Estimation of Mobile Robot Ego-Motion and Obstacle Depth Detection by Using Optical Flow

Xuebing Wang, Kenji Ban, Kazuo Ishii
Graduate School of Life Science and Systems Engineering
Kyushu Institute of Technology
Kitakyusyu, Japan
wang-xuebing@edu.brain.kyutech.ac.jp

Abstract—Estimation of ego-motion and obstacle depth detection plays an important role in environment recognition for navigation of mobile robots. In this paper obstacle perception is performed by using pattern matching between the planar flow and the normal optical flow observed by means of a monocular vision system installed on a mobile robot. The obstacle depth and robot velocity then can be estimated through the coordinate transformation between the image coordinate and the robot coordinate. An experiment is also performed under several different environment settings and shows a good performance for estimation of velocity and the obstacle depth for a mobile robot.

Keywords—ego-motion, depth detection, optical flow

I. INTRODUCTION

The autonomous mobile robot that could work under the living environment is requested and is suggested that they could become common in the future. In order to realize this purpose the environment recognition is indispensable for an autonomous robot such as the perception of obstacles, ego-motion and depth detection etc. In this paper we aim to develop an algorithm for estimation of ego-motion of a mobile robot and obstacle depth detection by using optical flow. Obstacle perception used to estimate the ego-motion and depth is performed by using pattern matching between the planar flow and the optical flow observed by means of a monocular vision system installed on a mobile robot.

For the autonomous navigation of mobile robots, usually vision sensors, infrared sensors or laser range sensors are used in environment recognition such as obstacle perception and ego-motion estimation. There are many research topics about the obstacle perception using infrared radiation sensor or ultrasonic sensor [1]. These sensors can provide the environment depth information directly. But using this method only the obstacle information can be obtained, and the other information for robot need to fuse another sensing system [2], also it is difficult to retrieve the large range simultaneously. For vision system, there are many methods of recognizing the environment such as analysis from information on the landmarks [3] or extract from edge detection by using monocular vision [4], which are depend on surrounding environment around a robot, and difficult to apply in general environments. Also the research that uses optical flow is done as one of other obstacle recognition techniques. A simple approach to obstacle perception is to examine the optical flow in the left and right halves of the visual field separately, and then to turn toward the direction in which the optical flow is smaller [5]. The motion segmentation of objects and background by using optical flow is described in [6]. In this paper we develop the detection of static obstacles from an image sequence observed from a mobile robot. As a previous work an algorithm about estimation of dominant plane is proposed in [7]. We aim to extend this work to estimation of robot’s ego motion and obstacle depth by using a mobile robot under several environments. The experimental evaluation using a real-time monocular vision system shows that this method is practicable for navigation of autonomous mobile robot.

In our work optical flow [8] is calculated by the Lucas-Kanade method with pyramids [9] [10]. Next, in optical flow field three random points are selected to calculate affine coefficient in order to obtain planar flow. Then for segmentation of background and objects, pattern matching is performed between optical flow and planar flow. Through the coordinate transformation between the image coordinate and the robot coordinate we can get obstacle depth from the optical flow in obstacle area. And the relation coefficient between planar flow and robot velocity is also estimated by using experimental planar data.

II. ALGORITHM

In this section, we develop an algorithm for the estimation of mobile robot ego-motion and obstacle depth detection by using optical flow which is obtained from a monocular vision system mounted on a mobile robot.

A. Optical flow estimation

Optical flow is motion vectors in the observer’s visual image, and also described by means of spatio-temporal distributions, or fields, of motion vectors [11]. There are several methods for estimation of optical flow based on partial derivatives of the image signal, such as Lucas–Kanade method, and Horn–Schunck method etc. In this paper the Lucas–Kanade method with pyramid is used in respect that for a big change of motion patterns in the visual image this method can solve the aperture problem and save processing time.

As shown in fig.1, the optical flow can be obtained through two successive images due to camera motion. The estimation equation of optical flow shows as (1),
The function of optical flow, start point \((u, v)\) described in the section of optical flow estimation. The brightness of present image. So an affine transformation as (3).

