



HAL
open science

Temporal model based robot navigation using optical flow

Naoya Ohnishi, Yukiko Kenmochi, Atsushi Imiya

► **To cite this version:**

Naoya Ohnishi, Yukiko Kenmochi, Atsushi Imiya. Temporal model based robot navigation using optical flow. International conference on Computer Vision / Computer Graphics Collaboration Techniques and Applications (Mirage 2005), 2005, Rocquencourt, France. pp.35-44. hal-00622235

HAL Id: hal-00622235

<https://hal.science/hal-00622235>

Submitted on 11 Jan 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

TEMPORAL MODEL BASED ROBOT NAVIGATION USING OPTICAL FLOW

¹Naoya Ohnishi, ²Yukiko Kenmochi and ³Atsushi Imiya

¹ohnishi@graduate.chiba-u.jp

¹School of Science and Technology, Chiba University, Japan
Yayoi-cho 1-33, Inage-ku, Chiba 263-8522

²y.kenmochi@esiee.fr

²CNRS/ESIEE/Universite de Marne-la-Vallee

Laboratoire A2SI, ESIEE Cite Descartes, B.P.99 93162 Noisy-le-Grand Cedex France

³imiya@faculty.chiba-u.jp

³Institute of Media and Information Technology, Chiba University, Japan
Yayoi-cho 1-33, Inage-ku, Chiba 263-8522

ABSTRACT

In this paper, we aim to develop an algorithm for the navigation of an autonomous robot using optical flow observed through a vision system mounted on a mobile robot. Optical flow is used for generating a temporal-model of the robot motion. The temporal-model enables us to detect a planar area on which robot moves by matching with optical flow, and estimate a ego-motion of the mobile robot. If we obtain the planar area and a ego-motion, we can navigate the mobile robot autonomously. We show experiments for the detection of the planar area and estimation of the ego-motion of the robot using the image sequence observed through the mobile robot.

1. INTRODUCTION

Model-based method for object recognition provides a model-based method for obstacle detection of mobile robots. CAD-based method detects obstacles by matching geons [1] in a database [2]. Environment-map-based method detects obstacles using a environmental map without obstacles. The first method requires to propose geons detect-base. The second method requires a learning stage for the construction of environmental map in which the robot works. In this paper, we propose a temporal-model-based navigation strategy for autonomous mobile robots.

Temporal-model-based method detects the feasible region where robot can move using three successive images observed from mobile robot. First two images are used for the generation of a short term model, second two images are used for data. Matching these two features from successive three images. Therefore, temporal-model-based method dose not require neither model in a database nor a learning stage for the generation model.

For the detection of obstacles or planar areas, there are many methods using vision systems [3]. For example, the methods of objects recognition using edge detection

or landmarks observation are dependent on the environment around a robot. These methods involve difficulties, such as the need to predetermine landmarks, when applied to general environments. If a robot captures an image sequence of moving objects, the optical flow [4] [5] [6] [7], which is the flow of movement in the scene, is obtained for the fundamental features to generate environment information around the mobile robot. Additionally, optical flow is considered to be fundamental information for obstacle detection in the context of biological data processing [8]. Therefore, the use of optical flow is an appropriate method for the detection of obstacles from the viewpoint of the affinity between the robot and human beings.

The obstacle detection using optical flow is proposed in [9] [10]. Enkelmann [9] proposed an obstacle-detection method using the model vectors from motion parameters. Santos-Victor and Sandini [10] also proposed an obstacle-detection algorithm for a mobile robot using the inverse projection of optical flow to ground floor. Those methods require the learning stage for the construction of the model of a robot motion and a environment. Figure 1 shows the algorithm for the model-based method. In this method in the learning stage, a model for the environment without obstacles is generated.

Our algorithm generates a short term model of the flow field using first two images from successive three images. Furthermore, second two images from successive three images generates features for obstacle detection. Therefore, matching the short term model and features, the algorithm detects a dominant plane as the feasible region on which the mobile robot moves. Figure 2 shows the algorithm for the temporal-model-based method. This method generates temporal-models.

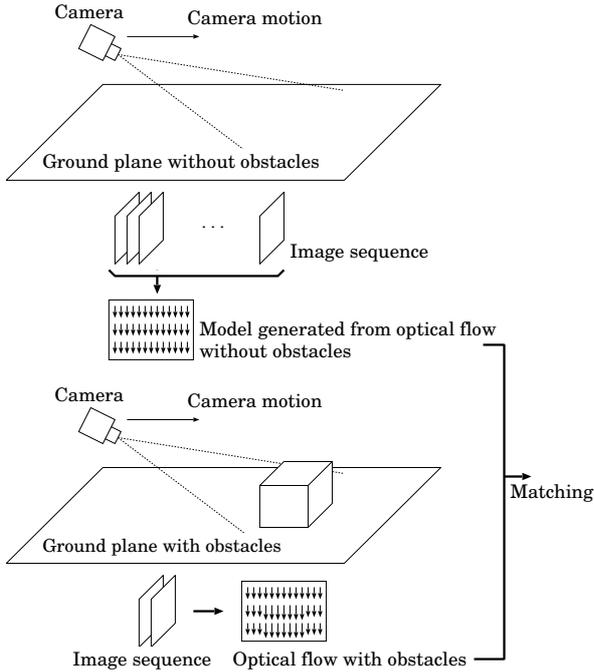


Figure 1: Model-based navigation using optical flow. The upper part is the learning stage for the construction of the model of a robot motion and a environment. During a navigation, obstacles are detected by matching optical flow of this model and observed ones.

