Measuring Corner Properties

Paul L. Rosin
Department of Computer Science and Information Systems,
Brunel University,
Uxbridge,
Middlesex UB8 3PH
email: Paul.Rosin@brunel.ac.uk

Abstract

We describe methods to measure the following properties of grey level corners: subtended angle, orientation, contrast, bluntness (or rounding of the apex), and boundary curvature (for cusps). Unlike most of the published methods for extracting these properties these methods are relatively simple, efficient and robust. They rely on the corner being pre-detected by a standard operator, thus making the measurement problem more tractable. Using 13000 synthetic images the methods are assessed over a range of conditions: corners of varying orientations and subtended angles, as well as different degrees of noise.

1 Introduction

The three most commonly used features in computer vision are regions, edges, and corners. Whereas regions are normally attributed properties to make them useful as input for subsequent processing stages such as matching, until recently edges and corners were described by little except for strength. However, over the past few years more attention has been paid to the properties of edges and corners too, since richer descriptions make them more effective features. For instance, in tracking, model matching, or model indexing, corner properties are capable of constraining the corner correspondences either as unary constraints or n-ary constraints between several corners. The latter enables viewpoint invariance. For example, under orthographic projection selected pairs of coplanar corners may be expected to have identical, but unknown, subtended angles or orientations. In [14] we demonstrated how the addition of relative colour and subtended angle corner properties (binary and unary constrains respectively) enabled the number of arcs in the association graph to be drastically reduced in a model recognition application using maximal clique graph matching.

Recent examples of edges properties being measured are scale [1], diffuseness, height and width [18]. Regarding corners, several interesting methods have been developed recently to measure scale, orientation, subtended angle, contrast, and blurring [2, 5, 7, 8, 11, 12]. A problem with these techniques is that they tend to be complex and iterative, which impacts on both their efficiency and reliability. In contrast, rather that attempt the complicated task of simultaneously detecting and describing corners, in a previous paper we decoupled these two stages [14]. Standard algorithms were used to identify corner locations, and simple non-iterative methods were then sufficient for extracting corner properties. Many of the techniques we developed were based on local intensity or orientation histograms generated in the corner's neighbourhood. Before corner properties could be measured the histograms were smoothed by an automatically determined amount and the two main peaks located. A weakness of this approach is that it depends on correctly locating the peaks, which in turn depends on the appropriate level of smoothing being correctly determined. In this paper we develop further techniques for measuring corner properties more directly without recourse to histograms. By eliminating one potential source of error this has the potential for improving reliability and accuracy. In addition to the previously extracted corner properties we describe methods for measuring bluntness (or degree of rounding of the apex) and boundary shape (i.e. straight or curved sides).

2 Measuring Properties

2.1 Contrast and Subtended Angle

Ghosal and Mehrotra [4] showed how Zernicke moments could be employed to recover many properties of corners and edges. Here use the simpler standard moments to determine corner properties in a similar manner to Tsai's [17] image thresholding scheme. We define the moments of the image intensities $I(x, y)$ as

$$m_p = \int \int I(x,y)^p \, dx dy$$

which we calculate within a circular window about the corner. Disregarding spatial information we model the corner in one dimension by two constant populations of gray levels b and d (bright and dark) containing m and $n-m$ elements respectively. The model's moments are

$$\begin{aligned} m_p &= \int_0^m b^p dt + \int_m^n d^p dt \\ &= m b^p + (n-m) d^p \end{aligned}$$

where n is the number of pixels in the window. The required parameters are obtained using the method of moments. Taking the first three moments the resulting set of simultaneous equations can be solved to determine the values of the background and foreground intensities:

$$\begin{aligned} b &= \tfrac{m_1 m_2 - n m_3 + t}{2(m_1^2 - n m_2)} \\ d &= \tfrac{m_1 m_2 - n m_3 - t}{2(m_1^2 - n m_2)} \end{aligned}$$

where

$$t = \sqrt{-3m_1^2 m_2^2 + 4m_1^3 m_3 + 4m_2^3 n - 6m_1 m_2 m_3 n + m_3^2 n^2}.$$

