Moving Objects Detection and Classification Based on Trajectories of LRF Scan Data on a Grid Map

Taketoshi Mori, Takahiro Sato, Hiroshi Noguchi, Masamichi Shimosaka, Rui Fukui and Tomomasa Sato

Abstract— Laser based environment recognition technologies have been developed recently. Especially moving objects detection and classification by laser scanners mounted on a mobility is required for mobile robots and autonomous cars. In this paper, we propose a moving objects detection and classification method based on grid trajectories acquired from sequential laser scan data. Grid trajectories are obtained by voting sequential laser scan points on a grid map, and these trajectories not only work for a correct scan segmentation, but also represent the size and the speed of moving objects. We classify a moving object into either a person, a group of people, a bike, a car based on its grid trajectory. In our experiments, our mobility mounted laser scanners acquired scan data in the university campus, and the experimental results illustrate the effectiveness of the proposed method in outdoor environments.

I. INTRODUCTION

Environmental recognition and understanding technologies are important for intelligent vehicles such as mobile robots and autonomous cars. Especially, a detection of moving objects (pedestrians, cars, bicycles, etc. ...) is available to realize a safer path planning, an informative support for the driver and so on. In environments that don't have clear discrimination in traffic regions between pedestrians and cars, many moving objects exist with different states, such as speed, direction, and size. Therefore, moving objects classification is effective for motion predictions of each object and an intuitive informative support to the driver.

Many research efforts have shown the possibility of detecting objects in front of a mobility. Approaches using a video camera [1] can utilize much information about the object such as its color and shape, but robust methods against outdoor lighting condition haven't been practical yet. Laser range finder(LRF) can detect objects in an extensive area because wide angle and long range data is required. Further, LRF is robust sensor against outdoor lighting condition.

A simple moving objects detection method is a way of using an occupancy grid map [2]. In an occupancy grid map, an environment is represented by many grids, each grid has the occupancy which indicates the probability of the object existence. This method is utilized in mobile robot localization and map construction. In the Vu's method [3], scan points appeared on a free grid are considered as points taken from moving objects. This approach is effective in

Taketoshi Mori, Takahiro Sato, Hiroshi Noguchi, Masamichi Shimosaka, Rui Fukui and Tomomasa Sato are with Graduate School of Information Science and Technology, The University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo, Japan {tmori, takahiro, noguchi, simosaka, fukui}@ics.t.u-tokyo.ac.jp, tomomasasato@jcom.home.ne.jp

an environment surrounded buildings, but it is difficult to acquire enough free spaces in an environment such as a park which has many open space.

In other moving objects detection methods, LRF scan points are processed as a segment: a set of continuous points. Many approaches have used segments because each object is often represented by one segment [4]–[6]. In their proposed methods, the segment is considered as a moving object when its enough movement is observed by tracking it a certain time. But in general, scan segments repeats combining and splitting through the observation, and disappears temporary. Therefore, it is difficult to track scan segments robustly.

Such noises of scan segments affect moving objects classification. In moving objects classification, the size, the shape and the speed of each segment are used as features of classification. In the Zhao's method [7], moving objects segments are classified to four groups: a pedestrian, a group of pedestrians, a bicycle, and a car. They focused on scan segments are divided into four groups by the size, the moving speed and the number of axis composing the segment. But in their paper, they presented as a problematic case that split car's scan segments are misclassified to other object if they are processed as separate objects. To avoid such segments splitting, merging segments by distance threshold between scan segments is in general, but it is difficult to decide the threshold appropriately.

As another scan segmentation method for each object, a model based approach is proposed. In this approach, a target segment contour shape is assumed by a rectangle. Streller et al. [8] focused on a car's scan contour was modeled a rectangle, so they fit a rectangle to scan segments and merge the segments on the same rectangle. By this way, they realized robust scan segmentation of cars. However, a
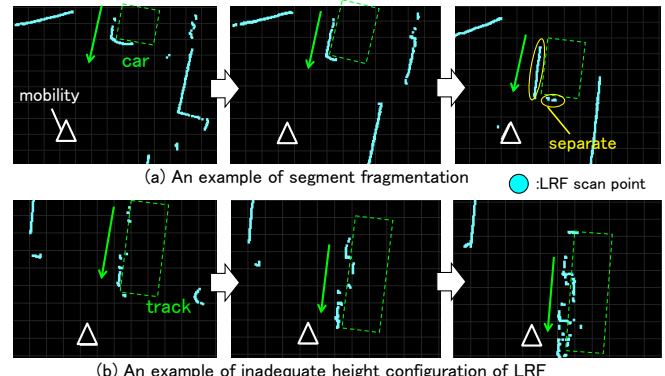

(a) An example of segment fragmentation

● :LRF scan point

(b) An example of inadequate height configuration of LRF

Fig. 1.   Examples of splitting LRF scan segments

scan contour of a car is not always modeled by a rectangle. A contour shape of a car varies in a kind of cars and the height where it is scanned. Therefore, rectangle fitting scan segmentation doesn't always works correctly.

