2016 IEEE 19th International Conference on Intelligent Transportation Systems (ITSC)
Windsor Oceanico Hotel, Rio de Janeiro, Brazil, November 1-4, 2016

Robust and Low Complexity Obstacle Detection and Tracking

Meiqing Wu, *Student Member, IEEE,* Chengju Zhou and Thambipillai Srikanthan, *Senior Member, IEEE*

Abstract— **Obstacle detection and tracking is essential module for autonomous driving. Vision based obstacle detection and tracking faces huge challenges due to factors like cluttered background, partial occlusion, inconsistent illumination, etc. In this paper, we propose a robust and low complexity stereo-vision based obstacle detection and tracking framework. Low complexity techniques are employed to detect obstacles in the u-v-disparity image space. In addition, effective strategies are proposed to construct a distinctive object appearance model for data association efficiently. Finally, an online multi-object tracking framework is proposed by integrating the obstacle detection and data association modules in a robust way. Extensive experimental results on the well-known KITTI tracking dataset demonstrate that the proposed method is able to detect and track various obstacles robustly and efficiently in diverse challenging scenarios.**

I. INTRODUCTION

Obstacle detection and tracking is essential module for automotive applications. Vision based obstacle detection and tracking faces huge challenges due to a number of factors [1][2]. For example, the urban environment is highly dynamic and cluttered. Obstacles can be static or subjected to regular or abrupt motion. Their appearances can also vary from time to time due to partial occlusion, illumination change and so on. In addition, the obstacle detection and tracking algorithm must be of low computational complexity as real-time response of vehicle to the environment is essential for safety related applications like collision avoidance.

In this paper, a robust and low complexity stereo-vision based obstacle detection and tracking method is proposed. Unlike the existing works that focus only on the detection of vehicles or pedestrians, the proposed obstacle detection method relies on u-v-disparity space to detect all obstacles in the scene. A Space of Interest (SOI) is defined to greatly reduce the search space of obstacles prior to employing adaptive connected component labeling techniques to segment SOI into sets of obstacles on the u-disparity image. To associate obstacles across frames, a color histogram based appearance model is constructed for each obstacle. Color histogram is employed due to its simplicity and high tolerance to scale change and partial occlusion [3]. In order to incorporate robustness of the model to inconsistent illumination, *L*a*b** color space is utilized. Moreover, pixels belonging to the background are excluded based on the depth information when constructing the histogram, which further increases the distinctiveness of the appearance model. A chessboard pattern based sparse sampling technique is also adopted to significantly reduce the number of operations and memory accesses for constructing the histogram. Finally, an online multi-object tracking framework is proposed by integrating the obstacle detection and data association modules in a robust way.

The rest of the paper is structured as follows. Section II reviews the existing works. The proposed method is presented in Section III and the experiment results are shown in Section IV. Finally, Section V concludes the paper.

II. RELATED WORKS

1) Obstacle Detection: Obstacle detection methods can generally be divided into two categories: monocular vision based and stereo vision based. Monocular vision based methods often resort to a general object recognition framework, where an implicit representation of object is learned from training samples. Due to the high inter- and intra-class variation of appearance, it is difficult to find a feature pattern that is distinctive for obstacles of all types. Existing monocular based methods mainly focus on detecting vehicles [4] or pedestrians [5]. On the other hand, since stereo vision is able to provide additional depth information, stereo vision based solutions have been regarded as the primary choice for obstacle detection [1]. Detailed reviews for obstacle detection can be found in [1], [4], [5].

2) Obstacle Tracking: Generally, a specific object tracker is characterized by several aspects: object localization, appearance model for object association and filtering [6]. A popular taxonomy is made according to object appearance model, which divides the existing works into three main categories [2][7][8]: point based tracker, contour based tracker and kernel based tracker.

The first is point based tracker. In this category, objects are modeled as a set of points [9][10][11]. In general, the performance of point based tracker is tightly related to the chosen number of feature points [11]. Small number of points may not be able to accurately model the scene while larger number of feature points requires huge computation power. Therefore, finding a suitable tradeoff between accuracy and speed is crucial. The second category is contour based tracker. Contour based tracking tightly depends on the performance of the chosen shape detector, and therefore only shows vitality in dedicated domain for certain objects [7]. The last category is kernel based tracking. Kernel based tracking has been widely studied in the literature and has demonstrated promising results in many areas [6][8][12][13][14].