Since $b > d$ the contrast is

$$c = b - d = \frac{t}{(m_1^2 - n m_2)}.$$

To find the subtended angle we use the fractions of the foreground and background populations within the window: $\frac{m}{n}$ and $\frac{n-m}{n}$. This is the same approach that we previously described for the thresholding method [13] (although we determined contrast differently by thresholding first and then applying some post-processing). Using the first moment we solve for m giving:

$$m = \left((m_1^2 - nm_2)m_1 - \frac{n}{2}(m_1 m_2 - nm_3 - t) \right) /t.$$

Assuming that the subtended angle lies in $[0, 180)$ it is simply calculated as

$$\theta = \min \left(\frac{m}{n}, \frac{n-m}{n} \right) \times 360^\circ.$$

2.2 Orientation

2.2.1 Intensity Centroid

Using geometric moments it is straightforward to determine the corner orientation (without having to use the method of moments). Defining the moments are as:

$$m_{pq} = \sum_{x,y} x^p y^q I(x, y)$$

the centroid is then determined as:

$$\mathbf{C} = \left(\frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}} \right).$$

Assuming the co-ordinate frame has been set so that the image window containing the corner is centred at the origin $\mathbf{O}$, the corner orientation is the angle of the vector $\overrightarrow{\mathbf{OC}}$ with a 180° correction to cater for corners which are darker than their background:

$$\phi = \begin{cases} \tan^{-1} \frac{m_{01}}{m_{10}} & \text{if colour} = \text{bright} \\ \tan^{-1} \frac{m_{01}}{m_{10}} + 180^\circ & \text{if colour} = \text{dark} \end{cases}$$

2.2.2 Gradient Centroid

Rather than use the moments of the image intensities it is possible to use the moments of the intensity gradient magnitude $G(x, y)$ instead:

$$\phi = \tan^{-1} \frac{\sum_{x,y} yG(x, y)}{\sum_{x,y} xG(x, y)}.$$

Although this requires the additional stage of edge detection (we use the Sobel operator to calculate $G(x, y)$) it has the advantage that unlike the intensity centroid method no special care needs be made concerning bright and dark corners, eliminating the need to predetermine corner colour.

2.2.3 Symmetry

Another method for determining corner orientation is based on the symmetry of simple corners. This implies that the orientation maximising symmetry will coincide with the corner orientation. In a similar vein, Ogawa [9] detected corners (i.e. dominant points) of curves by measuring local symmetry. We measure corner symmetry by requiring that intensities on either side of the corner bisector should be equal. This is implemented by rotating the image window by the hypothesised orientation ϕ using bilinear interpolation to obtain $I_\phi(x, y)$. Corner orientation can then be found as the rotation angle that minimises the summed absolute differences in corresponding intensity values:

$$\phi = \min_{\phi} \sum_{y=0}^{W} \sum_{x=-W}^{W} \left| I_\phi(x, y) - I_\phi(x, -y) \right| .$$

Since noise, quantisation, and the process of image rotation will introduce local errors we also consider differencing not individual pixel values but sums of row intensities or the two sums of all intensities on either side of the bisector. Of course, both ϕ and $\phi + 180°$ will give the same symmetry value. Therefore we only search over a small range of orientations (e.g. $45°$) centred at an initial estimate obtained using another technique such as those above. Note that unlike the methods above we can use a square rather than circular window for convenience.

2.3 Bluntness

Corner detectors usually assume that the corners are perfectly sharp (i.e. pointed). Not only is this invalidated by noise, blurring, and quantisation, but objects – both natural and man-made – often have rounded corners. Few detectors explicitly cater for this (but see Davies [3]) and none measure the degree of rounding or bluntness of the corner.