In the past scan segmentation approaches, every method utilizes only a single frame scan data. Our scan segmentation method utilizes sequential scan data. In the proposed method, we acquire moving objects trajectories by voting sequential scan data on a grid map, and utilize these trajectories for scan segmentation. In addition, moving objects are detected and classified based on these trajectories because these indicate the target moving speed, size (length and width). In our research's target environment, cars and motorcycles don't run too fast because they share roads with another moving objects such as a pedestrian and a bicycle. Therefore, in the proposed method, we regard following four classes as target moving objects: a person, a group of people, a bike, a car. A group of people represents two or three pedestrians walking concurrently and a bike contains both a bicycle and a motorcycle.

This paper is organized as follows. We mention our scan segmentation method using sequential scan data on a grid map in section II. In section III, we define features of a grid trajectory and utilize them for machine learning based detection and classification of moving objects. We show experimental results of the proposed method in section IV. Finally, conclusion is discussed in section V.

II. A SCAN SEGMENTATION METHOD BASED ON GRID TRAJECTORIES OF MOVING OBJECTS

In this section, we describe about a scan segmentation method using grid trajectories of moving objects. Compared to the past segmentation methods, robust segmentation can be achieved by using a grid map and sequential LRF scan data. First, we describe details about acquisition of grid trajectories for scan segmentation. Then we mention about a scan segmentation method based on the grid trajectory.

A. Acquisition of a Grid Trajectory

Fig. 2 illustrates the process of obtaining a grid trajectory. In our method, we utilize past N frames LRF scan and odometry data. In a current frame, scan points of the past

frame are voted on a local grid map based on a mobility pose obtained from an odometry estimation. The origin of the grid map is the mobility pose N frames before and each grid occupancy is a binary value: occupied or free. We refer to parameters setting of N and the grid size on IV. Then, by clustering occupied grids adjacent each other, we get grid trajectory clusters $C_{\text{traj}} = \{C_{\text{traj},1}...C_{\text{traj},N_{\text{traj}}}\}$ illustrated by white grids in Fig. 2(a). In our research's target environments which don't have clear discrimination in traffic regions between pedestrians and cars, cars and motorcycles don't run too fast. Therefore, LRF scan time is short enough to the relative speed between own mobility and each moving object, so C_{traj} represent N frames grid trajectories of moving objects. Similarly, new frame scan points are voted on the same grid map. By clustering only the grids where new scan points are voted, we obtain new grid clusters $C_{\text{new}} = \{C_{\text{new},1}...C_{\text{new},N_{\text{new}}}\}$ illustrated by green grids in Fig. 2(a).

Then, C_{traj} and C_{new} are taken correspondence by evaluating overlaps and adjacent degrees between grid clusters. This evaluation score is calculated by the equation (1). Grid clusters which don't have any overlap or adjacent grids aren't taken correspondence. $n_{ij}^{\text{same}}, n_{ij}^{\text{neighbor}}$ represents respectively the number of which a grid or its adjacent of $C_{\text{new},i}$ is identical to one of $C_{\text{traj},j}$.

$$\text{score}(C_{\text{new},i}, C_{\text{traj},j}) = 2n_{ij}^{\text{same}} + n_{ij}^{\text{neighbor}} \qquad (1)$$

B. Scan Segmentation Based on a Grid Trajectory

Large size moving objects such as cars are often misclassified to other objects due to split of segments obtained from these objects. Segments merging methods in a single frame such as a distance threshold and a rectangle fitting doesn't work correctly in general. So sequential information about scan data is effective for scan segmentation and we use grid trajectories as the sequential information. When C_{new} and C_{traj} are obtained as shown in Fig. 2(b), C_{new} near the same C_{traj} is more likely acquired from the same object, even if C_{new} are distributed separately. Therefore, C_{new} taken correspondence to the same C_{traj} are considered as acquired from the same object and re-clustered. By this way, split scan data obtained from the same objects clustered correctly.