2. DOMINANT PLANE DETECTION FROM IMAGE SEQUENCE

In this section, we develop an algorithm for the detection of the dominant plane from the image sequence observed by a camera mounted on a mobile robot. When the mobile robot moves over the ground plane, we obtain successive images which include a dominant plane area and obstacles. The dominant plane is adapted as the model of own navigation strategy. In own method, from a triplet of successive images, the model is estimated and renewed temporally.

First, we show that the corresponding points in a pair of successive images, which are a projection of the dominant plane in a space, are connected by an affine transformation. Therefore, the computed optical flow from the successive images describes the motion of the dominant plane and obstacles on the basis of camera displacement. Since the corresponding points in dominant plane motion are combined by an affine transformation, we can compute the affine coefficients in Eq.(1) from the optical flow on the dominant plane. Once the affine coefficients are computed, we can estimate the dominant plane motion in the image from the affine coefficients. The dominant plane

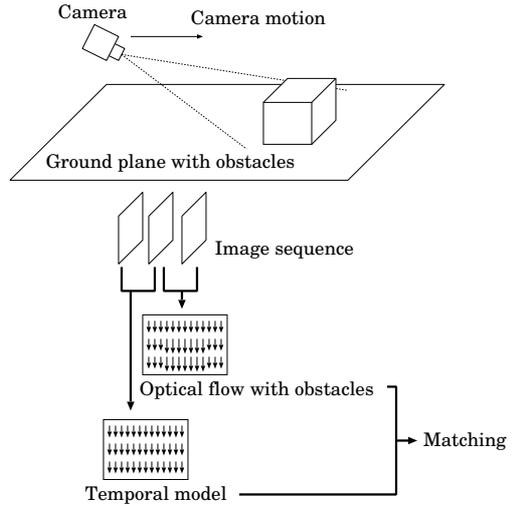


Figure 2: Temporal-model-based navigation using optical flow. This method detects obstacles without a model. The first two images from triplet of images are used for the construction of temporal-models.

motion is described by a *planar flow field*, as shown in Fig.3. The difference between the estimated planar flow field and the computed optical flow field enables us to detect the dominant plane area on the image by matching the flow vectors on the image planes.

2.1. Approximation of dominant plane motion by affine transformation

We show that the corresponding points in a pair of successive images, which are a projection of the dominant plane in a space, are connected by an affine transformation. Therefore, the corresponding points (u, v) and (u', v') on the dominant plane are expressed by

$$\begin{pmatrix} u' \\ v' \end{pmatrix} = \begin{pmatrix} a & b \\ d & e \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix} + \begin{pmatrix} c \\ f \end{pmatrix}, \quad (1)$$

when the camera displacement is small. This means that homography between the two images of a planar surface is approximated by an affine transformation when the camera displacement is small.

Setting \mathbf{H} to be the 3×3 matrix [11], the homography between the two images of a planar surface is expressed as

$$\mathbf{p} = \mathbf{H}\mathbf{p}', \quad (2)$$

where $\mathbf{p} = (u, v, 1)^\top$ and $\mathbf{p}' = (u', v', 1)^\top$ are the corresponding points on the two images. The matrix \mathbf{H} is expressed as

$$\mathbf{H} = \mathbf{K}(\mathbf{R} + \mathbf{t}\mathbf{n}^\top)\mathbf{K}^{-1}, \quad (3)$$

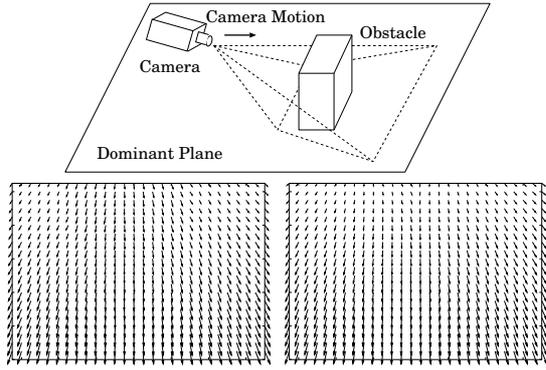


Figure 3: Planar flow of the image sequence. top: Example of camera displacement and the environment. left-bottom: Computed optical flow. right-bottom: Estimated planar flow. In a top-middle area, where exists the obstacle, the length of optical flow is bigger than planar flow.