Despite the tremendous progress in recent decades, object detection and tracking remains a challenging problem as

Meiqing Wu, Chengju Zhou and Thambipillai Srikanthan are with Hardware and Embedded Systems Lab, Nanyang Technological University, Singapore. e-mail:wume0007@e.ntu.edu.sg

978-1-5090-1889-5/16/$31.00 ©2016 IEEE

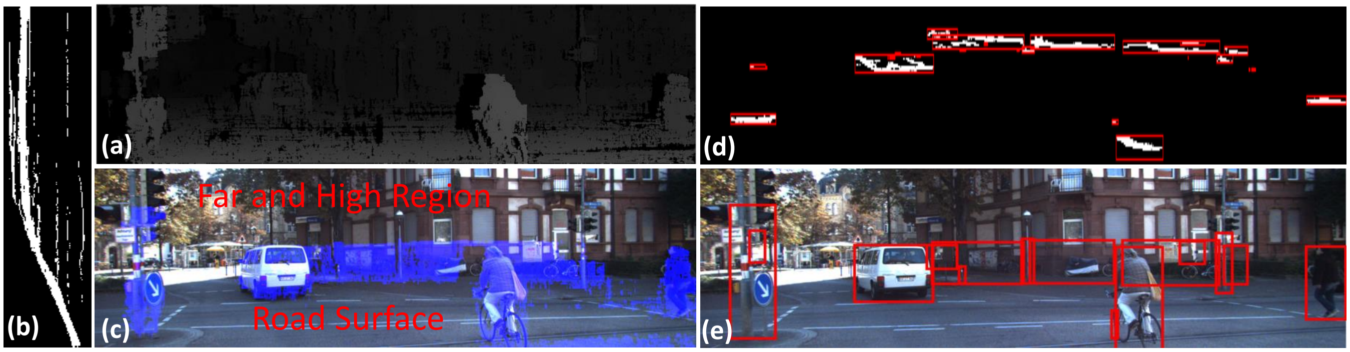


Fig. 1: Obstacle detection:(a) a disparity map; (b) the corresponding v-disparity image; (c) space of interest highlighted in blue;(d) segmented clusters on the u-disparity image; (e) obstacles detected with bounding box in red.

the object's appearance is easily affected by factors like inconsistent illumination, partial occlusion, shape deformation and change of view angle [7][15][16]. Devising a distinctive object representation model that can lead to efficient and robust object tracking is still an unresolved problem.

III. PROPOSED ALGORITHM

In this section, the proposed techniques for obstacle detection and appearance model setup are presented respectively, which are followed by an description of the proposed online multi-object tracking framework.

A. *Obstacle Detection*

Obstacles in the scene are not only restricted to vehicles and pedestrians but also traffic lights, sign posts, trees, barriers, etc. The obstacles can be standing still or be in motion. Unlike previous works which only focus on detecting vehicle or pedestrian, the proposed method detects all obstacles in the scene. This is achieved with the help of the u-v-disparity image space, which is an variant of the probabilistic occupancy grid [1]. One example is shown in Fig.1.

Given the disparity map, Space of Interest (SOI), which refers to the space where the concerned obstacles reside, is generated by removing irrelevant regions based on the knowledge of the geometrical structure of the scene. As shown in Fig.1(c), the SOI excludes the road surface and the far-away scene. The road surface in the scene is detected using the method proposed in [17]. Next, segmentation of SOI into set of obstacles is performed on the u-disparity image. As illustrated in Fig.1(d), u-disparity image provides a bird-eye's view of the scene and the peak regions in the u-disparity image correspond to potential obstacles. These peak regions can be identified using the adaptive connected component labeling technique. Each cluster identified in u-disparity image corresponds to one obstacle in the scene. Fig.1(e) shows the finally detected obstacles.

B. *Appearance Model Setup*

Appearance model refers to the representation of object based on specific features. The appearance of obstacle in urban scene is easily affected by many factors like inconsistent illumination, partial occlusion, scale and view point change and so on. Hence, the design of a good appearance model needs to take into account these factors.

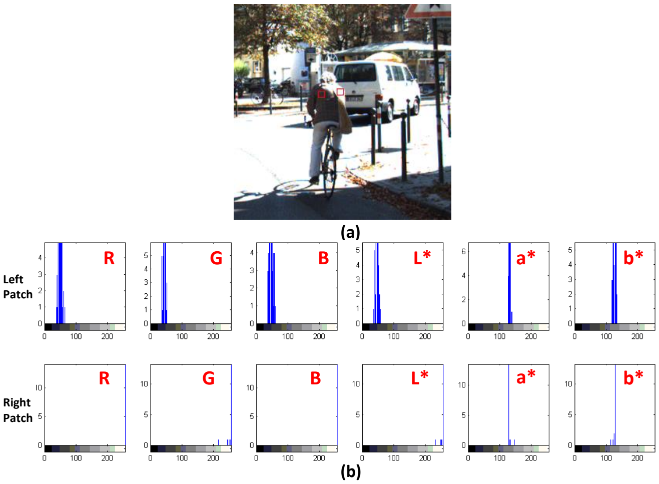


Fig. 2: Color histogram distributions for the same object in *RGB* and *L*a*b** color spaces.