III. MOVING OBJECTS DETECTION AND CLASSIFICATION BASED ON A GRID TRAJECTORY

In this section, we mention details about the moving objects detection and classification method. First, we define features of a grid trajectory, and these features are utilized in moving objects detection and classification mentioned later. Second, we utilize AdaBoost framework in moving objects detection, so we show the AdaBoost algorithm and our implementation. Last, we describe about the moving objects classification method using Naive Bayes classifier.

A. Features of a Grid Trajectory

In Fig. 3, we show some examples of grid trajectories acquired from target moving objects: a person, a group of people, a bike, a car. Compared to the grid trajectory
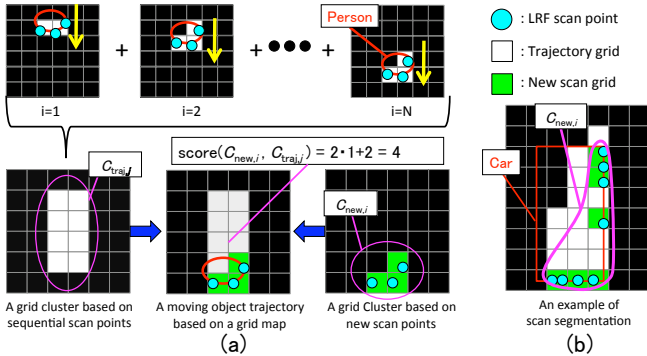

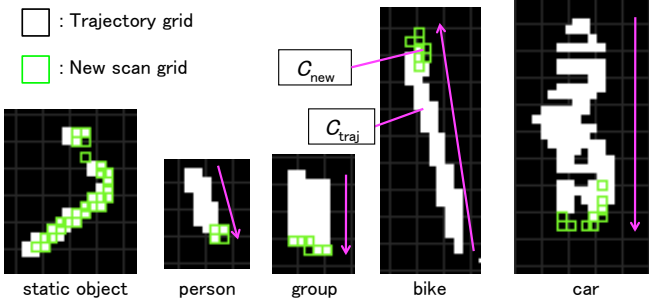Fig. 2. (a) Acquisition of a grid trajectory and (b) scan segmentation based on a grid trajectory
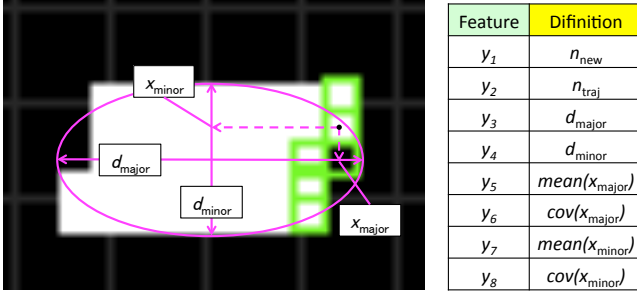
Fig. 3. Grid trajectory examples of target moving objects

Fig. 4. Features definition of a grid trajectory

Feature	Difinition
y_1	n_{new}
y_2	n_{traj}
y_3	d_{major}
y_4	d_{minor}
y_5	$mean(x_{major})$
y_6	$cov(x_{major})$
y_7	$mean(x_{minor})$
y_8	$cov(x_{minor})$

- Input : set of example $(\mathbf{y}_1,z_1),...,(\mathbf{y}_{N_{train}},z_{N_{train}})$
- Let m be the number of negatives examples and l be the number of positive exmaples. Initialize weights $w_{1,n} = \frac{1}{2m}, \frac{1}{2l}$ depending on the value of z_n
- For $t = 1,...,T$:
  1) Normailze the weights $w_{t,n}$ so that $\sum_{n=1}^{N_{train}} w_{t,n} = 1$.
  2) For each feature f_i, train a weak classifier h_i.
  3) The error ε_i of a classifier h_i is determined with respect to the weights $w_{t,1},...,w_{t,N_{train}}$:
  $$\varepsilon_i = \sum_n^{N_{train}} w_{t,n}|h_i(\mathbf{y}_n) - z_n|.$$
  4) Choose the classifier h_i with the lowest error ε_i and set $(h_t,\varepsilon_t) = (h_i,\varepsilon_i)$.
  5) Update the weights $w_{t+1,n} = w_{t,n}\beta_t^{1-e_n}$ where $\beta_t = \frac{\varepsilon_t}{1-\varepsilon_t}$ and $e_n = 0$, if example $\mathbf{y}_n$ is classified correctly by h_t and 1, otherwise.
- The final strong classifieris given by :
  $$h(\mathbf{y}) = \begin{cases} 1 & \text{if } \sum_{t=1}^T \log\frac{1}{\beta_t}h_t(\mathbf{y}) \geq \frac{1}{2}\sum_{t=1}^T \log\frac{1}{\beta_t} \\ 0 & \text{otherwise.} \end{cases}$$

as a face detector [9], and construct strong moving objects detector by choosing effective features from $\mathbf{y}$

