VISUAL ODOMETRY BASED ON OMNIDIRECTIONAL IMAGES

GABRIEL LEIVAS; ERICKSON NASCIMENTO; MARIO CAMPOS*

*Departamento de Ciência da Computação - Universidade Federal de Minas Gerais (UFMG) - Minas Gerais - Brasil

Emails: gabriel@dcc.ufmg.br, erickson@dcc.ufmg.br, mario@dcc.ufmg.br

Abstract— We propose a method for visual odometry using optical flow with a single omnidirectional (catadioptric) camera. We show how omnidirectional images can be used to perform optical flow, discussing the basis of optical flow and some restrictions needed to it and how un warp these images. In special we describe how to un warp omnidirectional images to Bird’s eye view, that correspond to scaled orthographic views of the ground plane. Catadioptric images facilitate landmark based odometry, since landmarks remain visible for longer time, as opposed to a small field-of-view standard camera. Also, providing adequate representations to support visual odometry with a fast processing time. We perform tests to measure robustness and performance of our approach with analysis of the data acquired.

Keywords— Visual Odometry, Optical Flow, Catadioptric Images.

Resumo— Neste trabalho proponos um método de odometria visual, empregando fluxo óptico, com um imagens omnidirecionais. Mostramos que imagens omnidirecionais podem ser usadas para realizar fluxo óptico, discutindo fluxo óptico e algumas restrições necessárias para realizar tal tarefa, em especial como retificar imagens catadiópticas para um tipo específico de vista denominado Bird’s eye view que corresponde a uma escala de visão ortográfica do terreno. Imagens omnidirecionais facilitam a odometria baseada em marcos, uma vez que estes marcos se mantêm visíveis por mais tempo neste tipo de imagem, diferentemente de cameras padrões que apresentam um pequeno campo de visão. Além disso, este tipo de imagem fornece uma representação adequada para odometria visual com baixo custo computacional. Testes foram realizados para medir a robustez e desempenho da técnica desenvolvida com relação aos dados adquiridos.

Keywords— Odometria Visual, Fluxo óptico, Imagens Catadiópticas

1 INTRODUCTION

Odometry is the process of inferring position and orientation of a mobile robot. This can be done by measuring the wheel rotations with, for example, rotary encoders. This type of information is useful for wheeled vehicles, and it is generally not applied to robots with non-standard locomotion. However, classical odometry suffers from errors due to issues such as wheel size variation, slipping and terrain conditions. These errors also accumulates over time, making odometry estimation unreliable.

When only images are used to determine odometry, the process is called visual odometry. Compared to classical odometry, visual odometry does not suffer from motion constrains and can be used on any robot with at least one camera. Visual sensors are usually low cost devices suitable to navigation in several types of environments. Particularly, omnidirectional cameras are efficient sensors due to its inherent characteristic to obtain panoramic view of its surroundings from a single image. Given a sequence of panoramic images obtained from the environment, these images are rectified and the odometric information is obtained using optical flow.

In this paper we present a system that performs visual odometry using one omnidirectional camera. Figure 1 describes the steps performed by our system to obtain displacement information based on omnidirectional images. Initially images are acquired by the omnidirectional system in the pre-processing step and calibration is accomplished. Images are unwarped and following to feature extraction module. These features feed the optical flow method that estimate motion between consecutive pictures. The motion estimation are used in the odometry process where the motion will be transformed to robot odometry. These last four steps are defined like Main loop.

![Figure 1: Overview of the proposed system. After the images are acquired, they are unwarped and features are extracted. The optical flow uses these features to estimate motion between two consecutive frames. Lastly, the estimated motion is used to compute the robot odometry.](image)

This paper is organized as follows. Section 2 presents related works on visual odometry and optical flow using omnidirectional images. Section 3 describes our methodology, which basically consists of calibration, feature extraction and optical flow. Section 4 validates our method by experimental results using real data sets. Section 5 discusses the effectiveness of our method and proposes the continuation of this research in future works.
2 RELATED WORKS

A major problem with robotic navigation based on vision systems is obtaining matches between images taken from different points of view. In literature, feature matching is extensively researched for standard cameras (Lowe, 2004; T. and Gool, 2004; Calonder et al., 2010). These methods have been successfully implemented in perspective camera, but such methods cannot be directly applied to images obtained from omnidirectional systems, because of the nonlinear distortions introduced by the wide field of view (e.g., decreased radial image resolution).

