Object Detection for Neighbor Map Construction in an IoV System

[1,a]Kuan-Wen Chen, [2]Shen-Chi Chen, [3]Kevin Lin, [5]Ming-Hsuan Yang, [3,4]Chu-Song Chen, [2,3,b]Yi-Ping Hung
[1]Intel-NTU Connected Context Computing Center
[2]Dept. of Computer Science and Information Engineering, National Taiwan University
[3]Graduate Institute of Networking and Multimedia, National Taiwan University
[4]Institute of Information Science, Academia Sinica, Taipei, Taiwan
[5]Electrical Engineering and Computer Science, University of California at Merced, Merced, USA
[a]kuanwenchen@ntu.edu.tw, [b]hung@csie.ntu.edu.tw

***Abstract*—Many applications of machine-to-machine (M2M) based intelligent transportation systems highly rely on the accurate estimation of neighbor map, where neighbor map mentions the locations of all nearby vehicles and pedestrians. To build the neighbor map, it usually integrates multiple sensors, such as GPS, odometer, inertial measurement unit (IMU), laser scanners, cameras, and RGB-D cameras. In this paper, we build a M2M framework to estimate the neighbor map and focus on the improvement of vehicle and pedestrian detection of most popular sensors, camera. We propose a novel grid-based object detection approach and deal with cameras on both roadside units and vehicles. It adapts to the environments and achieves high accuracy, and can be used to improve the performance of neighbor map estimation.**

Keywords-object detection, neighbor map, internet of vehicles, intelligent transportation system

I. INTRODUCTION

Intelligent transportation system (ITS), which has been extensively researched in the last decade, complies advanced mechanisms to provide innovative, proactive services relating to traffic management and driving safety. For example, drivers' behaviors are limited to their line of sight, and most of car accidents are caused by the driver's view being occluded by other vehicles. Fig. 1 shows one example where three vehicles A, B, and C enter an intersection. As the view of vehicles A and B is partially occluded by vehicle C, there is a potential collision danger. If all vehicles are connected, vehicle C is in a position to send warning signals to vehicle A and B to avoid traffic collision.

Connected vehicles cannot only share their sensory information, but also actively send out alerts to nearby vehicles in danger [1]. Forming an even larger vehicular network, comprising connected vehicles and infrastructures, make it possible to proactively perform load balancing across multiple routes. It is anticipated that traffic accidents can be eliminated from one of the leading causes of death [2] and the catastrophic ones can be effectively prevented. Thus, a machine-to-machine (M2M) based ITS has the following benefits. First, it expands the sensor coverage. Second, it increases the time that allowed driver to react, because the vehicles can pass through the warning signals. Third, it help doing some medication of bidding for right of way, for example, our vehicle can tell us whether it is safe or not to change lane or we should wait another vehicle passing first. To achieve such M2M-based ITS, one of the main research topics is to build the neighbor map, i.e. to estimate the positions of all surrounding targets, by integrating the information obtained from nearby vehicles or infrastructure via V2V (vehicle-to-vehicle) or V2I (vehicle-to-infrastructure) communications. For example, it helps generating warnings when two vehicles are too close.

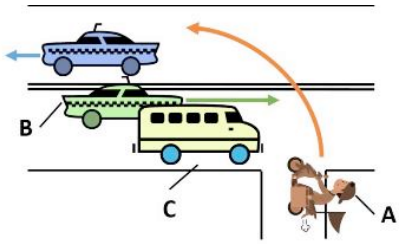


Fig. 1. An example of M2M-based ITS

In the following sections, we will introduce the M2M-framework of neighbor map estimation first. Section 3 describes the object detection by cameras. Experimental results are shown in Section 4. We draw the conclusion in Section 5 finally.

