Pacific Graphics (2011)
Bing-Yu Chen, Jan Kautz, Tong-Yee Lee, and Ming C. Lin (Editors)

Short Papers

An Effective and Efficient Contour-based Corner Detector using Simple Triangular Theory

R. M. N. Sadat, S. W. Teng and G. Lu

Gippsland School of IT, Monash University, Australia
{najmus.sadat, shyh.wei.teng, guojun.lu}@monash.edu

Abstract

Corner detection is an important operation in many computer vision applications. Among the contour-based corner detectors in the literature, the Chord-to-Point Distance Accumulation (CPDA) detector is reported to have one of the best repeatability and lowest localization error. However, we found that CPDA detector often fails to accurately detect the true corners in some situations. Furthermore, CPDA detector is also computationally expensive. To overcome these weaknesses of CPDA detector, we propose an effective but yet efficient corner detector using a simple triangular theory. Our experimental results show that our proposed detector outperforms CPDA and six other existing detectors in terms of repeatability. Our proposed detector also has one of the lowest localization error. Finally it is computationally the most efficient.

Categories and Subject Descriptors (according to ACM CCS): I.4.6 [IMAGE PROCESSING AND COMPUTER VISION]: Segmentation—Edge and feature detection

1. Introduction

Detecting corners is one of the most important operations in various computer vision and image processing applications such as motion tracking, shape representation, image registration, camera calibration, object recognition and stereo matching. A corner can be defined as a location on an edge where the slope changes abruptly. We can broadly classify corner detectors into two groups - intensity-based and contour-based. Intensity-based corner detectors [HS88, SB97, RD06] directly deal with the intensity values of the image. On the other hand, contour-based detectors [MS98, MM01b, HY04, AGM07, ZLY*07] first extract the curves (or contours) from the image and then identify the locations which have salient information or maximal curvature. Most intensity-based corner detectors are based on image derivatives, that is why they are more sensitive to noise. Contour-based corner detectors, however, are generally less sensitive to noise as they apply Gaussian smoothing to remove the noises from the contours. This paper focuses on contour-based detectors.

The Curvature Scale Space (CSS) [MS98] corner detector is one of the earliest contour-based detectors. It first uses a coarse smoothing scale to identify approximate locations of the corners. Next, it uses a finer scale to track these locations to improve the localization of these corners. The main weakness of this CSS detector is in selecting an appropriate scale for identifying the approximate locations of the corners. If a coarser scale was used, the detector would be robust to noise, but might miss many potential corners. However, if a finer scale was used, the detector would be sensitive to noise and would detect many spurious corners. The enhanced CSS [MM01b] detector attempted to solve this weakness by using different scales for curves with different lengths. However, choosing the right set of scales for various curves' length is still difficult. Furthermore, these CSS detectors estimate curvature values using the derivatives which are computed based on a small neighbourhood. This makes the detectors very sensitive to the local variations and noise.

To overcome the weaknesses of the CSS detectors, several detectors which use multiple scales for curvature estimation are proposed. Awrangjeb et al. proposed a multiscale detector (ARCSS) [AGM07] which uses affine-length parametrizations instead of the arc-length to detect the corners. However, this detector is computationally expensive due to the affine-length parametrizations calculation. The multiscale curvature product (MSCP) [ZLY*07] detector is another CSS-based detector which multiplies the curvature values derived using three scales to make the strong cor-

DOI: 10.2312/PE/PG/PG2011short/037-042

ners more distinguishable from the noise and weak corners. He and Yung [HY04] modified the original CSS detector by using an adaptive local threshold according to its neighbourhood region's curvature and then detecting the angle on proper region of support.

Zhang et al. [ZWH*09] proposed a detector which applies multiple levels of Difference of Gaussian (DoG) on a curve and used these planar curves for detecting the corners. Since, derivative is used for curvature estimation, this is still sensitive to noise. Another groups of detectors [QG02, GSQV07] apply wavelet transform to the curve for representing its contour orientation. However, as the wavelet transform is similar to second derivatives of the curve, these detectors can still be sensitive to noise. A few other detectors use various forms of matrix manipulation, e.g. Eigenvalues of the covariance matrix [THS99] and Gradient Correlation Matrix [ZWS*10], for processing the curve. Generally, these detectors are computationally expensive due to the matrix manipulation.

