Detecting and matching feature points

Etienne Vincent and Robert Laganière*

School of Information Technology and Engineering, University of Ottawa, Ottawa, Canada K1N 6N5

Received 22 March 2002; accepted 6 May 2004
Available online 24 June 2004

Abstract

This paper proposes a new feature point detector which uses a wedge model to characterize corners by their orientation and angular width. This detector is compared to two popular feature point detectors: the Harris and SUSAN detectors, on the basis of some defined quality attributes. It is also shown how feature points between widely separated views can be matched by using the information provided by the detector to approximate local affine transformations between them.

Keywords: Feature point; Corner detection; Feature-based matching; Widely separated view matching

1. Introduction

The robust detection of feature points constitutes a fundamental step in image characterization and matching. Feature points usually correspond to particular patterns exhibiting significant intensity variations in more than one direction. Although they might belong to a large variety of high frequency textures, these special points are often designated as corners.

Many feature point detectors can be found in the literature, and the results that they produce vary considerably. To evaluate their performance, the desirable properties that a good feature point detector should exhibit must first be established. Different criteria have been considered in past works; these may be summarized as follows:

* Corresponding author.
E-mail addresses: evincent@site.uottawa.ca (E. Vincent), laganier@site.uottawa.ca (R. Laganière).

1047-3203/$ - see front matter © 2004 Elsevier Inc. All rights reserved.
doi:10.1016/j.jvcir.2004.05.001
(1) **Accuracy**, or the ability to consistently detect a given image pattern, at the exact same location, in spite of minor variability in intensity values, in orientation, or in scale. This property is significant when the detector is tuned to detect well-defined structures, such as specific types of junctions or corners. Accuracy can be assessed by measuring the alignment of extracted features located on a straight line, or, as in (Laganière, 1998), by measuring the distance between a detected junction and the point of intersection of the two lines defining it.

(2) **Robustness**, or insensitivity to noise. Detection on noisy images can produce false positives, corresponding to noise patterns rather than true feature points. Also, in the presence of noise, some feature points can be lost or not localized properly. Robustness can be evaluated empirically, or theoretically, as in (Smith and Brady, 1995), where a probabilistic analysis of a corner model is used.

(3) **Sensitivity** of the detection, that is the ability to detect feature points in low contrast conditions. Usually, there are some parameters which control the sensitivity of a detector, and most often, a tradeoff exists between sensitivity and robustness.

(4) **Stability** of the detection. A detected feature point should continue to be detected after an image undergoes some geometrical transformation (especially the perspective deformation due to a change in viewpoint), and under different conditions of illumination. This is essential in the context of multi-view matching. A good measure of stability is the repeatability of detected features across several views (see Section 4.2).

(5) **Controllability** of a detector. This is mainly determined by the number of parameters which control its behavior, and their relative sensitivity. It is certainly useful to be able to control the number of feature points which will be selected in an image, ideally using a single control parameter. Other parameters might also be used to filter out certain kinds of features, not of interest in a specific application. However, the effect of each parameter should be specific and predictable enough to allow an easy tuning.

(6) **Richness** of the information provided about the detected feature points. When a detector returns various characteristics of feature points, and not just a strength measure, the additional information can be exploited in the task to follow. For example, it could be used to classify feature points into categories (e.g., type of junction Parida et al., 1998), or in matching, as a tool to normalize the patterns being matched (as in Section 5).

(7) **Variability** in the characteristics of detected feature points. A high variability ensures that several feature points are detected, regardless of the nature of the image under analysis. Good variability is also critical for matching, where the feature points must be easily distinguishable from each other.

(8) **Complexity** of the detector, or the speed at which it identifies corners in an image. In many applications, feature point detection is a preprocessing operation that must be performed at frame rate. However, a comparison of detection speeds is difficult to achieve, since the efficiency of a given feature point detector depends on its implementation. Nevertheless, the complexity of the operations which must be applied to each image point must remains sufficiently low.
These are the main quality attributes that should be used to evaluate feature point detectors. Depending on the application, some of these might become more or less important. However, it is expected that a good feature point detector will behave well with respect to all of these criteria.

