MVA2007 IAPR Conference on Machine Vision Applications, May 16-18, 2007, Tokyo, JAPAN

Extraction of Corresponding Points from Stereo Images by Using Intersections of Segments

Hiroshi Unno, Keikichi Hayashibe, and Hitoshi Saji
Graduate School of Informatics, Shizuoka University
3-5-1 Johoku, Hamamatsu 432-8011 Japan
saji@inf.shizuoka.ac.jp

Abstract

In many applications, it is necessary to extract corresponding points automatically. In this study, we present a method of extracting corresponding points automatically from stereo images. First, we use the Combinatorial Hough Transform to extract feature points that are intersections of segments. This method is robust to the image noise. Then we correspond them by using backmatching method. Backmatching is robust to the occlusion. In conventional studies, mismatch is likely to occur. This is because they have used limited information in the image for extracting feature points. In our method, we use information of wide region in the image. Our proposed algorithm enables stable extraction of corresponding points in images composed of many edges, and this extraction is independent of minute threshold variations.

1 Introduction

Image processing technologies show development in many fields in recent years as the performance of computers has improved. In particular, technologies of three-dimensional graphics are useful in various applications, such as games, movies, and simulation software.

Stereo systems can measure three-dimensional coordinates of an object. However, searching for corresponding points is difficult problem. Moreover, it is necessary to extract corresponding points automatically in several self-calibration methods[1]. It is a problem that a disparity exists in stereo images. An occlusion is occurred by the disparity, and it causes correspondence errors because the accuracy of local correlation falls remarkably. A multi-baseline stereo method[2] that uses two or more stereo images has been reported in order to solve this problem. However, it uses many cameras and the correspondence process is complex.

In conventional studies, feature points are extracted from the picture using Harris operator[3] or SUSAN operator[4], and there are matched by using template matching in the correspondence process. However, these methods depend on minute threshold variations, and their results are influenced by the local intensity distribution around the points. In this study, we use information of wide region in the image. We use the Combinatorial Hough Transform to extract feature points that are intersection of segments. This extraction is independent of minute threshold variations and is hard to be influenced by the local intensity distribution. Then, we correspond them by backmatching that is robust to the occlusion. We evaluate the efficacy of these approaches by several experiments.

2 Proposed Method

2.1 Algorithm

Corresponding points are extracted by the following processes.

- Extraction of edge pixels
- Extraction of feature points by using the Combinatorial Hough Transform
- Matching of feature points in image rows

2.2 Extraction of Edge Pixels

First, we use smoothing process in order to reduce noises in the image. Then we apply an edge extraction method (Sobel operator) to the image. Edge angles are obtained only on a pixel with high edge intensity, and they are used in the following processes. The edge angles can be obtained by the following expression.

$$\theta(x, y) = \tan^{-1} \frac{\delta y(x, y)}{\delta x(x, y)} \tag{1}$$

Here (x, y) is a pixel in the image, $\delta x(x, y)$ is the edge intensity to the direction of x, and $\delta y(x, y)$ is the edge intensity to the direction of y. These intensities are obtained by the Sobel operator.

The edge angles obtained by the equation (1) are dispersion widely by quantization errors, and hence we average the edge angles. The edge obtained by the Sobel operator have a width of several pixels. Then reliable edge pixels are extracted by restraining the non-maximum of the edge intensity. This process reduces the number of edge pixels and the computation amount of the following processes.