The original Adaboost algorithm is a supervised learning algorithm designed to construct a strong binary classifier. The input of the algorithm is a set of training examples $(\mathbf{y}_n, z_n), n = 1, ..., N_{train}$, where each $\mathbf{y}_n$ is an example and z_n is an boolean value indicating whether $\mathbf{y}_n$ is a positive or negative example. AdaBoost boosts the classification performance by combining a collection of weak classifiers. Each weak classifier is given as a function $h_i(\mathbf{y})$ which returns boolean value. The output is 1, if $\mathbf{y}$ is classified as a positive example and 0 otherwise. The weak classifier only need to be slightly better than a random guess. To boost a weak classifier, it solves a sequence of learning problems. After each learning, the examples are re-weighted in order to increase the importance of those which were incorrectly classified by the previous weak classifier. The final strong classifier takes the form of perceptron. Large weights are assigned to good classification functions whereas poor functions have small weights.

In our implementation, a weak classifier has the form (2) similar to the one proposed by Mozos et al. [10].

$$h_i(\mathbf{y}) = \begin{cases} 1 & \text{if } p_i f_i(\mathbf{y}) < p_i \theta_i \\ 0 & \text{otherwise.} \end{cases} \quad (i = 1,...,8) \quad (2)$$

$f_i(\mathbf{y})$ is a function which returns the i-th feature of $\mathbf{y}$. θ_i is a threshold and p_i is either -1 or 1 and thus representing the direction of the inequality. The optimal values for θ_i and p_i are chosen by minimizing the number of misclassified training examples as shown (3). To achieve this, the algorithm considers all possible combinations of θ_i and p_i.

$$(p_i, \theta_i) = \operatorname*{argmin}_{(p_i,\theta_i)} \sum_{n=1}^{N_{train}} |h_i(\mathbf{y_n}) - z_n| \quad (3)$$

The resulting algorithm is given by Table I.

When a strong classifier is constructed by one training data set, features which chosen more than the other are $d_{major}, d_{minor}, \text{mean}(x_{major})$ and $\text{cov}(x_{major})$.

of a static object, trajectory grids of moving objects are distributed trailing to new scan grids. This is common features between moving objects. In addition, compared grid trajectories of four objects each other, the number of grids in C_{new} and C_{traj}, and the major/minor axis length of the approximated ellipse of a grid trajectory are characteristic. From these perspective, we use following features for moving objects detection and classification.

n_{new}, n_{traj} are the number of grids composing $C_{new,i}, C_{traj,j}$ respectively. d_{major}, d_{minor} represents the length of a major/minor axis of the approximated ellipse of $C_{traj,j}$. These features indicate the size and speed of moving objects. Additionally, when each new scan grid are projected on the major/minor axis of a grid trajectory, let projected values notate x_{major}, x_{minor} respectively. The origin of the axis is a centroid of $C_{traj,j}$ and we define the moving direction of an object as positive. Mean and covariance of x_{major}, x_{minor} are used additional features. These represents characteristics of that trajectory grids of a moving object are distributed trailing to new scan grids. In easy representation, let these eight features notate $\mathbf{y} = \{y_i\}(i = 1, ..., 8)$

B. Moving Objects Detection

Moving objects detection is identical to an binary classification problem whether each grid cluster is a moving object or not. However the size and speed of target moving objects differ respectively, so it is difficult to decide threshold experimentally to detect four classes commonly. In the proposed method, decisions of features' thresholds are performed automatically with a set of training data obtained in advance. We use AdaBoost framework utilized in such

$d_{\mathrm{major}}, d_{\mathrm{minor}}$ represents the similarity to an ellipse shape of grid trajectory. $\mathrm{Mean}(x_{\mathrm{major}})$ and $\mathrm{cov}(x_{\mathrm{major}})$ represents that trajectory grids of a moving object are distributed trailing to new scan grids. These four features are considered as characteristics common to target four moving objects.