The corresponding points in a pair of successive images which lie on a projection of a plane in a space are connected by an affine transformation as (3).

The relation of optical flow, start point \((u, v)\) and end point \((u', v')\) of optical flow shows in fig.2, and then we have the relation

\[
\begin{align*}
\begin{pmatrix}
  u' \\
  v'
\end{pmatrix}
  &=
\begin{pmatrix}
  u \\
  v
\end{pmatrix}
  +
\begin{pmatrix}
  u' \\
  v'
\end{pmatrix}
\end{align*}
\]

If we can obtain the affine coefficients through (4), the planar flow \((u, v)\) can be calculated by (5).

\[
\begin{align*}
\begin{pmatrix}
  u \\
  v
\end{pmatrix}
  &=
\begin{pmatrix}
  a & b \\
  d & e
\end{pmatrix}
\begin{pmatrix}
  u \\
  v
\end{pmatrix}
  +
\begin{pmatrix}
  c \\
  f
\end{pmatrix}
\end{align*}
\]

C. Segmentation of background and obstacles

Next, we use the estimated planar flow and the computed optical flow to segment the background and obstacles. Setting \(c\) to be the tolerance of the difference between the optical flow vector and the planar flow vector, if

\[
\left|\begin{pmatrix}
  u' \\
  v'
\end{pmatrix}
  -
\begin{pmatrix}
  u \\
  v
\end{pmatrix}
\right| < \epsilon
\]

is satisfied, we accept point \((u, v)\) is in the background. In the case that we select at least one point on the obstacle area in the image, the estimated planar flow is no longer the background motion. Therefore, the detected background is very small. In such cases, it becomes evident that the selection of points is incorrect. In those cases, consequently, we select three points randomly again. In our work, we assume that the estimation is success if the detected background area is more than half of image. Otherwise, we need to select the random points again.

Till to segmentation of background and obstacles, the algorithm can be summarized as follows.

1. Compute optical flow \((\hat{u}, \hat{v})\) from two successive images.
2. Compute affine coefficients in (3) by random selection of three points.
3. Estimate planar flow \((\hat{u}, \hat{v})\) from affine coefficients.
4. Matching the computed optical flow and planar flow by using (6).
5. Segment the background and the obstacles. If the background area occupies less than half of the image, then go to step 2.

We test these steps by using real successive images and the result shows as Fig.3.
D. Ego-motion estimation

The definition of ego-motion is reconstruction of the observer’s own motion. In the psychophysical literature, three perceptions are most commonly studied, those of posture, heading and velocity, i.e. the sense of being transported [12]. In this paper, the main work is emphasized on the estimation of observer’s velocity.

The estimated planar flow denotes the background velocity in the image. Assume OY is the direction of movement of observer in the real world, according to the relation between camera coordinates and world coordinates in YZ plane as shown in Fig.4 can give us the coordinate transformation equation as (7).

\[ y = 0.0067x \]

Here, \( f \) is focal length of camera, \( H \) the height of camera from ground, and \( \theta \) the angel of view. Then we can get the connection equation (8) between camera velocity and planar flow.

\[ V = \frac{15vH}{f \cos^2 \theta + v \sin \theta \cos \theta} \]

Here we set \( H \) to 0.22m, \( f \) to 0.187m and \( \theta \) to 60° same with experimental set-up. Then the relation of camera velocity and planar flow can be approximated to a line through origin of coordinate. So (8) also can be expressed as (9).

\[ V = \frac{1}{n} \sum_{i=1}^{n} \sqrt{u_i^2 + v_i^2} \times \alpha \]

Here, \( \alpha \) is the proportionality coefficient which can be estimated using experimental successive images data described in the section of experiment.

