in many false alarm in practice.

III. Method

A. Overall Framework

B. Background Modeling Based Method

With the hypothesis that after almost anomaly, there will be an immobile car, the authors propose a system for detecting that type of anomaly, which is composed of three modules and illustrated in figure 1. The first module remove foreground out of every frame by using MOG2. Then, we apply Faster-RCNN to detect car in the background images. Finally, we design a module to assess the probability of an abnormal event based on result from the second module.

1) ***Background Extraction****:* In the first module, our purpose is removing moving vehicles and there are several algorithms for that. Particularly, we apply the MOG2 in our system, which is implemented by OpenCV, and is introduced by two papers [19] [20]. It is a Background/Foreground Segmentation Algorithm, which is based on Gaussian mixture. It is better than MOG, because with each pixel, it choose appropriate number of gaussian distribution. Therefore, it can adapt to varying scenes.

The probable background is the color that stay more static in a period of time T. In this problem, traffic congestion is a common case, so if T is not large enough, we cannot filter vehicles that move slowly, but if T is too large, the vehicles will be blur and cannot detect correctly in the second module. After researching, we decide to set the parameter T to 120 frames, that is equivalent to 4 seconds (30 fps).

2) ***Vehicle Detection****:* After removing moving vehicles from extracted background images from previous step, we need to locate the positions of static vehicles. In this step, we use a Faster RCNN as a detector. The lack of computation resource does not allow us to train Faster RCNN network with traffic and vehicle image. Therefore, we decided to use a pretrained model on COCO Dataset [21]. The pretrained model we used can be found in this *Github repository*.

The pretrained model, which was trained on COCO Dataset, has its benefits and drawbacks. The model was trained well enough to learn the representation as well as location of the object in images. However, the ability of the model that can perform detection a wide range of objects also raise chance that it gives wrong label for object detected. In our experiments, the pretrained model sometimes confused between

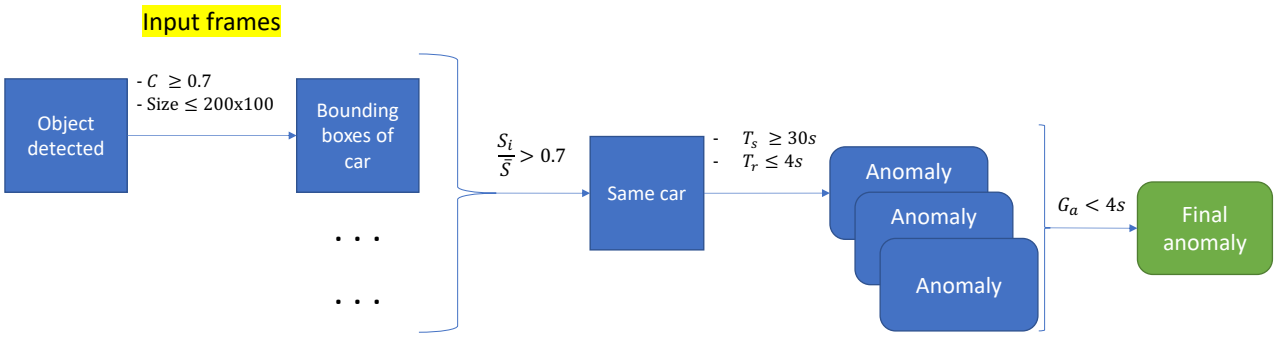

Fig. 2. **Decision module**

cars and trees or roads and airplanes. Furthermore, Vehicles in NVIDIA AI CITY CHALLENGE usually occupy small portion of area compare to the size of background extracted images (which is 800×410). This could be the reason why the model is inability to detect some vehilces in some videos of NVIDIA AI CITY CHALLENGE Dataset.

To solve the problem with the size of vehicles in extracted images, we propose a method based on the analysis of Tsung-Yi Lin in [22]. They proposed that higher resolution images could imporve the performance of Faster RCNN. We divide the images into 9 parts with equal width and equal heights. Each part of the original images is zoomed with ration $3x$, then feed to Faster RCNN to get the location of vehicles. Each location is stored with three information: coordinates of the bounding box contains detected object, the confidence score and the label of that object (we use class id of COCO Dataset). The coordinates of each part is transformed to match the coordinates in the original images.

*3) **Anomalies Detection**:* The module is shown in figure 2. After the second module, result for each image is a list of vehicles' position, and we just accept the vehicle if its confidence score C is higher than 0.7 and its size is smaller than $1/16$ of the image. In each frames, we compared its vehicles with the current vehicle list and two vehicles are consider same vehicle if the gap of time T_{r} between two frames less than 4 seconds and $intersectarea/averagearea >= 0.7$. In order to reduce the effect of false detection, traffic congestion, or stopping by traffic light, an abnormal vehicle must be stay in the background longer than 30 seconds (T_{s}). Finally, we merge some abnormal events if the beginning of the event G_{a} is no longer than 4 seconds from the ending of the previous event.