A novel feature point detector is proposed in this work. It aims to find feature points which can be adequately represented by a simple wedge corner model, and is described in Section 3. To assess the performance of this detector, experiments are proposed in Section 4. Then, in Section 5, it is seen how the corner information provided by this detector can be used to improve the reliability of sparse feature point matching, in the context of widely separated views.

2. Feature point detection

The most commonly used feature point detectors rely on image intensity derivatives to identify high curvature points. These will be reliable, as long as they include some smoothing operation to reduce sensitivity to noise. The most popular detector in this category was proposed by Harris and Stephens (1988), and is based on the image gradient's autocorrelation matrix:

\[ C(x,y) = S \ast (\nabla I \cdot \nabla I^T) = S \ast \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}, \]

where \( S \) is a smoothing operator. At the point where it is computed, the greatest eigenvalue of this matrix corresponds to the image's rate of change in the direction of highest variation, while its smallest eigenvalue corresponds to the rate of change in the perpendicular direction. If the smallest eigenvalue has a high magnitude, it means that, at the considered point, the image has a high rate of variation in at least two directions, and thus, that the point is in a high curvature region. Originally, this condition was tested using the following strength measure:

\[ \text{cornerStrength}(x,y) = \text{Det}(C(x,y)) - 0.04 \cdot \text{Trace}^2(C(x,y)), \]

with \( C(x,y) \) defined in Eq. (1). However, direct computation of the eigenvalues is now preferred.

In other approaches, detection is based on the direct comparison of intensity values inside small predefined windows. The SUSAN operator Smith and Brady, 1995, proposed by Smith and Brady, fits into this category. This detector compares the intensity of a pixel with its neighbors, to define a univalue segment assimilating nucleus (USAN), i.e., the set of points inside a circular region having similar intensities as its center. The size and the symmetrical axis of this nucleus then determine the presence of a feature point.

Finally, some detectors define feature point models, and search for image regions which match them (Deriche and Blaszk, 1993; Guiducci, 1998; Rohr, 1992). The selected parameterized models generally represent ideal corners, and the detection consists in determining the value of the model's parameters which
best fit the underlying intensity pattern. Depending on the approach taken to optimize these parameters, the complexity of this kind of detector can be quite high.

3. The proposed detector

3.1. The corner model

The corner detector presented here uses an intensity-based approach that makes use of a simple corner model to detect corner shapes from intensity patterns. The goal was to devise a stable and efficient corner detector, suitable for feature matching across different views. The corner model consists of a wedge (the corner), having its origin on the center of a circular neighborhood (the background). This idealized corner is described by two parameters: an angular position, $\theta$, and an angular width, $\phi$ (see Fig. 1). Furthermore, the wedge’s width must be within a reasonable angular range, i.e., $\phi_{\text{min}} < \phi < \phi_{\text{max}}$.

This model is similar to the one used by Rohr (1992), who however, also explicitly includes the strength of a gaussian blurring on the corner, and the image intensity of the wedge and of the background, which are specifically required to be uniform. Rohr’s corner model was in turn inspired by Berzins (1984), who worked on the problem that Laplacian edge detectors encounter near corners. Guiducci (1998) uses a model which is similar to Rohr’s, but only looks at the difference in intensity between the wedge and its background. The model is then fitted analytically from image derivatives rather than directly from the image intensities. Deriche and Blaszka (1993) use the same model as Rohr, but consider a more efficient implementation to the fitting. On the other hand, Marr (1976) proposes a much simpler model, only allowing for wedges with a width of 90°.