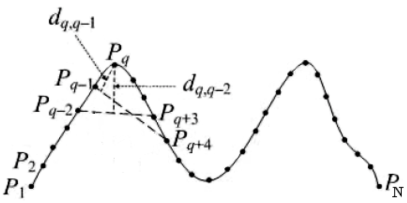


Figure 1: *Curvature estimation at a point using CPDA with chord L_5*

One of the best contour-based corner detectors reported in the literature is based on Chord-to-Point Distance Accumulation (CPDA) technique [HP01]. Although the CPDA [AG08] corner detector is reported to achieve one of the highest repeatability accuracy and lowest localization error among other existing compatible detectors in the literature, we found that CPDA detector has several weaknesses. Firstly, it is prone to detecting weak or false corners on rounded curves. Secondly, it has the potential to miss some corners on curves which have several corners closely located to each other. Thirdly, CPDA detector is computationally expensive due to the complexity of its curvature estimation. How the CPDA detector works and its weaknesses will be discussed in greater detail in Section 2.

In this paper, we propose a contour-based corner detector which uses a simple triangular theory for curvature estimation. As our proposed detector is simpler and more intuitive than CPDA for determining the presence of corners, it is also able to overcome all the aforementioned weaknesses of the CPDA detector. Our experimental results show that our proposed detector achieves the best repeatability and comparable low localization error among CPDA and other existing compatible detectors. It is also the most efficient.

The rest of the paper is organized as follows. An overview of CPDA detector and its weaknesses are discussed in Sections 2. Section 3 presents the proposed corner detector. Next, Section 4 discuses the complexity of the corner detectors and Section 5 shows the experimental results. Finally, Section 6 concludes the paper.

(a) (b)

Figure 2: *(a) Lena image; (b) Corners (denoted by 'o') detected by CPDA after removing weak corners.*

2. Overview of CPDA Detector

Similar to other CSS-based corner detectors, CPDA detector [AG08] also starts by detecting curves from the images and finding the T-junctions. Each extracted curve is smoothed with an appropriate Gaussian kernel (i.e. $\sigma = 1$, 2, or 3) depending on its length to remove the noise from it.

Next, three chords which are defined as, L_i where $i \in \{10, 20, 30\}$, are moved along each curve. In Figure 1, let $P_1, P_2, P_3, \ldots, P_N$ be the N points on a curve. So, value i of L_i chord defines a straight line joining points P_j and P_{j+i} on the curve. To estimate the curvature value $h_{L_i}(q)$ at point P_q using a chord which is i pixels apart, the chord is moved on each side of P_q for at most i points while keeping P_q as an interior point and the distances $d_{q,j}$ from P_q to the chord is calculated. Finally, CPDA accumulates the curvature estimation using the Equation 1.

$$h_{L_i}(q) = \sum_{j=q-i+1}^{q-1} d_{q,j} \tag{1}$$

The curvature values estimated using each chord are normalized using Equation 2 and then multiplied together using Equation 3.

$$h'_{L_i}(q) = \frac{h_{L_i}(q)}{max(h_{L_i})}, \text{for } 1 \leq q \leq N, i \in \{10, 20, 30\} \tag{2}$$

$$H(q) = h'_{L_{10}}(q) \times h'_{L_{20}}(q) \times h'_{L_{30}}(q), \text{for } 1 \leq q \leq N \tag{3}$$

h_{L_i} is the set of curvature values estimated for all points on the curve using chord L_i. The local maxima of $H(q)$ determine the locations of the candidate corners. Finally, to filter out the weak and false corners among the candidate corners, a two-step refinement process is used.