IV. Result

In Background Modeling Based Method, after many experiments, we realize that the performance of our proposed system mostly depends on the vehicle detection module. Therefore, we have tried our system with some different detector and the result is shown in table [PUT TABLE REF HERE].

As we can see, the performance when using Faster-RCNN is better than YOLO and OpenCV classification, and the performance of OpenCV classification is the worst of three detector. At the beginning, we tested our detection system with Cv2.classification and the performance was very low, just 0.17 in F1-score and 0.076 in S2-score. Then the authors changed to use YOLO as detecting vehicle model, which got a significant improvement in both F1 and S2 score (approximately 4 times). Finally, we chose faster-RCNN and that is a trade-off between running time and performance. In comparison with YOLO, faster-RCNN gives a slightly higher in F1-score (from 0.6 to 0.667) and considerable higher in S2-score (from 0.309 to 0.565). When using faster-RCNN, we also make some experiments with different confidence threshold and reach the highest performance with threshold around 0.65. However, there is still a distance between our result and the result of the winner in NVIDIA AI CITY 2018 CHALLENGE, and we still have a long way to go.

References

[1] S. Mohammadi, A. Perina, H. K. Galoogahi, and V. Murino, "Angry crowds: Detecting violent events in videos," in *ECCV*, 2016.

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

[3] M. Hasan, J. Choi, J. Neumann, A. K. Roy-Chowdhury, and L. S. Davis, "Learning temporal regularity in video sequences," *CoRR*, vol. abs/1604.04574, 2016.

[4] W. Liu, W. Luo, D. Lian, and S. Gao, "Future frame prediction for anomaly detection - A new baseline," *CoRR*, vol. abs/1712.09867, 2017.

[5] Y. Cong, J. Yuan, and J. Liu, "Sparse reconstruction cost for abnormal event detection," in *CVPR 2011*, June 2011, pp. 3449–3456.

[6] J. R. Medel and A. E. Savakis, "Anomaly detection in video using predictive convolutional long short-term memory networks," *CoRR*, vol. abs/1612.00390, 2016.

[7] Y. S. Chong and Y. H. Tay, "Abnormal event detection in videos using spatiotemporal autoencoder," *CoRR*, vol. abs/1701.01546, 2017.

[8] W. Sultani and J. Y. Choi, "Abnormal traffic detection using intelligent driver model," in *2010 20th International Conference on Pattern Recognition*, Aug 2010, pp. 324–327.

[9] S. Wu, B. E. Moore, and M. Shah, "Chaotic invariants of lagrangian particle trajectories for anomaly detection in crowded scenes," *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 2054–2060, 2010.

[10] F. Tung, J. S. Zelek, and D. A. Clausi, "Goal-based trajectory analysis for unusual behaviour detection in intelligent surveillance," *Image Vision Comput.*, vol. 29, pp. 230–240, 2011.

[11] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 1, June 2005, pp. 886–893 vol. 1.

[12] N. Dalal, B. Triggs, and C. Schmid, "Human detection using oriented histograms of flow and appearance," in *Computer Vision – ECCV 2006*, A. Leonardis, H. Bischof, and A. Pinz, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 428–441.

[13] C. Lu, J. Shi, and J. Jia, "Abnormal event detection at 150 fps in matlab," *2013 IEEE International Conference on Computer Vision*, pp. 2720–2727, 2013.

[14] B. Zhao, L. Fei-Fei, and E. P. Xing, "Online detection of unusual events in videos via dynamic sparse coding," *CVPR 2011*, pp. 3313–3320, 2011.

[15] J. Redmon, S. K. Divvala, R. B. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," *CoRR*, vol. abs/1506.02640, 2015.

[16] R. B. Girshick, "Fast R-CNN," *CoRR*, vol. abs/1504.08083, 2015.

[17] S. Ren, K. He, R. B. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," *CoRR*, vol. abs/1506.01497, 2015.

[18] W. Luo, W. Liu, and S. Gao, "A revisit of sparse coding based anomaly detection in stacked rnn framework," in *2017 IEEE International Conference on Computer Vision (ICCV)*, Oct 2017, pp. 341–349.

[19] Z. Zivkovic, "Improved adaptive gausian mixture model for background subtraction," *Pattern Recognition, ICPR 2004. Proceedings of the 17th International Conference*, vol. 2, pp. 28–31, 2004.

[20] Z. Zivkovic and F. V. D. Heijden, "Efficient adaptive density estimation per image pixel for the task of background subtraction," *Pattern recognition letters*, vol. 27, pp. 773–780, 2006.

[21] T.-Y. Lin, M. Maire, S. J. Belongie, L. D. Bourdev, R. B. Girshick, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *ECCV*, 2014.

[22] Q. Fan, L. M. Brown, and J. Smith, "A closer look at faster r-cnn for vehicle detection," *2016 IEEE Intelligent Vehicles Symposium (IV)*, pp. 124–129, 2016.