2.3.1 Kurtosis

One possible way to measure bluntness is to use the statistical measure of kurtosis which quantifies the “peakness” of a distribution. It is defined using central moments as

$$\kappa = \frac{m_4}{m_2^2}.$$

Assuming the corner orientation has already been found using one of the previously described techniques we generate the projection of the image window along the direction of the corner orientation see figure 1. We first rotate the image window by $-\phi$ using bilinear interpolation as before to align the corner along the X axis, and the projection is then given by

$$P_\phi(y) = \sum_x I_\phi(x, y).$$

Before calculating kurtosis the projection function is shifted so as to zero the tails

$$P'_\phi(y) = P_\phi(y) - \min_y P_\phi(y).$$

Furthermore, because high order moments are being used, kurtosis is very sensitive to noise. As a precaution small values of $P'_\phi(y)$ are zeroed:

$$P''_\phi(y) = \begin{cases} P'_\phi(y) & \text{if } P'_\phi(y) > \tau \max_y P'_\phi(y) \\ 0 & \text{otherwise.} \end{cases}$$

where $\tau \in [0, 1]$ is a threshold we have set to 0.1.

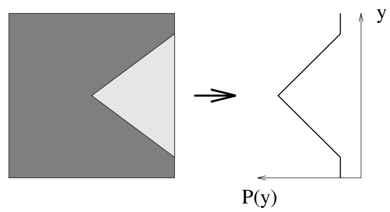


Figure 1: Projection of image window along corner orientation

We can analytically determine expected values of kurtosis for simple distributions [15]. For example, a triangular function, a parabola, and a semicircle produce values of $\kappa = 2.4$, 2.143, and 2, respectively. Thus it can be seen that rounding the corner decreases the value of the measured kurtosis. For convenience we normalise the measure as $\kappa_N = \frac{\kappa - 2}{0.4}$ to return an expected value in the range $[0, 1]$.[1]

2.3.2 Model Fitting

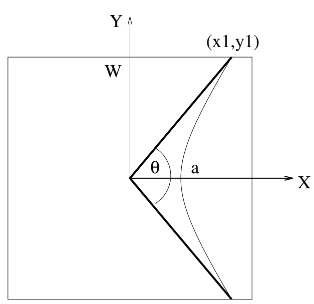


Figure 2: Hyperbola model

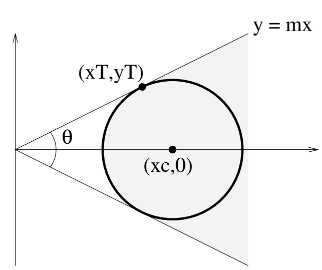


Figure 3: Circular arc model

A more direct way to measure corner bluntness is to fit a parametric model to the pixels in the image window. Rather than perform multivariate fitting we assume that most of the corner properties have already been determined by other simpler methods such as those we have already described. This allows a one-dimensional fit for bluntness to be carried out that is both efficient and robust. In particular, we require orientation, subtended angle, and foreground and background intensities to be known. This enables the image window to be rotated

[1] Of course, other distributions that might be produced by oddly shaped corners could produce values outside this range.

to align the corner with the X axis, and the model is then fitted using Brent's method [10] to minimise

$$\sum_{x=0}^{W} \sum_{y=-W}^{W} |I(x,y) - \mathrm{corner}(x,y,p)|$$

to obtain the value of the parameter p.

We model a rounded corner by a hyperbola aligned along the X axis. Knowing the subtended angle θ means that the locations of the expected boundaries of the perfect sharp corner are known. We constrain the model hyperbola to pass though the intersection of the image window and the corner boundaries. These two points are found as

$$(x_1, y_1) = \begin{cases} \left(W, \pm W \tan\frac{\theta}{2}\right) & \text{if } \theta < 90^\circ \\ \left(\frac{W}{\tan\frac{\theta}{2}}, \pm W\right) & \text{otherwise.} \end{cases}$$

so that the implicit equation of the hyperbola is

$$Q(x,y) = x^2 - \frac{x_1^2 - a^2}{y_1^2} y^2 - a^2 = 0.$$

The free parameter a specifies the distance of the rounded apex from the ideal sharp point of the corner, and increasing values of a imply increased rounding. The complete model for a blunt model is then

$$\mathrm{corner}(x,y,a) = \begin{cases} \text{foreground} & \text{if } x \geq 0 \text{ and } Q(x,y) \geq 0 \\ \text{background} & \text{otherwise.} \end{cases}$$

