Small Object Change Detection for Small Obstacle Avoidance in Everyday Robot Navigation

Koji Takeda[1], Kanji Tanaka[2] and Yoshimasa Nakamura[1]

Abstract— **This paper addresses the problem of small object change detection for small obstacle avoidance in everyday robot navigation. Despite recent research progress in the field of object detection and change detection, the problem of detecting semantically non-distinctive and visually small objects is still a challenging problem. We developed a practical image processing pipeline by combining state-of-the-art techniques from image retrieval, image registration, and image change detection. We then integrated the image processing pipeline into a traditional plan-sense-act cycle to realize a reactive collision avoidance system. Experiments using a real mobile robot verified the effectiveness of the proposed approach.**

I. Introduction

This paper considers the problem of small object change detection during everyday robot navigation and its application to small obstacle avoidance. Avoiding collisions with small objects (e.g., nails, cables, smartphones, glasses, handkerchiefs) is undoubtedly an important capability for an indoor mobile robot to avoid damaging itself and its surroundings. In this study, we consider everyday navigation scenarios, in which the robot may encounter unseen small objects in a familiar environment such as “convenience store,” “flooring,” and “office room”.

The problem of image change detection becomes challenging when changes are semantically *non-distinctive* and visually *small*. In these cases, an image change detection model (e.g., semantic segmentation [1], object detection [2], anomaly detection [3], and differencing [4]), which is trained in a past domain to discriminate between the foreground and the background, may fail to classify an unseen object into the correct foreground or background class. Typical alternative solutions, such as visual object detection [5], assume pre-trained object categories and are not valid for unknown objects.

We address the above issue by introducing a plan-sense-act (PSA) pipeline, as shown in Fig. 1. Specifically, the motion planning is formulated as a batch process and the system operates on a relatively long-term planning horizon. We developed a practical image processing pipeline by combining state-of-the-art techniques from image retrieval, image registration, and image change detection. We then integrated the image processing pipeline into a traditional plan-sense-act cycle for a reactive collision avoidance system. Experiments using a real mobile robot verified the effectiveness of the proposed approach.

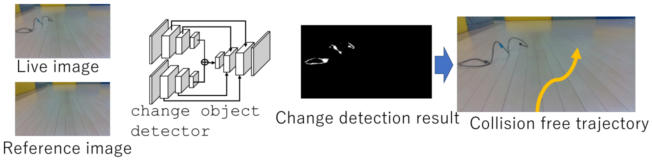

Fig. 1. Floor object avoidance application using change detection

II. Related Work

Image change detection is a long standing issue of computer vision and it has various applications such as satellite image [6], [7], and autonomous driving [4], [8]. Existing studies are divided into 2D or 3D, according to the sensor modality, and we focus on image change detection in 2D perspective views from an on-board front-facing camera in this study. Since the camera is a simple and inexpensive sensor, our 2D approach can be expected to have an extremely wide range of applications.

Pixel-wise differencing techniques for image change detection rely on the assumption of precise image registration between live and reference images [9]. This method is effective for classical applications such as satellite imagery [9], in which precise registration is available in the form of 2D rotation-translation. However, this is not the case for our perspective view applications [10], in which precise pixel-wise registration itself is a challenging ill-posed problem. This problem may be alleviated to some extent by introducing an image warping technique, as we will discuss in Section III-B. However, such pixel warping is far from perfect, and may yield false alarms in image change detection.

Novelty detection is a major alternative approach to image change detection [11]. In that, novelties are detected as deviations from a nominal image model that is pre-trained from unlabeled images in a past training domain. Unlike pixel-wise differencing, this technique can naturally capture the contextual information of the entire image to determine whether there are any changes in the image. However, on the downside, the change regions cannot be localized within the image even if the existence of the change is correctly predicted. Therefore, existing researches of novelty detection in the literature have focused on applications such as intruder detection [12], in which the presence or absence of change, not the position of the changing object, is the most important outcome information.

