View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by Okayama University Scientific Achievement Repository

Path Tracking by a Mobile Robot Equipped with Only a Downward Facing Camera

Isaku Nagai and Keigo Watanabe

***Abstract*—This paper presents a practical path-tracking method for a mobile robot with only a downward camera facing the passage plane. A unique algorithm for tracking and searching ground images with natural texture is used to localize the robot without a feature-point extraction scheme commonly used in other visual odometry methods. In our tracking algorithm, groups of reference pixels are used to detect the relative translation and rotation between frames. Furthermore, a reference pixel group of another shape is registered both to record a path and to correct errors accumulated during localization. All image processing and robot control operations are carried out with low memory consumption for image registration and fast calculation times for completing the searches on a laptop PC. We also describe experimental results in which a vehicle developed by the proposed method repeatedly performed precise path tracking under indoor and outdoor environments.**

I. Introduction

Schemes for controlling mobile robot navigation can be categorized into two types: occasional path planning, such as that employed by a cleaning robot; and traveling on a path specified by a control system adapted for planned motion. One approach for the latter type of control involves teaching and playing back motions. In such an approach, a path is recorded by manual operation while the robot estimates its own position in the teaching phase, and the robot is guided toward the recorded path by correcting position errors in the playback phase [1]. Some methods involve memorizing the positions of landmarks (e.g., edges in an image captured by a camera mounted on the robot for recording a path) and correcting the error in position estimated by wheel odometry [2]. In such a method, the view of the camera and brightness of the surrounding landmarks need to be secured. Information on characteristic shapes, such as the flat or angular part of a wall, can be obtained by a laser range finder and used for localization [3]. Methods that use a laser range finder require that landmarks are at the same height as the sensor.

For motion estimation, there are several advantages of using visual odometry to track feature points in images of scenes from a camera mounted on the robot [4–7]. One advantage is that it is applicable to mobile robots with various types of traveling mechanism because its non-contact sensing method precludes mechanistic problems including wheel slippage and skids [8, 9]. However, appropriate feature points are not always available in various environments.

Isaku Nagai is with the Department of Intelligent Mechanical Systems, Okayama University, Graduate School of Natural Science and Technology, 1-1 Tsushima-naka, 3-Chome, Okayama 700-8530, Japan (corresponding author to provide phone/fax: 81-86-251-8065; e-mail: in@sys.okayama-u.ac.jp).

Keigo Watanabe is with the Department of Intelligent Mechanical Systems, Okayama University, Graduate School of Natural Science and Technology, 1-1 Tsushima-naka, 3-Chome, Okayama 700-8530, Japan (e-mail: watanabe@sys.okayama-u.ac.jp).

Other visual odometry methods estimate position by using a downward camera facing the ground [10]. This method, which is also independent of the traveling mechanism, has another advantage in that it requires neither the presence nor visibility of landmarks, the latter of which is often degraded by light condition and other objects that surround them. In such methods, mosaic-based localization is one technique for positioning a vehicle [11]. However, the mosaicing process requires a large memory for storing images; in addition, the process of matching the image to a map is time-consuming. The tracking of local feature points extracted by an algorithm such as SIFT, SURF, KLT, or Blob is a method commonly used for visual odometry [12–16]. However, the problems with this novel method are its high calculation costs and sensitivity of the feature extraction parameter to surface texture. Moreover, the number of feature points to be extracted is uncertain and varies depending on the texture of the ground image because such points are extracted only when some conditions in the algorithm are satisfied.

In contrast, the image tracking method we have developed for estimating planar motions including rotation is robust because no parameter adjustments are required for passage planes with various textures. In addition, it is fast because it uses a group of reference pixels as in conventional template matching. This image tracking method was applied to mobile robots with a device in which an algorithm for the positioning on an indoor floor and outdoor terrain was implemented [17, 18]. Here we extend our method to correct errors by using a reference pixel group of another shape as an image of the passage plane. An image with position and angle information is registered at every time the robot travels 0.05 m during the first run before tracking the same path. The frequent correction of errors is enabled by a rapid search of the registered image based on the reference pixel group mentioned above. The calculation algorithm is further improved to be executable at sufficient speed even by software on a laptop PC. In the following sections, we describe in detail the tracking algorithm for detecting the relative translation and rotation between frames, as well as the registration and search of a reference pixel group for correcting errors. We also report experimental results in which a vehicle developed with only a downward facing camera repeatedly tracked paths initially recorded by manual control.

