World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:6, No:8, 2012

Feature Point Detection by Combining Advantages of Intensity-based Approach and Edge-based Approach

Sungho Kim, Chaehoon Park, Yukyung Choi, Soon Kwon, In So Kweon

***Abstract*—**In this paper, a novel corner detection method is presented to stably extract geometrically important corners. Intensity-based corner detectors such as the Harris corner can detect corners in noisy environments but has inaccurate corner position and misses the corners of obtuse angles. Edge-based corner detectors such as Curvature Scale Space can detect structural corners but show unstable corner detection due to incomplete edge detection in noisy environments. The proposed image-based direct curvature estimation can overcome limitations in both inaccurate structural corner detection of the Harris corner detector (intensity-based) and the unstable corner detection of Curvature Scale Space caused by incomplete edge detection. Various experimental results validate the robustness of the proposed method.

***Keywords*—**Feature, intensity, contour, hybrid.

I. Introduction

An interest point is one that has a location in space but no spatial extent. The presence of interest points can drastically reduce the required computation time; as such, these points are frequently used to compensate for many vision problems such as camera calibration, 3D reconstruction, stereo matching, image registration, structure from motion, image mosaicing, motion tracking, mobile robot navigation, and object recognition to find correspondences [1]. Many different interest point detectors have been proposed with a wide range of definitions. Some detectors find points of high local symmetry [2], [3], whereas others locate corner points. Corner points are more frequently used to solve correspondence problems, as they are formed from two or more edges that define the boundary between different objects or parts of the same object.

Corner detector should have to satisfy several criteria. First, all true corners should be detected. Second, no false corners should be detected. Third, the corner points should be well localized. Fourth, the most important property of a corner detector should be its high repeatability rate. Fifth, the corner detector should be robust with respect to noise and should be computationally efficient [1].

S. Kim is with the LED-IT Fusion Technology Research Center and Dept. of Electronic Engineering, Yeungnam University, Korea (e-mail: sunghokim@ ynu.ac.kr).

C. Park, Y. Choi, and I. S. Kweon are with the Dept. of Electrical Engineering, KAIST, Korea.

S. Kweon is with the DGIST, Korea.

To achieve these kinds of criteria, a number of corner detectors were proposed, such as the intensity-based approach, contour-based approach, biologically motivated approach, color-based approach, model- or parameter-based approach, segmentation-based approach, viewpoint invariant approach, and machine learning-based approach [4], [1]. In this paper, we focus on the first two approaches (the intensity-based and the contour-based approaches) since they are basic methods used for corner detection problems. The Harris corner detector, one of the most successful algorithms in the intensity-based approach [5], is based on a matrix related to the autocorrelation function. Corner points are detected if the autocorrelation matrix has two significant eigenvalues. Recently, Schmid et al. improved the original Harris corner detector using a Gaussian derivative kernel instead of simple derivative kernel [4]. In this paper, we call it impHarris. The impHarris shows the highest repeatability among the conventional Harris, Foerstner, Cottier, heitger, and Horaud corner detectors.

Likewise, contour-based methods have existed for a long time. These were originally applied to line drawings and machine parts rather than natural scenes. Another popular contour-based corner method is the Curvature Scale Space (CSS)-based algorithm [6]. Corner points are curvature maxima of contours at a coarse level and are tracked locally up to the finest level. The two sets are compared and close interest points are merged. Recently, He and Yung improved the original CSS corner detector by introducing the adaptive curvature threshold and a dynamic region of support. We call this method impCSS.

In this paper, we propose a novel corner detector by combining the advantages of both approaches by directly estimating curvature on the intensity image using spatial filtering methods. An orientation field is obtained and a curvature field is then generated by application of an approximated curvature estimation filter to the orientation field. Local maxima and thresholding can detect structurally important corners for both structural and textured images.

This paper is organized as follows. Section 2 explains the key idea of the proposed method including the overall corner detection framework. Section 3 presents details of the spatial filtering and detection method for good corner detection. Section 4 shows various performance evaluations and results. And finally, Section 5 concludes this paper.

II. Motivation and Proposed Method

In this section, we briefly introduce corner detector basics and explain our key idea to improve corner detection

performance. We then present the framework of the proposed corner detection method.