where \mathbf{K} , \mathbf{R} , \mathbf{t} , and \mathbf{n} are the projection matrix, the rotation matrix, the translation vector, and the plane normal of the planar surface, respectively. The matrixes \mathbf{K} and \mathbf{K}^{-1} are affine transformations since these matrixes are the projection matrix of the pinhole camera. Assuming that the camera displacement is small, the matrix \mathbf{R} and the matrix $\mathbf{t}\mathbf{n}^\top$ are approximated by an affine transformation. This approximation is illustrated in Fig.4. These geometrical and mathematical assumptions are valid when the camera is mounted on a robot moving over the dominant plane. Then above assumptions allow us to describe the relationship between (u', v') and (u, v) as the affine transformation. Therefore, setting a, b, c, d, e, f to be affine coefficients, Eq.(2) can be expressed as Eq.(1). In the next section, we develop an algorithm for the estimation of these six parameters from a sequence of images.

2.2. Planar flow estimation

Using a pair of successive images from a sequence of images obtained from the camera during robot motion, we compute the optical flow field (\dot{u}, \dot{v}) . Since optical flow is the correspondence of dense points between an image pair, Eq.(1) can be applied to each point on the dominant plane. Let (u, v) and (u', v') be the points of the corresponding image coordinates in the two successive images. Point (u', v') can be calculated by adding the flow vector (\dot{u}, \dot{v}) to (u, v) , if (u, v) and (u', v') are a pair of corresponding points. Thus, by setting (\dot{u}, \dot{v}) to be the optical flow at point (u, v) in the image coordinate system, as shown in Fig.5, we have the relation

$$\begin{pmatrix} u' \\ v' \end{pmatrix} = \begin{pmatrix} u \\ v \end{pmatrix} + \begin{pmatrix} \dot{u} \\ \dot{v} \end{pmatrix}. \quad (4)$$

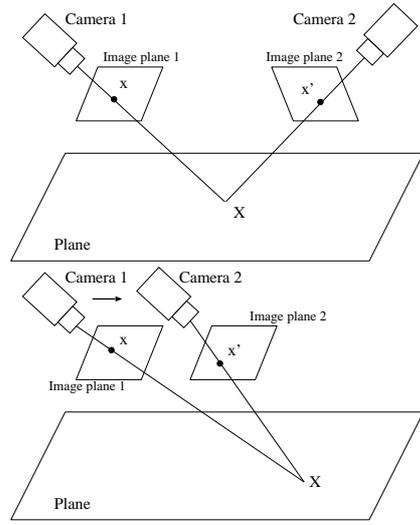


Figure 4: Approximation of dominant plane motion by affine transformation. top: Homography is expressed by $\hat{x}' = \mathbf{H}\hat{x}$. bottom: Affine transformation is expressed by $\hat{x}' = \mathbf{A}\hat{x}$.

If we obtain (u, v) and (u', v') , we can compute the affine coefficients in Eq.(1). If three points are non-collinear, Eq.(1) has a unique solution. Since the dominant plane occupies the largest domain in the image, we select three points randomly to compute the affine coefficients in the successive pair of images. After we compute the affine coefficients, using Eq.(1) again, we can compute the motion of the image sequence of the dominant plane, that is, the collection of corresponding points

$$\begin{pmatrix} \hat{u} \\ \hat{v} \end{pmatrix} = \begin{pmatrix} u' \\ v' \end{pmatrix} - \begin{pmatrix} u \\ v \end{pmatrix} \quad (5)$$

in the image sequence can be regarded as motion flow of the dominant plane on the basis of camera displacement

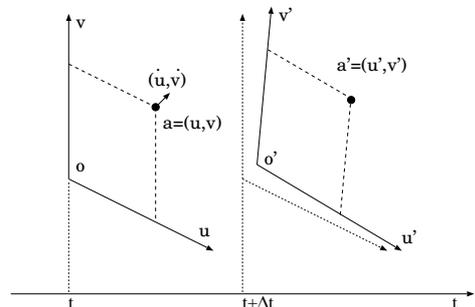


Figure 5: Optical flow generated by affine transformation. Optical flow (\dot{u}, \dot{v}) on the dominant plane is described as an affine transformation.

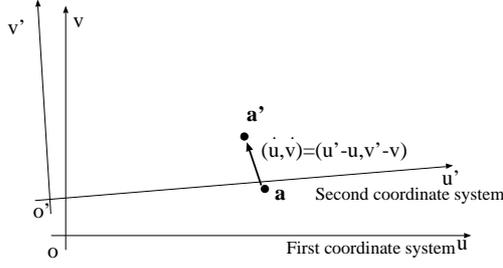


Figure 6: Optical fbw computed from corresponding points in two successive images. Optical fbw is described as the correspondence of the same points in the two successive images. Therefore, point (u', v') is $(u + \hat{u}, v + \hat{v})$ and planar fbw (\hat{u}, \hat{v}) is $(u' - u, v' - v)$.

as shown in Fig.6. We call this fbw *planar flow* (\hat{u}, \hat{v}) . This planar fbw is used as temporal-model for dominant plane detection.