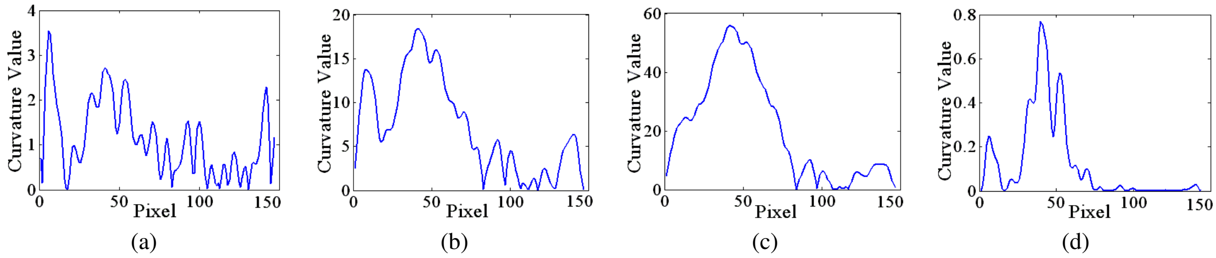


Figure 3: *(a)-(c) Curvature estimation of Curve A of Lena image using chord 10, 20 and 30 respectively; (d) combined curvature estimation using Equations 2 and 3*

The first refinement step is to remove all weak corners by thresholding the candidate corners. The second step is to remove all false corners. Here, the angle from a candidate corner to its two neighbouring candidate corners is calculated. The candidate corner is considered as a false corner if the angle is greater than an angle-threshold. Figure 2 shows the resultant corners detected by CPDA detector.

From our analysis, we found a few weaknesses of CPDA Detector. CPDA detector is prone to detect false and weak corners on curves which have rounded slopes. This is because after normalizing the curvature values, information about the actual magnitude differences of the corresponding curvature values of one chord to the other two chords are lost. Therefore the final curvature values derived by multiplying the corresponding curvature values might no longer reflect the true cornerness of the points. To illustrate this, we will use *Curve A* of Lena image shown in Figure 2(b). As *Curve A* is with no abrupt slope changes, many of the normalized curvature values derived using each chord will be closer to 1. Therefore, a good portion of the final curvature values derived will continue to be closer to 1 (see Figure 3(d)), thereby resulting in false or weak corners to be detected as final set of corners. Although, CPDA uses the second refinement step to discard the false corners, this step is not robust in discarding false corners on round curves. Moreover, the refinement process is computationally expensive.

We have also found that, CPDA detector might potentially miss obvious corners if they were located closely. This is due to the use of longer chords. For example, Figure 4 shows a hand-drawn shape where the Corner 'C' is not detected by the CPDA detector. The chords of length 10 and 20 can detect the local maxima on the location 'C' (Figure 4 (b) and (c)), however, the third chord cannot (Figure 4 (d)). After normalizing the curvature values and multiplying them, the final curvature value representing the location of this corner will be too low to be detected as a corner by CPDA detector.

3. Proposed Corner Detector

In this section, we propose a corner detector that can overcome the weaknesses of CPDA and other CSS-based corner detectors. First, the measure used to estimate the curvature value is discussed. Next, a corner detector using the curvature estimation measure is proposed.

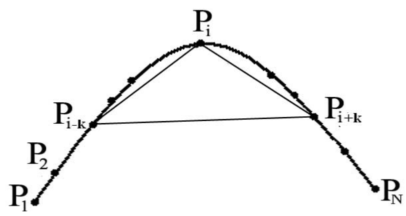
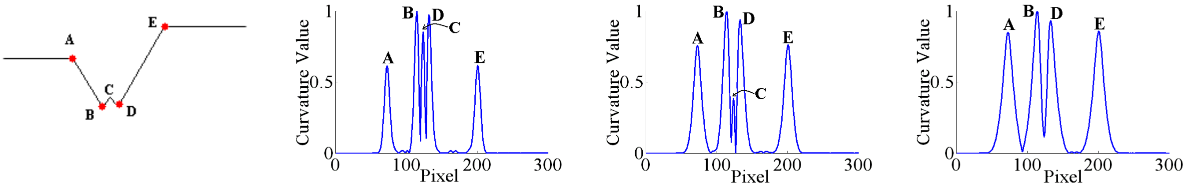