II. M2M-BASED NEIGHBOR MAP ESTIMATION

We build a M2M framework [3] to estimate the neighbor map based on sensor fusion and belief merge algorithm [4]. The system architecture is shown in Fig. 2. For each vehicle or roadside unit (RSU), the **signal processing** component analyzes the data from multiple sensors, such as GPS, odometer, camera, LIDAR, and etc., to estimate its own global position and the relative positions of observed nearby targets. Then, **sensor fusion** component integrate the results of multiple sensors, and further Kalman filter [5] and multi-hypothesis tracking algorithm are used to generate the local belief. After that, the local belief is broadcast by the **communication** component to other nearby vehicles. For broadcast communications, we use monitoring mode WiFi which enables devices to connect to each other without any access point. Finally, each vehicle will apply the global **belief merge**, which fuse its own local belief with other receiving beliefs to get the final estimation results. More details can be seen in [3].

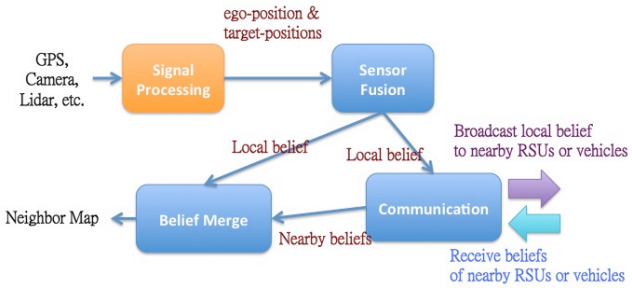


Fig. 2. System architecture of the M2M-based neighbor map construction

Although the global belief merge can improve the performance of neighbor map estimation, the framework still relies on the high accuracy of signal processing results. Furthermore, the prior works [3][4] focus on the belief merge algorithm without considering how to obtain a high accurate estimation of object detection. Thus, in this paper, we focus on the most popular sensors on RSUs and vehicles, cameras, to detect vehicles and pedestrians, and further estimates the global or relative positions of nearby targets. The object detection is still an open problem in computer vision researches [6][7], especially for the heavy occlusion cases. We propose a grid-based object detection approach and deal with cameras on both RSUs and vehicles. It adapts to the environments and can achieve high accuracy.

III. Object Detection & Position Estimation

In this section, we introduce two kinds of object detection methods. One is applied to the roadside camera, and the other is for the camera on vehicle. Then, how to estimate the global or relative positions in real world will be described finally.

A. *Object Detection for Roadside Camera*

Unlike the traditional learning-based approaches [6][7], which learns generic detectors trained offline and are usually not work well in a large diversity of scenes. We propose an adaptive grid-based approach, which discovers location-dependent features and simultaneously detects and classifies moving objects in a fixed camera.

We first employ background modelling and subtraction [8] in pre-processing. It models the background intensity of each pixel as a Gaussion mixture model. After extracting the foreground regions, each region is referred as a blob and represented by a rectangular bounding box. Then, We design a grid-based perspective dependent model (GPDM) to classify objects into three categories: pedestrian, scooter, or car. We divide the scene into $R \times C$ relatively small girds, as shown in Fig 3(a). The appearance, motion, and shape features are used to build three classifiers for different categories in each grid. Each classifier in the grid is trained by using the blobs with its center position in the grid. This method enhances the coherency of the collected features, as shown in Fig 3(b). Therefore, the grid-based model can adapt to the specific perspective and scene.

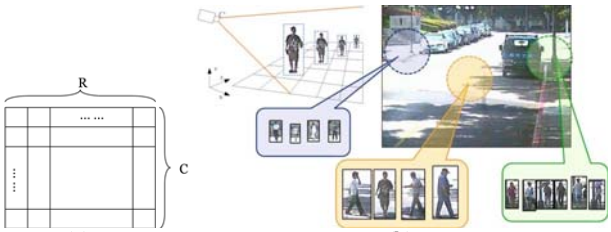


Fig. 3. (a) Divide the image into grids. (b) The GPDM enhances the coherency of the collected features.

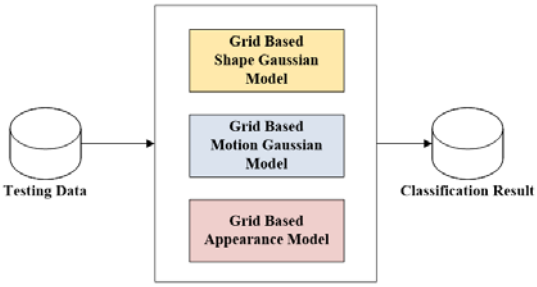


Fig. 4. The testing data classified by the combination of the results obtained from three models of the grid.