A. Related works

This paper is motivated from well-known corner detectors such as the intensity-based impHarris [4] and contour-based impCSS [7]. In this section, we briefly introduce the basics of these methods. The impHarris method is an improved version of the original Harris corner detector [5]. As shown in Fig. 1(a), the impHarris computes image derivatives (I_x, I_y) using Gaussian derivatives ($\sigma = 1$) which is the improvement point. An autocorrelation matrix **A** is then calculated using a Gaussian convolution ($\sigma = 2$) to weight the derivatives summed over the window. Instead of direct eigenvalue extraction of **A** , the corner strength of an interest point is calculated using $det(\mathbf{A}) - \alpha \cdot \mathbf{trace}(\mathbf{A})^2$. The second term is used to remove edge points with one strong eigenvalue. α is normally set to 0.06 . After non-maximum suppression using 3×3 window, final impHarris corners are detected with a threshold.

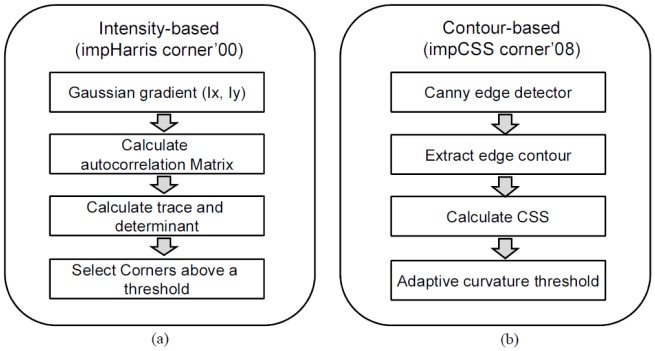


Fig. 1 Corner detection flows of previous works: (a) Intensitybased method (impHarris), and (b) contour-based method (impCSS)

The contour-based impCSS corner detector improved the conventional CSS method [6] by carefully designing the selection mechanism as shown in Fig. 1(b). The philosophy of the impCSS method is to use global and local curvature properties. The first step is to obtain a binary edge map using a Canny edge detector. Then, edge contours are extracted by edge linking as the original CSS method. After the contours are extracted, the curvature is calculated. The adaptive threshold is then estimated using support regions. Finally, the end points of the open contours also considered corner points.

There are also image-based curvature estimation methods. Donias et al. proposed implicit curvature calculation using differential geometry as (1) where I_x, I_y denote the 1st derivatives along row-direction and column direction, respectively. The theoretic derivations are useful, but application results are quite disappointing as shown in Fig. 2(b) testing of the input image of Fig. 2(a). This produces strong curvature responses around slanted edges.

The intensity-based Gaussian curvature calculation method such as $(I_{xx}I_{yy} - I_{xy}^2)/(1+I_x{}^2 + I_y{}^2)^2$ [8] also produces strong double curvature responses along the slanted edges as shown in Fig. 2(c).

Ginkel et al. also proposed an image-based curvature estimation method using geometric analysis and showed good performance on low signal to noise ratio but weak to strong curvature [9].

$$k = \frac{2I_x I_y I_{xy} - I_x^2 I_{yy} - I_y^2 I_{xx}}{(I_x^2 + I_y^2)^{3/2}} \quad (1)$$

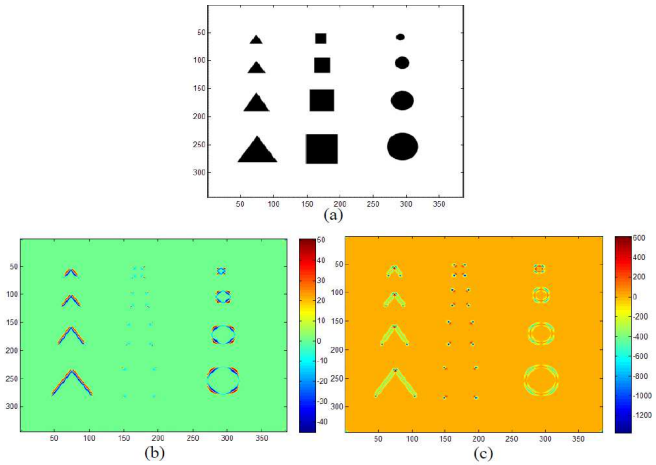


Fig. 2 Image-based curvature estimation results: (a) Input test image; (b) implicit curvature method [10]; and (c) Gaussian curvature.