Figure 5: *Curvature estimation measure used in proposed detector*

Instead of using distance accumulation, a simple but yet effective measure based on a triangular theory is used to estimate the curvature values. To apply this measure, a chord is first moved along the curve in a way which is similar to CPDA curvature estimation. Every time the chord is placed on the curve, a new triangle can be formed using two ends of the chord and the middle point on the curve segment between the two ends of the chord. The ratio of the length of the chord to the summation of the length of the other two arms of the triangle, which are from the middle point to each respective ends of the chord, is computed. The value of this ratio is the estimated curvature value for the middle point on the curve. Since, this measure does not use any derivative based measurements and it also uses a relatively bigger neighbourhood. Thus, it is less sensitive to noise which is one of the weaknesses of CSS-based corner detectors.

We illustrate the above measure with an example in Figure 5. Let $P_1, P_2, \ldots, P_N$ be the N points of a curve and P_i be the point where the curvature value is to be estimated. Now, we traverse k pixels from P_i in the right direction to pixel P_{i+k} and then, k pixels from P_i in reverse direction to pixel P_{i-k}. If the three pixels P_{i-k}, P_i and P_{i+k} are collinear, the ratio of the length of the chord from P_{i-k} to P_{i+k}, to the summation of the length of the other two arms of the triangle, from P_i to P_{i-k} and P_{i+k} respectively, is 1, otherwise the ratio is less than 1. The value of the ratio will decrease as the sharpness



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

Figure 4: *(a) Corners detected by CPDA detector; (b)-(d) Curvature values estimated using the three chords after normalization*

of the corner at P_i increases. Now, the curvature value of point P_i on the curve is estimated using

$$R_L(i) = \frac{d_1}{d_2 + d_3} \tag{4}$$

where,

$$d_1 = \sqrt{(x_{P_{i-k}} - x_{P_{i+k}})^2 + (y_{P_{i-k}} - y_{P_{i+k}})^2}$$
$$d_2 = \sqrt{(x_{P_i} - x_{P_{i-k}})^2 + (y_{P_i} - y_{P_{i-k}})^2}$$
$$d_3 = \sqrt{(x_{P_i} - x_{P_{i+k}})^2 + (y_{P_i} - y_{P_{i+k}})^2}$$

Next, we describe how the curvature estimation measure described above is used in our proposed detector. Similar to other CSS-based corner detectors, our proposed detector also starts with detecting the curves from the image and finding out the T-junctions. Next, we apply the Gaussian smoothing scale ($\sigma = 3$) to reduce the noise on the curve.

We use a smaller k so that the detector does not lose the maxima at two nearby corners. We have chosen the value of k as 3. After estimating the curvature values, we found the local minima from each curve's estimation and consider the minima as corners if the curvature value is less than a threshold ($T = 0.989$). Finally, the T-junctions are added to the final set of corners if any location near (5×5 window) the T-junctions is not detected as a corner. We name our proposed detector as Chord to Triangular Arms Ratio (CTAR) detector.