The R-HOG [6] features are used as the appearance features to learn the Linear SVM classifier for each grid. Given an object blob, we extract the average motion flow and its aspect ratio and size as motion features and shape feature, respectively, and model them as Gaussian models. After building the appearance, motion, and shape model for each grid, the object can be classified by the combination of the results obtained from three models, as show in Fig 4.

B. *Object Detection for Camera on Vehicle*

The camera on vehicle is usually moving, and so to detect foreground object based on background modeling and subtraction, as that we used for roadside camera, is unviable. Thus, we use the sliding window technique to find the object-of-interest in the given frame. It employs a filter (or classifier) to every position and every scale of an image. Several studies have proposed sliding window detection method [6][9]. Motivated by the good performance and occlusion handling strategy, while heavy occlusion usually happens when camera mounted on vehicle, we use deformable part-based model [9] with acceleration algorithm [10] to detect the target.

The deformable part-based model is constructed with the pictorial structure framework, which represent an object using a root filter and some part filters. The root filter depicts the overall appearance pattern of an object. Each part filter further describes the local appearance of the object. The object is localized when the location is voted with highest score by root and part filter.

However, compare to the object detection in the fixed camera, which mainly relies on the foreground detection to segment the interesting region, the computational cost of sliding window technique is much higher, because it has to

use multiple sizes of filter kernel to scan the image. To overcome this, we consider the sliding window approaches approximately as convolution computation. It is then speeded up by the Furrier transformation, which can be seen as simple multiplication in frequency domain.

C. Position Estimation

After detecting targets in an image, the next question is to estimate the global or relative 3D positions in real world. Here, we make two assumptions. First, the cameras can be calibrated in advance, i.e. the 3x3 perspective transform matrix, homography, between image plane and ground plane of the real world is known. The assumption is reasonable, because the roadside camera is fixed after being installed, and the camera on vehicle is also usually mounted on a fixed bracket, so the relative pose between camera and the ground plane in real world is unchanged even when the vehicle is moving. The homography can be estimate [11], when we have at least four corresponding points between the ground plane in image and the ground plane in real world. Second, we suppose all the targets are standing on the ground plane in real world.

With these assumptions, once the objects are detected in an image, we can compute its global or relative 3D position by using only one camera. First, we estimate the foot or tire position in 2D image of camera from the object detection result. Second, we project the 2D point [*x, y*] to a 3D point [*X, Y,* 0] on the ground plane in real world by the following equation:

$$s\begin{bmatrix} X & Y & 1 \end{bmatrix}^T = H \cdot \begin{bmatrix} x & y & 1 \end{bmatrix}^T, \qquad (1)$$

where H is the homography between the ground plane in image and the ground plane in real world, and s is a scale factor.

IV. Experimence

A. Object Detection by the Roadside Camera

To evaluate the performance of multi-class object detection and classification, we conduct experiments on two datasets. One is the publicly available dataset PETS2001 (Fig. 5(a)). Here, we selected the training and testing sequences recorded by camera#1 in dataset#1, #2 and #4, and recorded by camera#2 in dataset#3, respectively. The other is the datasets collected by our own in a campus, as shown in Fig. 5(b). The amount of targets in the training and testing data are depicted in Table 1. We compare our method with the generic detector using HOG [6]. Table shows the experimental results. It shows our approach is with higher accuracy than the generic detector. Some examples of detection results are shown in Fig. 6. The rectangle in red, yellow, or blue color represents it is car, bike, or pedestrian, respectively.

B. Object Detection by the Camera on Vehicle

To evaluate the object detection by the camera on vehicle, we record a video sequence of 5 minutes on road. Some examples of detection results are shown in Fig. 7. As we can see, our system can detect vehicles well even when heavy occlusion happens (Fig. 7(d)(e)).

(a) (b)

Fig. 5. Experimental datasets: (a) PETS2001, and (b) our dataset.