Both impHarris and impCSS corner detectors have their own advantages and limitations. In general, the impHarris corner detector is robust to textured images due to image filtering but offers poor detection of obtuse corners and shows shifted corner positions (Fig. 3). The shifted corner detection as shown in Fig. 3(a) is originated from the Gaussian derivatives and the additional smoothing in the computation of autocorrelation matrix. The impHarris detects only strong corners such as those with an "L" shape or "T" junction, which have two significant eigenvalues. An obtuse angular structure generates only one significant eigenvalue, which leads to the corner missing problem shown in Fig. 3(b). Conversely, use of the CSS corner detector is powerful for structured objects or line drawings due to its edge-based curvature estimation but is poor in textured images with inaccurate edge extraction (Fig. 4).

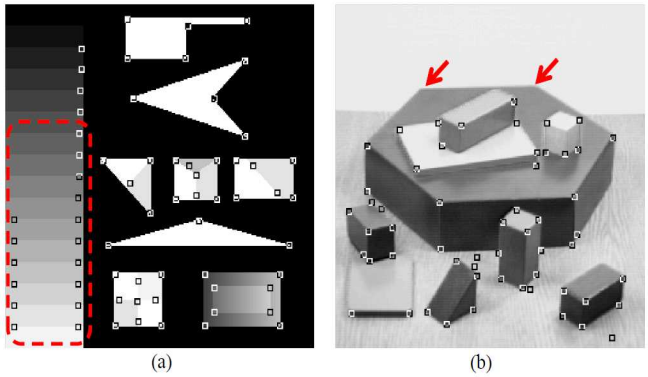
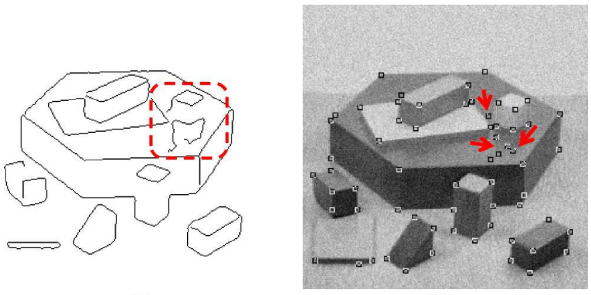


Fig. 3 Limitations of the impHarris corner detector: (a) Inaccurate corner locations; and (b) missing obtuse angular corners ($\alpha = 0.06$)



(a) (b)

Fig. 4 Limitations of the impCSS corner detector: (a) Canny edge detector used in the impCSS; and (b) false corner detection due to unstable edge detection.

B. Proposed feature detection system

As discussed above, the intensity-based corner detection is robust when used in textured images due to its image filtering scheme but weak when used to detect structurally meaningful corners such as obtuse angles and has low positional accuracy. In contrast, the contour-based corner detector is powerful when used to detect structured objects due to its curvature estimation strategy but is weak when used to detect textured images due to its fragile Canny edge detector. The motivation of our research starts at this point: how can we use the advantages of both approaches to detect corners stably in general scenes? Since evidence exists that the human visual system pays strong spatial attention to contour curvatures [11], we use the curvature-based approach as a basic corner detector. The next question is how to stably extract curvature information from textured or noisy images. Our approach is to adopt the underlying scheme of the intensity-based approach to alleviate the edge extraction process problem. The intensity-based method is usually based on a spatial filter. In the case of impHarris corner detector, it uses image-based filters such as the derivative filter or the autocorrelation filter. As such, we estimate the curvature information directly in the image space by eliminating the edge detection process. Fig. 5 summarizes the key idea and the proposed corner detection system. All tables and figures you insert in your document are only to help you gauge the size of your paper, for the convenience of the referees, and to make it easy for you to distribute preprints.