II. Localization

A. Relative Search for Detecting Movement Between Frames

Position estimation is performed by integrating relative movements of translation and rotation between frames. Two reference pixel groups are used for estimating the motion. The location and illuminance of each pixel are independently stored and updated. Despite the independent architecture, the coordinates of all pixels are moved and rotated like a rigid body for the search. One reference pixel group has a shape different from the rectangular shape in conventional template

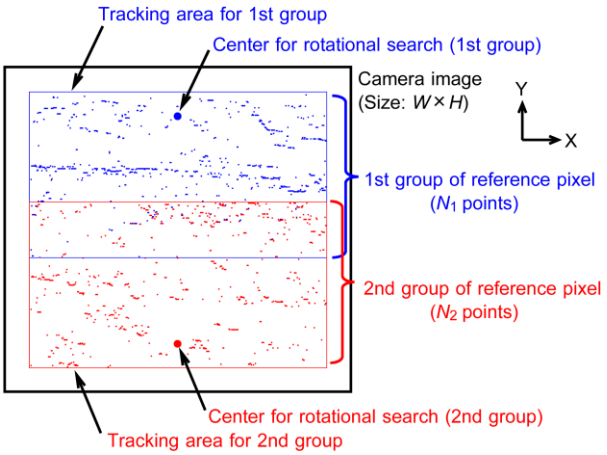


Figure 1. Two groups of reference pixels for detecting relative translation and rotation.

matching, with scattered locations for each pixel (Fig. 1). The two groups (although they partly overlap) are located at different tracking areas in a frame, and the actual number of pixels is fixed to 768 for both N_1 and N_2. The pixels are initially selected by finding the maximum and minimum illuminance in each vertical line of the tracking area. This algorithm is faster than others for feature point extraction. Each group has a different center of rotation (O_X, O_Y) that is used to perform three-dimensional relative searches. For this search procedure, $E(i,j,k)$ based on the sum of absolute differences correlation is evaluated according to

$$E(i,j,k) = \sum_{n=1}^{N} \left| I(n) - C(\hat{T}_X(n,i,k), \hat{T}_Y(n,j,k)) \right| \tag{1}$$

$$\begin{bmatrix} \hat{T}_X(n,i,k) \\ \hat{T}_Y(n,j,k) \end{bmatrix} = \begin{bmatrix} \cos k & \sin k \\ -\sin k & \cos k \end{bmatrix} \begin{bmatrix} T_X(n) - O_X \\ T_Y(n) - O_Y \end{bmatrix} + \begin{bmatrix} O_X \\ O_Y \end{bmatrix} + \begin{bmatrix} p_X + i \\ p_Y + j \end{bmatrix}$$

where N is the pixel number of a reference pixel group; $I(n)$, $T_X(n)$, and $T_Y(n)$ represent the illuminance, x coordinate, and y coordinate, respectively, of pixel n; $C(u,v)$ is the illuminance of a pixel at coordinates u, v in a camera image; and p_X, p_Y is the predictive translation, which is the same as the relative translation of the previous frame. Although all variables for x- and y-coordinates consist of 9-bit integer part and 7-bit decimal part, only the integer part of u, v is valid at the reference to $C(u,v)$. The relative motion is determined as i_d, j_d, and k_d by finding a minimum of $E(i,j,k)$ in the search ranges (d_{min} to d_{max} for translation i, j, and k_{min} to k_{max} for rotation k) as shown in

$$i_d, j_d, k_d = \arg\min_{i,j,k} E(i,j,k) \tag{2}$$

The predictive translation allows tracking up to a practical velocity (± 36 pixel/frame) with low calculation costs, although the actual range for both i and j is ± 8 pixels. For the rotational search, the actual range for k is $\pm 2.24°$ (11 steps by 0.448°).

Although each group undergoes translation and rotation according to the motion determined by the search, the location of pixel outside each tracking area; shown as the blue and red rectangular regions in Fig. 1, jumps to an opposite side of the area for permanently staying in the area. The new pixel is selected by finding the maximum or minimum illuminance in the side. The illuminance $I(n)$ of each pixel is updated independently in case such a jump occurs. Therefore, each reference pixel group forms an irregular arrangement, as shown in Fig. 1. Using the relative displacement of each group (v_{1X}, v_{1Y}) and (v_{2X}, v_{2Y}), which is the sum of translation by both prediction and search, the resultant motion (translation v_X, v_Y, and rotation ω) is determined by

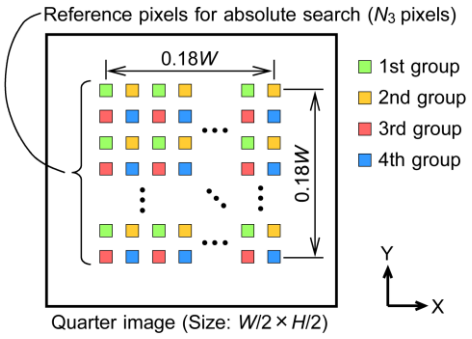


Figure 2. Grouped reference pixels with discrete coordinates for error correction.