In this work, color histogram is employed to describe the appearance of obstacles due to its simplicity and its high tolerance to scale and view angle change and partial occlusion. A number of strategies to further increase the distinctiveness and reduce the computational complexity for constructing the object model are also adopted, which will be described in detail in the following sub-sections.

*1) Utilizing L*a*b* Color Space to Increase Robustness to Illumination Change:* There are many ways to encode color. Compared to the popular *RGB* color model, *L*a*b** color is closer to human visual perception. In particular, the *L** component closely matches human perception of brightness. As shown in Fig.2(a), two patches with the same size and texture but subjected to different illumination are sampled from the back of the bicyclist. The histogram of the *RGB* and *L*a*b** color components for each patch is depicted in Fig.2(b). Fig.2(b) clearly illustrates that when illumination changes, the corresponding histogram change drastically for all of the three components of *RGB* color. On the other hand, the histograms for *a** and *b** component

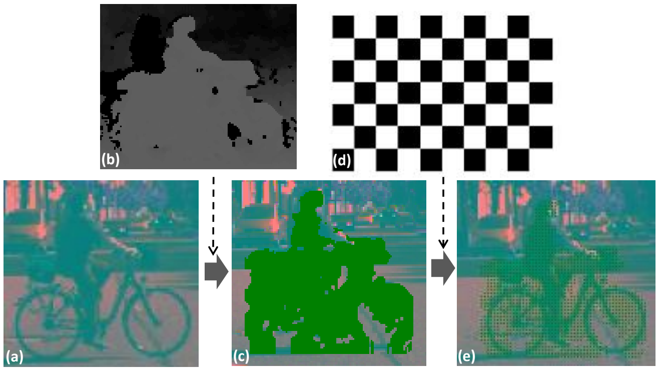


Fig. 3: Illustration of appearance model setup: (a) a *L*a*b* patch corresponding to one obstacle; (b) the corresponding disparity map; (c) background pixels are excluded; (d) a chessboard pattern; (e) only the pixels that don't belong to background and are not masked by the chessboard pattern will contribute to the final histogram construction.

are stable, and only $L*$ component is affected. This means that the *RGB* color space is sensitive to illumination change while the $a*$ and $b*$ components of $L*a*b*$ are insensitive. This motivates us to construct the color histogram in $L*a*b*$ color space. The number of bins for $L*$ component is only half of those required for $a*$ and $b*$ components. By doing this, the interference from illumination change is notably mitigated and robustness to inconsistent illumination are therefore increased.

2) Excluding Background Information to Increase Distinctiveness of Appearance Model: When building the histogram for a kernel based model, the interference from the background is another big concern. The inclusion of background pixels will result in inconsistency in the histograms of the same object when the background varies across frames.

In order to overcome this problem, the depth information are exploited to exclude the background pixels when constructing histograms. This is possible as generally, obstacles and background are associated with different depth value. In addition, the corresponding depth range of the obstacles are made available during obstacle detection as discussed in section III-A. Therefore, as illustrated in Fig.3(c), instead of considering all the pixels inside the bounding box, only pixels within the bounding box whose depth are in the range of the corresponding obstacle will contribute to the generation of color histogram for the obstacle. This strategy can effectively enhance the distinctiveness of the obstacle's appearance model.

3) Reducing Computational Complexity using Sparse Sampling Technique: The authors in [18] find that the sparse census transform configuration presents low computational complexity without compromising on correlation accuracy. Inspired by this idea, a chessboard pattern sampling technique when constructing histogram is adopted. The idea of sparse sampling technique is illustrated in Fig.3(d) and (e). Only the pixels that don't belong to background and are not masked by the chessboard pattern will contribute to the final histogram construction.