2.3. Dominant plane detection

Next, we compute the dominant plane area using the estimated planar fbw and the computed optical fbw. Setting ε to be the tolerance of the difference between the optical fbw vector and the planar fbw vector, if

$$\left| \begin{pmatrix} \dot{u}_t \\ \dot{v}_t \end{pmatrix} - \begin{pmatrix} \hat{u}_t \\ \hat{v}_t \end{pmatrix} \right| < \varepsilon \quad (6)$$

is satisfied, we accept point (u_t, v_t) as a point in the dominant plane.

In the case that at least one point on the obstacle area in the image is selected, the estimated planar fbw is no longer the dominant plane motion. Therefore, the detected dominant plane area is very small. Since the dominant plane occupies the largest domain in the image, in such cases, it becomes evident that the selection of points is incorrect. In those cases, we consequently select another three points randomly. Figure 7 shows examples of each case.

Once we have detected the dominant plane in a certain frame of the image sequence, the planar fbw of subsequent images can be estimated robustly using the least-squares method, because dense optical fbws are used for the estimation of affine coefficients. Assuming that the robot displacement is small, the dominant plane of the successive images changes negligibly. Therefore, using the optical fbw on the estimated dominant plane in the previous image, we estimate the affine coefficients using the least-squares method, as shown in Fig.8. Setting (u_i, v_i) and (u'_i, v'_i) ($0 \leq i \leq n$) to be corresponding points, the

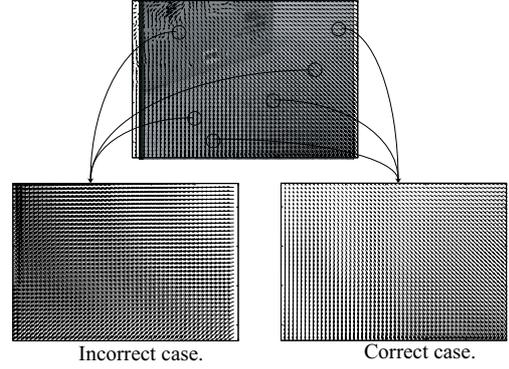


Figure 7: Examples of random sampling. Bottom-left is incorrect case, since the point is selected on the obstacle area. Select another points randomly. Bottom-right is correct case. This planar fbw is equal to optical fbw in 50% or more of area.

mean-squared errors E_u and E_v associated with Eq.(1) are

$$E_u = \sum_{i=1}^n \{u'_i - (au_i + bv_i + c)\}^2, \quad (7)$$

$$E_v = \sum_{i=1}^n \{v'_i - (du_i + ev_i + f)\}^2, \quad (8)$$

where n is the number of points used for estimation. Therefore, we can compute affine coefficients which minimize errors E_u and E_v . This algorithm generates planar fbw as temporal-model at subsequent frames

2.4. Procedure for dominant plane detection

Our algorithm is summarized as follows.

1. Compute optical fbw (\dot{u}, \dot{v}) from two successive images.
2. Compute affine coefficients in Eq.(1) by random selection of three points.
3. Estimate planar fbw (\hat{u}, \hat{v}) from affine coefficients.
4. Match the computed optical fbw (\dot{u}, \dot{v}) and estimated planar fbw (\hat{u}, \hat{v}) using Eq.(6).
5. Detect the dominant plane. If the dominant plane occupies less than half of the image area, then return to step(2).

Figure 9 shows the procedure of dominant plane detection from the image sequence.

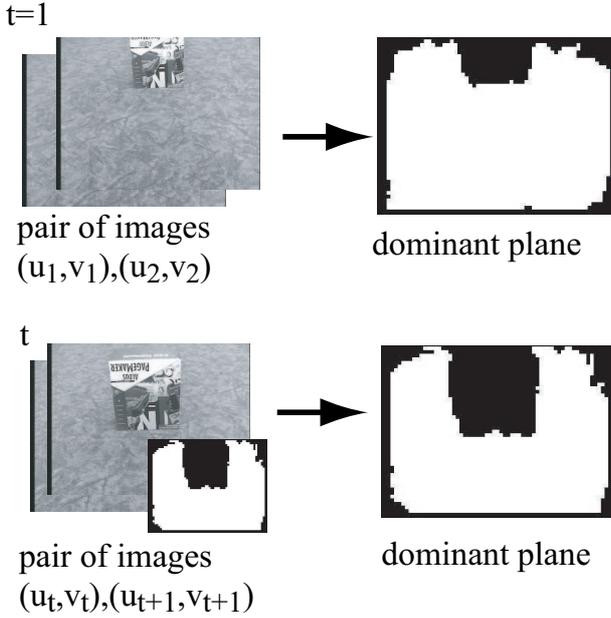


Figure 8: For the first frame, our algorithm uses a pair of images for the detection of the dominant plane. For subsequent frames, the dominant plane in the previous frame is used for the application of the least-squares method.