[1] K.Takeda and Y.Nakamura are with Tokyo Metropolitan Industrial Technology Research Institute, Tokyo, Japan {takeda.koji_1, nakamura.yoshimasa}@iri-tokyo.jp

[2] K. Tanaka is with Faculty of Engineering, University of Fukui, Japan. tanakakanji@gmail.com

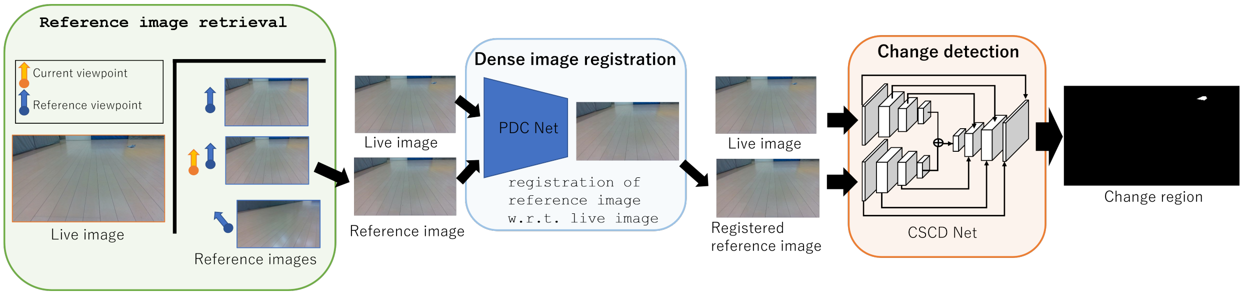

Fig. 2. Small object detection pipeline using change detection

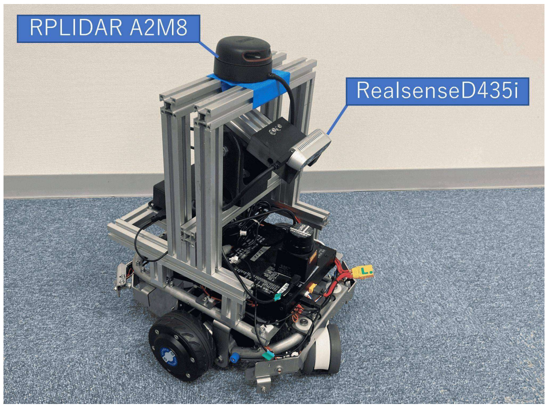

Fig. 3. Robot used in the experiment

Several new architectures targeting small object change detection have recently been presented. For example, Klomp et al. proposed to use Siamese CNN to detect markers for improvised explosive devices (IEDs) [13], where they tackled the resolution problem by removing the output-side layer of ResNet-18 [14] to improve the detection performance of small objects. Our approach differs from these existing approaches in that (1) it extends the plan-sense-action cycle to realize a reactive collision avoidance system, and (2) it is able to incorporate the background model.

III. Small Object Detection Framework

Figure 2 shows the proposed pipeline of small object change detection. First, we perform a preprocessing to compensate for the viewpoint error and the resulting uncertainty in non-linear mapping from the 3D real environment to a 2D image plane of the on-board camera. This preprocessing consists of LRF-SLAM based viewpoint estimation (Section III-A) followed by pixel-wise warping (Section III-B). However, even with such a preprocessing, the images are often affected by unpredictable nonlinear mapping errors. To address this, we introduce a plan-sense-act cycle for stable collision avoidance. That is, the robot operates on a relatively long-term planning horizon, in which muliti-view knowledge integration is performed to keep a 2D obstacle map up to data, and replanning is performed to avoid obstacles safely and efficiently. Each of the above modules/subsystems will be described in detail in Subsections III-A, III-B, and III-C.