$$\begin{bmatrix} v_{1X} \\ v_{1Y} \end{bmatrix} = \begin{bmatrix} p_{1X} \\ p_{1Y} \end{bmatrix} + \begin{bmatrix} i_1 \\ j_1 \end{bmatrix}, \quad \begin{bmatrix} v_{2X} \\ v_{2Y} \end{bmatrix} = \begin{bmatrix} p_{2X} \\ p_{2Y} \end{bmatrix} + \begin{bmatrix} i_2 \\ j_2 \end{bmatrix},$$

$$\begin{bmatrix} v_X \\ v_Y \\ \omega \end{bmatrix} = \begin{bmatrix} (v_{1X} + v_{2X})/2 \\ (v_{1Y} + v_{2Y})/2 \\ \tan^{-1}\left(\dfrac{v_{1X} - v_{2X}}{D + v_{1Y} - v_{2Y}}\right) \end{bmatrix} \tag{3}$$

where D is the distance between the rotational centers for determining the relative motions of the first and second reference pixel groups. The rotation estimated by this dual-viewpoint tracking, based on our previous results, marginally improves the measurement accuracy. Position (x, y) and orientation θ of a robot are obtained by negatively integrating the relative motion (v_X, v_Y, ω) determined at every frame, as shown in (4).

$$\begin{bmatrix} x^{f+1} \\ y^{f+1} \\ \theta^{f+1} \end{bmatrix} = \begin{bmatrix} x^f \\ y^f \\ \theta^f \end{bmatrix} - \begin{bmatrix} \cos\theta^f & \sin\theta^f & 0 \\ -\sin\theta^f & \cos\theta^f & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} v_X^f \\ v_Y^f \\ \omega^f \end{bmatrix} \tag{4}$$

where f is the frame number of a corresponding image. The coordinates (X, Y) in metric units are derived by multiplying the position (x, y) in pixel units by the resolution (m/pixel) of the camera image.

B. Registering Images for Error Correction

A ground image with the vehicle position and angle information is registered every time the vehicle travels 0.05 m. For this image, only the central area (size: $0.18W \times 0.18W$) in a quarter image (size: $W/2 \times H/2$) is used for registration in consideration of the search time, as shown in Fig. 2. Here, W and H are the horizontal and vertical dimensions, respectively, of the original image. All pixels located at discrete coordinates in the square area are simply selected regardless of the illuminance, and only the illuminances are stored for registration. Although the image also consists of a reference pixel group similar to that for the relative search, the square shape of the image is invariant.

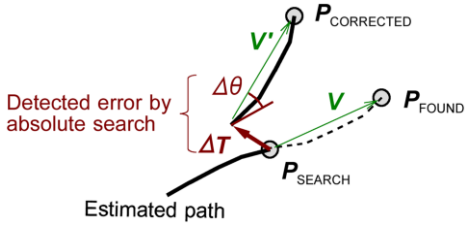


Figure 3. Error correction from $\boldsymbol{P}_{\text{FOUND}}$ to $\boldsymbol{P}_{\text{CORRECTED}}$ by using both translation $\Delta\boldsymbol{T}$ and rotation $\Delta\theta$ detected by absolute search started at $\boldsymbol{P}_{\text{SEARCH}}$.

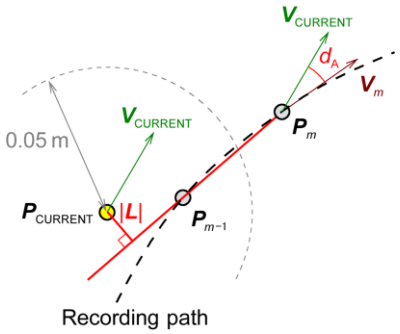


Figure 4. Positional and angular differences from recorded path for controlling front and rear steering of vehicle.

This means each pixel's location of an image is able to be reconstructed from the position and angle information. The actual number of pixels N_3 for an image is 1936(=44×44). Therefore, the amount of memory required for an image is only 1960 bytes, including position and angle information (three double precision variables).

The registered image is searched in the playback phase when the estimated vehicle position enters an area with a radius of 0.025 m from the registered position. The search is performed by using images divided into four groups shown in Fig. 2, and the detected error is corrected only when four results of the search are close to one another. This rule enhances the reliability of rejecting an erroneous error correction when the texture of a passage plane widely deteriorates compared to the registered image. The actual ranges of this absolute search are ±0.025 m (±32 pixel) for translation and ±6.7° (31 steps by 0.448°) for rotation. Therefore, an error correction is possible by finding the difference between the expected position and the detected position of a registered image. However, an error that occurred in the recording phase remains even after this correction.