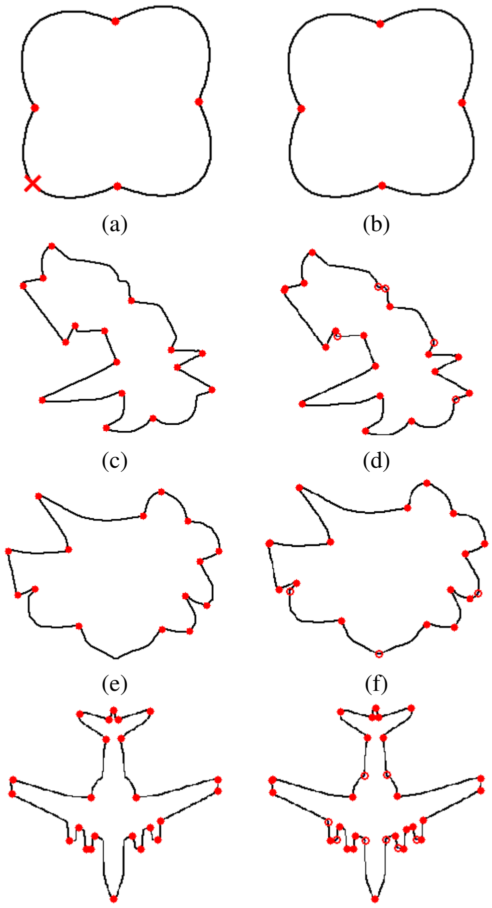
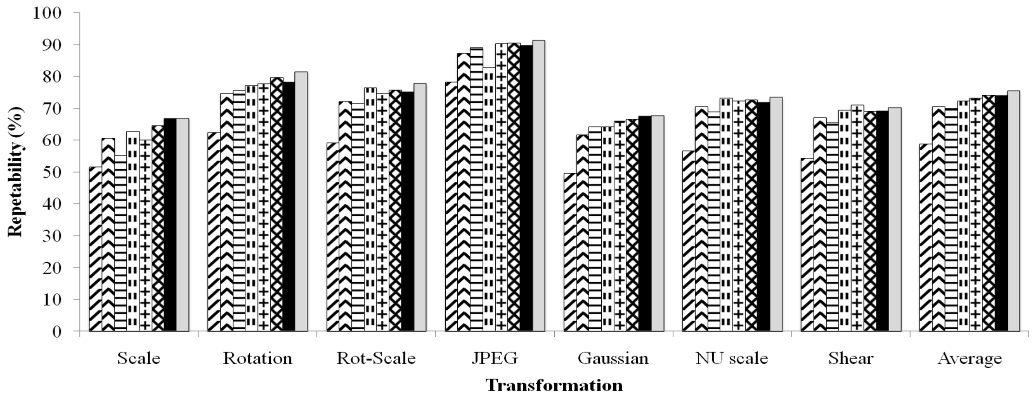


Figure 6: *(a),(c),(e),(g) Corners detected by CPDA detector; (b),(d),(f),(h) Corners detected by CTAR detector*

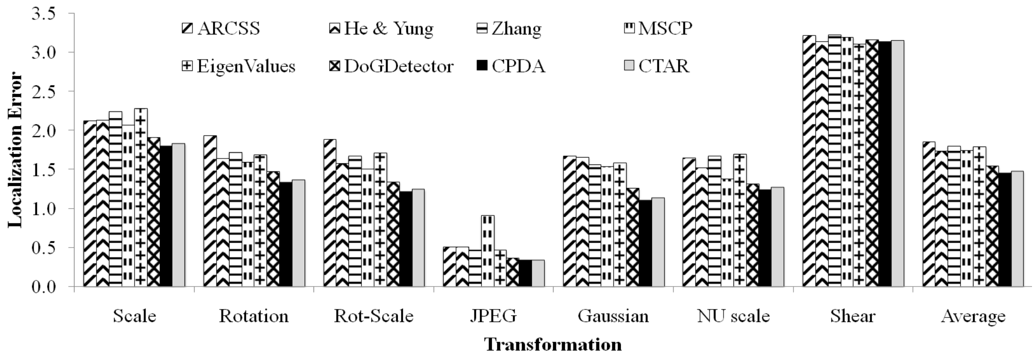
4. Discussion on Complexity

We compare the complexity of the CPDA detector with our proposed detector in this section. Among the arithmetic operations commonly used in CPDA and CTAR for estimating the curvature values, the most computationally expensive operation is the square root operation. Since the computational time required by other common arithmetic operations is relatively insignificant, we compare the complexity of these two detectors by counting the number of square root operations used in these two detectors.