In each frame, static objects such as tree, bush and polls are often detected as moving objects because of LRF scan noises and disturbance of the mobility's attitude by road bump. These miss-detection happens momentarily, so we suppress these using sequential detection results. In details, each grid cluster preserves its N_{detect} frames history which it was detected as a moving object by tracking each cluster. Tracking a grid cluster is performed based on a correspondence of C_{traj} between frames by equation (1). If the cluster was detected as a moving object more than threshold frames in the past, it is detected as a moving object in a current frame.

C. Moving Objects Classification

Moving objects classification is effective for a prediction of a object's moving direction and intuitive informative supports to the driver. In each frame, given a feature set $\mathbf{y}$, the objective is to classify the object into a certain class c_i, where c_i might be either a person, a person group, a bike, or a car. Let $c_{t,i}$ notate a class of a moving object at time t, the problem is formulated as follows.

$$c_{t,i} = \underset{i}{\mathrm{argmax}}\, p(c_i|y_1, ...y_8) \quad (i = 1, ...4) \qquad (4)$$

According to Baysian rule, (4) can be parsed to

$$c_{t,i} = \underset{i}{\mathrm{argmax}}\, p(c_i) \prod_{j=1}^{8} p(y_j|c_i) \quad (i = 1, ...4) \qquad (5)$$

$p(c_i)$ is a prior of each object. In our implementation, $p(c_i)$ is a equivalent for each class. $p(y_j|c_i)$ is a likelihood function obtained from a training data set previously. We assume that the function is a gaussian mixture model.

Fig. 5 shows likelihood functions acquired from a certain training data. The number of each class contained in the training data was as bellow: a person 12, a group of people 4, a bike 12, a car 10. Fig. 5 illustrates that each class has the different likelihood distribution of its feature, so these features are likely to work effectively in classification of moving objects.

However, misclassifications happen momentarily because of LRF scan noises and speed changes of objects. Similar to moving objects detection, these temporary misclassifications are suppressed by the past sequential classification results. In details, each grid cluster preserves its N_{classify} frames history which class it was classified by tracking the cluster. In the past N_{classify} frames history, the label classified the most is an output result in a current frame.

In addition, there is an impossible transition of a classification label between each class. For example, a moving object considered as a car for the past long time should be classified to a car even if features similar to other objects are observed. By utilizing knowledge like this, the classifier
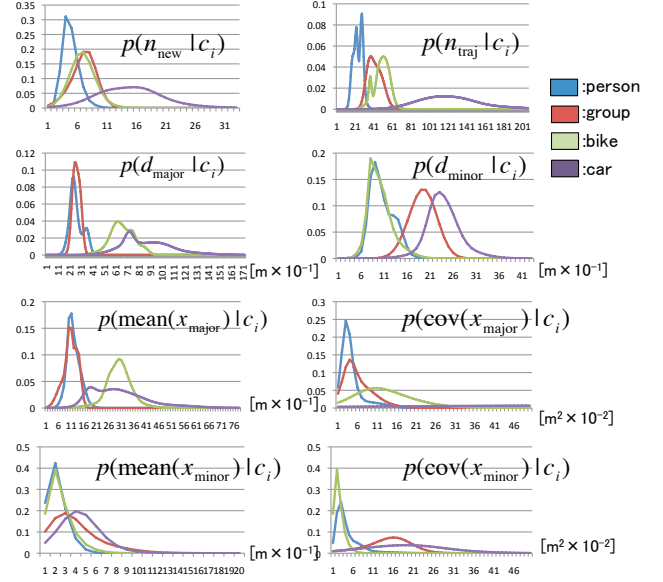

Fig. 5. Likelihood functions obtained from a training data set

TABLE II

TRANSITION MODEL BETWEEN TARGET FOUR CLASSES

		previous frame class			
		person	group	bike	car
current frame class	person	O	O	×	×
	group	O	O	×	×
	bike	O	O	O	×
	car	O	O	O	O

gets robust against scan noises and the change of the object's speed. When an object is classified c_{trans} more than N_{trans} in the past N frames, it was applied a transition model defined in Table II and classified by (6). $I_{c_{\mathrm{trans}}}$ is a set of indices based on the transition model of c_{trans}.

$$c_{t,i} = \underset{i \in I_{c_{\mathrm{trans}}}}{\mathrm{argmax}}\, p(c_i) \prod_{j=1}^{8} p(y_j|c_i) \qquad (6)$$