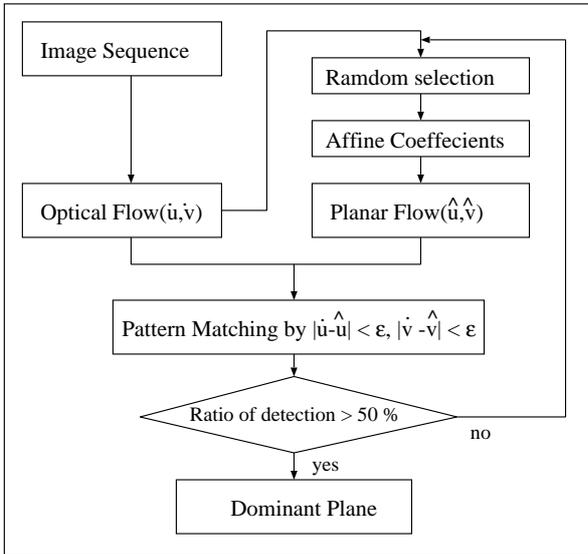


Figure 9: Procedure for dominant plane detection. (1) Compute optical flow (\hat{u}, \hat{v}) from two successive images. (2) Compute affine coefficients in Eq.(1) by random selection of three points. (3) Estimate planar flow (\hat{u}, \hat{v}) from affine coefficients. (4) Match the computed optical flow (\hat{u}, \hat{v}) and estimated planar flow (\hat{u}, \hat{v}) using Eq.(6). (5) Detect the dominant plane. If the dominant plane occupies less than half of the image area, then return to step(2).

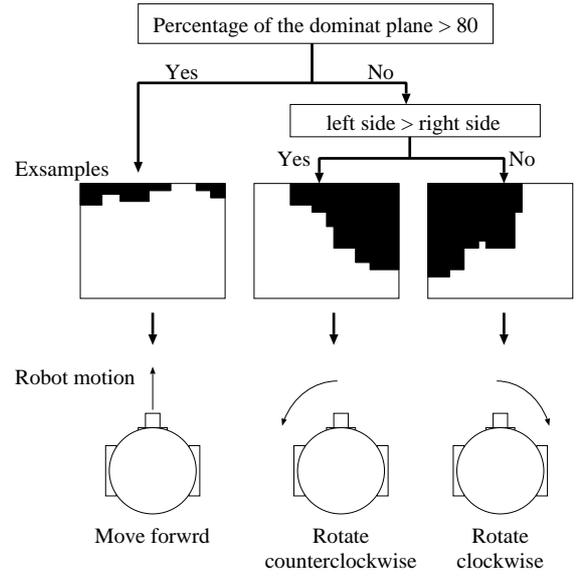


Figure 10: Strategy of a mobile robot. left: Robot moves forward. middle: Robot rotates counterclockwise. right: Robot rotates clockwise.

2.5. Navigation using the dominant plane

We describe an algorithm for navigation of a mobile robot using the dominant plane. The percentages of the dominant plane in the image determine for the robot a strategy of a robot motion. If the percentage of the dominant plane in the image is greater than 80% and the dominant plane of the left side of the image is greater than that of right side, the robot moves forward. If the percentage of the dominant plane in the image is less than 80% and the dominant plane of the left side of the image is greater than that of right side, the robot rotates counterclockwise to avoid collision of obstacles. If the percentage of the dominant plane in the image is less than 80% and the dominant plane of the right side of the image is greater than that of left side, the robot rotates clockwise. These algorithm are summarized in Fig. 10.

3. EXPERIMENT

In this section, we evaluate the performance of our method for steps (2) and (5) of the algorithm listed in Fig.9, since the estimation of affine coefficients and dominant plane detection are essential to the resolution of our problem. In Step (2), the parameters are estimated from the image sequence. In Step (5), the area of the dominant plane in the images is evaluated. These experiments are described in subsections 4.1 and 4.2, respectively. For the computation of optical flow, we use the Lucas-Kanade method

with pyramids [12]. The tolerance for the matching of flow vectors in Eq.(6) is set to be $\varepsilon = 0.2$, which was determined experimentally.

To evaluate the accuracy of the dominant plane detection, we define the error ratio $E(t)$. Setting $D(u, v, t)$ and $M(u, v, t)$ to be the dominant planes on the image coordinate system (u, v) at frame t detected from the optical flow and detected manually, respectively, and setting

$$D(u, v, t) = \begin{cases} 0 & \text{for dominant plane} \\ 1 & \text{for other area} \end{cases}, \quad (9)$$

$$M(u, v, t) = \begin{cases} 0 & \text{for dominant plane} \\ 1 & \text{for other area} \end{cases}, \quad (10)$$

we evaluate the ratio

$$E(t) = 100 \frac{\sum_{u,v} |M(u, v, t) - D(u, v, t)|}{\sum_{u,v} 1} \quad (11)$$

as a function of time t .

