Evaluation of Compact Real-time Descriptors for Camera Motion Estimation

Shota Ogawa, Kenichi Asami, and Mochimitsu Komori

Abstract
This paper introduces the image processing system of camera motion estimation for autonomous mobile robots. Most autonomous mobile robots need to quickly recognize the surroundings so as to move to the space. Moreover, the image processing algorithm requires low computation complexity. Accordingly, we evaluate an image processing system based on Compact and Real-time Descriptors (CARD)[1]. Proposed system has scale-invariance and rotation-invariance for local image feature. Our method uses a multi-scale image pyramid to extract coordinates for keypoint and a log-polar binning pattern for a patch around the keypoint. The results of correspondence for keypoint between two images were good enough to estimate the camera motion.

Keywords: scale- and rotation-invariance, gradient features, feature description, log-polar binning pattern.

1. Introduction
Local feature descriptor is effective for visual correspondence between two images. Scale-Invariant Feature Transform (SIFT)[2] is representative method for local feature descriptors and has been verified to be useful for image recognition application. But it is unsuitable to real-time processing for autonomous mobile robots. It is important for robot application to reduce computational cost for the mobile ability.

This paper focuses on scale- and rotation-invariant feature descriptor by using a multi-scale image pyramid and a log-polar binning pattern, instead of a big amount of computation of gradient extremum search and a binning pattern manipulation of SIFT descriptor.

2. CARD Algorithm

2.1 Keypoint Detection
Keypoint detection is an important function for scale-invariance. SIFT is scale-invariant by using Difference of Gaussians (DoG) pyramid. But a DoG pyramid requires heavy processing because an image needs to be applied different sizes of Gaussian filter.

Instead of DoG pyramid, the method uses a multi-scale image pyramid to detect keypoint which is robust under scale change. That image pyramid is structured in the fast way of downsizing image. The size of the n-th level corresponds to the $1/\sqrt{2}$ size of the (n-1)-th level. The 0-th level represents original image. In practice, n-th level is represented by making half size of (n-2)-th level. Then Good Features to Track (GFTT)[3] method detects the coordinates for keypoint from each level of the scale pyramid. GFTT is based on Harris corner detector.

2.2 Description with Gradient Features
The descriptor uses the gradient of corner points with the scale- and rotation-invariant distinctiveness. The gradient represents magnitude and direction for local pixel value. It is found from the difference of brightness value for neighborhood around the sampling point. Also, this method is strong in a local illumination change. The set of gradients in a regional block characterizes the local feature description.

The descriptor use a log-polar binning pattern which shows the patch around a keypoint for feature description, in the same way of GLOH[4]. A patch around a keypoint is usually relocated before feature description when the image is rotated. The employment of a log-polar pattern for the patch around a keypoint avoids many bilinear interpolations.
ψ(u, v, ρ) = \begin{cases} 0, & 0 \leq r < r_1 \\ 1 + Q_6(\theta(u, v) - ρ), & r_1 \leq r < r_2, \\ 9 + Q_6(\theta(u, v) - ρ), & r_2 \leq r \leq r_3 \end{cases}

(4)

where \( r \) denotes the distance between \((u, v)^T\) and \((p_u, p_v)^T\). The thresholds \( r_1, r_2, \) and \( r_3 \) are set to the radial distances.

The feature descriptor \( d \) is obtained as below:

1. Calculate the block number \( k \) from (1) and the gradient direction \( l \) from (3) for all pixels in a block.
2. Accumulate the gradient histogram \( h_{k,l} \leftarrow h_{k,l} + m(u, v)G_\sigma(u, v) \)
3. Normalize \( d \) according to the block area.

where \( m(u, v) = \sqrt{u^2 + v^2} \) denotes the magnitude of the gradient and \( G_\sigma \) denotes the weighted values as a normal distribution with variance \( \sigma \). In the case of 17 blocks and 8 quantized directions, the 136 dimensional descriptor is constructed.

In practice, computation of (1) can be done beforehand because \( u, v, \) and \( ρ \) take only a finite number. Therefore, the lookup table for \( ψ(u, v, ρ) \) is prepared to reduce the computational cost.

3. Matching Algorithm

3.1 Rotation of a Binning Pattern

As mentioned in the preceding section, the descriptor rotate a binning pattern for rotation invariance. For that purpose, a principal orientation is calculated from the whole gradient histogram by extracting \( d \) for all the pixels in 17 blocks. Consequently, the whole gradient histogram can be easily accumulated the values of gradient magnitude. The principal orientation is assigned to the gradient direction with the maximum gradient magnitude as shown in Fig. 2.

![Fig. 1. A log-polar binning pattern.](image)

![Fig. 2. Calculation of a principal orientation.](image)
In the matching, there are two images which are called reference image and alternative image respectively. A binning pattern of the alternative image is rotated to adjust the principal orientation of it to the reference image. At the same time, the orientation of all pixels in the binning pattern is rotated as shown in Fig. 3.

3.2 Matching by Euclidean Distance

The similarity among feature descriptions is measured by the ratio of the Euclidean distances to the two nearest neighbors so as to make robust matches. Accordingly, most mismatches are rejected by the matching result.

For every image of a multi-scale pyramid, the feature descriptions is evaluated for matching keypoints. Therefore, the scale change by a camera motion can be handled.

4. Experiments

Several experiments measured the correct matching rate and the error matching rate for changing the location and turning camera. Before showing the result of experiments, the correct matching rate and the error matching rate are defined. They are represented as follows:

\[
\text{correct matching rate} = \frac{\text{number of true correspondances}}{\text{number of keypoints}} \times 100
\]

\[
\text{error matching rate} = \frac{\text{number of false correspondances}}{\text{number of keypoints}} \times 100
\]

where the number of keypoints is counted in the reference image at the 0-th level. The number of true correspondences represents the true positive of the matching result. The number of false correspondences represents the false positive.

4.1 Location Change

The camera position was changed to right, left, forward, and backward perspectives. The matching correspondences in the images are shown in Fig. 4. The results of matching rate are shown in Table 1.
The correct matching rate achieved 82.0% and error matching rate showed 0.0% for right movement. The correct matching rate achieved 55.0% and the error matching rate showed 1.1% for forward movement. Accordingly, this algorithm verified robust for scale and view change.

4.2 Angle Change

The camera angle was changed to clockwise incrementally. The matching correspondences are shown in Fig. 5. The results of matching rate are shown in Table 2.

<table>
<thead>
<tr>
<th>Right perspective change</th>
<th>Correct matching rate[%]</th>
<th>Error matching rate[%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>15 degree angle change</td>
<td>56.2</td>
<td>0.0</td>
</tr>
<tr>
<td>30 degree angle change</td>
<td>58.4</td>
<td>0.0</td>
</tr>
<tr>
<td>45 degree angle change</td>
<td>70.8</td>
<td>0.0</td>
</tr>
</tbody>
</table>

The correct matching rate achieved 56.2% and the error matching rate showed 0.0% for angle change of 15 degree. The correct matching rate increased 70.8% for angle change of 45 degree. That's because a binning pattern is divided by 45 degree. As the result, this algorithm verified robust for rotation.

Conclusion

In this paper, we evaluated an image processing system based on CARD. For three different scenes, experiments including Fig. 4 and Fig. 5 were operated to confirm the robustness for the camera motion changes. The experiments for location and angle changes showed that the correct matching rate was high and the error matching rate limited to 1.1%. This rates are sufficient for autonomous mobile robots. The processing time of this algorithm should be reduced. We would like to propose the binarized algorithm of this descriptor based on FREAK (5).

References