IV. EXPERIMENTAL RESULTS

A. Experimental Setup

The proposed method described above has been implemented and evaluated on real data acquired with Hokuyo UTM-30LX laser range finders mounted on a powerchair. UTM-30LX covers an angular area of $270°$ at a resolution of about $0.25°$ and measures distance of 30 meters with a nominal system error of ± 50 mm. 1081 data points are obtained at 40 fps. In our experiment, we used it at 20 fps considering the computation efficiency. The two LRFs were mounted at a height of about 50 cm covering all around the mobility shown in Fig. 6. The maximum translational velocity of the powerchair during data acquisition was 1.1
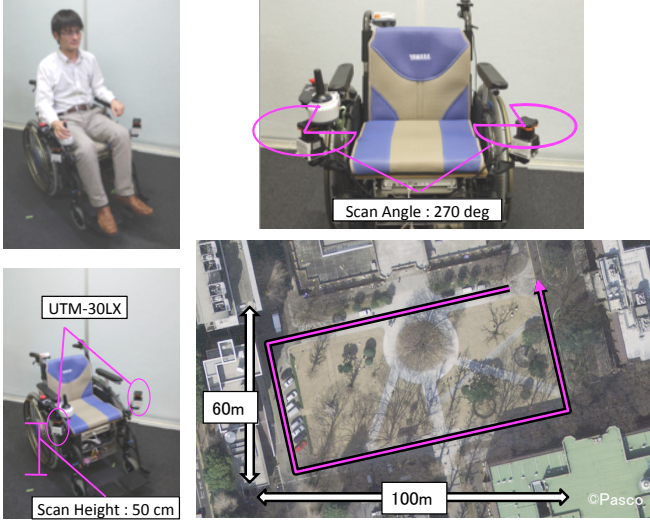
Fig. 6. Our platform mobility and the experimental environment

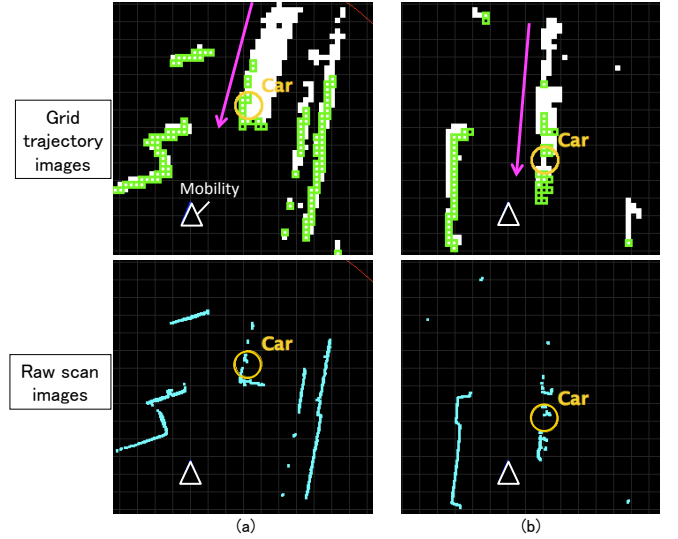|	Total	Total frame	Total detected frame	Correctly detected frame	Precision [%]	Recall [%]	F-Measure [%]
Person	16	2395	2783	2083	76	87	81
Group	2	497	689	478	69	96	81
Bike	6	646	666	613	92	94	93
Car	6	981	1036	890	86	91	88

Fig. 8. Examples of correct scan segmentation results

m/s. Experimental environment was inside the campus of the University of Tokyo shown in Fig. 6, whose characteristics are that many trees, bushes exist other than buildings.

In this experiment, we set the size of the grid map 40×40 m^2. In our preliminary experiment, the grid size should be set about person's waist width (30-60 cm) to decrease error scan segmentation and N should be set about a time length which some trajectory grids are observed from a person whose the velocity is the minimum of all four classes. Considering above, we set all parameters as below: the grid size 30 cm, N=30, N_{detect}=18, N_{classify}=30, N_{classify}=30, N_{trans}=25. Intel Core 2 Duo 2.8GHz PC was used for the computation.

The object's reference class label in the test data was acquired manually. In an area more than 20 m away from the mobility, it is difficult to measure the objects stably because of LRF scan noises. Further, in our method, moving objects must be kept observed a certain time to detect them. So we limited evaluated moving objects existing in an area within 15 m away from the mobility.