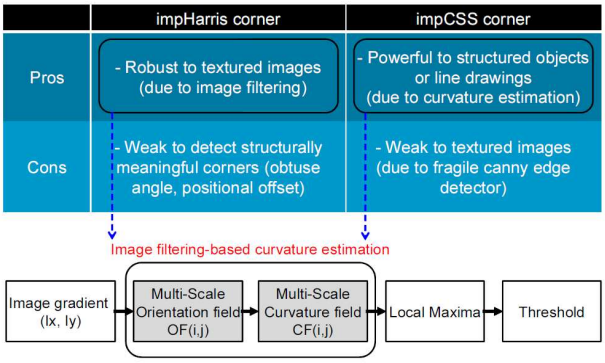


Fig. 5 Motivation of the proposed method and the related block diagram

The proposed corner detection system consists of spatial filtering part and detection part. The filtering part conducts direct curvature estimation by applying the curvature filter after the orientation filter. The corner detection part performs local maxima on the curvature field and the final corners are extracted by the application of a threshold. The key contribution of this paper is conducting a multi-scale curvature estimation on the intensity image space instead of the edge-based contour space to detect the structurally accurate corners for both textureless and textured objects. The orientation filter produces orientation flow image, called the orientation field (OF), from an input image. Pixel-wise approximate curvature filtering on the OF generates the curvature intensity image, called the curvature field (CF). The global thresholding method detects the final corner points after the local maxima. The spatial filter and corner detection process is repeated for the next pyramid image to detect larger structural corners. We call the proposed corner detector CF corner in the following sections. Since the CF corner detector combines the advantages of both approaches, we can expect both robust detection of structurally meaningful corners and accurate localization of the corner position, even in textured or noisy environments. This method will be validated in the experimental section.

III. Estimation of OF and CF

A. OF

The proposed spatial filter consists of two steps. The OF ($OF(i,j)$) is obtained in advance and then CF ($CF(i,j)$) is estimated. Since we do not use the edge extraction process, the orientation calculation is critical to the consecutive processes. As such, an initial input of $I(i,j)$ is pre-processed using using Gaussian smoothing with $\sigma = 1.4$. The orientation of each pixel can be calculated simply using (2).

$$OF_{simple}(i,j) = mod\left\{tan^{-1}\left(\frac{I_y}{I_x}\right), \pi\right\} + \frac{\pi}{2} \quad (2)$$

where I_x, I_y denote the row and column directional gradient, respectively, with a kernel coefficient $[-101]$. We use the orientation range of $[0, \pi]$ instead of $[-\pi, \pi]$ to consider shape direction only and not polarity.

A simpler orientation estimation method proposed by Kass and Witkin [12] can directly calculate orientation flow without the use of a modulus operator. They derived image flow orientation in terms of power spectrum analysis as shown in (3). This can be easily derived by vector analysis. Assume a gradient vector $G = I_x + I_y i$ whose power is $G^2 = (I_x + I_y i)^2 = I_x^2 - I_y^2 + 2I_x I_y i$. As a result, the angle of gradient power is defined as shown in (3). Fig. 6 shows OF examples calculated using the OF_{simple} and OF_{flow} methods. Note that both methods produces the same results. In this paper, we use (3) since it needs not the modulus computation.

$$OF_{flow}(i,j) = \frac{1}{2} tan^{-1}\left\{\frac{2I_x I_y}{I_x^2 - I_y^2}\right\} + \frac{\pi}{2} \quad (3)$$

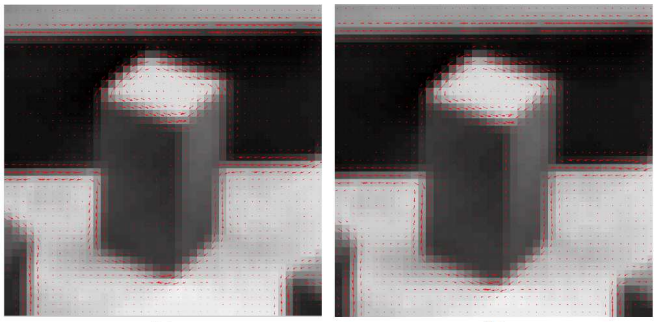


Fig. 6 Orientation field estimation results: (a) OF_{simple} method; and (b) OF_{flow} method. The arrows indicate calculated orientations

B. CF

The next step is to estimate a CF ($CF(i, j)$). Curvature is originally defined as the rate of change of orientation over spatial variation as shown in Fig. 7(a) [13]. Given an extracted contour, the ideal curvature is defined as (4), where ΔS denotes infinitesimal contour length and $\Delta\theta$ denotes orientation variation on the contour position.