4) Similarity Measure for Data Association: The popular histogram intersection distance to is adopted to measure the similarity between two obstacles. Additionally, the similarity between two obstacles is weighted by their distance. The final similarity measure is shown in (2).

$$H(\mathbf{p},\mathbf{q}) = \sum_{i=1}^{n} min(p_i, q_i) \qquad (1)$$

$$C(\mathbf{p},\mathbf{q}) = \begin{cases} H(\mathbf{p},\mathbf{q}), & \Delta < \tau \\ 0, & \Delta > \tau \end{cases} \qquad (2)$$

where $\mathbf{p}$ and $\mathbf{q}$ are two normalized $L*a*b*$ color histograms for the two obstacles. $H(\mathbf{p},\mathbf{q})$ is the histogram intersection distance between $\mathbf{p}$ and $\mathbf{q}$. Δ refers to the distance between the two obstacles. τ is a predefined threshold.

C. Online Multi-Object Tracking Framework

Given a set of tracks $T = \{t_i\}$ identified from earlier frames and a set of detections $D = \{d_j\}$ in current frame, the whole tracking framework is presented as follows.

Step 1 - Similarity Computation: Compute the similarity matrix $C = \{c_{ij}\}$ between the tracks T and detections D using the metric defined in Eq. (2). c_{ij} refers to the similarity value between track t_i and detection d_j.

Step 2 - Tracks Assignment: Assign the detections D to the tracks T by solving a bipartite matching problem with the Hungarian method.

Step 3 - State Management: The total states for tracks can be: *stable, new, lost*. Through the maintenance of these three states, the context of the scene is well understood.

After *Step 2*, there are three types of assignment: tracks $T_1 = \{t_i^1\}$ are assigned with detections $D_1 = \{d_j^1\}$, unassigned tracks $T_2 = \{t_i^2\}$, and unassigned detections $D_2 = \{d_j^2\}$.

1) For each track t_i^1 in T_1, its state is updated as *stable*.

2) For each track t_i^2 in T_2, the following two cases are checked in the order listed: t_i^2 is merged with other track; and t_i^2 is lost.

There are several reasons as to why a track is unable to find its correspondence in current frame. Firstly, the track can be merged with other track in current frame due to close proximity or the inaccuracy in obstacle detection. Secondly, the corresponding object physically disappears in current frame. Measures should be taken to differentiate these two cases. An example for the first case is given in Fig.4. Fig.4(a) shows the detection and tracking results for frame 0085, where the bicyclist with id 1 and the vehicle with id 240 are separated and treated as individual tracks. When the new frame 0086 comes, the road surface is texture-less and the corresponding disparity map as shown in Fig.4 is inaccurate. This causes the bicyclist and the vehicle to be detected as one entity as shown in Fig.4(b). As illustrated in Fig.4(c), if no countermeasure is taken, the bicyclist with track id 1 gets unassigned and the vehicle with track id 240 will be updated wrongly with an new object as a result of merging the bicyclist and vehicle.

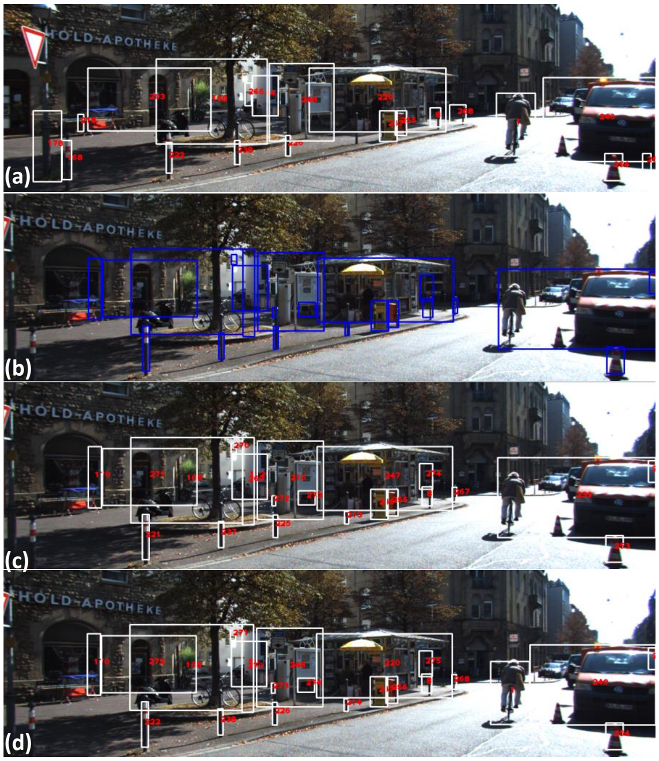


Fig. 4: The obstacle detection and data association modules are integrated to form an online multi-object tracking system in a robust way. For the detailed description of this figure, please refer to the text in Section III-C.