Figure 3 shows the experimental robot platform. The robot is equipped with a highly-accurate positioning system based on a two-dimensional laser range finder (2D LRF). Without losing generality, a highly accurate two-dimensional environment map is assumed to be constructed in advance using the LRF-based SLAM algorithm. In online, the robot estimates the robot's viewpoint by building a local map in a similar way and map-matching the local map with the environment map. It was found that this LRF-based map matching is sufficiently robust against dynamic changes caused by dynamic obstacles and pedestrians, and it provides state and accurate function of self-localization. It is also assumed that the front-facing monocular camera is the only sensor that can be used for change detection. That is, our target small objects are too small and cannot be detected by the LRF system mentioned above.

A. Image Retrieval

A change detection algorithm requires a pairing of live and reference images as input. We developed an image retrieval system for aligning live images with the reference images. An input live image is paired with a reference image if its angle deviation from the live image is less than the threshold of 3.6 degree. If no such reference image exists, it is paired with the nearest neighbor viewpoint to the live image's viewpoint, without considering the angle information.

B. Image Registration

We further compensate for the viewpoint misalignment in LRF-SLAM by introducing an image warping technique. A warp is a 2D function, $u(x, y)$, which maps a position (x, y) in the reference image to a position $u = (x', y')$ in the live image. A method for dense image alignment, PDC-Net, which is recently proposed in [15], is employed to find an appropriate warp, by minimizing an energy function in the form:

$$-\log p(Y|\Phi(X;\theta)) = \sum_{ij} \log p(y_{ij}|\varphi_{ij}(X;\theta)) \qquad (1)$$

where X is input image pair $X = (I^q, I^r)$, Y is ground-truth flow, Φ and φ are predicted parameters. PDC-Net is a neural network that takes two images as input and finds the correspondence between the pixels of the two images. Compared with the conventional methods, the uncertainty of the prediction of the correspondence between pixels can

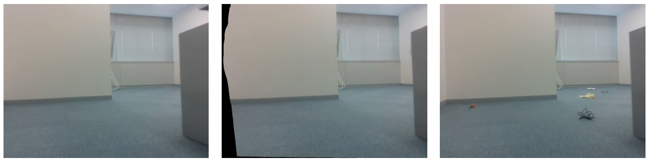

Fig. 4. An example of pixel warping.

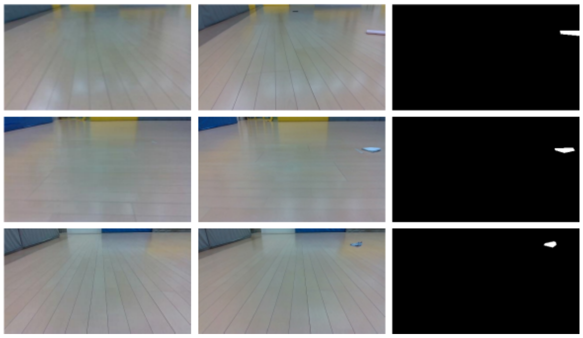

Fig. 5. Image used for training CSCD Net. From the left, reference image, live image, teacher image.

be obtained at the same time, so that the prediction of pixels with high uncertainty can be unreliable. Uncertainty is expressed as a real number from 0 to 1, and this time image alignment was performed using only pixels with an uncertainty of less than 0.5. PDC-Net used a network trained in the MegaDepth dataset [16]. An example of pixel warping is shown in Fig. 4.

C. Image Change Detection

The state-of-the-art Siamese model for image change detection, CSCDNet [1], is used as our base architecture. The network is initialized with the weight pre-trained on ImageNet [17]. The pixel-wise binary cross-entropy loss is used as loss function as in the original work of CSCDNet [1]. PDCNet [15] is used to align reference images. Adam optimizer [18] is used for the network training. Learning rate is 0.0001. The number of iterations is 20,000. The batch size is 32.

D. Plan-Sense-Act Cycle

The proposed image processing pipeline was implemented on a collision avoidance system for an indoor mobile robot. The collision avoidance system uses the traditional plan-sense-act cycle. In the planning phase, a static map of obstacles is used to find an collision-free route to the goal. The plan is updated in real time by feeding back the results of sense and action during navigation. Specifically, the map is updated by incorporating the information of new changing objects provided by the image processing pipeline into the obstacle map, assuming that the pose relationship between the floor surface and the camera coordinate system is a-priori known. The next best action is then generated via a shortest path algorithm on the updated map.