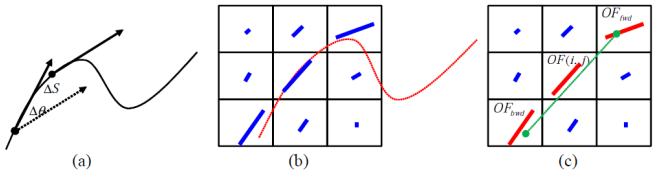


Fig. 7 Curvature field estimation procedures: (a) ideal curvature estimation given a contour; (b) calculated orientation field (over which the ideal contour is overlaid.); and (c) approximate curvature estimation diagram

Since we do not use edge or contour extraction process, we have to use approximate curvature estimation method in image domain. As shown in Fig. 7(b), tthe ideal contour is quantized into pixels and the OF has implicit contour information. As such, if we carefully design a certain filter to be applied on in the OF, we can then obtain approximated curvature information. As shown in Fig. 7(b), we do not have any information about contour pixels in advance, so all pixels in the OF are considered candidate contours. Curvature approximation in the OF can be achieved as shown in Fig. 7(c). Assume that the current pixel of an OF is (i, j). We can then make a local contour pixel segment using the orientation information, $OF(i, j)$. Extending along that direction, contour segment pixels are selected in a 3×3 window. If we use the direction information of neighboring pixels ($OF_{fwd}(i, j), OF_{bwd}(i, j)$), the approximate curvature (k_{sel}) can be estimated using (5), where ΔS can be considered as 2 (pixel distance) and $\Delta\theta$ can be approximated as the neighboring orientation difference ($\theta_{fwd} - \theta_{bwd}$). k_{sel} denotes curvature estimation by neighboring pixel selection. Neighboring pixel pairs are selected by quantizing the direction of the center pixel into four angles such as $0^\circ, 45^\circ, 90^\circ, 135^\circ$.

$$k_{sel}(i, j) = \frac{\Delta\theta}{\Delta S} = \frac{\left\| OF_{fwd}(i, j) - OF_{bwd}(i, j) \right\|}{2} \quad (5)$$

We can also consider another curvature estimation as shown (6), in which orientation differences between neighboring pixels and a center pixel are calculated and summed. W denotes a local window around (i, j). In this approach, we need not find the contour segments. k_{sum} denotes the curvature estimation by summing the orientation differences of the neighboring pixels. A performance comparison of the curvature estimation methods will be presented in the experimental results section.

$$k_{sum}(i, j) = \frac{1}{8} \sum_{k,l \in W, k \neq i, l \neq j} \left\| OF(k,l) - OF(i, j) \right\| \quad (6)$$

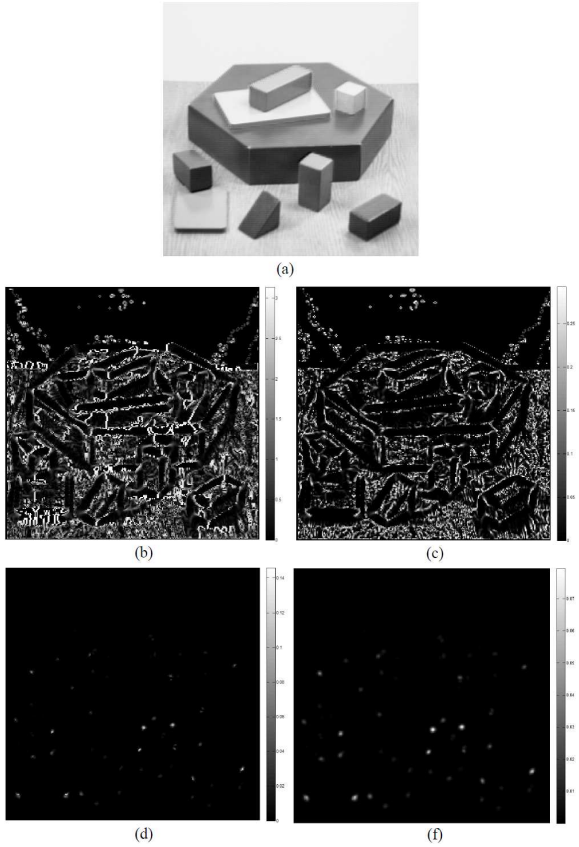


Fig. 8 Curvature estimation results using (b) k_{sel}, (c) k_{cosine}, (d) proposed, and (e) additional smoothing, for a given test image (a)