To identify corners in an image, each pixel location should be examined by comparing its surrounding circular neighborhood with the ideal corner model. To do so, the following process is proposed:

![Fig. 1. The corner model: a wedge $W_{\theta,\phi}$ at angular position $\theta$ and having an angular width of $\phi$.](image)
The circular window's intensity mean and variance are computed. The variance must be over a given threshold $\sigma^2_{\text{min}}$. This first parameter controls the sensitivity of the corner finder by discarding low contrast areas which do not correspond to corners but might otherwise generate false positives.

The circular area around the potential corner is segmented into a background and a foreground, following a simple classification scheme described in Section 3.2.

The corner model is fitted to the extracted foreground, by finding the values for $\varphi$ and $\theta$ that best approximate the area, as explained in Section 3.3. The strength of the corner is then determined by comparing the segmented area to the parameterized corner model.

Applying this procedure to all pixels results in a corner map where each feature point is associated with a corner strength value, and the parameters of the fitted model at that location. The final set of corners is obtained by imposing a threshold on the corner strength values, which must be preceded by non-maxima suppression to eliminate clusters. The non-maxima suppression step simply involves discarding all corner locations immediately adjacent to other locations with higher corner strength scores.

The efficiency of this detection will be essentially determined by the ease with which corner information is extracted from the raw image intensity values. This is accomplished here with a simple but effective foreground/background classification scheme. The result is a segmented area from which the ideal model can be incrementally reconstructed. This is explained in the next two sections.

### 3.2. Background/foreground segmentation

To obtain a simple foreground/background segmentation of circular areas around potential corners, the surrounding pixels are classified as having an intensity above or below the mean area's intensity. The group covering the smallest area becomes the foreground, and the other group forms the background. This segmentation is inspired from rudimentary block truncation which was introduced by Delp and Mitchell for the purpose of image compression (Delp and Mitchell, 1979).

In practice, the smallest region of the segmented area is not necessarily its foreground (in the case of a concave foreground element). In fact, both areas might be part of the same scene element (in the case where the corner is part of a color pattern on a single object). The terms foreground/background are simply used because they represent the common situation of corners defined by a convex foreground object. Although some types of corners might be more useful when it comes to matching, the fact that the different foreground and background areas of the corners might be misclassified will have no effect on the following matching step described in Section 5.

Instead of using a hard threshold, a sigmoidal function is actually used to determine the degree to which each pixel belongs to one of the two groups. This sigmoidal function has the form:
The model which best fits the segmented area is found from the following similarity criteria:

\[
c_s(x, y, W_{\theta}) = \frac{1}{\text{Area of } C_{xy}} \int \int_{C_{xy}} |W_{\theta}^n(i, j) - \text{sig}(I(x + i, y + j) - \bar{T}(x, y))| \, di \, dj,
\]

where \( C_{xy} \) is a small circular window centered on the point \((x, y)\), \( W_{\theta}^n \) is a binary map of the wedge model being fitted as in Fig. 1, \( \text{sig()} \) is the sigmoid function defined in Eq. (3), and \( \bar{T}(x, y) \) is the average pixel intensity in \( C_{xy} \). In the context of feature point detection, it would be too costly to find the optimal \( \theta \) and \( \phi \) for this pseudo Hamming distance using functional minimization. Instead, we propose to use the following strategy to approximate the optimal values:

1. The circular area around the potential corner is subdivided into small wedges, \( W_{\phi}^{n\Delta\theta} \) having a width of \( \phi_{\text{min}} \) and rotated around the circular area by increments of \( \Delta\theta \). Note that for a better accuracy in the delimitation of the wedge model, \( \Delta\theta \) will generally be smaller than \( \phi_{\text{min}} \), implying that adjacent elementary wedges will overlap.

2. The foreground coverage of these elementary wedges is then computed:

\[
c_{\text{cover}}(x, y, W_{n\Delta\theta}) = \int \int_{W_{n\Delta\theta}} \text{sig}(I(x + i, y + j) - \bar{T}(x, y)) \, di \, dj.
\]

An elementary wedge will be considered as a part of the potential corner’s foreground if its corner coverage is greater than some predetermined threshold value, \( c_{\text{min}} \). This process is similar to the rotated wedge filtering presented by Yu et al. (1998), where it is used to compute intensity mean values, from which one-dimensional angular derivative are computed.