TABLE I

EXPERIMENTAL RESULTS

Scene ID	Depth Camera	Change Detection
1	× (smartphone)	✓
2	✓	✓
3	× (pliers, bolts)	× (bolts)
4	✓	✓
5	× (S-shaped hook)	✓

IV. Experimental Results

The proposed method was implemented on a real indoor mobile robot and verified experimentally.

A. Settings

The number of training images for the CSCD-Net (Section III-C) was 20,000, and the learning rate was 0.0001. The number of iterations was 20,000, The training process took about 20 hours using the NVIDIA GeForce RTX 3090.

A trained CSCD-Net outputs for each pixel, the probability of the pixel belonging to change regions as a real value between 0 to 1. We binarize the probability into 0 or 1 using a preset threshold of 0.5.

The number of training and test images were 184 and 52.

For the small change objects, two types of wallets and three types of handkerchiefs, were placed at random locations on the floor.

The experimental platform is shown in Fig. 3. The size of the robot is width 0.345 [m] × depth 0.335 [m] × height 0.450 [m]. It is equipped with a 2D Lidar sensor RPLIDAR A2 M8 and an RGB-D camera RealSense D435i. The resolution of the camera was 424 × 240.

The camera was mounted on the robot at yaw angle -8.6 [deg] and height 0.245 [m]. An alternative possible solution would be to mount the camera horizontally (i.e., yaw angle 0 [deg]). However, it would have large occluded regions, and fail to detect nearby small obstacles. On the contrary, mounting the camera vertically (i.e., yaw angle -90 [deg]) would have narrow field of views, and would not be effective for detecting distant obstacles. As an another drawback, images taken by such a downward-looking camera often fail to capture features of background objects, yielding poor performance in image change detection.

The training set was annotated in a semi-supervised way. The annotation process consists of two steps. First, PaddleSeg [19], [20], a highly-efficient automatic segmentation model was applied to a training image. Next, each of the segmented regions was manually labeled as either change or no-change. An example of training image and its annotated regions is shown in the Fig. 5.

The planning module for the collision avoidance system was based on the autonomous traveling package of ROS [21]. Specifically, teb_local_ planner and global_planner [22] were employed as the local and global planners.

An additional depth image information from the on-board RGB-D camera was used for precisely measuring the small change objects detected with respect to the obstacle map.