TABLE I. Specification of the training and testing data

		Ours	PETS01
Train	Ped#	2799	31178
	Bike#	1207	1432
	Car#	8810	2734
Test	Ped#	3528	44992
	Bike#	897	1358
	Car#	4071	6166

TABLE II. Confusion matrix of detection result

	Ours			Generic detector [6]		
	Ped	Bike	Car	Ped	Bike	Car
Ped	93.02	6.47	0.49	91.3	8.3	0.3
Bike	24.85	72.88	2.25	34.9	51.0	13.9
Car	3.33	5.44	91.22	3.6	7.3	89.0
Overall	85.7%			77.14%		

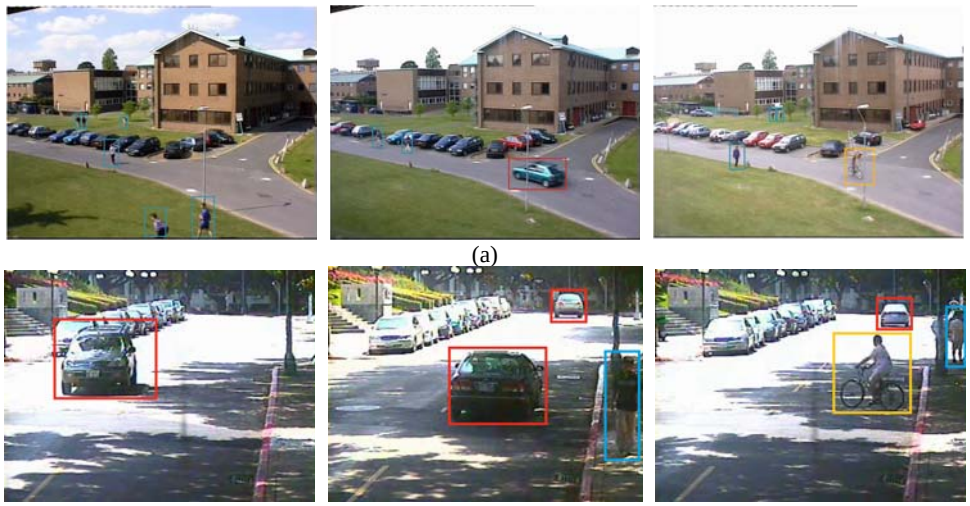
	Ours			Generic detector [6]		
	Ped	Bike	Car	Ped	Bike	Car
Ped	87.70	8.13	4.15	86.8	13.1	0
Bike	18.88	70.39	10.72	52.9	47.0	0
Car	2.26	5.15	92.58	3.4	5.4	91.0
Overall	83.55%			74.95%		

V. Conclusion

In this paper, we built a M2M framework to estimate the neighbor map, including signal processing, sensor fusion, communication, and belief merge. Furthermore, we proposed a novel grid-based object detection approach and deal with cameras on both RSUs and vehicles. Unlike traditional object detection approaches, which estimated a generic detection model, our method learnt individual model for each environment and even for each region. It is more adaptive to the appearance differences of vehicles or pedestrians in different environments, and experiences showed its reliability. A better estimation of target positions improves the performance of neighbor map construction and will be evaluated in the future.