3. The wedge with the highest coverage is selected.
From this initial wedge, all adjacent foreground wedges (those such that \( c_{\text{cover}}(x, y, W_{x,y,W_{u}}_{\min}) > c_{\min} \)) are retained. The corner model is then the one formed by the union of all the selected elementary wedges.

The union of all the wedges selected in the above procedure determines \( W_{u,h} \), which is used by Eq. (4) to compute the strength of the detected corner. Thus, \( \varphi \) is the angle spanned by the union, and \( \theta \) is its bisector. However, before going further, it is verified that \( \varphi \) is compatible with the corner model described in Section 3.1, i.e., \( \varphi_{\min} < \varphi < \varphi_{\max} \).

4. Experimental comparisons

To test the proposed corner detector on the basis of the previously described quality attributes, experiments that compare our corner detector to the SUSAN and Harris detectors were conducted. Results are presented in the following two sections.

4.1. Corner localization

The first set of experiments measures the robustness and the accuracy of each detector. To this end, we used a synthetic image, where the position of each corner (here the vertices of synthetic shapes) is known.

Fig. 2 shows the original test image, and the feature points found by the three different detectors. Note the variability in contrast conditions at different corner locations, caused by the smooth gray level transition of the background. This allows the comparison of the respective sensitivity of the operators, each of them having been tuned to obtain the best possible results. In Fig. 2D, note that it was impossible to find a threshold for the Harris detector that would allow the detection of the corners located in the low contrast area, without significantly increasing the number of false positives. In this area, the difference in intensity values between the background and the shapes is as low as 11.

The behavior of the same corner detectors in the presence of noise was also tested. Fig. 3 shows the feature points detected when a gaussian noise of variance 50 is added to the original synthetic image. The detection thresholds for these experiments were chosen to visually give the best results. For each vertex in image Fig. 2A, the closest detected corner was determined, and the distances between the vertices and their closest detected corner were measured. These measurements were taken for all three detectors, and for noisy versions of the test image. Results are presented in Fig. 4, where the graphs show, for different noise levels, the number of vertices with a detected corner within 1, 2, 3, and 4 pixels of it.

In light of these experiments, SUSAN appears to be a more accurate detector. However, as the level of noise is increased, the proposed wedge-based detector leads to superior results. The poor performance of the Harris detector in the presence of noise can be attributed to its reliance on image intensity derivatives which are unstab-
ble with respect to noise. As for the SUSAN detector, the segmentation it uses selects as foreground intensities in a small range around the central pixel’s intensity. This represents a narrower range of possible intensities than the proposed segmentation, making the segmentation into USANs more sensitive to noise. The fact that the proposed segmentation is not binary, but rather based on the sigmoid function of Eq. (3) also reduces the effect of misclassifications within the segmentation. Finally, USANs are less stable because they depend on the intensity of the model’s central pixel which is unstable in noisy circumstances.

However, it must be noted that with the noisy image, the proposed detector produced more false positives. For example, the proposed detector found points within three pixels, for 39 out of the 47 true corners in the image, as opposed to 35 and 34 for Harris and SUSAN, respectively. But also found 22 false positives, compared to 13 and 8 for Harris and SUSAN, respectively. It should nevertheless be noted that each detector produced about the same number of real false positives, while most of the proposed detector’s and some of Harris’ false positives where found in clusters around the true corners. These clusters survive the step of non-maxima suppression since they are formed of points which are not immediately adjacent, but separated by

Fig. 2. Corner detection on a synthetic image. (A) The original image. (B) The wedge models found using the proposed detector. (The black foreground areas of the corners, as described in Section 3.2, do not necessarily correspond to the physical foregrounds.) (C) Corners found using SUSAN. (D) Corners found using Harris.
one or two pixels with lower corner strength. In some applications such as matching (see Section 5), the number of false positives is far less important than the robustness of a detector.
4.2. Repeatability of the corner detection

In this subsection, a second set of experiments concerning the stability of the different detectors is presented. This is the most important attribute of a corner detector in the context of feature point matching. The repeatability of a detector was introduced by Schmid et al. (1998). It consists in the proportion of features that are consistently detected in different views of a scene.