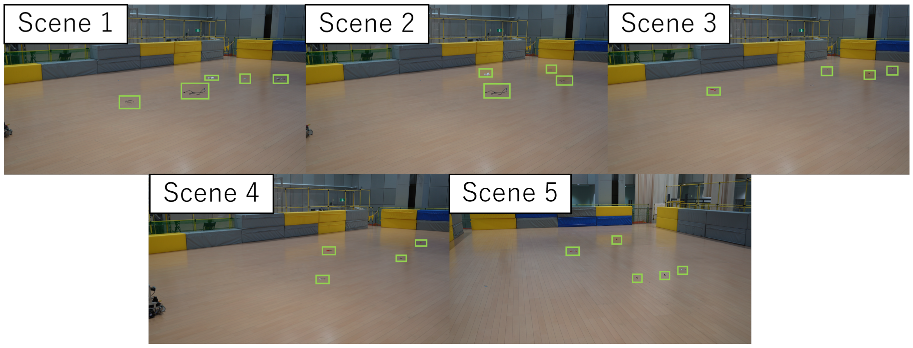

Fig. 6. test environment

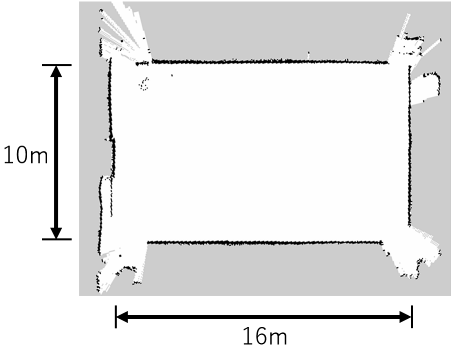

Fig. 7. 2D map created in the test environment

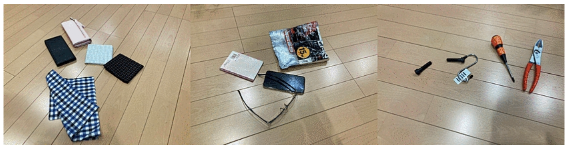

Fig. 8. Small object dropped on the floor this time

This choice allows us to eliminate the influence of depth noises in obstacle avoidance. However, we believe that recent 3D depth prediction models would provide sufficiently noise-robust depth information without relying on such an additional depth sensor, which is a future direction of research.

An alternative depth-based change detection system was developed and used as a baseline method for change detection. Specifically, depth image was reused for this purpose. This is an application that estimates the relative 3D position of a small object on the floor from the depth image of the RGB-D camera and adjusts it so that an object with a height of about 10 [mm] or more from the bottom of the robot is detected as an obstacle.

The environment is a flooring space with an area of 10m × 16m. Figs. 6 and 7 show the test environment and map created in the test environment. The test environment was different from the training environment only in the robot's trajectories and the appearance of small change objects.

Figure 8 shows small objects were used in the experiments. For scene #1, glasses, cable, notepad, smartphone, and bag were used as the small objects. For scene #2, cable, notepad, glasses, and smartphone were used. For scene #3, pliers, bolts, and Phillips screwdriver were used. For scenes #4 and #5, a character hook, pliers, bolts, and a screwdriver were used. For all scenes, the robot moves from left to right in the Fig. 7. For fair comparison, the start position, goal position, and object placement of the robot were set the same between the proposed and baseline methods.

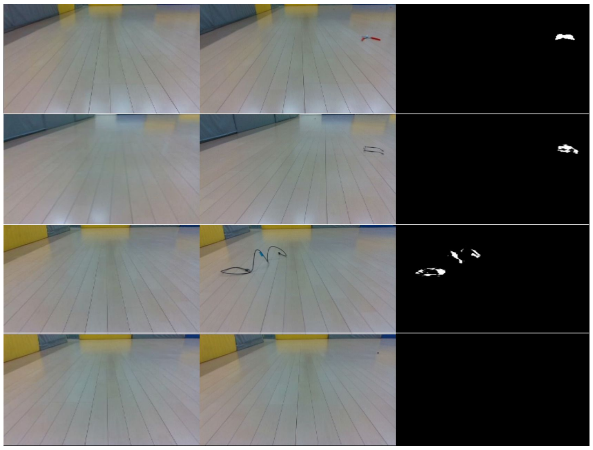

Fig. 9. Change detection result. From the left, reference image, live image, change detection result.

B. Results

Table I lists all the small objects the method failed to avoid collision with during the five sessions of navigation tasks. The proposed method was successful for most small objets the robot encountered in the five different scenes. As can be seen, the proposed method outperforms the baseline method for most scenes and small objects considered here. Bolt was the only object that the proposed framework failed to avoid a collision.

Figure 9 shows results of change detection.

It should be noted that the depth-based method (i.e., baseline method) has a narrow effective distance range in which changes can be stably detected. In contrast, the proposed method was able to detect visually small objects at distance more stably.

The processing cycle of the proposed pipeline was about 12 Hz. Accelerating the real-time processing towards highspeed mobile robot applications is an important direction of future research. We also plan to collect large datasets and conduct large-scale experiments.

V. Conclusion

Despite recent research progress in the field of object detection and change detection, the problem of detecting semantically non-distinctive and visually small objects is still a challenging problem. We developed a practical image processing pipeline by combining state-of-the-art techniques from image retrieval, image registration, and image change detection. We then integrated the image processing pipeline into a traditional plan-sense-act cycle to realize a reactive collision avoidance system. Experiments using a real mobile robot verified the effectiveness of the proposed approach.