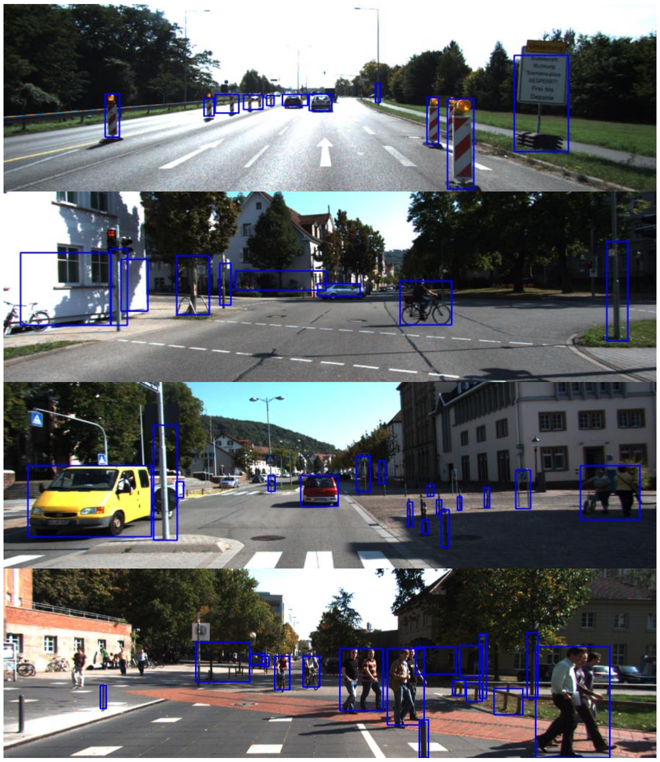


Fig. 5: The proposed obstacle detection method is capable of detecting various obstacles including pedestrians, vehicles, traffic poles, barriers etc.

In order to check whether t_i^2 falls under the first case mentioned above and to perform the correction if it happens, we find the corresponding detection d_j where the similarity value between t_i^2 and d_j is highest. We then denote the corresponding track that d_j is assigned to as t_m^1. If $c(t_i^2, d_j)$ and $c(t_m^1, d_j)$ are similar and are in close proximity, the objects corresponding to t_i^2 and t_m^1 are deemed to have merged. At this time, t_i^2 is assigned with a detection d_{new}, which is the predicted position of t_i^2 in current frame. d_j is corrected by excluding the part that corresponds to d_{new}. t_i^2 is updated with state *stable* and added to T_1. d_{new} is added to D_1. As illustrated in Fig.4(d), with the proposed correction strategy, the bicyclist with track id 1 and the vehicle with track id 240 are correctly tracked. If the first case doesn't happen, t_i^2 belongs to the second case where the corresponding obstacle disappears in current frame. For this case, the state of t_i^2 is labeled as *lost*.

3) For each detection d_j^2 in D_2, create a track with state *new*.

Step 4 - Appearance Model Update: Create a new Kalman filter for each of the tracks with *new* state. Update the appearance model for each of the tracks with *stable* state using the corresponding detection via Kalman filter. For every *new* or *stable* track, predict its position in the next frame via Kalman filter. Delete the tracks which have been *lost* for n frames.

IV. Evaluation

We have chosen the well-known *KITTI* tracking benchmark [19] to evaluate the proposed algorithm. The benchmark consists of 21 training sequences and 29 test sequences, which cover various challenging road scenarios. The semi-global matching algorithm is utilized to generate the required disparity map [20].

A. Accuracy Evaluation

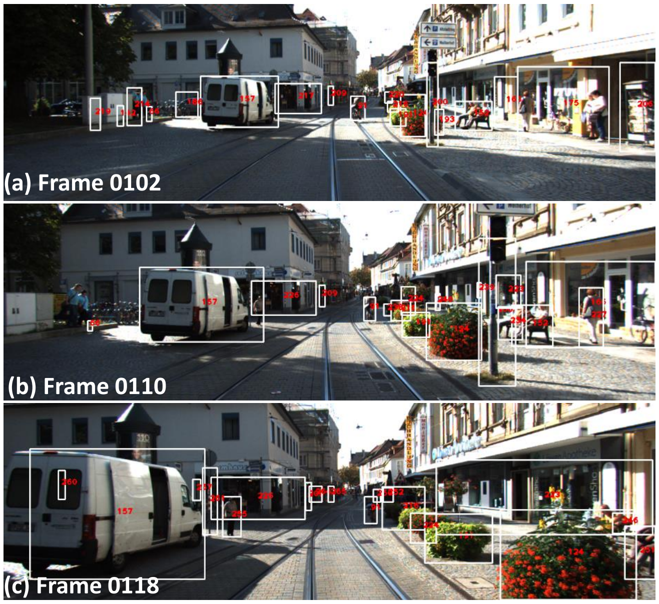
By exploiting the geometrical topology of the scene, the proposed method is able to detect obstacles of various types in diverse traffic scenarios. This is exemplified in Fig.5. As can be observed, the detected obstacles include not only vehicles and pedestrians but also traffic lights, sign posts, traffic barriers, etc.