To compute the repeatability, the location of a corresponding point in a second view must be known for every feature point in a first view of the same scene. It is easy to determine this correspondence, when the views are related by an homographic transformation. It is well known that images of a planar surface are related by homographies (Hartley and Zisserman, 2000). Such a transformation can be represented by a $3 \times 3$ matrix, $H$, which relates corresponding points, $p = [x, y, 1]^T$ and $p' = [x', y', 1]^T$ by:

$$p' = Hp.$$  \hspace{1cm} (6)

This relation also holds between images where the difference in viewpoint corresponds to a pure rotation. Two image pairs, corresponding to these two situations where the views are related by homographies are shown in Fig. 5. Having estimated the values of $H$ for these image pairs, it is easy to determine the expected position of

![Fig. 5. (A and B) Two views of a plane and (C and D) two views separated by pure rotation.](image-url)
a corresponding point in the other image using Eq. (6), and to see if a feature point was also detected at that location.

The computed repeatability for each considered corner detectors are given in Fig. 6. It shows the percentages of detected feature points in the left images for which the corresponding points in the right image have also been detected. These repeatability rates were computed for different corner acceptance thresholds. In this case, SUSAN clearly demonstrates the worst performance, while the other two detectors present a similar behavior.

5. Matching feature points

One common application of feature point detection is in selecting points towards sparse matching. When the change in viewpoint between images is small, matching can be accomplished through normalized correlation. This results in a set of matches which may contain false positives. Most of these can be filtered out using basic constraints (Vincent and Laganère, 2001), and the epipolar or trifocal geometry (Deriche et al., 1994; Torr and Zisserman, 1998) computed using a robust estimator such as RANSAC (Fischler and Bolles, 1981).

However, the situation is more difficult when there is significant deformation due to changes in viewpoint between the images to be matched. Then, normalized correlation between windows around feature points may not provide a meaningful measure of the similarity of possibly corresponding corners. But in these cases, the information conveyed by the corner detector could contribute to improve the matching results. This is where the use of a corner detector that provides rich descriptive information about the detected corners becomes advantageous. Here, a matching scheme which is robust to the perspective deformation induced by changes in viewpoint is required.

A possible solution is to find the local 2D projective transformation between two image patches before measuring their similarity (Zoghliami et al., 1997). The local neighborhood of points can be approximated as planar, so that the required trans-
formulation becomes an homography as in Pritchett and Zisserman (1998). To simplify the problem, it is often assumed that shear effects between views can be neglected, and that simple similarity transformations (rotation + scale) can be used (Jung and Lacroix, 2001), but a better approximation of the homography due to perspective effects consists in an affine transformation, as used in works such as (Baumber, 2000; Tuytelaars et al., 1999). These affinities can be represented with a 6 DOF, $3 \times 3$ non-singular matrix which geometrically transform the coordinates $(x, y)$ of an image patch into coordinates $(x', y')$ in another view, as follows (Hartley and Zisserman, 2000):

\[
\begin{bmatrix}
x' \\
y' \\
1
\end{bmatrix} = \begin{bmatrix}
a_{11} & a_{12} & t_1 \\
a_{21} & a_{22} & t_2 \\
0 & 0 & 1
\end{bmatrix} \begin{bmatrix}
x \\
y \\
1
\end{bmatrix},
\]

(7)

where homogeneous coordinates are used. This transformation approximates the varying viewpoint distortion with a translation (given by vector $(t_1, t_2)$), a rotation, and a non-isotropic scaling (skew). If such a transformation can be found for a given pair of image patches, then matching can be accomplished by correlating one patch with the transformed version of the other patch.