2.3 Feature Points Extraction by the Combinatorial Hough Transform

Intersections are extracted from an edge pixel pair using the Combinatorial Hough Transform. Originally the Hough Transform[5] and the Combinatorial Hough Transform[6] are the methods of extracting straight lines from images. In the Combinatorial Hough Transform, the parameters of a line are obtained from two pixels on the line, and the voting process is used. It

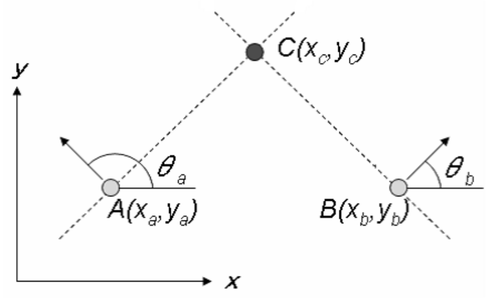


Figure 1: Intersection extraction

is a method that can reduce the computation amount compared with usual Hough Transform. In this study, we use the Combinatorial Hough Transform to extract intersections in the images, and we use the intersection as the feature points.

2.3.1 Intersection Extraction

Intersections of two straight lines are extracted by using coordinates of two points and directions of edge pixels. The intersection is shown in Fig.1.

First, we prepare two vote spaces whose sizes are identical to the sizes of the object images. Points $A(x_a, y_a)$ and $B(x_b, y_b)$ are edge pixels, and those edge angles are θ_a and θ_b respectively. If $C(x_c, y_c)$ exists on the extension line of the edge segment containing $A(x_a, y_a)$, the following equations can be obtained.

$$x_c = x_a + s \cdot \cos(\theta_a + \pi/2), \tag{2}$$

$$y_c = y_a + s \cdot \sin(\theta_a + \pi/2), \tag{3}$$

where s is the parameter. Similarly, if $C(x_c, y_c)$ exists on the extension line of the edge segment containing $B(x_b, y_b)$, the following equations can be obtained.

$$x_c = x_b + t \cdot \cos(\theta_b - \pi/2), \tag{4}$$

$$y_c = y_b + t \cdot \sin(\theta_b - \pi/2), \tag{5}$$

where t is the parameter. Coordinates $C(x_c, y_c)$ that is the intersection of two straight lines can be obtained by equations (2,3,4,5), and we vote the coordinates on the vote space.

In this study, we select all pairs of edge pixels in the area of 50×50 pixels, and we compute only when the edge angle difference θ between two points is $\pi/3 < \theta < 2\pi/3$. On these conditions, we can shorten the computation time.

2.3.2 Feature Points Extraction from Vote Spaces

After the vote processing, vote spaces are divided into lattice small areas, and the points whose vote values are local maxima and are more than a threshold are extracted as the feature points.

2.4 Matching of Feature Points by Backmatching

Occlusions may be occurred because the disparity exists in stereo images. If the point is occluded, it is necessary to remove as a candidate of corresponding point

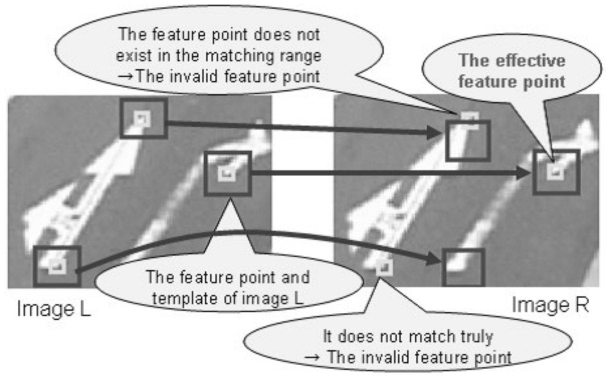


Figure 2: Corresponding result

for avoiding matching errors in the following process. In this removal process, we use the backmatching.

Moreover, improvement in the speed of processing is expectable by limiting the search range of template matching. In order to limit the search range, we compute the gap between left and right images by template matching around center part of the images.

2.4.1 Template Matching

In the backmatching, we use the template matching method. The template matching is a method for searching out the position of one image (template) in another image.