However, we cannot use this curvature information because it produces many false responses around the homogeneous area including the edges as shown in Fig. 8(b) for a given test image (Fig. 8(a)). If we use cosine angle distance [14] as shown in (7) instead of the angle difference, we can enhance the curvature response while maintaining strong curvature around the homogeneous region and edges as shown in Fig. 8(c). As such, we modify (7) by adaptive weighting using gradient magnitude (M_{fwd}, M_{bwd}) as defined in (8). Fig. 8(d) shows the obtained CF estimation using (8). Note that there are strong responses around the true corners. Some noisy curvature responses can be reduced further by a simple smoothing as shown in Fig. 8(e).

$$k_{cosine}(i, j) = (1 - cos(k_{sel}(i, j))) \quad (7)$$

$$CF(i,j) = k_{cosine}(i,j) \cdot M_{fwd} \cdot M_{bwd} \quad (8)$$

Fig. 9 shows the overall corner detection procedures using the proposed orientation and curvature filters for the OF and CF calculations. An input image consists of three regions with different shades in the rectangle. In Step 1, the modified orientation estimation filter produces the OF. In Step 2, the proposed curvature estimation filter produces the final CF. Note the curvature responses around the interior regions. We can obtain final corner detection results through a local maxima (3×3 window) and threshold as shown in the block diagram of Fig. 5.

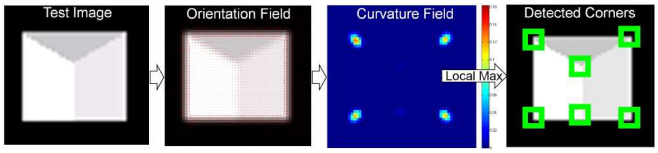


Fig. 9 Example of corner detection process using the proposed direct curvature estimation method

IV. Experimental Results

The first evaluation is the corner localization accuracy, which can be important to camera calibration, 3D reconstruction, and so on. We use the "synthetic" test image since we can know the exact corner location. The ground truth location is prepared by human vision to evaluate the location error. In addition, thresholds are tuned to produce almost the same number of corners for the CF, impHarrs, and impCSS corners. Fig. 10 represents the evaluation results. The squares denote the detected corners while the crosses indicate the ground truth corner locations. The average localization error of the CF corner is 1.17 pixel, that of impHarris corner is 1.74 pixel, and that of impCSS corner is 1.44 pixel. As a result, the proposed CF corner has the lowest localization error, followed by the impCSS corner, and then the impHarris corner.

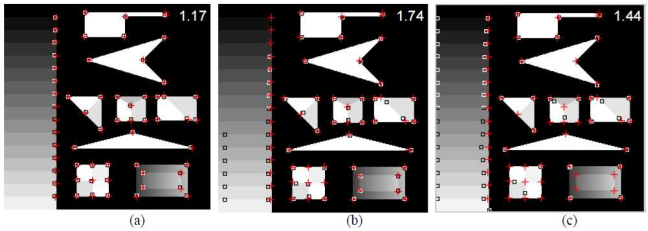


Fig. 10 Comparison of corner localization error using (a) proposed CF corner, (b) impHarris corner, and (c) impCSS corner, where the squares represent the detected corners and the crosses (+) represent the ground truth locations

The second evaluation is of noise sensitivity of the corner detectors. Gaussian noise is added by changing the standard deviation from 0 to 20. In this case, we use the "blocks" image and check the recall vs. (1-precision) as a comparison measure. The threshold of each method is tuned to produce the same number (around 58) of corners at noise level 0. Fig. 11 shows the comparison results.

At a glance, impHarris seems to be robust to noise, CF corner reacts normally, and impCSS performs the worst. However, if we inspect the corner detection images as shown in Fig. 12, impHarris generates a lot of corner detection, which leads to a high recall rate. The impCSS corner detector also produces many false corners in noisy homogeneous regions. The proposed CF corner detector shows more stable detection around corners compared with other methods.

