An Efficient Event Detection Through Background Subtraction and Deep Convolutional Nets

Kahlil Muchtar[1,2](✉), Faris Rahman[1],
Muhammad Rizky Munggaran[1], Alvin Prayuda Juniarta Dwiyantoro[1],
Richard Dharmadi[1], Indra Nugraha[1], and Chuan-Yu Chang[3]

[1] Nodeflux, Jakarta, Indonesia
kahlil@nodeflux.io
[2] Department of Electrical and Computer Engineering,
Syiah Kuala University, Banda Aceh, Aceh, Indonesia
[3] Department of Computer Science and Information Engineering,
National Yunlin University of Science and Technology, Yunlin, Taiwan

Abstract. The smart transportation system is one of the most essential parts in a smart city roadmap. The smart transportation applications are equipped with CCTV to recognize a region of interest through automated object detection methods. Usually, such methods require high-complexity image classification techniques and advanced hardware specification. Therefore, the design of low-complexity automated object detection algorithms becomes an important topic in this area. A novel technique is proposed to detect a moving object from the surveillance videos based on CPU (central processing units). We use this method to determine the area of the moving object(s). Furthermore, the area will be processed through a deep convolutional nets-based image classification in GPU (graphics processing units) in order to ensure high efficiency and accuracy. It cannot only help to detect object rapidly and accurately, but also can reduce big data volume needed to be stored in smart transportation systems.

Keywords: Background subtraction · Deep convolutional nets · Smart city

1 Introduction

In practice, the cloud computing based smart transportation applications face significant challenges. Although they require real-time object detection, transferring the massive amount of raw full frame data to cloud centers not only leads to uncertainty in timing but also poses extra workload to the communication networks [1, 2].

Related to this problem, this research aims to accelerate this process by comprehensively considering both algorithm design and implementation. For algorithm design, a block texture-based approach is chosen for the determination of moving object areas. For algorithm implementation, a deep convolutional nets is parallelized on modern graphics processing units (GPU) to improve the recognition efficiency.

© Springer Nature Singapore Pte Ltd. 2019
C.-Y. Chang et al. (Eds.): ICS 2018, CCIS 1013, pp. 163–167, 2019.
https://doi.org/10.1007/978-981-13-9190-3_16

The most relevant related work is proposed by Kim et al. [3] which introduces a hybrid framework to detect the moving person rapidly. The method utilizes a GMM background subtraction to find the region of interest (ROI) [4]. However, the GMM is prone to noise and very sensitive to illumination changes. In our preliminary work, we aim to leverage a block texture-based foreground extraction as a new ROI extractor. Finally, the obtained ROI will feed the deep convolutional nets to classify objects more efficient and accurate. Our proposed system might be useful as a surveillance system, which has a strict hardware and network specification in the end-user side.

2 Proposed Method

2.1 ROI Extractor

Our proposed model is based on the model uses a non-overlapping block to extract the foreground (FG) [5]. The initial step is to divide the current frame into *n* x *n* non-overlapping blocks. The process converts each block into a binary bitmap. When a new observed bitmap, BM_{obs}, comes in, it will be compared against the several numbers of weights BG model BM_{mod}, where the total number of weights is set as parameter *K* in the algorithm. From [5], we only perform the 1-bit mode due to its efficiency.

$$Dist(BM_{obs}, BM_{mod}) = \sum_{i=1}^{n} \sum_{j=1}^{n} (b_{ij}^{obs} \oplus b_{ij}^{mod}) \quad (1)$$

The corresponding frame rate for original GMM (before post-processing) [4] and 1-bit mode [5] are about 30 and 62 frames/second, respectively. For clarity purposes, we draw the detailed step of ROI generation in Fig. 1.

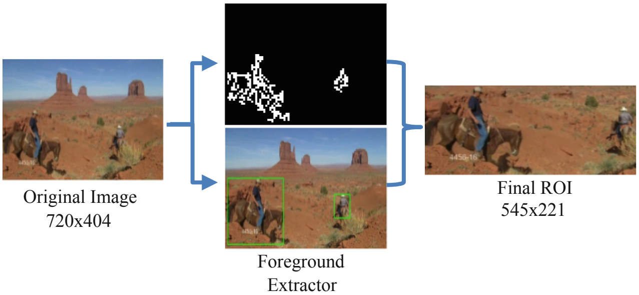


Fig. 1. Illustration of ROI generation

2.2 Deep Detector

To achieve our goal to design a real-time unified technique, we select YOLO as a deep detector. YOLO [6, 7], a state-of-the-art detection method, which first divides an image into grid cells. It then performs the prediction of the coordinates of bounding boxes and the probabilities of each cell. Every bounding boxes have their own confidence score, which is calculated by aggregating the probabilities. Therefore, a smaller number of grid cells of an incoming frame, faster detection will be (Fig. 2).