Another corner model we have experimented with uses the perfect wedge and replaces the apex by a circular section. For a circle of radius r we wish to locate it so that it smoothly joins the straight sections of the corner. This is obtained if the circle is positioned at $x_c = \frac{r}{m}\sqrt{m^2+1}$, and so the tangent point can be determined:

$$(x_T, y_T) = \left(\frac{x_c}{m^2+1}, m x_T\right),$$

where θ is the subtended angle and $m = \tan\frac{\theta}{2}$. The model is then

$$\mathrm{corner}(x,y,r) = \begin{cases} \text{foreground} & \text{if } [x \geq x_T \text{ and } |y| \leq mx] \text{ or } \left[(x-x_c)^2 + y^2 \leq r^2\right] \\ \text{background} & \text{otherwise} \end{cases}$$

2.4 Boundary Shape

So far we have assumed that corners have straight sides (although possibly a rounded apex). Now we consider curved sides, and distinguish between concave and convex, although the following approach is restricted to symmetric corners, i.e. both sides are either concave or convex. We work with the projection along the corner orientation again. Since the precise shape of the corner boundary is unknown we do not fit a parametric model. Instead we look to see how much the boundary is indented into or out of the foreground. Each half of the projection along the spatial axis should correspond to one side of the corner. The indentation of each side is measured by dividing the non-zero elements into two halves and fitting straight lines to each. The angle ψ between the lines then indicates the degree of concavity or convexity. If positive angles are calculated in the counterclockwise direction then the measures are simply

$$\text{concavity} = \frac{\psi}{\text{contrast}}; \quad \text{convexity} = \frac{-\psi}{\text{contrast}}$$

where large values imply greater curvature. Even though the projection is obtained by integrating the image it may still be noisy. We therefore require the line fitting to be robust, and use the least median of squares (LMedS) method [16].

3 Results

Ideally we would like the methods for measuring corner properties to work reliably over a range of conditions: varying orientations, subtended angles, degrees of noise, etc. To assess the accuracy and robustness of the new methods described here and also some of the previous ones given in [14] they have been extensively tested on synthetic data. Idealised corners were generated for 13 different subtended angles ($\theta = [30°, 150°]$) as 160×160 images, and then averaged and subsampled down to 40×40. Different levels of Gaussian noise ($\sigma = [5, 60]$) were then repeatedly added to create a total of 13000 test images. Some examples are shown in figure 4 with $\theta = \{30°, 60°, 90°, 120°, 150°\}$ and $\sigma = \{5, 10, 20, 40, 60\}$ respectively. The contrast between corner foreground and background was kept constant at 200, so that the images contained signal to noise ratios ranging from 40 to $3\frac{1}{3}$. Note that unless otherwise stated all tests use a window size of 31×31 and the corner properties were measured at the ideal location: $(20, 20)$.

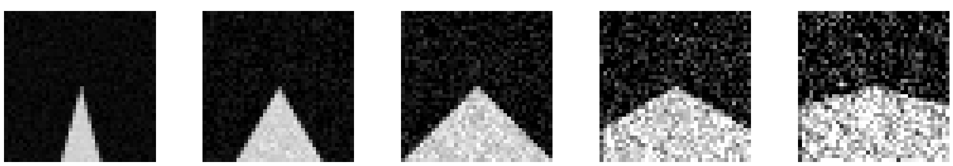


Figure 4: Sample test corner images

Table 1 shows the error rates of each method averaged over all 13000 test images. The tests were carried out twice. The first time the corner properties were measured at the known true location of the corner $(20, 20)$. The second time the Kitchen-Rosenfeld [6] detector was applied to the images, and the properties were measured at the corner that was detected closest to the true location.