We use the template whose size is $2s \times 2s$, and center coordinates of the template are the coordinates of each feature point obtained from stereo image L. Here, $L_R(x,y), L_G(x,y), L_B(x,y)$ are pixel color values of the stereo image L, and $R_R(x',y'), R_G(x',y'), R_B(x',y')$ are pixel color values of stereo image R. We compute difference J in $2s \times 2s$ area in each position of the image. If this value is minimum, the similarity is the highest, and the position is obtained as the matching position.

$$\begin{aligned} J \;=\; & \sum_{\Delta x=-s}^{s} \sum_{\Delta y=-s}^{s} \\ & |L_R(\Delta x, \Delta y) - R_R(x' + \Delta x, y' + \Delta y)| \\ & + |L_G(\Delta x, \Delta y) - R_G(x' + \Delta x, y' + \Delta y)| \\ & + |L_B(\Delta x, \Delta y) - R_B(x' + \Delta x, y' + \Delta y)| \,. \end{aligned} \tag{6}$$

In this study, we allocate 10 to "s".

When feature points obtained from the stereo image R exist in the matched template area, these are extracted as the effective feature points. The method is shown in Fig.2.

2.4.2 Backmatching

We create the template around the effective feature points in image R. Then we apply backmatching process to the stereo image L. The method is shown in Fig.3.

When feature points that are obtained from the stereo image L exist in the matching areas, these areas are processed. By this processing, only the two points with locally high similar level are corresponded. This method is robust to the occlusion.

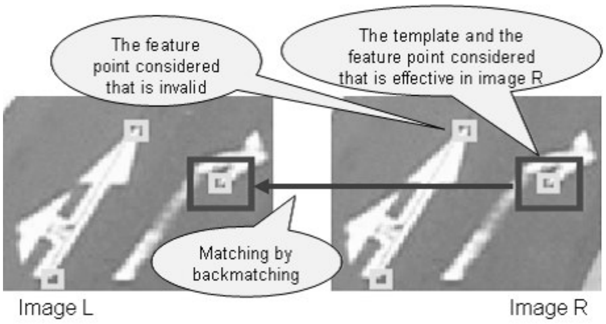


Figure 3: Backmatching

3 Experiments

3.1 Experimental Condition

The computer and images which are used in this study are shown below.

- CPU Pentium(R)D 3.4GHz
- RAM 1GByte
- Image size 720×480pixels（1 byte each of RGB）

Object images are road images taken with the stereo camera on the street. Distance between these cameras is 1m.

3.2 Experimental Results

3.2.1 Creation of Vote Spaces

We show left image and right image in Fig.4. After extracting reliable edge pixels from the left and right images, vote spaces of the same size as an input images are obtained by using the Combinatorial Hough Transform algorithm, and these are shown in Fig.5. The intensity of each pixel indicates the number of vote.

3.2.2 Feature Points Extraction

Feature points are extracted from the vote spaces of the left and right images. These vote spaces are divided into the lattice small areas and the points whose vote values are local maxima and more than a threshold are extracted as the feature points. Feature points in the left and right images are shown as the white squares in Fig.6.

3.2.3 Matching of Feature Points

Feature points in the left and right images are matched by using backmatching. Corresponding points obtained from the left and right images are shown in Fig.7.

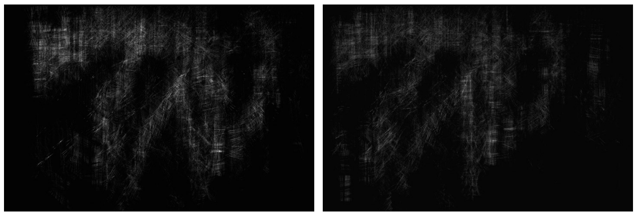
3.2.4 Extraction of Corresponding Points by a Comparison Method

Feature points are extracted by using the Harris operator as the comparison method. In the Harris operator, (x, y) are the coordinates in the image and $I(x, y)$ is the pixel value of (x, y), and we compute the following matrix$M$.

$$M = \begin{pmatrix} \left(\frac{\partial I}{\partial x}\right)^2 & \frac{\partial I}{\partial x}\frac{\partial I}{\partial y} \\ \frac{\partial I}{\partial x}\frac{\partial I}{\partial y} & \left(\frac{\partial I}{\partial y}\right)^2 \end{pmatrix}. \tag{7}$$

(a) Left image (b) Right image

Figure 4: Input images



(a) Left image (b) Right image

Figure 5: Vote results

(a) Left image (b) Right image

Figure 6: Feature points

When the two eigenvalues computed from the matrix M is more than a threshold, the points are extracted as the feature points.

