Image Corner Detection Using Hough Transform

Sung Kwan Kang, Young Chul Choung, and Jong An Park

Dept. of Information & Communications Engineering,
Chosun University, Gwangju, Korea
japark@chosun.ac.kr

Abstract. This paper describes a new corner detection algorithm based on the Hough Transform. The basic idea is to find the straight lines in the images and then search for their intersections, which are the corner points of the objects in the images. The Hough Transform is used for detecting the straight lines and the inverse Hough Transform is used for locating the intersection points among the straight lines, and hence determine the corner points. The algorithm was tested on various test images, and the results are compared with well-known algorithms.

Keywords: corner detection, Hough Transform, Detecting Straight Lines, curvature scale, corner points.

1 Introduction

Corners have been found to be very important in human perception of shapes and have been used extensively for shape description, recognition, and data compression [1]. Corner detection is an important aspect of image processing and finds many practical applications. Applications include motion tracking, object recognition, and stereo matching. Corner detection should satisfy a number of important criteria. It should detect all the true corners, and the corner points should be well localized, and should be robust with respect to noise, and should be efficient. Further, it should not detect false corners.

There is an abundance of literature on corner detection. Moravec [2] observed that the difference between the adjacent pixels of an edge or a uniform part of the image is small, but at the corner, the difference is significantly high in all directions. Harris [3] implemented a technique referred to as the Plessey algorithm. The technique was an improvement of the Moravec algorithm. Beaudet [4] proposed a determinant (DET) operator which has significant values only near corners. Dreschler and Nagel [5] used Beaudet's concepts in their detector. Kitchen and Rosenfeld [6] presented a few corner-detection methods. The work included methods based on gradient magnitude of gradient direction, change of direction along edge, angle between most similar neighbors, and turning of the fitted surface. Lai and Wu [7] considered edge-corner detection for defective images. Tsai [8] proposed a method for boundary-based corner detection using neural networks. Ji and Haralick [9] presented a technique for corner detection with covariance propagation. Lee and Bien [10] applied fuzzy logic to corner detection. Fang and Huang [11] proposed a method which was an improvement on the gradient magnitude of the gradient-angle method by Kitchen and Rosenfeld. Chen and Rockett utilized Bayesian labeling of corners using a gray-level corner image model in [12]. Wu and Rosenfeld [13] proposed a technique which examines

J.S. Marques et al. (Eds.): IbPRIA 2005, LNCS 3523, pp. 279–286, 2005.

the slope discontinuities of the x and y projections of an image to find the possible corner candidates. Paler et al. [14] proposed a technique based on features extracted from the local distribution of gray-level values. Rangarajan et al. [15] proposed a detector which tries to find an analytical expression for an optimal function whose convolution with the windows of an image has significant values at corner points. Arrebola et al. [16] introduced corner detection by local histograms of contour chain code. Shilat et al. [17] worked on ridge's corner detection and correspondence. Nassif et al. [18] considered corner location measurement. Sohn et al. [19] proposed a mean field-annealing approach to corner detection. Zhang and Zhao [20] considered a parallel algorithm for detecting dominant points on multiple digital curves. Kohlmann [21] applied the 2D Hilbert transform to corner detection. Mehrotra et al. [22] proposed two algorithms for edge and corner detection. The first is based on the first-directional derivative of the Gaussian, and the second is based on the second-directional derivative of the Gaussian. Davies [23] applied the generalized Hough transform to corner detection. Zuniga and Haralick [24] utilized the facet model for corner detection. Smith and Brady [25] used a circular mask for corner detection. No derivatives were used. Orange and Groen [26] proposed a model-based corner detector. Other corner detectors have been proposed in [27-30]. Mokhtarian [31] used the curvature-scale-space (CSS) [32], [33] technique to search the corner points. The CSS technique is adopted by MPEG-7. The Kitchen and Rosenfeld detector [6], the SUSAN detector [25] and the CSS [32] corner detector have shown good performance. These detectors are therefore chosen as our test detectors.