Traditional methods for feature matching cannot be applied to omnidirectional images without first performing unwarping to first remove distortions. Therefore, in this work we incorporate the Bird’s eye view unwarp technique to unwarp images due to (Gaspar and Santos-Victor, 2000), which corresponds to scaled orthographic views of the ground plane.

Some methods were proposed to work directly on non-rectified images. (Scaramuzza and Siegwart, 2009) presented a method for visual odometry using a new method for removing outliers in the matching process. Another case of visual odometry methods using omnidirectional cameras is the work of (Corke and Strelow, 2004), that developed two methods for visual odometry for a planetary rover. (Demonceaux and Rizzi, 2008) proposed an adaptation of optical flow for omnidirectional images without rectification. Their approach consists in solving the problem of loss correction remains approximately constant for a short period of time a when frames are acquired.

The mirror radius can be easily measured, but the camera-mirror distance \( L \), the focal length \( f \) and the principal point \((\mu_0, v_0)\), can only be determined up to some error:

\[
\delta \Theta = [\delta L \  \delta f \  \delta \mu_0 \ \delta v_0]^T
\]

To estimate \( \delta \Theta \), a set of known 3D points, \( P' \), and the corresponding image projections, \( p' \), are used in a minimization of the following cost function:

\[
\delta \Theta = \arg \min_{\delta \Theta} \sum_i ||p' - P(P', \theta_0 + \delta \Theta)||^2
\]

This procedure defines a mapping between radial distances measured on the ground plane and the respective image coordinates. This process can be efficiently implemented by means of a look-up table. It allows us to unwarp omnidirectional images to bird’s eye view (Figure 3). A detailed description of the omnidirectional vision system and the image unwarping technique can be found in (Gaspar and Santos-Victor, 2000).

3 Methodology

To unwarp the images acquired with omnidirectional systems, one needs to estimate the intrinsic and extrinsic parameters of the camera, given by the matrix:

\[
\Theta = [L \ f \ \mu_0 \ v_0]^T
\]

where \( L \) is camera-mirror distance, \( f \) the focal length, and \((\mu_0, v_0)\) the principal point.

To compute the matrix \( \Theta \) the method proposed by (Scaramuzza and Siegwart, 2006) was applied. Figure 2 illustrates one image of a set of 17 images that were used on the calibration step.

Having computed the \( \Theta \) matrix, the Bird’s Eye View unwarping process is then applied.

3.1 Bird’s Eye View unwarping

A point \( P \), in the 3D space, are projected to \( p \) in the image domain as:

\[
p = \mathbb{P}(P, \Theta)
\]

where \( \mathbb{P} \) is the projection operator and \( \Theta \) the matrix containing intrinsic and extrinsic parameters.

To unwarp omnidirectional images to bird’s eye view (Figure 3). A detailed description of the omnidirectional vision system and the image unwarping technique can be found in (Gaspar and Santos-Victor, 2000).

3.2 OPTICAL FLOW

The methods for computing optical flow can be categorized into three major groups: differential, matching and frequency energy techniques. The initial hypothesis of differential optical flow is that the scene illumination remains approximately constant for a short period of time a when frames are acquired.