The feature points obtained in left image and right image by the Harris operator are matched by using our backmatching method. Corresponding points obtained from the left and right images are shown in Fig.8. White squares indicate the correct result, and black squares indicate the errors.

3.2.5 Another Experiment

We evaluate the efficacy of the methods by another road images. Corresponding points obtained by our method are shown in Fig.9, and corresponding points obtained by the Harris operator are shown in Fig.10. White squares indicate the correct result, and black squares indicate the errors.

4 Evaluation and Consideration

The result of corresponding point extraction by our method is shown in Table.1, and the execution time is shown in Table.2. The result of corresponding point extraction by the comparison method (Harris operator) is also shown in Table.1. We set a threshold value of the Harris operator so that the number of feature points extracted by the two methods are almost identical . In the result, there is no error in extracting corresponding points by our method. On the other hand, there is a few errors by the Harris operator. This is because the feature points are computed from image information of wide region in our method. On the other hand, in

Table 1: The result of corresponding point extraction

		Feature points		Corresponding points	Mismatch points
1	Proposed method	Left	89	16	0
		Right	48		
	Comparison method	Left	91	34	4
		Right	101		
2	Proposed method	Left	79	26	0
		Right	76		
	Comparison method	Left	88	33	4
		Right	91		

Table 2: Execution time

	Vote process (sec)	Feature points extraction (sec)	Matching (sec)
1	0.625	0.015	17.265
2	0.89	0.015	18.703

(a) Left image (b) Right image

Figure 7: Corresponding points

(a) Left image (b) Right image

Figure 8: Corresponding points by the Harris operator

(a) Left image (b) Right image

Figure 9: Corresponding points

(a) Left image (b) Right image

Figure 10: Corresponding points by the Harris operator

the Harris operator, the feature points are computed from image information of neighborhood of one pixel, and hence the extraction results are influenced by the image noise. In the result, the similar points may be mistaken for the corresponding points.

5 Conclusions

In this study, we proposed the method that extracts corresponding points from stereo images by using intersections of segments. The result of our experiments shows that our proposed algorithm enables stable extraction of corresponding points in the images composed of many edges.

As the future work in this study, we would like to experiment using many images in several situations, such as several seasons, several times, several places, and so on.

Acknowledgements

We would like to thank Japan traffic management technology association for their helps on this study.

References

[1] F.Tomita and H.Takahashi: Self-calibration of stereo cameras, Proceedings of the 2nd International Conference on Computer Vision, Issue, pp.123-128, 1988.

[2] M.Okutomi and T.kanade: A multiple-baseline stereo, IEEE Transactions on Pattern Analysis and Machine Intelligence, vol.15, No.4, pp.353-363, 1993.

[3] C.Harris and M.Stephens : A combined corner and edge detector, Proc. of the 4th Alvey Vision Conference, pp.147-151, 1988.

[4] S.M.Steve and M.Brady: SUSAN-a new approach to low level image processing, International Journal of Computer Vision, vol.23, No.1, pp.45-78, 1997.

[5] R.O.Duda and P.E.Hart: Use of the Hough Transformation to Detect Line and Curves in Pictures, ACM, vol.15, No.1, pp.11-15, 1972.

[6] D.Ben-Tzvi and M.B.Sandler: A combinatorial Hough transform, Pattern Recognition Letters, vol.11, No.1, pp.94-98, 1990.