In this paper, a new corner detection based on the forward and inverse Hough Transform is presented. The straight lines in the images are detected and their intersection points are used to locate the corner points. This paper is organized as follows. Section 2 describes the method to determine the straight lines in the images using Hough Transform and section 3 describes the algorithm to detect the intersection points among the straight lines using the inverse Hough Transform. Section 4 describes the simulation results. At the end, we will conclude our paper with few final remarks.

2 Detecting Corners Using Inverse Hough Transform

Ideally, a corner is an intersection of two straight lines. However, in practice, corners in the real world are frequently deformed with ambiguous shapes. As corner represent certain local graphic features at abstract level, corners can intuitively be described by some semantic patterns (see Fig. 1). A corner can be characterized as one of the following four types:

- Type A: A perfect corner as modeled in [30], i.e., a sharp turn of curve with smooth parts on both sides.
- Type B: The first of two connected corners similar to the END or STAIR models in [30], i.e., a mark of change from a smooth part to a curved part.
- Type C: The second of two connected corners, i.e., a mark of change from a curved part to a smooth part.
- Type D: A deformed model of type A, such as a round corner or a corner with arms neither long nor smooth. The final interpretation of the point may depend on the high level global interpretation of the shape.

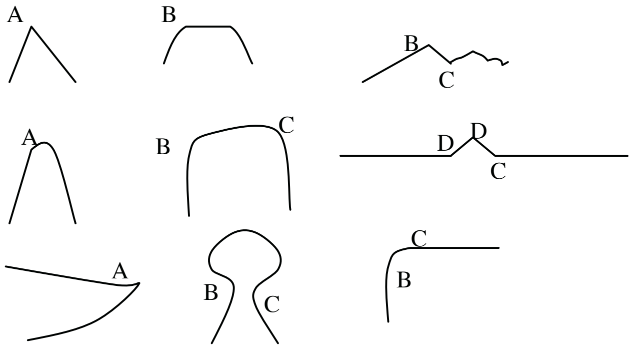


Fig. 1. Four types of corners.

Figure 2 shows some examples of the four types of the corner. It is obvious from the Fig.2 that the corner points at very small level are the intersection points of the two straight lines. To detect the intersection points between the straight lines, and hence the corner points in an image, we do the following procedure.

- Quantise (x,y) space into a two-dimensional array C for the original size of the image in appropriate steps of x and y.
- Initialise all elements of C(x,y) to zero.
- For each pixel (ρ',θ') in parameter space, we add 1 to all elements of C(x,y) whose indices x and y satisfy $\rho' = x\cos\theta' + y\sin\theta'$.
- Search for elements of C(x,y) which have large values than one. Each one found corresponds to a possible candidate for corners in the original image.

Because of many intersections of lines, false corners are also detected. To avoid false candidates, the detected corners whose vicinity does not contain any edge point are discarded. Now consider the case of a corner in an image as shown in Fig. 2.