Let \( I(x, y) \) be the image intensity at time \( t \), and that \( dt \), the time interval between two consecutive images, is very short. Then the intensity constancy between two frames may be expressed as:

\[
I(x, y, t) = I(x + dx, y + dy, t + dt).
\]
A quick inspection of the optical flow equation (Eq. 8) shows that there are more unknowns than equations. This is known as the problem of openness in optical flow algorithms. Therefore, in order to compute the optical flow, another set of equations is necessary (Lucas and Kanade, 1981), which are given by additional constraints. The solution given by Lucas and Kanade is by a non iterative method that assumes a constant local optical flow. Assuming that the flow \( (V_x, V_y, V_z) \) is constant at the center of a small image area of size \( m \times m \), with \( m > 1 \), and numbering the pixels \( 1, \ldots, n \), a set equations can be written:

\[
\begin{align*}
I_{x1}V_x + I_{y1}V_y + I_{z1}V_z &= -I_{t1} \\
I_{x2}V_x + I_{y2}V_y + I_{z2}V_z &= -I_{t2} \\
I_{x3}V_x + I_{y3}V_y + I_{z3}V_z &= -I_{t3} \\
I_{xn}V_x + I_{yn}V_y + I_{zn}V_z &= -I_{tn}.
\end{align*}
\]

With this restriction, there are more equations than unknowns and the system becomes over determined.

\[
\begin{bmatrix}
I_{x1} & I_{y1} & I_{z1} & V_x & -I_{t1} \\
I_{x2} & I_{y2} & I_{z2} & V_y & -I_{t2} \\
I_{x3} & I_{y3} & I_{z3} & V_z & -I_{t3} \\
\vdots & \vdots & \vdots & \vdots & \vdots \\
I_{xn} & I_{yn} & I_{zn} & V_z & -I_{tn}
\end{bmatrix}
\begin{bmatrix}
V_x \\
V_y \\
V_z
\end{bmatrix}
= \begin{bmatrix}
-I_{t1} \\
-I_{t2} \\
-I_{t3} \\
\vdots \\
-I_{tn}
\end{bmatrix},
\]

that can be summarized below:

\[
A\tilde{v} = -b.
\]

A least squares method is used to solve the above over determined system of equations:

\[
A^TA\tilde{v} = A^T(-b).
\]

This means that the optical flow can be computed by deriving the image in all four dimensions.

### 3.2.2 Shi-Tomasi Feature Tracker

No feature-based vision system can work correctly unless good features can be identified and tracked from frame to frame. Although the tracking problem has been tackled by several approaches, the selection of features that can be well tracked and which correspond to actual points in the world is still hard (Shi and Tomasi, 1994). In the Shi-Tomasi approach, the following matrix is computed:

\[
\begin{bmatrix}
\sum (\frac{\partial I}{\partial x})^2 & \sum (\frac{\partial I}{\partial x}\frac{\partial I}{\partial y}) \\
\sum (\frac{\partial I}{\partial x}\frac{\partial I}{\partial y}) & \sum (\frac{\partial I}{\partial y})^2
\end{bmatrix}
\]

Figure 3: Original image (a) and unwarped image (b). Note the straight lines that are bent in image (a) and rectified in (b).
Figure 4: In image (a) the region defined by the square is low textured, with gradients with small magnitude and small eigenvalues. In (b) the highlighted square is a high texture region, with gradients of large magnitude and large eigenvalues.

where $I$ is the intensity of the pixel with image coordinates $[x, y]^T$, $\partial x$ and $\partial y$ and are the horizontal and vertical displacements of the center of the window containing the neighborhood in the next frame.

(Shi and Tomasi, 1994) et al. define that a good feature should have two distinctive qualities, texture and corner. Lack of texture causes ambiguities in tracking, since pixels become very similar, resulting in outliers and false matches. Fig. 4 (a) shows a low texture region, with small eigenvalue, and Fig. 4 (b) depicts a high texture region, with a large eigenvalue. Therefore it may be concludes that a good feature has a large eigenvalue, which may lead to more reliable results. Another problem in images, known as the aperture problem, happens in images with reduced number of corners.

4 EXPERIMENTS

There are several datasets publicly available, such as (Blanco and Gonzalez, 2009), (Wang and Duggins, 2004) and (Project, Rawseeds Project, 2010.), which were obtained during robotic navigation. We choose (Project, Rawseeds Project, 2010.) dataset since it supplies information about the omnidirectional images, wheel odometry and GPS data.

We choose two examples of navigation in the dataset. One with variation between natural and artificial light and the other has an environment composed of low texture regions, that present a challenge for the proposed system.