3.1 Bluntness

Now we look at the effectiveness of the methods for measuring bluntness. Each of the three methods assumes a different model of corner rounding. We cannot therefore generate synthetic blunt corners using any of these models since this would

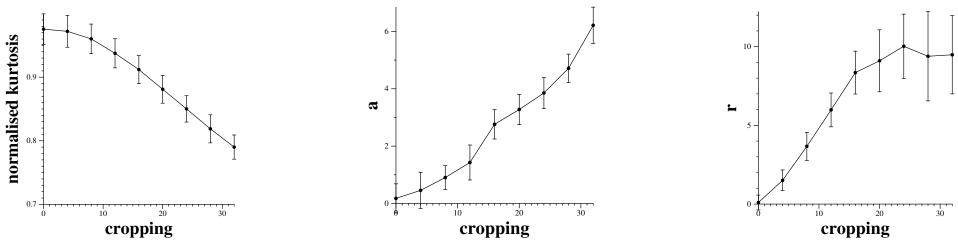
Table 1: Summary of error rates for each method of measuring corner properties

METHOD	AVERAGE ERROR			
	true position		approximate position	
	μ	σ	μ	σ
Orientation				
average orientation	1.289°	0.949°	1.314°	0.957°
multi-scale orientation histogram	3.674°	10.552°	3.608°	10.318°
thresholding	1.223°	1.175°	1.515°	1.795°
gradient centroid	3.675°	4.090°	4.085°	4.469°
intensity centroid	1.285°	1.173°	1.629°	1.845°
symmetry (pixel differences)	1.763°	0.574°	1.976°	1.172°
symmetry (line differences)	1.224°	0.670°	1.505°	1.406°
symmetry (region differences)	1.114°	0.755°	1.593°	1.682°
Subtended Angle				
single scale orientation histogram	7.688°	13.653°	7.589°	13.911°
multi-scale orientation histogram	5.178°	6.416°	5.179°	6.353°
thresholding	8.353°	4.064°	11.711°	5.165°
moments	8.353°	4.064°	11.711°	5.165°
Contrast				
multi-scale intensity histogram	35.898	52.800	34.134	51.851
thresholding and average	17.816	14.101	17.212	13.751
thresholding and median	7.443	12.019	6.989	11.667
moments	8.622	7.459	8.245	7.218

favour the corresponding bluntness measurement method. Instead we blunten the corners by cropping the apex by a straight edge. For each degree of cropping 1000 examples of 90° corners containing various amounts of noise were generated for testing. Results are shown in figure 5. Both the kurtosis and hyperbola fitting methods appear to do well over a range of degrees of cropping, displaying a fairly linear behaviour. The circle fitting method does not fare so well as it breaks down under large amounts of cropping, losing its linear response, and showing substantial variance. We can quantify the linearity of the methods using Pearsons's correlation coefficient. For the kurtosis method we test $1 - \kappa_N$ against cropping, while for the other two we simply test a and r respectively. The coefficients are 0.98364, 0.98507, and 0.93383, verifying that the first two methods are superior to the third.

3.2 Cusps

To test the measurement of concavity and convexity of cusps one synthetic symmetric example was generated of each, and Gaussian noise added as before to provide two sets of 1000 test images. The results of testing the method on test images of each corner type are shown in figure 6a. For convenience the spatial axis



(a) kurtosis method (b) hyperbola fitting method (c) circle fitting method

Figure 5: Measuring bluntness

has been scaled which also causes the angles to be scaled. Even at low SNRs the concave and convex corner types can be reliably discriminated. For comparison the method was also tested on 1000 examples of a noisy 90° straight corner. The measured angle is close to zero, allowing it to be confidently classified as a straight edge.