B. Detection and Classification Accuracy Results

Table III shows experimental results in the test data and we evaluated recall rate, precision rate, and F-measure defined as below.

$$\text{recall rate} = \frac{\text{correctly detected frames}}{\text{total frames}}$$

$$\text{precision rate} = \frac{\text{correctly detected frames}}{\text{total detected frames}}$$

$$\text{F-measure} = \frac{2 \times \text{recall} \times \text{precision}}{\text{recall} + \text{precision}}$$

Each frame was processed at about 30 fps, so the proposed method can work online. Fig. 7 shows examples of LRF raw scan points, grid trajectories, reference video camera images when objects were detected. Fig. 7(d) was the case that a person walking near a parking car was misclassified to a group of people. This is because trajectory grids are miss-clustering when a person walks near other objects as shown Fig. 7(d) middle row. Therefore, F-measure values of person and group were less than the other.

Fig. 8 shows cases considered that scan segmentation such as a distance threshold and rectangle fitting are difficult. Yellow circles in Fig. 8 represents the centroid of new grids clustered as one object. These objects were classified correctly based on the grid trajectory despite that their scan segments were distributed separately.

Further, Fig. 9 shows detected moving objects other than target four classes. Fig. 9(a) were two bicycles running concurrently and (b)(c) were pedestrians carrying a bicycle and a carrier cart respectively. Whereas each of them wasn't contained in the training data, they were detected as moving objects. Therefore, it indicates that the proposed method has generality as moving objects detector.

V. CONCLUSION

In this paper, we proposed a moving objects detection and classification method based on grid trajectories using LRFs mounted on a mobilty. Grid trajectories are obtained by voting sequential LRF scan points on a grid map, and these trajectories not only work for a correct scan segmentation,
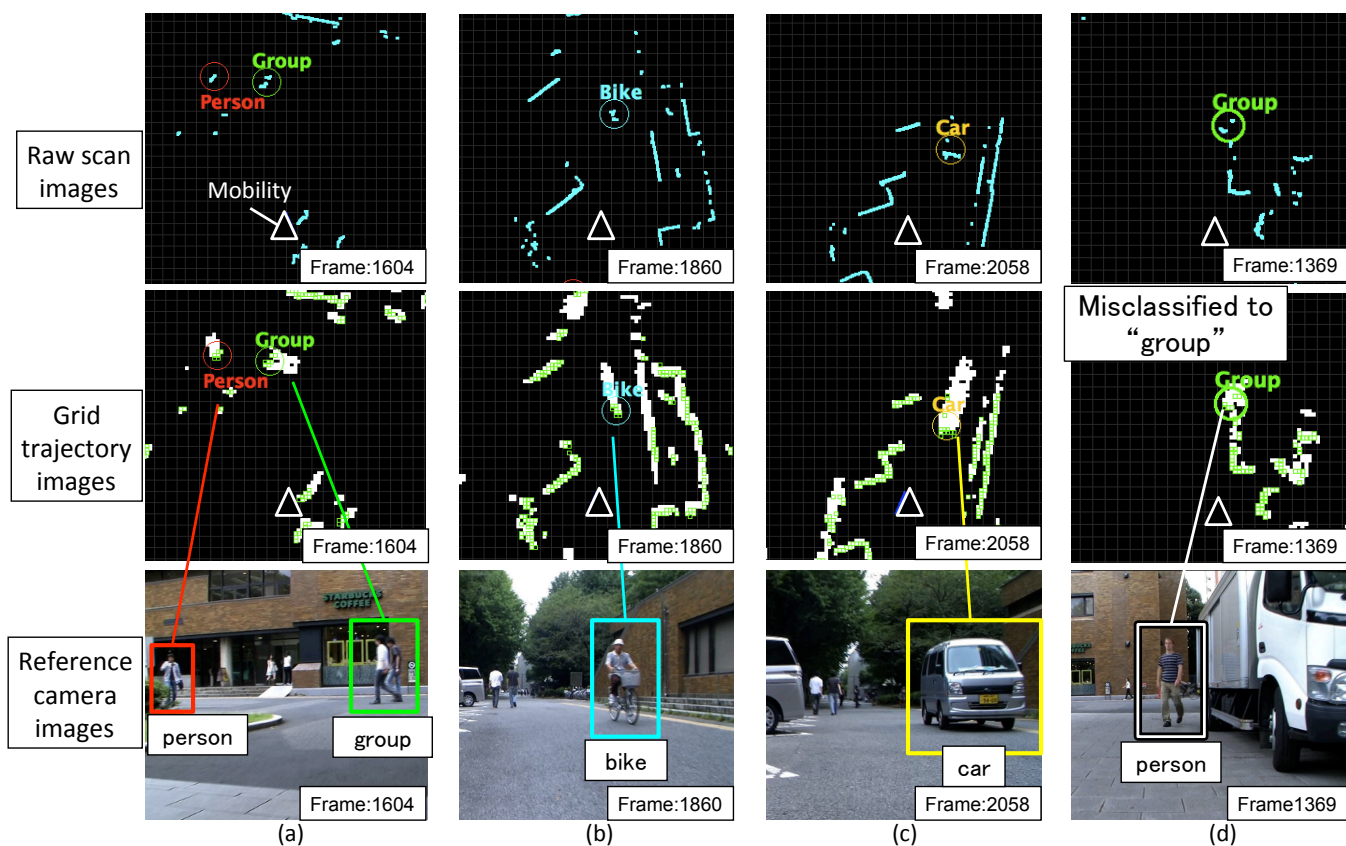
Fig. 7. Examples of (a)(b)(c) correct classification results and (d) an incorrect one
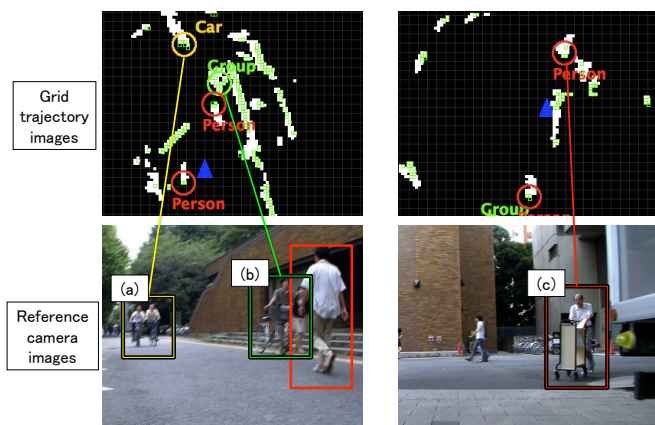