When matching is done between feature points, the image patches are simply small circular windows centered on these points. For a given pair of candidate points, the translation is therefore known. Four degrees of freedom remain, to completely describe the affinity. Some extra information can be provided by the wedge corner model. Assuming that the difference in scale between the two images may be neglected, the affinity can be estimated using the two following simplifications: (1) the rotation angle is given by the difference in angular position between the two detected wedges, $\theta - \theta'$. (2) The non-isotropic component of the affinity is applied in the bisector direction of the rotated wedge, with a ratio given by the angular widths of the two wedges, $\varphi : \varphi'$.

Under these assumptions, given a point $(x, y)$, with a corner model represented by $\theta$ and $\varphi$, which corresponds to a point $(x + t_1, y + t_2)$, with a corner model represented by $\theta'$ and $\varphi'$, then $(x + \cos(\theta + \varphi/2), y + \sin(\theta + \varphi/2))$ should correspond to $(x + t_1 + \cos(\theta' + \varphi'/2), y + t_2 + \sin(\theta' + \varphi'/2))$ and $(x + \cos(\theta - \varphi/2), y + \sin(\theta - \varphi/2))$ should correspond to $(x + t_1 + \cos(\theta' - \varphi'/2), y + t_2 + \sin(\theta' - \varphi'/2))$. By solving the resulting system of linear equations, the coefficients from Eq. (7) are found to be:

\[
\begin{align*}
a_{11} &= \frac{\cos(\theta' + \varphi'/2) \sin(\theta - \varphi/2) - \cos(\theta' - \varphi'/2) \sin(\theta + \varphi/2)}{\cos(\theta + \varphi/2) \sin(\theta - \varphi/2) - \cos(\theta - \varphi/2) \sin(\theta + \varphi/2)}, \\
a_{12} &= \frac{\cos(\theta + \varphi/2) \cos(\theta' - \varphi'/2) - \cos(\theta - \varphi/2) \cos(\theta' + \varphi'/2)}{\cos(\theta + \varphi/2) \sin(\theta - \varphi/2) - \cos(\theta - \varphi/2) \sin(\theta + \varphi/2)}, \\
a_{21} &= \frac{\sin(\theta' + \varphi'/2) \sin(\theta - \varphi/2) - \sin(\theta' - \varphi'/2) \sin(\theta + \varphi/2)}{\cos(\theta + \varphi/2) \sin(\theta - \varphi/2) - \cos(\theta - \varphi/2) \sin(\theta + \varphi/2)}, \\
a_{22} &= \frac{\cos(\theta + \varphi/2) \sin(\theta' - \varphi'/2) - \cos(\theta - \varphi/2) \sin(\theta' + \varphi'/2)}{\cos(\theta + \varphi/2) \sin(\theta - \varphi/2) - \cos(\theta - \varphi/2) \sin(\theta + \varphi/2)}.
\end{align*}
\]

(8)
The proposed matching scheme, therefore proceeds as follow:

1. Wedge corners are detected in each view.
2. For all possible pairs of wedge corners, the affine transformation between them is estimated using (8).
3. Normalized correlation is applied to the intensity values found in the vicinity of the corner in the first view and the corresponding intensity values, in the second transformed image.
4. All matches with a correlation score higher than a given threshold are retained.

This process is illustrated in Fig. 7 where two views of a feature point are shown. In this work, the commonly used normalized correlation measure described in Deriche et al. (1994) was used. It is seen in Fig. 7 that the use of the estimated affine transformation improves greatly this correlation score.

Obviously, this simple approach will still produce a large number of false matches. However, in the case of widely separated views, it should produce many more good matches than would be obtained through direct correlation of the image patches. Mismatches can still be filtered out using some robust estimator of the camera system’s geometry. In the following examples, a random sampling consensus (RANSAC) scheme was used to estimate fundamental matrices or homography matrices.