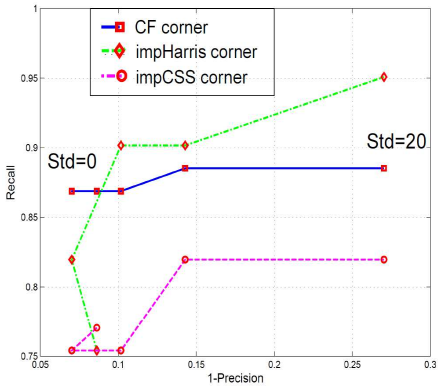


Fig. 11 Comparison of the image noise sensitivity in terms of recall vs. (1-precision) curve

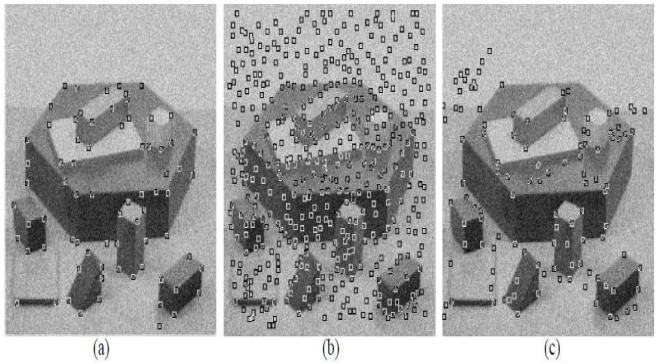


Fig. 12 Corner detection examples at a noise level 20 using: (a) CF corner; (b) impHarris corner; and (c) impCSS corner

The fourth evaluation is the consistency of corner detection in image transformations. We use the repeatability measure to quantify the consistency. Repeatability is important to detect corners in sequences where correspondence should be achieved among image transformations. The test images include "blocks," "house," and "lab" data. The repeatability tests are conducted in terms of image rotation and scale change. As a result, we compute the repeatability by counting matched corners between a reference image and transformed images. Since the transformation value is available, we can predict the ground truth of the corner positions. Fig. 13 summarizes the repeatability comparisons in terms of image rotation and scale for the standard test images. We use rotation range of $[0^\circ, 90^\circ]$ with an interval 0.5° and scale range of $[1, 2]$ with an interval of 0.1. The proposed CF corner detector shows upgraded repeatability performance compared with the impHarris and impCSS methods.

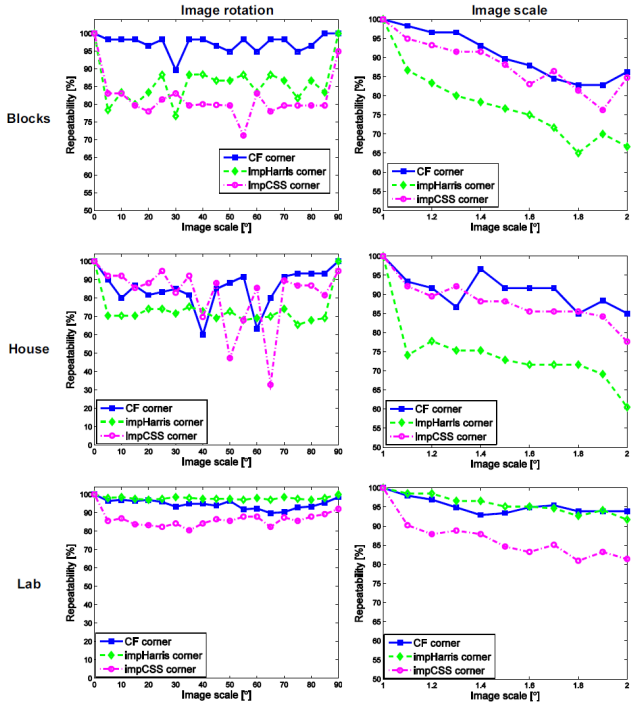


Fig. 13 Repeatability evaluation results for image rotation and scale using standard test images (blocks, house, lab). The row represents image type, the first column represents image rotation, and the second column represents image scale

V. Conclusion

This paper proposed a new simple but powerful corner detection method for detecting structurally important corners using direct curvature estimation filters. As validated by a set of experiments, use of the OF estimation filter followed by approximated curvature estimation filter can effectively find true corners, including obtuse corners with stable corner positions and image variations, such as image rotation and scale changes. Due to the simplicity of the algorithm, the proposed corner detection method can be used in various vision applications.