The absolute search for registered images is performed by using another thread of a CPU, so that it can be processed in parallel to the relative search. However, this search takes longer than a frame time (14.3 ms) because the search range is large, as described above. For this reason, positional and angular differences arise by the time the search ends, that is ($\boldsymbol{P}_{\text{SEARCH}}$, θ_{SEARCH}) and ($\boldsymbol{P}_{\text{FOUND}}$, θ_{FOUND}) as shown in Fig. 3. By using the detected errors of $\Delta\boldsymbol{T}$ for translation and $\Delta\theta$ for orientation, the position $\boldsymbol{P}_{\text{CORRECTED}}$ and the orientation $\theta_{\text{CORRECTED}}$ after correction are determined by (5).

$$\begin{aligned} \boldsymbol{V} &= \boldsymbol{P}_{\text{FOUND}} - \boldsymbol{P}_{\text{SEARCH}} \\ \boldsymbol{P}_{\text{CORRECTED}} &= \boldsymbol{P}_{\text{SEARCH}} + \Delta\boldsymbol{T} + \mathbf{Rot}(\Delta\theta)\boldsymbol{V} \\ \theta_{\text{CORRECTED}} &= \theta_{\text{FOUND}} + \Delta\theta, \end{aligned} \tag{5}$$

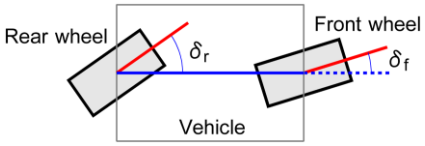
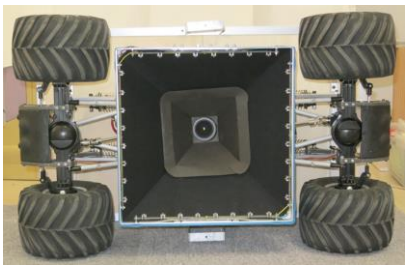


Figure 5. Angles of front and rear steering in vehicle.

(a) Exterior



(b) Bottom view

Figure 6. Vehicle developed for experiment.

where $\mathbf{Rot}(\cdot)$ is a two-dimensional rotating matrix.

III. Path Tracking of Vehicle

A. Path Recording

When the REC button of a vehicle is held down, the control system switches to the recording mode until the STOP button is pressed. In the recording mode, the vehicle, which is manually controlled by a wireless transmitter, executes localization and image registration every time it travels 0.05 m. The image data includes information of the estimated position $\boldsymbol{P}_t$ and orientation θ_t of the vehicle as a way point; therefore, a path is memorized at the same time. t is the serial number of the way point and increases every time the vehicle travels 0.05 m. The starting point and goal of a path are defined when the REC and STOP buttons, respectively, are pressed.

B. Path Tracking

The vehicle starts forward-path tracking when the PLAY button is pressed at the starting point, and reverse-path tracking when it is pressed at the goal. The vehicle automatically repeats the travel between the starting point and goal a certain number of (repeat-count) times specified in advance. The following is a description of forward-path tracking.

Fig. 4 shows the difference between the registered way point and the current position $\boldsymbol{P}_{\text{CURRENT}}$, which is used for steering control. $\boldsymbol{P}_m$ is the closest of the forward way points that are more than 0.05 m away from $\boldsymbol{P}_{\text{CURRENT}}$. First, L is defined as distance between line $\boldsymbol{P}_m\boldsymbol{P}_{m-1}$ and $\boldsymbol{P}_{\text{CURRENT}}$, which is found by (6).

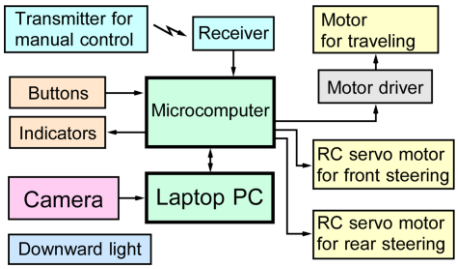


Figure 7. Components of mobile robot vehicle.

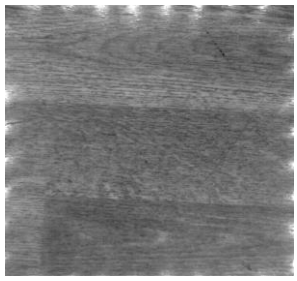


Figure 8. Example of floor image captured by vehicle camera.

