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].

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 \( \langle I_x, I_y \rangle \) 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 \( \det(A) - \alpha \cdot \text{trace}(A)^2 \). The second term is used to remove edge points with one strong eigenvalue. \( \alpha \) is normally set to 0.06. After non-maximum suppression using 3x3 window, final impHarris corners are detected with a threshold.

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].

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

(1)

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).

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).
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 \((\text{OF}(i,j))\) is obtained in advance and then CF \((\text{CF}(i,j))\) 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 Gaussian smoothing with \(\sigma = 1.4\). The orientation of each pixel can be calculated simply using (2):

\[
\text{OF}_{\text{simple}}(i,j) = \text{mod} \left( \tan^{-1}\left( \frac{I_x}{I_y} \right), \pi \right) + \frac{\pi}{2}
\]

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 \(G = I_x + i I_y\) whose power is \(G^2 = (I_x + i I_y)^2 = I_x^2 + I_y^2 + 2I_x I_y\). As a result, the angle of gradient power is defined as shown in (3). Fig. 6 shows OF examples calculated using the \(\text{OF}_{\text{simple}}\) and \(\text{OF}_{\text{flow}}\) methods. Note that both methods produces the same results. In this paper, we use (3) since it needs not the modulus computation.

\[
\text{OF}_{\text{flow}}(i,j) = \frac{1}{2} \tan^{-1} \left( \frac{2I_x I_y}{I_x^2 - I_y^2} \right) + \frac{\pi}{2}
\]
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 ($CF_{ij}$). 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 $\Delta S$ denotes infinitesimal contour length and $\Delta \theta$ denotes orientation variation on the contour position.

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), the 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 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, $OF_{ij}$. Extending along that direction, contour segment pixels are selected in a $3 \times 3$ 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 $\Delta S$ 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 $0^\circ, 45^\circ, 90^\circ, 135^\circ$.

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

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. $\Psi$ 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.

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

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.

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_f,

$$M_b$) 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).

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

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

The first evaluation is the corner localization accuracy, which can be important for 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, impHarris, 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.

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.

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, 90] 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.
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.

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