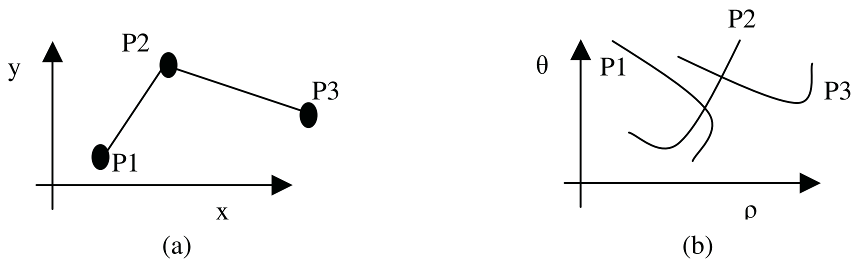
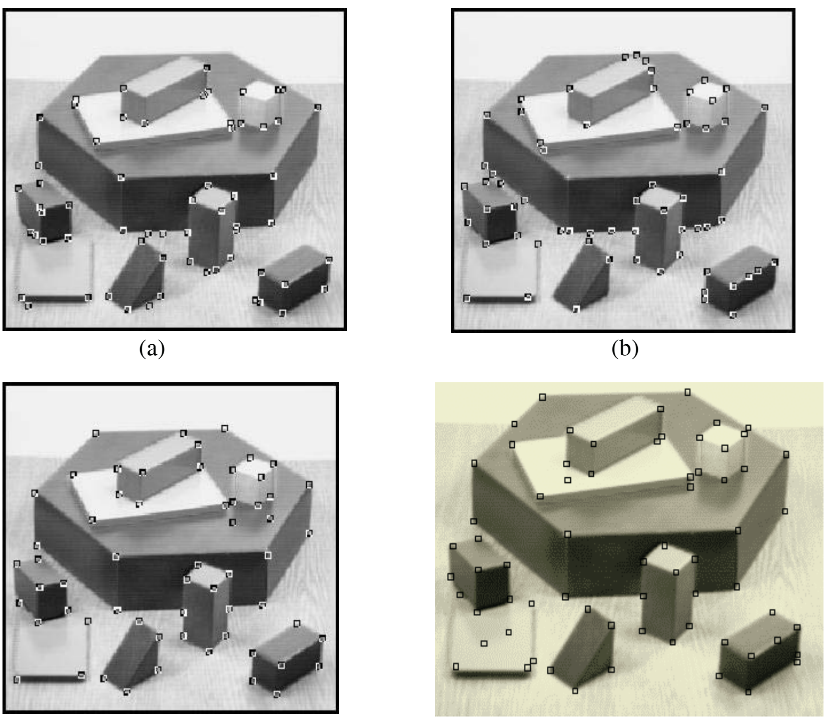


Fig. 2. (a) Points in image space (b) Corresponding points in parameter space.

The corner formed by three points P1, P2 and P3 are transformed into (ρ, θ) parameter space. In parameter space, corner point P2 has two intersections with other lines, while P1 and P3 have only one. It means P2 is a corner point. The peaks in the parameter space correspond to the corners in the image space. We get many intersection points in the image space after getting the inverse transform of the peaks from the parameter space. To remove the unwanted intersection points (i.e., no corner points), we ANDED the intersection points with the edges of the image. The position of true corners and the edges will coincide and give the actual position of the corners. To get more accurate results and to avoid large number of intersections, corners are detected block by block processing with sliding overlapping window. The number of computations becomes higher, but the results are more accurate.