Acknowledgment

This research was supported by National Strategic R&D Program for Industrial Technology, Korea and by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (No. 2011-0009684). It was also supported by the DGIST R&D Program of the Ministry of Education, Science and Technology of Korea (12-BD-0202) and by the 2012 Yeungnam University Research Grants.

References

[1] T. Tuytelaars and K. Mikolajczyk, Local Invariant Feature Detectors: A Survey. Hanover, MA, USA: Now Publishers Inc., 2008.
[2] G. Loy and A. Zelinsky, "Fast radial symmetry transform for detecting points of interest," *PAMI*, vol. 25, no. 8, pp. 959–973, 2003.
[3] D. Reisfeld, H. Wolfson, and Y. Yeshurun, "Context-free attentional operators: The generalized symmetry transform," *IJCV*, vol. 14, no. 2, pp. 119–130, 1995.
[4] C. Schmid, R. Mohr, and C. Bauckhage, "Evaluation of interest point detectors," *IJCV*, vol. 37, no. 2, pp. 151–172, 2000.
[5] C. Harris and M. Stephens, "A combined corner and edge detector," in *4th Alvey Vision Conference*, 1988, pp. 147–151.
[6] F. Mokhtarian and R. Suomela, "Robust image corner detection through curvature scale space," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 12, pp. 1376–1381, 1998.
[7] X. C. He and N. H. C. Yung, "Corner detector based on global and local curvature properties," *Optical Engineering*, vol. 47, no. 5, p. 057008, 2008.
[8] W. Kuhnel, *Differential Geometry: Curves - Surfaces - Manifolds*. American Mathematical Society, 2002, vol. 2.
[9] M. van Ginkel, J. van de Weijer, L. J. van Vliet, and P. W. Verbeek, "Curvature estimation from orientation fields," in *SCIA*, 1999, pp. 545–551.
[10] M. Donias, P. Baylou, and N. Keskes, "Curvature of oriented patterns: 2-d and 3-d estimation from differential geometry." in *ICIP (1)*, 1998, pp. 236–240.
[11] J. D. Winter and J. Wagemans, "Perceptual saliency of points along the contour of everyday objects: a large-scale study." *Percept Psychophys*, vol. 70, no. 1, pp. 50–64, 2008.
[12] M. Kass and A. P. Witkin, "Analyzing oriented patterns," *Computer Vision, Graphics, and Image Processing*, vol. 37, no. 3, pp. 362–385, 1987.
[13] J. Stoker, *Differential Geometry*. John Wiley & Sons Inc., 1969.
[14] G. Qian, S. Sural, Y. Gu, and S. Pramanik, "Similarity between euclidean and cosine angle distance for nearest neighbor queries," in *Proceedings of 2004 ACM Symposium on Applied Computing*. ACM Press, 2004, pp. 1232–1237.
[15] M. Awrangjeb and G. Lu, "Robust image corner detection based on the chord-to-point distance accumulation technique," *IEEE Transactions on Multimedia*, vol. 10, no. 6, pp. 1059–1072, 2008.
[16] J. Klippenstein and H. Zhang, "Quantitative evaluation of feature extractors for visual slam," in *Proceedings of the Fourth Canadian Conference on Computer and Robot Vision*. Washington, DC, USA: IEEE Computer Society, 2007, pp. 157–164.
[17] L. Teixeira, W. C. Filho, and M. Gattass, "Accelerated corner-detector algorithms." in *BMVC*. British Machine Vision Association, 2008.
[18] A. Willis and Y. Sui, "An algebraic model for fast corner detection," in *ICCV*, 2009, pp. 2296–2302.
[19] S. M. Smith 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.
[20] S. C. Bae, I. S. Kweon, and C. D. Yoo, "COP: a new corner detector," *Pattern Recognition Letters*, vol. 23, pp. 1349–13 608, 2002.