E. Obstacle depth detection

In this work obstacle depth detection can be summarized in Fig. 6. There are 2 steps in this procedure. In step 1, segmentation of background and obstacle area is performed by using segmentation algorithm described in previous section. In step 2, we extract the obstacle area using Blob Extraction Library in OpenCV and calculate the obstacle coordinate point near to the observer in the image. Then in the end these coordinates are converted to the world coordinates as the depth of obstacles in the real world by using coordinate transformation equation (7) and (10).
III. EXPERIMENT

An experiment is performed by using a monocular vision system mounted on a 6-wheeled mobile robot named “Zaurus” in our lab. The aim of this experiment is to make a verification of algorithm in real-time. In order to obtain a robust ego-motion and obstacle depth estimation system the experiment is also performed in several environments by changing the background pattern, the height of camera and the velocity of the mobile robot.

A. Experimental set-up

The experiment diagrammatic illustration including camera, obstacle, mobile robot and robot coordinate system arrangement shows in Fig.7. The 6-wheeled mobile robot “Zaurus” can give us a lasting approximate constant velocity and less vibration during the experiment so that the error from optical flow estimation can be decreased. A commercially available web-camera with a resolution of 320*240 pixels is used and the length of focus is 187.5mm. The arrangement of camera’s height and elevation shows in Fig.8. And the floor patterns used in the experiment show in Fig.9. Barcode patterns are used to detect the background flow easily. Marble floor and carpet floor are common floor planes in our teaching building.

B. Robot ego-motion experimental result

As described in algorithm section, estimation of robot ego-motion needs average planar flow. A procedure of getting planar flow in experiment shows in Fig.10. First, get the 20*30 planar flows using optical flow. Then average 100 frames’ planar flow by using 3*3 flows in the center of the image. Finally, the proportionality coefficient, can be calculated using this average value and the robot velocity already known. The result shows in Tab.1. In coordinate between planar flow and robot velocity, the proportionality coefficient can be linearly-approximated by using linear least-squares method and the result shows in Fig.11. In order to evaluate the ego-motion estimation system, the experiments

\[ X_R(u,v) = \frac{H(u - u_0)}{f \cos \theta + (u - u_0) \sin \theta} \] (10)
are also performed by using calculated proportionality coefficient under several different robot velocities and background patterns. The result shows in Tab. 2 and error ratio in different environment is shown in Tab. 3. Robot velocity estimation according to the different patterns on the floor gives us a scattering result. Especially on the marble floor there is a big error by three major reasons. First, there has some reflection of light on the ground, so the optical flow estimation is misrecognition in the successive images sometimes. Second reason is the fewer features on the marble ground, which has led to be unrecognizable for optical flow estimation because optical flow is the distribution of apparent of movement of brightness patterns in an image. The last one is also the most important reason that the image processing speed is a little bit slow for on-line experiments as only 1 or 2 frames per second, which means the features in one frame move more than 1 pixel between this frame and next frame. So the aperture problem happened resulting in the big error of optical flow estimation. To solve this problem the image processing can be embedded in the system or use the high speed optical flow detection image sensor such as Alan stocker 2-D optical flow sensor.

### Table I. Coefficient Estimation Results

<table>
<thead>
<tr>
<th>Robot velocity</th>
<th>av_x (pixel)</th>
<th>av_y (pixel)</th>
<th>( \sqrt{\text{x}^2+\text{y}^2} ) (pixel)</th>
<th>Coefficient α</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.06 m/s</td>
<td>0.739056</td>
<td>10.2825</td>
<td>10.30903</td>
<td>0.00582</td>
</tr>
<tr>
<td>0.08 m/s</td>
<td>1.279331</td>
<td>13.35462</td>
<td>13.41576</td>
<td>0.005963</td>
</tr>
<tr>
<td>0.10 m/s</td>
<td>1.517734</td>
<td>15.62204</td>
<td>15.69559</td>
<td>0.006371</td>
</tr>
<tr>
<td>0.12 m/s</td>
<td>1.754797</td>
<td>17.77629</td>
<td>17.8627</td>
<td>0.006718</td>
</tr>
<tr>
<td>0.14 m/s</td>
<td>2.14845</td>
<td>21.56238</td>
<td>21.67583</td>
<td>0.006459</td>
</tr>
<tr>
<td>0.16 m/s</td>
<td>2.143345</td>
<td>23.4556</td>
<td>23.55332</td>
<td>0.006793</td>
</tr>
<tr>
<td>0.18 m/s</td>
<td>1.931728</td>
<td>25.98321</td>
<td>26.05492</td>
<td>0.006908</td>
</tr>
<tr>
<td>0.20 m/s</td>
<td>2.360653</td>
<td>28.99686</td>
<td>29.09279</td>
<td>0.006875</td>
</tr>
</tbody>
</table>