![Fig. 7. Affine matching of two corners. (A and B) The detected wedge models on the two corners. (C) The correlation window around the corner (A). (D) The corresponding window around (B) before an affine transformation is applied (the correlation score is ~0.31). (E) The window around (B) after applying the affine transformation defined by (8) (correlation score is now 0.96).](image-url)
towards filtering the match sets. RANSAC robustly estimates a fundamental matrix from a match set containing outliers, by iteratively selecting seven matches at random (the minimum number of matches needed) and computing a fundamental
matrix. In the end, the attempted fundamental matrix agreeing with the most matches is selected as the best estimate. Finally, all matches not agreeing with this estimated fundamental matrix can be rejected. A similar process can be used to estimate a homography, but in this case, only four matches need to be randomly selected at each iteration. In this work, the software\(^1\) described in Roth and Whitehead (2000) was used to perform the RANSAC estimations.

Fig. 8 shows a pair of images on which our simple matching procedure was applied. The resulting candidate match set contained 300 point pairs, and was used to estimate the epipolar geometry. A fundamental matrix was found which agreed with 35 of the candidate matches. However, when using also 300 matches, but found by direct normalized correlation, no valid fundamental matrix could be obtained, no matter how many RANSAC iterations were tried.

The same approach was also applied to the image pair shown in Fig. 9. In this case, the two images are related by a pure rotation. This time, after applying the proposed matching scheme, a homographic transformation was estimated from 300 point pairs. This homography agreed with 21 candidate matches which were drawn in the figure. A mosaic\(^2\) is shown in Fig. 9 which was constructed from the computer homography. Again, using the best 300 matches obtained by direct correlation produced no valid results.

6. Conclusion

The novel feature point detector introduced in this paper proceeds in two steps: it first performs a simple segmentation based on the intensity values found in the vicinity of each considered point. Then, it tries to fit a simple wedge corner model to the resulting segmented area. The combination of these two approaches contribute to the effectiveness of the detector. Indeed, intensity-based corner finders, like the SUSAN operator, tends to show good accuracy, a property also inherited by the operator proposed here. On the other hand, the stability of such approaches is usually poor because intensity patterns can undergo important modifications when geometrical transformations are applied. This is not the case of model-based approaches such as the Harris detector, which normally exhibit good stability. However, derivative-based approaches are usually more sensitive to noise than model-based approaches. These observations were confirmed by the presented experiments.

The controllability of the operator is managed by the following parameters, each having a specific role:

1. The minimum acceptable variance \(\sigma^2_{\text{min}}\), that controls the sensitivity of the operator. This parameter is normally fixed, but could be modified to deal with images

---

\(^1\) Available from <http://www.cv.iit.nrc.ca/~gerhard/>.

\(^2\) This figure was generated with the help of Rafy Terzian and Stéphanie Deguire.
containing more noise or artifacts. A value of 150 was used to produce the presented results.

(2) The step angle $\Delta \theta$, that discretizes the space of admissible angles for a wedge corner. The smaller this angle, the more precise the identification of the edges of the corner. A value of $10^\circ$ was used to produce the presented results.

(3) The minimal and maximal angle width of corners $\phi_{\min}$ and $\phi_{\max}$ which define the range of acceptable wedge angular widths that can be attributed to a corner. These control parameters can be selected once and for all. The presented results used values of $30^\circ$ and $120^\circ$.

(4) The radius of the wedge model is the last control parameter. It should be relatively small to reduce the computation time. The presented results were obtained with a radius of 7.

(5) The corner strength threshold $c_{\min}$. This is the main control parameter, and the only one that needs to be tuned according to the problem at hand. Its value is chosen as a compromise between selecting more corners, and selecting better corners. Values between 0.75 and 0.9 are reasonable.

References