Let us denote the three chords used by the CPDA detector to estimate a curve of n number of points as L_i where $i \in \{10, 20, 30\}$. Using a specific chord, $(L_i - 1)$ square root operations are required to estimate the curvature value at each point (Equation 1). So, to estimate the curvature values for all points on the curve, the total number of square root operations used by CPDA is $n \times (L_{10} - 1) + n \times (L_{20} - 1) + n \times (L_{30} - 1)$. On the contrast, CTAR uses far fewer number of square root operations compared to CPDA detector. Only



(a)



(b)

Figure 7: *(a)Repeatability and (b)Localization Error by the corner detectors for different geometric transformation*

$3n$ square root operations are required to estimate the curvature values of the n points on the curve. To provide an idea the computational time difference between the two detectors in processing real-life images, CPDA uses 84582 square root operations for processing the "Lena" image (Figure 2) while CTAR uses only 5418.

Furthermore, as described in Section 2, CPDA detector uses an additional normalization and a multiplication processes to bring all the chords' measurement to similar unit. This normalization process is already reported as a weakness of CPDA detector (Section 2). These two process takes $2n$ operations. On the other hand, the CTAR's ratios do not need to be normalized. Finally, CTAR does not need the second refinement step used in CPDA detector.

5. Experimental Results

This section presents the results from two experiments conducted to evaluate the performance of the proposed detector. First, we show the corners detected by CTAR and CPDA detectors. The corners are extracted from nine different shapes which are commonly used in several other research on contour-based corner detection [ZWS*10, HY04]. Figure 6 shows four shapes which have different sets of corner locations detected by the CPDA and CTAR detectors. The false corners are marked with cross (×) and the additional corners detected by CTAR have been marked with unfilled circle. Figure 6 shows that CPDA detects a false corner in the first shape and misses many strong corners in the other three shapes. However, CTAR detects almost all the strong corners (17 more than CPDA) on the shapes.

Second, we compare the robustness of the detected corners by CTAR with seven existing compatible detectors - ARCSS [AGM07], He and Yung [HY04], Zhang [ZWS*10], MSCP [ZLY*07] , EigenValues [THS99], DoGDetector [ZWH*09], and CPDA [AG08]. The robustness of the detectors is evaluated based on repeatability [MM01a] and localization error [AG08]. The repeatability takes into the consideration of the number of corners from original as well as corners from the transformed image, and it is computed as follows:

$$Repeatability = 100\% \times \frac{\frac{N_m}{N_o} + \frac{N_m}{N_t}}{2} \quad (5)$$

where N_o is the number of reference corners from the original image, N_t is the number of detected corners in the trans-

Detector	Execution Time (sec)
CTAR	0.0890
CPDA	0.5430
DoGDetector	0.6003
EigenValues	1.7236
Zhang	0.1348
He & Yung	0.4418
ARCSS	0.6587

Table 1: *Time to detect corners by different corner detectors*

formed image, and N_m is the number of matched corners between detected and reference corners. The corners detected in the original images are used as reference points so that human intervention, which is very subjective, is not needed to determine the reference points [AG08]. Localization error [AG08] shows the distances between the detected location of corners and their correct locations on the image. It is computed as follows:

$$LE = \sqrt{\frac{1}{N_m}\sum_{i-1}^{N_m}(x_{oi}-x_{ti})^2+(y_{oi}-y_{ti})^2} \qquad (6)$$

where (x_{oi}, y_{oi}) and (x_{ti}, y_{ti}) are the ith matched corners from the corners of reference image (N_o) and test image (N_t) respectively.

The test dataset consists of over 8700 images. They are derived by applying a wide range of transformations on 23 different base images which include real-life (e.g. Lena, House, and Lab) and synthetic images. Seven different sets of transformations are applied on the base images - Rotation, Uniform Scale, Non-uniform Scaling, Rotation and Scale, JPEG compression, Shear Transform and Gaussian noise. Figures 7 (a) and (b) show the average repeatability and the average localization error of each evaluated detector respectively. The average repeatability of all the transformations of CTAR is the best among all the corner detectors, followed by CPDA. As for localization error, CPDA detector has the lowest average localization error, closely followed by CTAR.