The lens used in the (Project, Rawseeds Project, 2010.) dataset is a omni-VS-C15MR-Vstone with hyperbolic mirror (Figure 6 (a)) mounted on a Prosilica GC1020C camera. The complete acquisition system (Lens+Camera) can be seen on Figure 6 (b).

Our hardware setup was a Core i7 920 (2.67 Ghz) with 12 Gb of RAM, 320 Gb Hd and a GTX 260 graphic card.

Figure 5: Vehicle employed for data acquisition in the Rawseeds Project.

4.1 Performance Results

The first experiment focused on estimating the time spent by the unwarping and optical flow modules. Table 1 clearly shows that the unwarping process is more time consuming than the optical flow module, where the overhead added by this routine is shown on the solution row. These values correspond to the

Figure 6: Lens omni-VS-C15MR-Vstone with hyperbolic mirror (a) and acquisition system composed of omni-VS-C15MR-Vstone and Prosilica GC1020C camera (b).
Table 1: Statistical values of modules, first row optical flow, then unwarping and complete solution.

<table>
<thead>
<tr>
<th>Module</th>
<th>Mean(milliseconds)</th>
<th>FPS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Optical Flow</td>
<td>21</td>
<td>47</td>
</tr>
<tr>
<td>Unwarping</td>
<td>28</td>
<td>33</td>
</tr>
<tr>
<td>Solution</td>
<td>49</td>
<td>20</td>
</tr>
</tbody>
</table>

mean processing time of five executions. Although the process of image rectification adds a considerable amount of time to solution, it can still be processed at 20 frames per second rate, which is not an issue for mobile robots with a slow dynamics.

### 4.2 Optical Flow Results

Figure 7 illustrates the optical flow computation performed in our tests. Figure 8 presents the number of matches during a test. These matches are computed from 400 features extracted from each frame. One may observe that 44 matches are the average number obtained by this technique, because many areas on the image present low texture, with small eigenvalue, hindering our feature matching process.

![Figure 7: Sample of the output of the optical flow module. In red is the estimated motion captured by our method. For visualization purposes the vector magnitudes are plotted four times larger than originally.](image)

![Figure 8: Number of matches during our tests, with a mean value of 44 matches, this low value is due to many areas with low texture.](image)

4.3 Visual Odometry Results

The first test consists of a robot moving in a straight line through 3.35 meters. Tests were executed with people passing by (which disturb optical flow) and artificial and natural lighting. On Figure 9, vertical axis represent the odometry values of the robot in x (blue line) and y axis (red line). The Figure show that our method estimates a final position of 3.10 meters, which compared to groundtruth of 3.35 meters get an error in position of 25 centimeters corresponding to 7.4% of error.

![Figure 9: Results of first test. Our method estimates a final position of 3.10 meters, which compared to groundtruth of 3.35 meters get an error in position of 25 centimeters corresponding to 7.4% of error.](image)

The second test was performed on a 7.55 meters route moving in straight line, through a corridor with uniform structure (low texture in some parts), people passing by and artificial lighting. On Figure 10, the vertical axis show the odometry values of the robot. The red line (y axis) his final position (6.64 meters), which favorably compares to groundtruth of 7.55 meters, which implies in 12% error.

![Figure 10: The odometry values of the robot in x and y axes.](image)

5 CONCLUSION AND FUTURE WORKS

In this paper, we presented a method for visual odometry using omnidirectional images. One key feature with wider view cameras when compared to standard cameras is that a landmark tracked on the former remains observable for a larger number of frames than in the later, which makes visual based odometry approaches more robust. We also described a method for obtaining a bird’s eye view of the ground floor, that by removing perspective effects greatly simplified omnidirectional optical flow strategies. In future work we plan to apply...
this methodology to more complex environments and test other approaches to solve visual odometry and use recent results in feature extraction for omnidirectional images, like in (Barreto and Lourenco, 2010).

Visual odometry is a powerful technique to estimate motion from image sequences. Many investigations are still underway to further improve its performance and reduce its computational time. This is a vast research area with room for improvements.

References