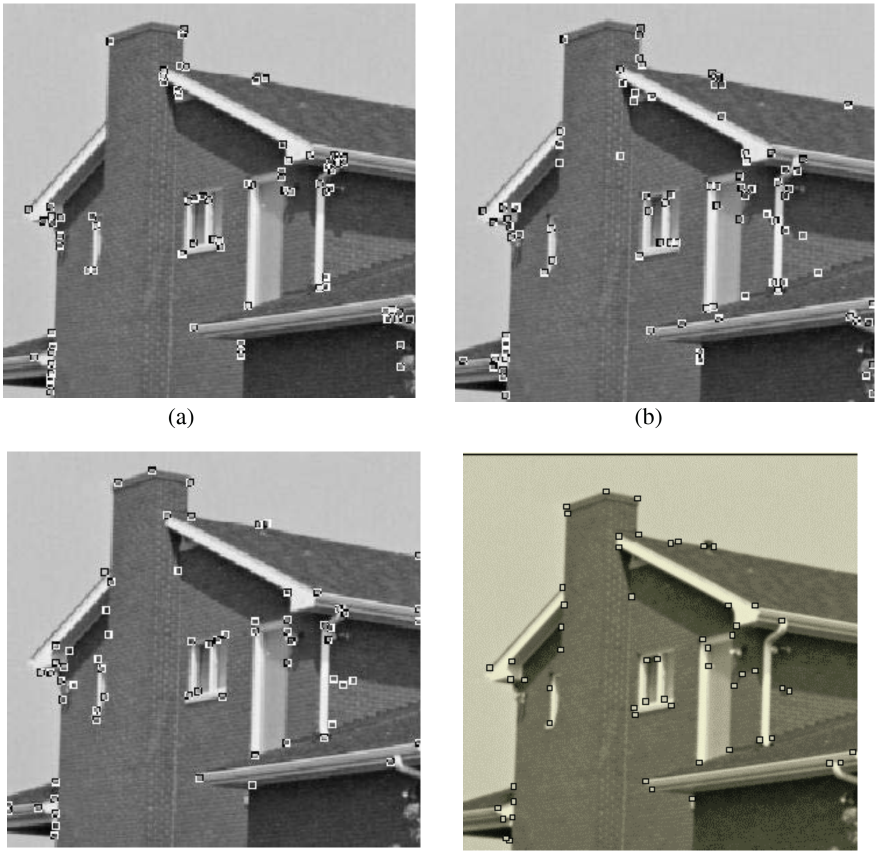
3 Simulations

The proposed algorithm was tested using three different images, the same images used in [31]. The results are compared with three different corner detectors, namely, Kitchen and Rosenfeld, SUSAN and CSS corner detectors. The results for the three corner detectors are also taken from [31]. The results for each detector were the best results obtained by searching the best parameters. The three test images show in Fig .3, Fig.4, and Fig.5 are called Blocks, House and Lab. The Blocks test image contains much texture and noise. The House image has a lot of small details and texture in the brick wall. The Lab image contains plenty of corners. The results show that the proposed algorithm gives better result than that of KR and SUSAN method and gives comparable results with that of CSS.



(a) (b) (c) (d)

Fig. 3. Blocks image. (a) Kitchen/Rosenfeld. (b) SUSAN. (c) CSS. (d) Proposed Algorithm.



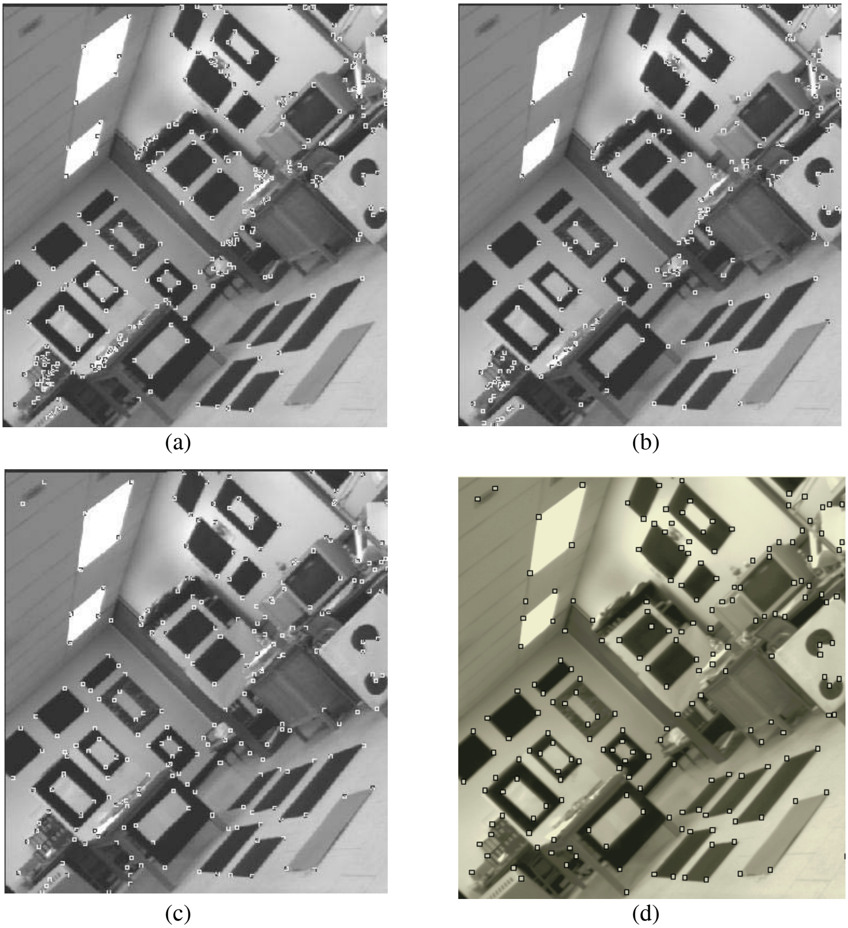
(a) (b) (c) (d)