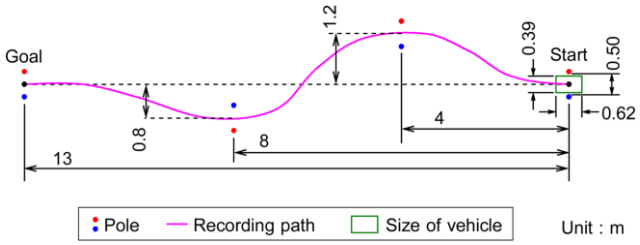


Figure 9. Dimensions of recorded path for indoor experiment.

$$\boldsymbol{u} = \boldsymbol{P}_{m-1} - \boldsymbol{P}_m$$
$$\boldsymbol{v} = \boldsymbol{P}_{\mathrm{CURRENT}} - \boldsymbol{P}_{m-1} \tag{6}$$
$$L = \frac{\boldsymbol{u} \times \boldsymbol{v}}{|\boldsymbol{u}|}$$

where L represents the deviation from the recorded path. Next, d_A is defined as a difference between the orientation θ_m memorized with $\boldsymbol{P}_m$ and $\theta_{\mathrm{CURRENT}}$, as found by (7).

$$d_\mathrm{A} = \theta_{\mathrm{CURRENT}} - \theta_m \tag{7}$$

Finally, the front and rear steering angles, δ_f and δ_r, of the vehicle are determined by (8).

$$\delta_\mathrm{f} = K_\mathrm{p} L - K_\mathrm{r} d_\mathrm{A}$$
$$\delta_\mathrm{r} = K_\mathrm{p} L + K_\mathrm{r} d_\mathrm{A} \tag{8}$$

where K_p is the gain of same-phase steering, and K_r is the gain of opposite-phase steering. The actual values of these gains are empirically determined as $K_\mathrm{p} = 0.6$ and $K_\mathrm{r} = 16$. The units for L and d_A are pixels and degrees, respectively. Furthermore, δ_f and δ_r are limited in the range of −100 to 100, which corresponds to $\pm$ 18.3°. Fig. 5 shows both steering angles in the positive direction.

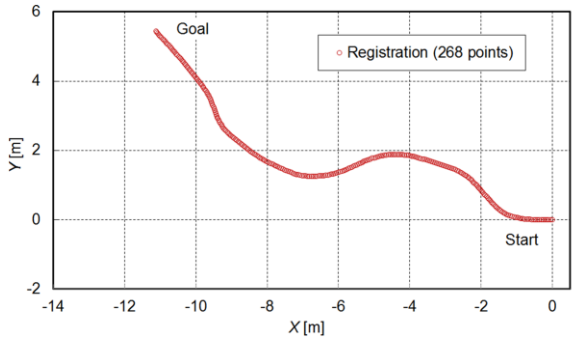


Figure 10. Points registered as a recording path.

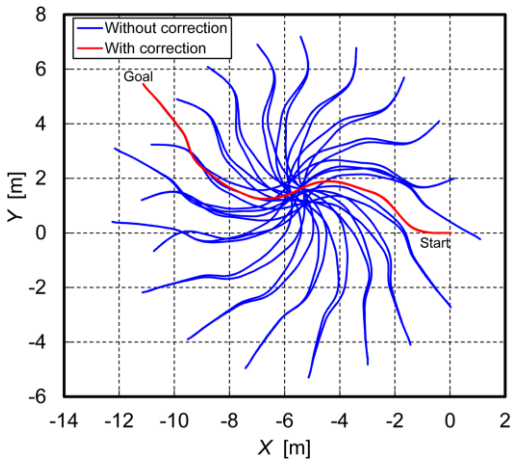


Figure 11. Estimated trajectory for path tracking of 10.5 round-way trips.

IV. Experiment

A. Experimental Setup

We manufactured a vehicle with a downward facing camera (Fig. 6). Images of a passage plane are captured by a camera (UB2D120, monochrome, USB 2.0) with an FOV of 200 ×188 [mm], resolution of 512×480 pixels, and frame rate of 70 Hz. The negative X-axis of the camera image corresponds to the forward direction of the vehicle. A lens with low distortion (LM3NC1M) is used with the camera. The luminance non-uniformity in lighting is corrected by using a reference image for the calibration, although white LEDs illuminate the ground surface at a specified illuminance with constant current circuits in an FOV cover for the camera. The vehicle is equipped with four-wheel steering and has a minimum turning radius of 0.75 m. Fig. 7 shows the components for configuring the vehicle as a mobile robot. A laptop PC (CF-J10, Core i5-2410M, Windows 7) is used to localize and control the vehicle. Servo signals from the receiver are used by a microcomputer without change, and the vehicle operates as a radio-controlled car in the path-recording mode.