A comprehensive qualitative evaluation of the proposed multi-obstacles tracking system in diverse challenging realistic environments is also conducted. Fig.6 shows a busy road scenario. It is evident that not only common obstacles like pedestrian, vehicle, bicyclist but also unexpected ones like traffic light and flowerbed are simultaneously tracked. The obstacles can be standing still or subjected to motion. In Fig.7, a train appears in the scene. The scale of the train varies drastically over the frames. However, the proposed algorithm is still able to robustly track it. The proposed algorithm is also insensitive to inconsistent illumination. Fig.8 illustrates a scenario where the illumination changes abruptly. Although subjected to different illumination conditions, the cyclist is continuously tracked over frames. The dataset also contains scenarios where obstacles are occluded by others. For example, in Fig.9, the man in grey shirt with id 41 walks towards a group of two people, gets merged and occluded by them, and finally appears again. The man is correctly tracked by the proposed method throughout the entire course. Therefore, the qualitative results confirm that the proposed algorithm is capable of tracking obstacles in challenging conditions.

Finally, an extensively quantitative evaluation of the proposed tracking algorithm is conducted based on the evaluation criteria proposed in [21], [22]. In addition, the object tracker proposed in [13][14] has been chosen as the baseline algorithm. The baseline algorithm is designed to detect vehicles only. In order to exclude the effect of object detection and evaluate the ability of data association for both the proposed and baseline tracking algorithm, we feed both the proposed and baseline tracking algorithms with the same detections as inputs, which correspond to the ground truth object bounding boxes with class *car* and *pedestrian* in the KITTI tracking dataset [19]. The corresponding evaluation results are shown in Table I. It is evident that the proposed tracking algorithm significantly outperforms the baseline algorithm.

TABLE I: Tracking accuracy evalution

Method	MOTA(%)	MOTP(%)	MT(%)	ML(%)	FM	IDS
Baseline	88.10	94.62	65.74	4.56	808	512
Proposed	95.11	98.24	99.78	0	736	731

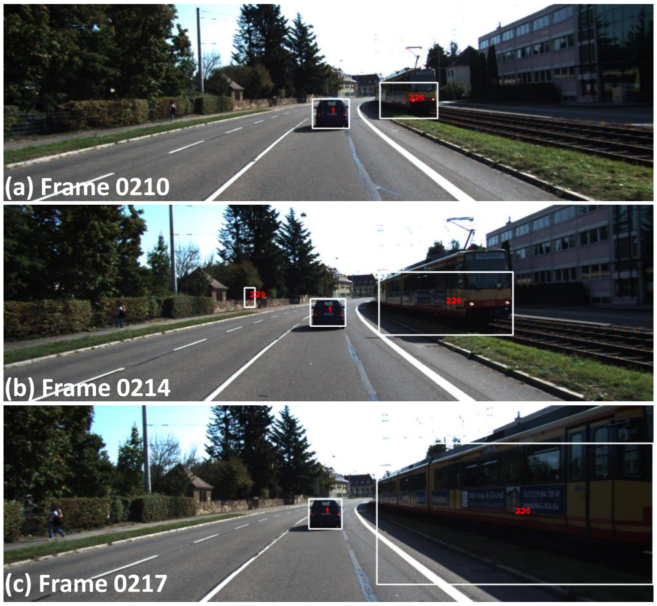


(a) Frame 0102

(b) Frame 0110

(c) Frame 0118

Fig. 6: Busy Road Scenario: Not only common obstacles like pedestrian, vehicle and bicyclist but also unexpected ones like traffic light and flowerbed are simultaneously tracked. The obstacles can be standing still or subjected to motion.