Fig. 4. House image. (a) Kitchen/Rosenfeld. (b) SUSAN. (c) CSS. (d) Proposed Algorithm.

The proposed method and CSS perform well on the Blocks image. Other detectors find difficulties in locating the corner points. Similarly, the proposed method and CSS method show good performance on the House and the Lab images, while others perform badly. Overall our proposed method and CSS are comparable. The most of the time of the proposed method is consumed by the Hough Transforms. By decreasing or increasing the criterion for peaks effect the detail information about the corners.

4 Conclusions

In this paper, a new corner detector based on the Hough Transform is proposed. The edges in the image are found and the edges are transformed from the image space to parameter space. The Hough Transform is used to find the straight lines in the image, and the inverse Hough Transform is used to find the intersection points among the straight lines. The intersection points are the corner points. The proposed method is compared with the previous methods. The results are comparable with the curvature scale space image corner detector.



(a) (b) (c) (d)

Fig. 5. Lab image. (a) Kitchen/Rosenfeld. (b) SUSAN. (c) CSS. (d) Proposed Algorithm.

Acknowledgement

This research was supported by the Program for the Training of Graduate Students in Regional Innovation which was conducted by the Ministry of Commerce, Industry and Energy of the Korean Government

References

1. Liyuan Li and Weinan Chen, "Corner detection and interpolation on planar curves using fuzzy reasoning," IEEE Trans. On Pattern Analysis and Machine Vision, vol. 21, no. 11, November, 1999.
2. H. P. Moravec, "Towards automatic visual obstacle avoidance," Proc. Int'l Joint Conf. Artificial Intelligence, p. 584, 1977.