B. Fundamental Tests

Measurement accuracy of the visual odometry tracking ground images was investigated, in which the vehicle traveled 10 m straight on a wood-grain indoor floor shown in Fig. 8. The errors of the estimated odometer and angle were −0.99% and 0.44°/m, respectively. We further confirmed the effectiveness of the reliability enhancement using four image

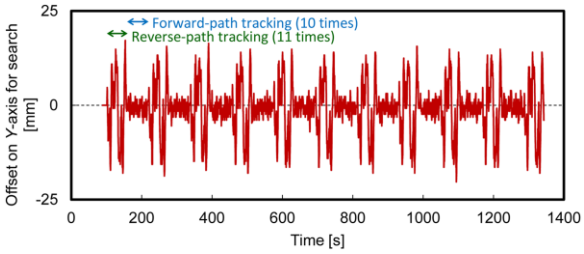


Figure 12. Deviation in position for indoor experiment.

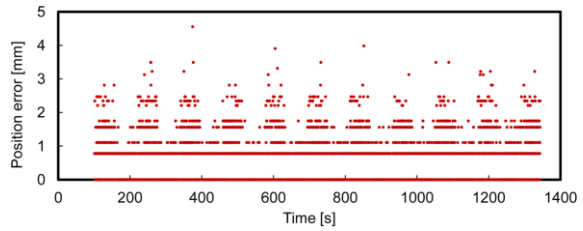


Figure 13. Position error of localization detected by absolute search.

groups, in which the passage plane was partially covered with a different texture (carpet tile) for a length of 0.5 m after the linear path of 10 m was recorded on the wood-grain floor. The vehicle deviated from the partially-covered path when the reliability enhancement was disabled. In contrast, the vehicle shuttled along the path without a large deviation when it was enabled.

C. Path Tracking on Indoor Floor

We conducted an indoor experiment on the same floor as used in the previous section. The path shown in Fig. 9 was recorded in the experiment. Pairs of poles spaced 0.50 m apart were placed at four locations along the path, and the vehicle was driven between the poles by manual operation in the recording phase. Because our vehicle is 0.39 m wide, the total clearance between the vehicle and pole is 0.11 m. After the one-way trip was recorded, the vehicle was instructed to track the path for 10.5 round-way trips in the playback phase.

Fig. 10 shows the estimated positions at which images were registered. The trajectory of the position is based on accumulation of the estimated movements by the relative search; hence, it curves slightly toward the right. The number of registrations was 268, and therefore, the one-way travel distance was estimated as 13.4 m.

During the path tracking of 10.5 round-way trips, the vehicle did not touch the poles at all and finally returned to the starting point after the entire run. The estimated trajectory with the proposed error correction in the playback phase is shown by the red line in Fig. 11. The blue line shows the estimated trajectory without the correction for reference. The trajectory with the correction corresponds well with the recorded path.

Fig. 12 shows the amount of preliminary offset on the *Y*-axis before starting each absolute search. This offset is the positional deviation of the vehicle from the recorded path. The maximum and average offsets were 20.3 and 4.4 mm, respectively. Hence, the distance between the poles and the vehicle was secured because the margin for one side of the vehicle in this experiment is about 50 mm. The cause that the vehicle deviated larger in the reverse-path tracking compared to the forward-path tracking may be coming from the asymmetric steering mechanism of the vehicle. However, further investigation is necessary for clarifying the reason.

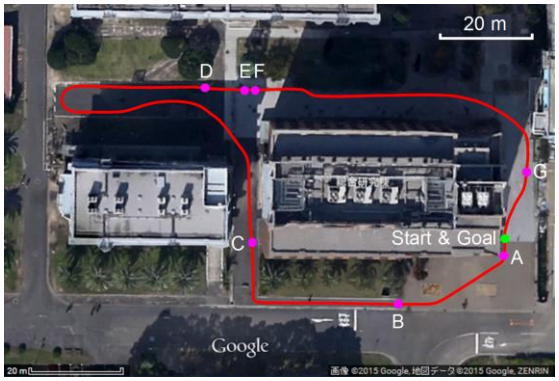


Figure 14. Recording path on a campus and labeled points.

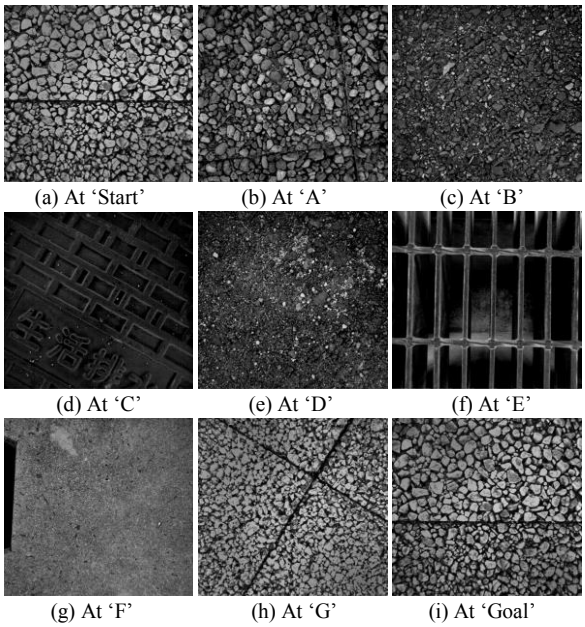


Figure 15. Examples of outdoor ground at labeled points.