Acknowledgment

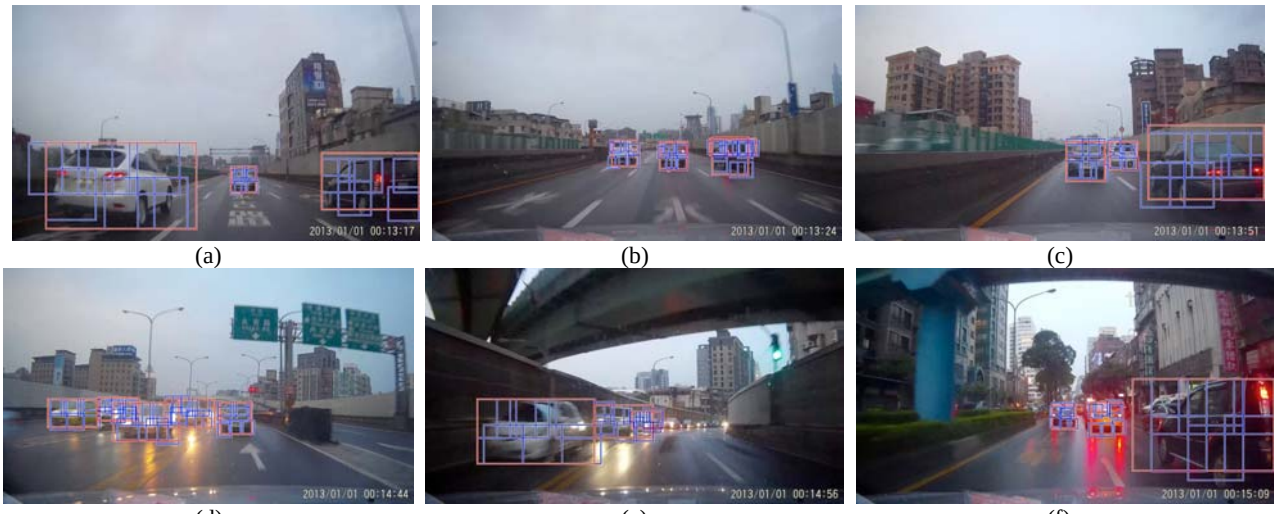
This work was also supported in part by National Science Council, National Taiwan University, and Intel Corporation under Grants NSC-102-2911-I-002-001 and NTU-102R7501.



(a)

(b)

Fig. 6. Examples of object detection in roadside cameras: (a) PETS2001, and (b) our datasets.



(a) (b) (c)

(d) (e) (f)

Fig. 7. Examples of vehicle detection in the camera on vehicle.

REFERENCES

[1] C. Stiller, G. Farber and S. Kammel, "Cooperative cognitive automobiles," in IEEE Intelligent Vehicles Symposium, 2007.

[2] R. Subramanian, "Motor vehicle traffic crashes as a leading cause of death in the United States, 2008 and 2009," National Center for Statistics and Analysis, 2012.

[3] K.W. Chen, H.M. Tsai, C.H. Hsieh, S.D. Lin, C.C. Wang, S.W. Yang, S.Y. Chien, C.H. Lee, Y.C. Su, C.T. Chou, Y.J. Lee, H.K. Pao, R.S. Guo, C.J. Chen, M.H. Yang, B.Y. Chen, and Y.P. Hung, "Connected Vehicle Safety - Science, System, and Framework," IEEE World Forum on Internet of Things, 2014.

[4] C.K. Chang, C.H. Chang, and C.C. Wang, "Communication Adaptive Multi-Robot Simultaneous Localization and Tracking via Hybrid Measurement and Belief Sharing," IEEE International Conference on Robotics and Automation, 2014.

[5] R.E. Kalman, "A new approach to linear filtering and prediction problems," Journal of basic Engineering, 82(1):35-45, 1960.

[6] N. Dalal and B. Triggs, "Histogram of oriented gradients for human detection," IEEE International Conference on Computer Vision and Pattern Recognition, 2005.

[7] P. Felzenszwalb, D. McAllester, and D. Ramanan, "A discriminatively trained, multiscale, deformable part model," IEEE International Conference on Computer Vision and Pattern Recognition, 2008.

[8] C. Stauffer and W.E.L Grimson, "Learning patterns of activity using real-time tracking," IEEE Transactions on Pattern Analysis and Machine Intelligence, 2000.

[9] P. Felzenszwalb, R. Girshick, D. McAllester, D. Ramanan, "Object Detection with Discriminatively Trained Part Based Models," IEEE Transactions on Pattern Analysis and Machine Intelligence, 32(9), 2010.

[10] C. Dubout and F. Fleuret, "Exact Acceleration of Linear Object Detectors," European Conference on Computer Vision, 2012.

[11] Z. Zhang, "A Flexible New Technique for Camera Calibration," IEEE Transaction on Pattern Analysis and Machine Intelligence, 22(11):1330-1334, 2000.