3. C. G. Harris, "Determination of ego-motion from matched points," Proc. Alvey Vision Conf., Cambridge, UK, 1987.
4. P. R. Beaudet, "Rotationally invariant image operators," Int'l Joint Conf. Pattern Recognition, pp. 579-583, 1978.
5. L. Dreschler and H. H. Nagel, "Volumetric model and 3D trajectory of a moving car derived from monocular TV frame sequences of a street scene," Int'l Joint Conf. Artificial Intelligence, pp. 692-697, 1981.
6. L. Kitchen and A. Rosenfeld, "Gray level corner detection," Pattern Recognition Letters, pp. 95-102, 1982.
7. K. K. Lai and P. S. Y. Wu, "Effective edge-corner detection method for defected images," Proc. Int'l Conf. Signal Processing, vol. 2, pp. 1,151-1,154, 1996.
8. D. M. Tsai, "Boundary based corner detection using neural networks," Pattern Recognition, vol. 30, no. 1, pp. 85-97, 1997.
9. Q. Ji and R. M. Haralick, "Corner detection with covariance propagation," Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 362-367, 1997.
10. K. J. Lee and Z. Bien, "Grey-level corner detector using fuzzy logic," Pattern Recognition Letters, vol. 17, no. 9, pp. 939-950, 1996.
11. J. Q. Fang and T. S. Huang, "A corner finding algorithm for image analysis and registration," Proc. AAAI Conf., pp. 46-49, 1982.
12. W. C. Chen and P. Rockett, "Bayesian labeling of corners using a grey-level corner image model," IEEE Int'l Conf. Image Processing, vol. 1, pp. 687-690, 1997.
13. Z. O. Wu and A. Rosenfeld, "Filtered projections as an aid to corner detection," Pattern Recognition, vol. 16, no. 31, 1983.
14. K. Paler, J. Foglein, J. Illingworth, and J. Kittler, "Local ordered grey levels as an aid to corner detection," Pattern Recognition, vol. 17, no. 5, pp. 535-543, 1984.
15. K. Rangarajan, M. Shah, and D. Van Brackle, "Optimal corner detector," Computer Vision, Graphics, and Image Processing, vol. 48, pp. 230-245, 1989.
16. F. Arrebola, A. Bandera, P. Camacho, and F. Sandoval, "Corner detection by local histograms of contour chain code," Electronics Letters, vol. 33, no. 21, pp. 1,769-1,771, 1997.
17. E. Shilat, M. Werman, and Y. Gdalyahu, "Ridge's corner detection and correspondence," Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 976-981, 1997.
18. S. Nassif, D. Capson, and A. Vaz, "Robust real-time corner location measurement," Proc. IEEE Conf. Instrumentation and Measurement Technology, pp. 106-111, 1997.
19. K. Sohn, J. H. Kim, and W. E. Alexander, "Mean field annealing approach to robust corner detection," IEEE Trans. Systems, Man, and Cybernetics, vol. 28B, no. 1, pp. 82-90, 1998.
20. X. Zhang and D. Zhao, "Parallel algorithm for detecting dominant points on multiple digital curves," Pattern Recognition, vol. 30, no. 2, pp. 239-244, 1997.
21. K. Kohlmann, "Corner detection in natural images based on the 2-D Hilbert Transform," Signal Processing, vol. 48, no. 3, pp. 225-234, 1996.
22. R. Mehrotra, S. Nichani, and N. Ranganathan, "Corner detection," Pattern Recognition, vol. 23, no. 11, pp. 1,223-1,233, 1990.
23. E. R. Davies, "Application of the generalized Hough Transform to corner detection," IEE Proc., vol. 135, pp. 49-54, 1988.
24. O. A. Zuniga and R. M. Haralick, "Corner detection using the facet model," Proc. Conf. Pattern Recognition and Image Processing, pp. 30-37, 1983.
25. S. M. Smith and J. M. Brady, "SUSAN—A new approach to low level image processing," Defense Research Agency, Technical Report no. TR95SMS1, Farnborough, England, 1994.
26. C. M. Orange and F. C. A. Groen, "Model based corner detection," Proc. IEEE Conf. Computer Vision and Pattern Recognition, 1993.

27. Bejamin Bell and L. F. Pau, “Contour tracking and corner detection in a logic programming environment,” IEEE Trans. On Pattern Analysis and Machine Vision, vol. 12, no. 9, September 1990.
28. H. C. Liu and M. D. Srinath, “Corner detection from chain-code,” Pattern Recognition, vol. 23, nos. ½, pp. 51-68, 1990.
29. H. Freeman and L. Davis, “A corner-finding algorithm for chain-coded curves,” IEEE Trans. Coputers, vol. 26, pp. 297-303, 1977.
30. A. Rattarrangsi and R. T. Chin, “Scale-based detection of corners of planar curves,” IEEE Trans. On Pattern Analysis and Machine Vision, vol. 14, no. 4, pp. 430-449, 1992.
31. Farzin Mokhtarian and Riku Suomela, “Robust image corner detection through curvature scale space,” IEEE Trans. On Pattern Analysis and Machine Vision, vol. 20, no. 12, December, 1998.
32. F. Mokhtarian and A.K. Mackworth, “A theory of Multi-Scale, Curvature-based shape representation for planar curves,” IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 14, no. 8, pp. 789-805, Aug. 1992.
33. F. Mokhtarian and R. Suomela, “Curvature Scale Space for robust image corner detection,” Int’l Conf. Pattern Recognition, Brisbane, Australia, 1998.