Table 5 shows every evaluated detector's computational time for detecting the corners in the 23 base images. All detectors are implemented in Matlab and executed in a Windows machine with a Core2Duo 2.0 GHz processor and 3 GB RAM. The time for curve extraction of every detector is same, so this time is excluded from the time presented. The result clearly shows the efficiency of the proposed corner detector. CTAR is the fastest corner detector and more than 6 times faster than CPDA detector.

6. Conclusions

In this paper, we have proposed an effective and efficient contour-based corner detector. In comparison to other detectors evaluated, the proposed detector achieves the best repeatability accuracy in detecting robust corners. However, its localization error is slightly higher than the CPDA detector. Our proposed detector is also computationally more efficient than other evaluated detectors.

References

[AG08] AWRANGJEB M., GUOJUN L.: Robust image corner detection based on the chord-to-point distance accumulation technique. *Multimedia, IEEE Transactions on 10*, 6 (2008), 1059–1072. 2, 5, 6

[AGM07] AWRANGJEB M., GUOJUN L., MURSHED M.: An affine resilient curvature scale-space corner detector. I–1233–I–1236. 1, 5

[GSQV07] GAO X., SATTAR F., QUDDUS A., VENKATESWARLU R.: Multiscale contour corner detection based on local natural scale and wavelet transform. *Image and Vision Computing 25*, 6 (2007), 890 – 898. 2

[HP01] HAN J. H., POSTON T.: Chord-to-point distance accumulation and planar curvature: a new approach to discrete curvature. *Pattern Recognition Letters 22*, 10 (2001), 1133 – 1144. 2

[HS88] HARRIS C., STEPHENS M.: A combined corner and edge detector. 50. Alvey vision conference. 1

[HY04] HE X. C., YUNG N. H. C.: Curvature scale space corner detector with adaptive threshold and dynamic region of support. 791–794. 1, 2, 5

[MM01a] MOHANNA F., MOKHTARIAN: Performance evaluation of corner detection algorithms under similarity and affine transforms. *British Machine Vision Conference* (2001), 353–362. 5

[MM01b] MOKHTARIAN F., MOHANNA F.: Enhancing the curvature scale space corner detector. *PROCEEDINGS OF THE SCANDINAVIAN CONFERENCE ON IMAGE ANALYSIS* (2001), 145–152. 1

[MS98] MOKHTARIAN F., SUOMELA R.: Robust image corner detection through curvature scale space. *IEEE Trans. Pattern Anal. Mach. Intell. 20*, 12 (1998), 1376–1381. 297831. 1

[QG02] QUDDUS A., GABBOUJ M.: Wavelet-based corner detection technique using optimal scale. *Pattern Recogn. Lett. 23* (January 2002), 215–220. 2

[RD06] ROSTEN E., DRUMMOND T.: Machine learning for high-speed corner detection. 430–443. 1

[SB97] SMITH S. M., BRADY J. M.: Susan:a new approach to low level image processing. *Int. J. Comput. Vision 23*, 1 (1997), 45–78. 258056. 1

[THS99] TSAI D. M., HOU H. T., SU H. J.: Boundary-based corner detection using eigenvalues of covariance matrices. *Pattern Recognition Letters 20* (1999), 31–40. 2, 5

[ZLY*07] ZHANG X., LEI M., YANG D., WANG Y., MA L.: Multi-scale curvature product for robust image corner detection in curvature scale space. *Pattern Recogn. Lett. 28*, 5 (2007), 545–554. 1223932. 1, 5

[ZWH*09] ZHANG X., WANG H., HONG M., XU L., YANG D., LOVELL B. C.: Robust image corner detection based on scale evolution difference of planar curves. *Pattern Recogn. Lett. 30* (March 2009), 449–455. 2, 5

[ZWS*10] ZHANG X., WANG H., SMITH A. W. B., LING X., LOVELL B. C., YANG D.: Corner detection based on gradient correlation matrices of planar curves. *Pattern Recogn. 43* (April 2010), 1207–1223. 2, 5