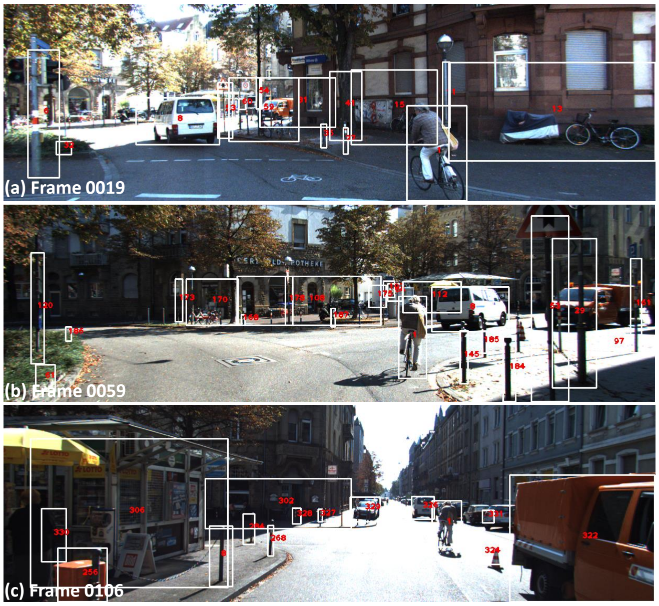


(a) Frame 0210

(b) Frame 0214

(c) Frame 0217

Fig. 7: Scale Change: The scale of the train varies drastically over the frames. But the train is tracked robustly.



(a) Frame 0019

(b) Frame 0059

(c) Frame 0106

Fig. 8: Inconsistent Illumination: Although subjected to different illumination conditions, the cyclist is continuously tracked over frames.

B. Runtime Performance Evaluation

A comprehensive runtime performance evaluation is conducted in this section. Given the input color image and the corresponding disparity map, the proposed algorithm yields low computational complexity due to the following strategies. Firstly, obstacles are detected efficiently in the *u-v-disparity* image space. SOI is generated to reduce the search space of obstacle. The algorithm we adopt to detect the road surface [17] is lightweight. Secondly, the generation of histogram is fast due to the sparse sampling technique.

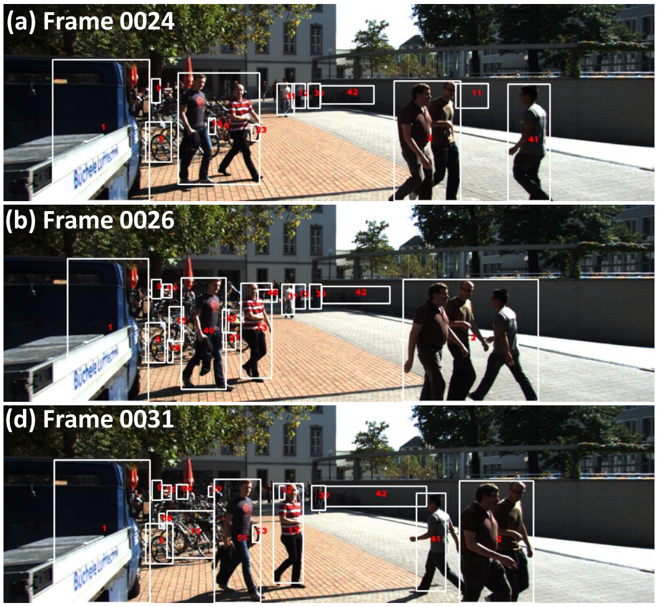


Fig. 9: Occlusion: the man in grey shirt with id 41 walks towards others, is merged with them and occluded by them, and finally appears again. The whole course is well understood.

Therefore, as illustrated in Table II, the proposed obstacle detection and tracking algorithm only needs 0.046 second/frame for detection and 0.003 second/frame for tracking.

TABLE II: Runtime evaluation of the proposed obstacle detection and tracking algorithms.

Method	Detection	Tracking	Total	Platform
Proposed	0.046s	0.003s	0.049s	CPU@3.5GHZ

V. CONCLUSIONS

Using the well-known benchmark, we have demonstrated that the proposed obstacle detection and tracking system is capable of detecting and tracking diverse obstacles in various challenging environments. This is achieved by detecting obstacles in every frame by exploiting the geometrical structure in the u-v-disparity image space and associating obstacles across frames with the aid of an distinctive object appearance model. A number of strategies to reduce the computational complexity for obstacle detection and appearance model setup and increase the robustness of the appearance model are proposed. In addition, the proposed obstacle detection and data association modules are integrated to form an online multi-object tracking framework in a robust way. Evaluations using the KITTI tracking benchmark confirm that the proposed obstacle detection and tracking method outperforms the baseline algorithm in terms of tracking accuracy. In addition, the proposed method lends well for real-time realization with 20 fps.