3.1. The error rate of affine coefficients with uniform motion

First, we show the validity of the affine transformation in two successive images. From Eq.(1), we define matrix \mathbf{A}_t as

$$\mathbf{A}_t = \begin{pmatrix} a & b & c \\ d & e & f \\ 0 & 0 & 1 \end{pmatrix}, \quad (12)$$

where t is the frame number in the image sequence. Assuming that the velocity of camera displacement is constant in the two successive images, the relation $\mathbf{A}_{t+1} \simeq \mathbf{A}_t$ is satisfied. Therefore, we define the error of affine coefficients as

$$e(t) = |\mathbf{A}_{t+1} - \mathbf{A}_t|^2. \quad (13)$$

In this experiment, we use the marbled-block image sequence [13], since the velocity of camera displacement should be constant. Figure 11 shows the error of affine coefficients. Figs.12 and 13 show the image sequence of the detected dominant plane and the error ratio of the dominant plane, respectively.

Figure 11 shows that $|\mathbf{A}_{t+1} - \mathbf{A}_t|^2$ approaches zero with time. Therefore, the affine coefficients are constant when the camera displacement is constant. We show the validity of the affine transformation in two successive images.

3.2. Detection of dominant plane

We show next the validity of the random selection of three points for the computation of affine coefficients. In this experiment, we use image sequences of forward motion

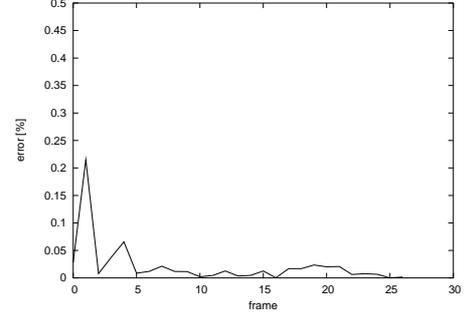


Figure 11: Error of affine coefficients. The vertical axis is the error of affine coefficients $e(t)$, as given by Eq.(13). The horizontal axis is the frame number of the image sequence.

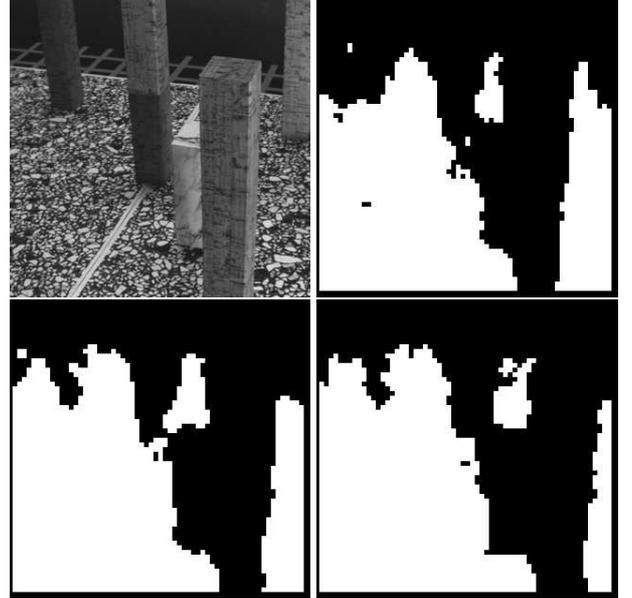


Figure 12: Detected dominant plane in frames 0, 10 and 20. The original image(top-left), and detected dominant plane in frames 0(top-right), 10(bottom-left) and 20(bottom-right). In the images of the dominant plane, the white areas are the dominant planes and the black areas are obstacles.

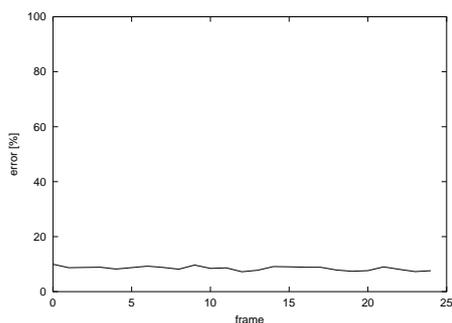


Figure 13: Error ratio of dominant plane detection using the image sequence shown in Fig.12. The vertical axis is the error ratio $E(t)$ expressed by Eq.(11). The horizontal axis is the frame number of the image sequence.

Table 1: Error ratios of dominant plane detection in Fig.14. In the second row, F90, F80 and F70 are the image sequences of forward motion in which the percentages of the dominant plane in the image are 90%, 80% and 70%, respectively. Rotation gives the image sequence of rotational motion. This image sequence is shown in the first row of Fig.14.

frame	error ratio $E(t)$ [%]			
	F90	F80	F70	Rotation
0	23.22	24.01	20.33	19.37
1	14.40	12.99	12.29	14.50
2	10.91	11.83	10.45	11.09
3	10.54	11.08	10.55	11.29

and rotational motion, because the motion of a mobile robot mainly consists of these motions. For forward motion, we prepared image sequences in which the percentages of the dominant plane are 90%, 80% and 70%, since we assume that the dominant plane occupies more than half of an image. These percentages of the dominant plane were computed using $M(u, v, t)$ in Eq.(10). For rotational motion, we use the rotating blocks image sequence [14].