The maximum and average angular offsets were 3.1° and 0.69°, respectively. These deviations may be further reduced by improving the control algorithm for path tracking.

Fig. 13 shows the error detected by the absolute search during path tracking, i.e., the deterioration of localization. The maximum and average errors are 4.6 and 0.63 mm, respectively. Such precise localization was realized by the frequent error corrections made every time the vehicle traveled 0.05 m.

D. Path Tracking in Outdoor Environment

We also conducted an outdoor experiment in which all settings for the vehicle were kept identical to those of the indoor experiment. In the recording phase, the vehicle was operated to run a path on a campus of Okayama University (Fig. 14). The path has various surface textures along the passage plane. Fig. 15 shows examples of the different textures, which correspond to the labeled points in Fig. 14. The number of registrations was 4868; therefore, the one-way travel distance was estimated as 243.4 m.

Fig. 16 shows the estimated trajectories in the recording, forward path tracking, and reverse path tracking phases. In this experiment, we conducted path tracking the day after recording the path. The vehicle shuttled along the path at an

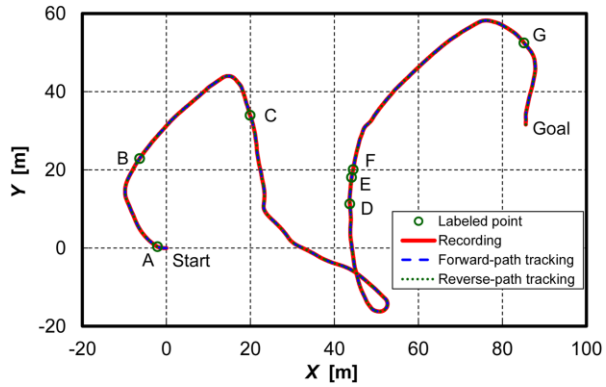


Figure 16. Estimated trajectories of outdoor path.

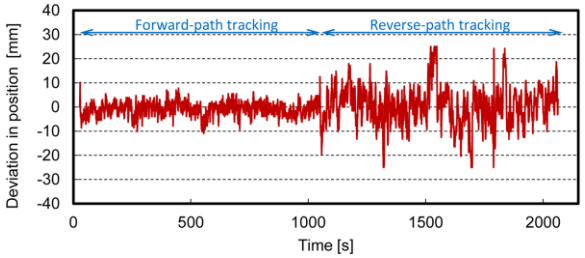


Figure 17. Deviation in position for outdoor experiment.

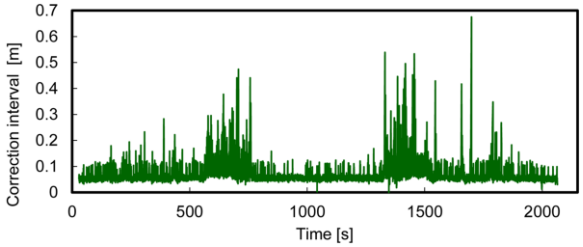


Figure 18. Interval between corrections for outdoor experiment.

average velocity of 0.26 m/s, and returned to the starting point. Fig. 17 shows the positional deviation measured as the offset on the *Y*-axis. The maximum and average offsets were 25 and 4.1 mm, respectively. Furthermore, the maximum and average angular offsets were 3.2° and 0.49°, respectively. Some absolute searches were skipped during reverse path tracking because the some deviations exceeded the search range of ±0.025 m. Of the absolute search results, 8.5% were rejected by the reliability enhancement described in the previous section. The interval between error corrections is shown in Fig. 18. The average interval was 0.062 m. This indicates that most registered images were used for error correction (i.e., without skipping). Thus, spatially dense corrections were performed for tracking. We also obtained a similar result even when the vehicle tracked the same path five days after the recording.

E. Processing Time and Memory Consumption

The average and maximum processing times for a relative search in the outdoor experiment were 3.5 and 6.3 ms, respectively. Because the cycle time of a camera frame is 14.3 ms, the processing time is sufficiently fast. This rapid search process is mainly accelerated by using programming techniques such as two-level pyramid search, fixed-point arithmetic, and loop-order shuffle. The average and maximum processing times for an absolute search were 196 and 360 ms, respectively.