### Table II. Coefficient Estimation Results

<table>
<thead>
<tr>
<th>Robot velocity</th>
<th>Barcode (2mm)</th>
<th>Barcode (5mm)</th>
<th>Marble</th>
<th>Carpet</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.05 m/s</td>
<td>0.640706</td>
<td>9.006215</td>
<td>9.028976</td>
<td>0.060494</td>
</tr>
<tr>
<td>0.08 m/s</td>
<td>0.299386</td>
<td>9.490064</td>
<td>9.494785</td>
<td>0.063615</td>
</tr>
<tr>
<td>0.10 m/s</td>
<td>0.325644</td>
<td>5.198402</td>
<td>5.208592</td>
<td>0.034898</td>
</tr>
<tr>
<td>0.12 m/s</td>
<td>1.279331</td>
<td>13.35462</td>
<td>13.41576</td>
<td>0.089886</td>
</tr>
<tr>
<td>0.14 m/s</td>
<td>0.670209</td>
<td>14.46787</td>
<td>14.48338</td>
<td>0.097039</td>
</tr>
<tr>
<td>0.16 m/s</td>
<td>0.259735</td>
<td>7.315211</td>
<td>7.319821</td>
<td>0.049043</td>
</tr>
</tbody>
</table>

### Table III. Relative Error [%]

<table>
<thead>
<tr>
<th>Velocity</th>
<th>Barcode (2mm)</th>
<th>Barcode (5mm)</th>
<th>Marble</th>
<th>Carpet</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.05 m/s</td>
<td>21.0</td>
<td>27.2</td>
<td>30.2</td>
<td>4.0</td>
</tr>
<tr>
<td>0.08 m/s</td>
<td>12.3</td>
<td>21.3</td>
<td>38.7</td>
<td>13.4</td>
</tr>
</tbody>
</table>

### C. Obstacle depth estimation result

As described in algorithm of obstacle depth estimation, an experiment is performed by calculating the obstacle area’s nearest point to the observer. The segmentation of obstacle from the background, labeling of obstacle areas and the estimation of obstacle nearest points’ coordinate between successive images in on-line experiment are shown in Fig. 12. The distance between two objects used in experiment is near to 50 cm and the distance from robot start point to obstacles is near to 2 m. We set the height of camera to 0.76 m and background to the caret floor. Experimental results in robot velocity at 0.05 m/s and 0.08 m/s are shown in Fig. 13 and Fig. 14 separately. Condition 1 means the corresponding points’ number (feature points) in the image is set to small, and oppositely, the number is high in condition 2. This means that we adjust the balance between precision and processing cost.
And from the result we can know the estimation points is back-and-forth and also disconnection happened sometimes because the false recognition of optical flow. To solve this one corresponding point’s correspondence become important [13].

IV. CONCLUSIONS

In this paper the planar flow is estimated from optical flow observed by means of a monocular vision system installed on a mobile robot. Through the relation of planar flow and robot velocity, the coefficient is estimated by using experimental data and tested in real time experiment. For obstacle depth perception, the segmentation of background and obstacles is done. The position coordinate is extracted so that we can use the coordinate transformation between image coordinate and world coordinate to calculate the depth of obstacle.

In the future work, the system can be improved on the side of processing cost decreasing and improvement of optical flow estimation precision. For example, use the optical flow sensor.