The result of dominant plane detection is shown in Fig.14. The first row is the original image, and from the second row to the fifth row, we show image sequences of the dominant plane in the first, second, third and fourth frames, respectively. In the three images of the dominant plane, the white areas are the dominant planes and the black areas are the obstacles. Table 1 lists error ratios evaluated with Eq.(11).

Figure 14 and Tab.1 show that the validity of regions detected as the dominant plane increases with time.

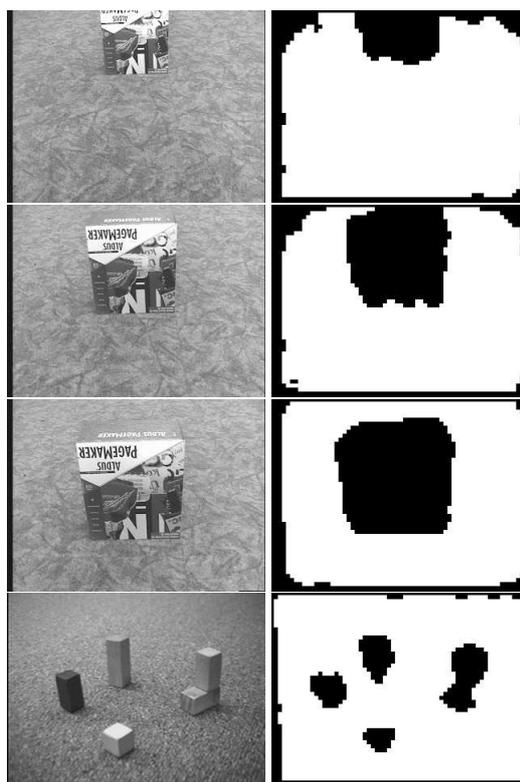


Figure 14: Detected dominant plane. The first and second column are the original image and the dominant plane, respectively.

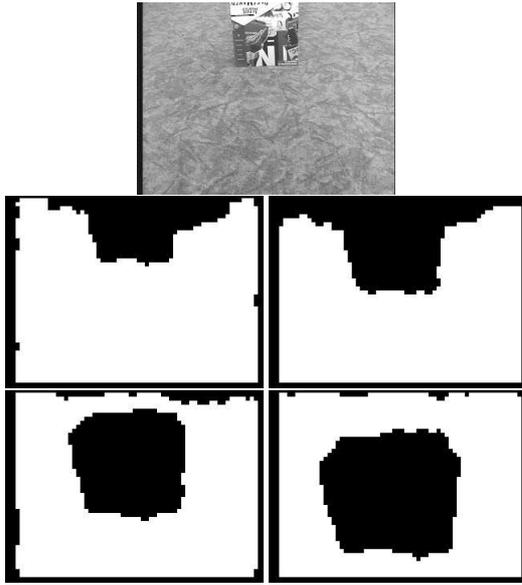


Figure 15: Detected dominant plane at the 50, 100, 150 and 200 frames.

3.3. Dominant plane detection in the long sequence

We present the dominant plane detection in a long image sequence for the navigation and path planning of the mobile robot. Figure 15 and Fig.16 show the results for the dominant plane detection and the error ratio, respectively. As evident from Fig.16, even if the error ratio increases in a sequence, the error ratio becomes small with a subsequent frame. Therefore, our method detects the dominant plane without accumulation of the error ratio. Our algorithm runs on a 2.0 GHz Pentium4 for 320×240 images and takes 0.25 seconds per frame for the detection of the dominant plane.

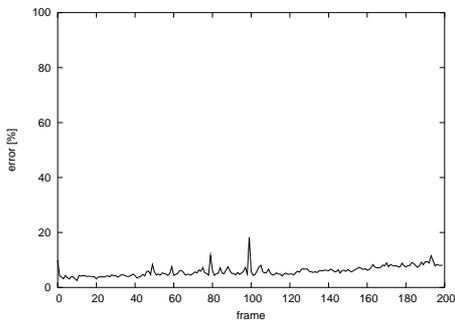


Figure 16: Error ratio $E(t)$. The vertical axis is the error ratio $E(t)$ in Eq.(11). The horizontal axis is the frame number of the image sequence.



Figure 17: The left column shows the images of the experimental environment, the middle column shows the images observed by the mobile robot, and the right column shows the dominant planes computed by our algorithm. The top, middle, and bottom rows are captured at frame 16, 68, and 100, respectively.

3.4. Autonomous navigation of the mobile robot

Using the algorithm of Section 3.4., we experiment for the obstacle avoidance of the mobile robot. Figure 17 shows the result of the experiment. In the top row of Fig.17, at the 16 frame, the robot rotates counterclockwise since the robot detects the obstacle in the right side of the image. In the middle row of Fig.17, at the 68 frame, the robot rotates clockwise since the robot detects the obstacle in the left side of the image. In the bottom row of Fig.17, at the 100 frame, the robot moves forward since the percentage of the dominant plane in the image is greater than 80%. Figure 18 shows the top view of the experimental environment. Figure 18 is generated by projecting dominant planes in the image plane onto the ground plane. Figure 18 is generated by projecting dominant planes in the image plane onto the ground plane. The trajectory of the mobile robot is also obtained by projecting the planar fbw in the image plane onto the ground plane.