References

[1] Ken Sakurada, Mikiya Shibuya, and Wang Weimin. Weakly supervised silhouette-based semantic scene change detection. *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2020.

[2] Lei Ma, Manchun Li, Thomas Blaschke, Xiaoxue Ma, Dirk Tiede, Liang Cheng, Zhenjie Chen, and Dong Chen. Object-based change detection in urban areas: The effects of segmentation strategy, scale, and feature space on unsupervised methods. *Remote Sensing*, 8(9):761, 2016.

[3] Trong-Nguyen Nguyen and Jean Meunier. Anomaly detection in video sequence with appearance-motion correspondence. pages 1273–1283, 2019.

[4] Pablo F Alcantarilla, Simon Stent, German Ros, Roberto Arroyo, and Riccardo Gherardi. Street-view change detection with deconvolutional networks. *Autonomous Robots*, 42(7):1301–1322, 2018.

[5] Zhigang Dai, Bolun Cai, Yugeng Lin, and Junying Chen. Up-detr: Unsupervised pre-training for object detection with transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1601–1610, 2021.

[6] Daifeng Peng, Yongjun Zhang, and Haiyan Guan. End-to-end change detection for high resolution satellite images using improved unet++. *Remote Sensing*, 11(11), 2019.

[7] Jie Chen, Ziyang Yuan, Jian Peng, Li Chen, Haozhe Huang, Jiawei Zhu, Yu Liu, and Haifeng Li. Dasnet: Dual attentive fully convolutional siamese networks for change detection in high-resolution satellite images. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 14:1194–1206, 2020.

[8] Ken Sakurada, Weimin Wang, Nobuo Kawaguchi, and Ryosuke Nakamura. Dense optical flow based change detection network robust to difference of camera viewpoints, 2017.

[9] Daifeng Peng, Yongjun Zhang, and Haiyan Guan. End-to-end change detection for high resolution satellite images using improved unet++. *Remote Sensing*, 11(11):1382, 2019.

[10] Ken Sakurada and Takayuki Okatani. Change detection from a street image pair using cnn features and superpixel segmentation. 61:1–12, 2015.

[11] Boris Sofman, Bradford Neuman, Anthony Stentz, and J Andrew Bagnell. Anytime online novelty and change detection for mobile robots. *Journal of Field Robotics*, 28(4):589–618, 2011.

[12] Waqas Sultani, Chen Chen, and Mubarak Shah. Real-world anomaly detection in surveillance videos. pages 6479–6488, 2018.

[13] Sander R Klomp, Dennis WJM van de Wouw, et al. Real-time small-object change detection from ground vehicles using a siamese convolutional neural network. *Electronic Imaging*, 2020(6):60402–1, 2020.

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[15] Prune Truong, Martin Danelljan, Luc Van Gool, and Radu Timofte. Learning accurate dense correspondences and when to trust them. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5714–5724, 2021.

[16] Zhengqi Li and Noah Snavely. Megadepth: Learning single-view depth prediction from internet photos. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2041–2050, 2018.

[17] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[18] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[19] Yi Liu, Lutao Chu, Guowei Chen, Zewu Wu, Zeyu Chen, Baohua Lai, and Yuying Hao. Paddleseg: A high-efficient development toolkit for image segmentation, 2021.

[20] PaddlePaddle Contributors. Paddleseg, end-to-end image segmentation kit based on paddlepaddle. `https://github.com/PaddlePaddle/PaddleSeg`, 2019.

[21] Stanford Artificial Intelligence Laboratory et al. Robotic operating system.

[22] Christoph Rösmann, Frank Hoffmann, and Torsten Bertram. Integrated online trajectory planning and optimization in distinctive topologies. *Robotics and Autonomous Systems*, 88:142–153, 2017.