The number of registrations was 4868, indicating that 9.54 MB of memory was used for storing ground images. This is equivalent to 38.8 monochrome images, each with a size of 512×480 pixels. Because ground images are registered with a form of reference pixel groups, the memory consumed for 4868 ground images is small for the outdoor path with a travel distance of 243.4 m.

V. Conclusion

A method was proposed for a mobile robot to track paths with only a downward facing camera as the sensor. Our image tracking algorithm is unique in that is not based on feature detection but uses groups of reference pixels to detect relative motion between frames and to correct the accumulated error. Our algorithm is fast, low in memory consumption, and enables frequent error corrections (i.e., every time a distance of 0.05 m is traveled) to track long paths precisely. The vehicle we developed was experimentally confirmed to repeatedly travel paths recorded at indoor and outdoor environments. In future work, we will evaluate our method under various outdoor conditions such as on ground surfaces that are wet or covered with leaves.

References

[1] T. Shioya, K. Kogure, and N. Ohta, "Minimal autonomous mover -MG-11 for Tsukuba Challenge-," *J. Robot. Mechatron.*, vol. 26, no. 2, pp. 225–235, 2014.

[2] A. Ohya, Y. Miyazaki, and S. Yuta, "Autonomous navigation of mobile robot based on teaching and playback using trinocular vision," *Proc. 27th Annu. Conf. IEEE Ind. Electron. Soc.*, vol. 1, 2001, pp. 398–403.

[3] S. Thrun, W. Burgard, and D. Fox, "Probabilistic robotics," MIT Press, Cambridge, MA, 2005.

[4] A. Howard, "Real-time stereo visual odometry for autonomous ground vehicles," *Proc. Int. Conf. Intell. Robots Syst.*, 2008, pp. 3946–3952.

[5] P. Furgale and T. D. Barfoot, "Visual teach and repeat for long-range rover autonomy," *J. Field Robot.*, vol. 27, no. 5, pp. 534–560, 2010.

[6] K. Irie, T. Yoshida, and M. Tomono, "Mobile robot localization using stereo vision in outdoor environments under various illumination conditions," *Proc. Int. Conf. Intell. Robots Syst.*, 2010, pp. 5175–5181.

[7] H. Lategahn and C. Stiller, "Vision-only localization," *Trans. Intell. Transp. Syst.*, vol. 15, no. 3, pp. 1246–1257, 2014.

[8] A. Mallet, S. Lacroix, and L. Gallo, "Position estimation in outdoor environments using pixel tracking and stereovision," *Proc. Int. Conf. Robot. Autom.*, 2000, pp. 3519–3524.

[9] M. Killpack, T. Deyle, C. Anderson, and C. C. Kemp, "Visual odometry and control for an omnidirectional mobile robot with a downward-facing camera," *Proc. Int. Conf. Intell. Robots Syst.*, 2010, pp. 139–146.

[10] N. Seegmiller and D. Wettergreen, "Optical flow odometry with robustness to self-shadowing," *Int. Conf. Intell. Robots Syst.*, 2011, pp. 613–618.

[11] A. Kelly, "Mobile robot localization from large scale appearance mosaics," *Int. J. Robot. Res.*, vol. 19, no. 11, pp. 1104–1125, 2000.

[12] S. Se, D. Lowe, and J. Little, "Vision-based mobile robot localization and mapping using scale-invariant features," *Proc. Int. Conf. Robot. Autom.*, 2001, pp. 2051–2058.

[13] X. Song, L. D. Seneviratne, K. Althoefer, "A Kalman filter-integrated optical flow method for velocity sensing of mobile robots," *Trans. Mechatron.*, vol. 16, no. 3, pp. 551–563, 2011.

[14] C. Tomasi and T. Kanade, "Detection and tracking of point features," *Carnegie Mellon University Technical Report*, CMU-CS-91-132, 1991.

[15] H. Bay, T. Tuytelaars, and L. V. Gool, "SURF: Speed up robust features," *Ninth Eur. Conf. Comput. Vis.*, 2006, pp. 404–417.

[16] T. Lindeberg, "Detecting salient blob-like image structures and their scales with a scale-space primal sketch: A method for focus-of-attention," *Int. J. Comput. Vis.*, vol. 11, no. 3, pp. 283–318, 1993.

[17] I. Nagai and Y. Tanaka, "Mobile robot with floor tracking device for localization and control," *J. Robot. Mechatron.*, vol. 19, no. 1, pp. 34–41, 2007.

[18] K. Nagatani, A. Ikeda, G. Ishigami, K. Yoshida, and I. Nagai, "Development of a visual odometry system for a wheeled robot on loose soil using a telecentric camera," *Adv. Robot.*, vol. 24, no. 8-9, pp.1149–1167, 2010.