4. CONCLUSION

We developed an algorithm for the navigation of a mobile robot by temporal-model-based planar area detection using optical fbw. The algorithm allows the positions of obstacles and the location of the robot to be detected using optical fbw using optical fbw computed from the image sequence observed through the camera mounted on the mobile robot. Furthermore, we showed that corresponding points on dominant planes in a pair of succes-

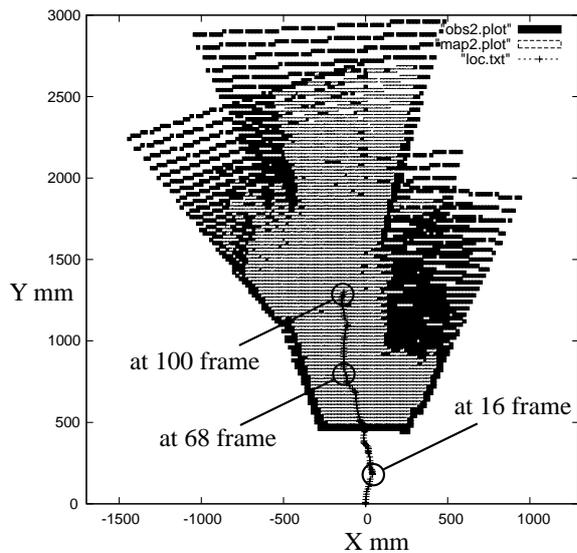


Figure 18: Environmental map computed from dominant plane. White and black area are planar area and obstacles, respectively. The line started at the origin is the trajectory of the location of the mobile robot. We set the vertical axis to coincide with the direction of the robot at initial location, and the horizontal axis to be orthogonal the vertical axis.

sive images are combined by affine transformation. Using this idea, if we compute the affine coefficients which relate the corresponding points in two successive images, we can easily obtain a dense planar fbw which expresses a camera motion. This property of the points in a dominant plane allows us to design an algorithm which enable the dominant plane to be detected by simple pattern matching of the fbw vectors in a series of dominant planes. Although model-based approach to dominant plane detection has been proposed in [9], our method is a non-model-based approach. In addition, our algorithm allows the dominant plane to be detected without camera calibration, since our algorithm uses short-term model of the robot motion. Results of experiments using real image sequences confirmed that the dominant plane can be detected accurately. These experiments allow the application of our method to the navigation and path planning of a mobile robot with a vision system.

5. REFERENCES

- [1] I. Biederman, "Human image understanding: Recent research and a theory," *Computer Vision, Graphics, and Image Processing*, vol. 32, pp. 29–73, 1985.
- [2] S.J. Dickinson and D. Metaxas, "Decoupling recognition and localization in cad-based vision," *CAD-Based Vision Workshop*, pp. 246–257, 1994.
- [3] N.D. Guilherme and C.K. Avinash, "Vision for mobile robot navigation: A survey," *IEEE Trans. on PAMI*, vol. 24, pp. 237–267, 2002.
- [4] J.L. Barron, D.J. Fleet, and S.S. Beauchemin, "Performance of optical fbw techniques," *International Journal of Computer Vision*, vol. 12, pp. 43–77, 1994.
- [5] B.K.P. Horn and B.G. Schunck, "Determining optical fbw," *Artificial Intelligence*, vol. 17, pp. 185–203, 1981.
- [6] B. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proc. of 7th IJCAI*, 1981, pp. 674–679.
- [7] H.-H. Nagel and W. Enkelmann, "An investigation of smoothness constraint for the estimation of displacement vector fields from image sequences," *IEEE Trans. on PAMI*, vol. 8, pp. 565–593, 1986.
- [8] H.A. Mallot, H.H. Bulthoff, J.J. Little, and S. Bohrer, "Inverse perspective mapping simplifies optical fbw computation and obstacle detection," *Biological Cybernetics*, vol. 64, pp. 177–185, 1991.
- [9] W. Enkelmann, "Obstacle detection by evaluation of optical fbw fields from image sequences," *Image and Vision Computing*, vol. 9, pp. 160–168, 1991.
- [10] J. Santos-Victor and G. Sandini, "Uncalibrated obstacle detection using normal fbw," *Machine Vision and Applications*, vol. 9, pp. 130–137, 1996.
- [11] A. Hartley and A. Zisserman, "Multiple view geometry in computer vision," 2000.
- [12] Bouguet J.-Y., "Pyramidal implementation of the lucas kanade feature tracker description of the algorithm," *Microprocessor Research Labs, OpenCV Documents*, 1999.
- [13] "Marbled-block sequence: recorded and first evaluated by otte and nagel," *website: http://i21www.ira.uka.de/image_sequences/*.
- [14] "Rotating blocks: Otago optical fbw evaluation sequences," *website: <http://www.cs.otago.ac.nz/research/vision/Research/>*.