(a) concave and convex cusps (b) straight boundary corner

Figure 6: Effect of noise on measurement accuracy for boundary curvature

4 Conclusions

We have described various methods for measuring corner properties. In order to compare them they have been extensively tested on synthetic data. This paper presents their average performance. A more detailed analysis is presented in Rosin [15] which helps demonstrate their strengths and weaknesses, and shows under what conditions each method is suitable. It concludes that some methods work very well all the time; for instance the thresholding followed by median and the moments methods consistently measure contrast better than the multi-scale intensity histogram or thresholding followed by averaging methods. Other methods perform better than others only over a restricted range of conditions. For example, in the range of subtended angles 90° ± 40° the multi-scale orientation histogram method for measuring subtended angle outperforms the other methods, but outside this range it breaks down. Another example is when applying the multi-scale orientation histogram method to measure corner orientation. Again it outperforms all the other methods, but breaks down when the SNR drops below 10.

Overall, the average orientation, thresholding, intensity centroid and all the symmetry methods are all good choices for measuring orientation. Although it breaks down for very narrow or wide corners the multi-scale orientation histogram method is probably the best choice for measuring subtended angle. The thresholding followed by median and the moments methods are best for measuring contrast. The moments method being more robust under severe noise or when a small measurement window is used.

A more limited set of tests have been applied to the methods for measuring bluntness and cusp boundary curvature. For measuring bluntness the kurtosis and hyperbola fitting methods do better than the circle fitting method when the corner is substantially rounded. Finally, the method for measuring boundary shape of cusps appears to work reliably.

References

[1] W.F. Bischof and T. Caelli. Parsing scale-space and spatial stability analysis. *CVGIP*, 42:192–205, 1988.

[2] T. Blaszka and R Deriche. Recovering and characterizing image features using an efficient model based approach. Technical Report RR-2422, INRIA, 1994.

[3] E.R. Davies. Application of the generalized Hough transform to corner detection. *IEE-P*, E: 135(1):49–54, 1988.

[4] S. Ghosal and R. Mehrotra. Zernicke moment-based feature detectors. In *Int. Conf. Image Processing*, pages 934–938, 1994.

[5] A. Guiducci. Corner characterization by differential geometry techniques. *PRL*, 8:311–318, 1988.

[6] L. Kitchen and A. Rosenfeld. Grey-level corner detection. *PRL*, 1:95–102, 1982.

[7] S.T. Liu and W.H. Tsai. Moment preserving corner detection. *PR*, 23:441–460, 1990.

[8] S.K. Nyar, S. Baker, and H. Murase. Parametric feature detection. Technical Report CUCS-028-95, Dept. Computer Science, Columbia University, 1995.

[9] H. Ogawa. Corner detection on digital curves based on local symmetry of the shape. *PR*, 22:351–357, 1989.

[10] W.H. Press, B.P. Flannery, S.A. Teukolsky, and W.T. Vettering. *Numerical Recipes in C*. Cambridge University Press, 1988.

[11] K. Rohr. Recognizing corners by fitting parametric models. *Int. J. Computer Vision*, 9:213–230, 1992.

[12] P.L. Rosin. Adding scale to corners. In *AI'92*, pages 171–176. World Scientific, 1992.

[13] P.L. Rosin. Acquiring information from cues. *PRL*, 14:599–609, 1993.

[14] P.L. Rosin. Augmenting corner descriptors. *GMIP*, 58(3):286–294, 1996.

[15] P.L. Rosin. Measuring corner properties. Technical Report CSTR-96-18, Brunel University, 1996.

[16] P. Rousseeuw and A. Leroy. *Robust Regression and Outlier Detection*. Wiley, 1987.

[17] W.H. Tsai. Moment-preserving thresholding. *CVGIP*, 29:377–393, 1985.

[18] W. Zhang and F. Bergholm. An extension of Marr's signature based edge classification and other methods determining diffuseness and height of edges, and bar edge width. In *Int. Conf. Computer Vision*, pages 183–191, 1993.