REFERENCES

[1] N. Bernini, M. Bertozzi, L. Castangia, M. Patander, and M. Sabbatelli, "Real-time obstacle detection using stereo vision for autonomous ground vehicles: A survey," in *Intelligent Transportation Systems (ITSC), 2014 IEEE 17th International Conference on*. IEEE, 2014, pp. 873–878.

[2] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *Acm computing surveys (CSUR)*, vol. 38, no. 4, p. 13, 2006.

[3] K. Nummiaro, E. Koller-Meier, and L. Van Gool, "Color features for tracking non-rigid objects," *ACTA Automatica Sinica*, vol. 29, no. 3, pp. 345–355, 2003.

[4] A. Mukhtar, L. Xia, and T. B. Tang, "Vehicle detection techniques for collision avoidance systems: A review," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 16, no. 5, pp. 2318–2338, 2015.

[5] P. Dollar, C. Wojek, B. Schiele, and P. Perona, "Pedestrian detection: An evaluation of the state of the art," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 34, no. 4, pp. 743–761, 2012.

[6] D. Comaniciu, V. Ramesh, and P. Meer, "Kernel-based object tracking," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 25, no. 5, pp. 564–577, 2003.

[7] A. W. Smeulders, D. M. Chu, R. Cucchiara, S. Calderara, A. Dehghan, and M. Shah, "Visual tracking: An experimental survey," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 36, no. 7, pp. 1442–1468, 2014.

[8] A. Adam, E. Rivlin, and I. Shimshoni, "Robust fragments-based tracking using the integral histogram," in *Computer vision and pattern recognition, 2006 IEEE Computer Society Conference on*, vol. 1. IEEE, 2006, pp. 798–805.

[9] U. Franke, C. Rabe, H. Badino, and S. Gehrig, "6d-vision: Fusion of stereo and motion for robust environment perception," in *Pattern Recognition*. Springer, 2005, pp. 216–223.

[10] C. Rabe, U. Franke, and S. Gehrig, "Fast detection of moving objects in complex scenarios," in *Intelligent Vehicles Symposium, 2007 IEEE*. IEEE, 2007, pp. 398–403.

[11] R. Danescu, C. Pantilie, F. Oniga, and S. Nedevschi, "Particle grid tracking system stereovision based obstacle perception in driving environments," *Intelligent Transportation Systems Magazine, IEEE*, vol. 4, no. 1, pp. 6–20, 2012.

[12] Z. Wu, J. Zhang, and M. Betke, "Online motion agreement tracking." in *BMVC*, 2013.

[13] H. Zhang, A. Geiger, and R. Urtasun, "Understanding high-level semantics by modeling traffic patterns," in *Proceedings of the IEEE International Conference on Computer Vision*, 2013, pp. 3056–3063.

[14] A. Geiger, M. Lauer, C. Wojek, C. Stiller, and R. Urtasun, "3d traffic scene understanding from movable platforms," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 36, no. 5, pp. 1012–1025, 2014.

[15] X. Li, W. Hu, C. Shen, Z. Zhang, A. Dick, and A. V. D. Hengel, "A survey of appearance models in visual object tracking," *ACM transactions on Intelligent Systems and Technology (TIST)*, vol. 4, no. 4, p. 58, 2013.

[16] S. Salti, A. Cavallaro, and L. D. Stefano, "Adaptive appearance modeling for video tracking: Survey and evaluation," *Image Processing, IEEE Transactions on*, vol. 21, no. 10, pp. 4334–4348, 2012.

[17] M. Wu, S.-K. Lam, and T. Srikanthan, "Nonparametric technique based high-speed road surface detection," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 16, no. 2, pp. 874–884, 2015.

[18] W. S. Fife and J. K. Archibald, "Improved census transforms for resource-optimized stereo vision," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 23, no. 1, pp. 60–73, 2013.

[19] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012, pp. 3354–3361.

[20] H. Hirschmüller, "Stereo processing by semiglobal matching and mutual information," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. 30, no. 2, pp. 328–341, 2008.

[21] K. Bernardin and R. Stiefelhagen, "Evaluating multiple object tracking performance: the clear mot metrics," *Journal on Image and Video Processing*, vol. 2008, p. 1, 2008.

[22] Y. Li, C. Huang, and R. Nevatia, "Learning to associate: Hybrid-boosted multi-target tracker for crowded scene," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 2953–2960.