Fig. 9. Examples of moving object detection results other than our targets

but also represent the size and the speed of moving objects. Based on these trajectories, our method realizes robust detection and classification against LRF scan noises. In our experiment, the proposed method detected and classified target four moving objects(a person, a group of people, a bike, and a car) more than 80 % at F-measure in each frame. Our future task is an avoidance of misclassifications when multi objects approach each other.

REFERENCES

[1] M. Guanglin, P. Su-Birm, and A. Ioffe and. A real time objects detection approach applied to reliable pedestrian detection. In *Proceedings of IEEE Intelligent Vehicles Symposium*, pages 755–760, 2007.

[2] D. Schulz, W. Burgard, D. Fox, and A. Cremers. Tracking multiple moving targets with a mobile robot using particle filters and statistical data association. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 1665–1670, 2001.

[3] T. Vu, O. Aycard, and N. Appenrodt. Online localization and mapping with moving object tracking in dynamic outdoor environments. In *the IEEE Intelligent Vehicle Symposium*, pages 190–195, 2007.

[4] H. Zhao, M. Chiba, R. Shibasaki, X. Shao, J. Cui, and H. Zha. Slam in a dynamic large outdoor environment using a laser scanner. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 1455–1462, 2008.

[5] C. C. Wang and C. Thrope. Simultaneous localization and mapping with detection and tracking of moving objects. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 2918–2924, 2002.

[6] J. H. Lee, T. Tubouchi, K. Yamamoto, and S. Egawa. People tracking using a robot in motion with laser range finder. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems(IROS)*, pages 2936–2942, 2006.

[7] H. Zhao, Q. Zhang, M. Chiba, R. Shibasaki, J. Cui, and H. Zha. Moving object classification using horizontal laser scan data. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 2424–2430, 2009.

[8] D. Streller and K. Dietmayer. Multiple hypothesis classification with laser range finders. In *Proceedings of Intelligent Transportation Systems Conference*, pages 195–200, 2004.

[9] P.Viola and M.J.Jones. Robust real-time object detection. In *Proceedings of IEEE Workshop on Statistical and Theories of Computer Vision*, 2001.

[10] O. M. Mozos, C. Stachniss, and W. Burgard. Supervised learning of places from range data using adaboost. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 1742–1747, 2005.