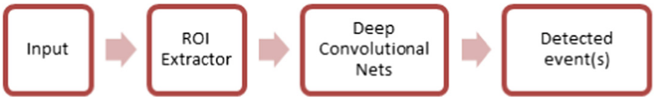
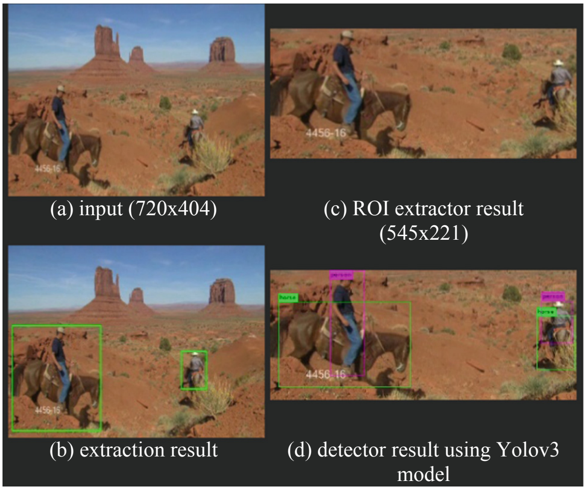


Fig. 2. The complete proposed workflow system

3 Experimental Results

In this section, we discuss a public datasets in order to evaluate the entire proposed system. The dataset is UCF-Sports dataset [8, 9] that provides several sports actions, for example, people riding a horse, walking, diving, etc.

3.1 UCF-Sports Dataset: Riding-Horse and Walk-Front



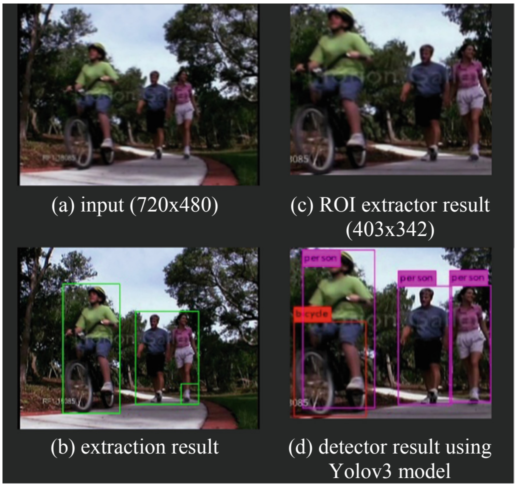
(a) input (720x404)

(c) ROI extractor result (545x221)

(b) extraction result

(d) detector result using Yolov3 model

Fig. 3. UCF-Sports Dataset: Riding-Horse (frame no: 23)



(a) input (720x480)

(c) ROI extractor result (403x342)

(b) extraction result

(d) detector result using Yolov3 model

Fig. 4. UCF-Sports Dataset: Walk-Front (frame no: 49)

3.2 Frame Rate and Confidence Scores

In Table 1, we provide confidence scores for all tested videos. The detection uses a common Yolov3 model, which is publicly available [6]. As depicted below, the scores represent the corresponding bounding boxes from Figs. 3 and 4. In Table 2, we evaluate the frame rate comparison. In [3], Kim et al. tested 64 × 64 pixels of the input frame, and the entire framework achieved 15 fps on an NVIDIA Tesla M40. Our workflow yields a significant improvement in terms of frame rate that achieve ~16,62 fps for relatively higher input size on an NVIDIA GTX 1050 Ti. Both devices are equipped with 12 GB memory size.

Table 1. Confidence scores of Figs. 3 and 4.

Frame no.	Datasets	Confidence scores using Yolov3 model
23	UCF-Sports (riding-horse)	horse: 97%, person: 93%, horse: 53%, person: 96%
49	UCF-Sports (walk-front)	bicycle: 99%, person: 100%, person: 100%, person: 100%

Table 2. Representative frame rate comparison.

	Yolov3 model (80 object classes)	Our vehicle model (24 object classes)
Entire-frame detection	14,74 frames/sec	22,02 frames/sec
ROI-based event detection	**16,62** frames/sec	**24,57** frames/sec

4 Conclusions

In this paper, we introduced an approach for low-cost smart transportation application. It was thoroughly evaluated that incorporating an ROI robust extractor and efficient deep learning is beneficial for a light and real-time smart system.

References

1. Xu, R., et al.: Real-time human objects tracking for smart surveillance at the edge. In: IEEE International Conference on Communications (ICC) 2018, Kansas City, MO, USA (2018)
2. Zheng, R., Yao, C., Jin, H., Zhu, L., Zhang, Q., Deng, W.: Parallel key frame extraction for surveillance video service in a smart city. PLoS ONE **10**, e0135694 (2015)
3. Kim, C., Lee, J., Han, T., Kim, Y.-M.: A hybrid framework combining background subtraction and deep neural networks for rapid person detection. J. Big Data **5**, 22 (2018)
4. Stauffer, C., Grimson, W.E.L.: Adaptive background mixture models for real-time tracking. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, New York, pp. 246–252 (1999)
5. Yeh, C.-H., Lin, C.-Y., Muchtar, K., Kang, L.-W.: Real-time background modeling based on a multi-level texture description. Inf. Sci. **269**, 106–127 (2014)
6. Redmon, J., Farhadi, A.: YOLO9000: better, faster, stronger. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 2017, Honolulu, HI, USA, pp. 6517–6525 (2017)
7. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: unified, real-time object detection. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, pp. 779–788 (2016)
8. Rodriguez, M.D., Ahmed, J., Shah, M.: Action MACH: a spatio-temporal maximum average correlation height filter for action recognition. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2008), Anchorage, AK, USA (2008)
9. Soomro, K., Zamir, A.R.: Action recognition in realistic sports videos. In: Moeslund, T.B., Thomas, G., Hilton, A. (eds.) Computer Vision in Sports. ACVPR, pp. 181–208. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-09396-3_9