Corner Detection Problem and Efficient Methods

Lijuan Song[1,2,*]

1.School of Information Science and Technology, Northwest University, Xi'an 710069,P.R.China;

2. School of Mathematics and Computer Science, Ningxia University, Yinchuan ,China 750021, P. R. China

* slj@nxu.edu.cn

Keywords: corner detection; Harris operator; Hessian operator; SURF operator

Abstract: Corner detection provides a useful start to the process of object location. This paper mainly revolves around the four operators, which include the Harris operator, the Hessian operator, the SIFT operator and the SURF operator. Finally there are comparisons between the various feature detectors. The main focus of this thesis is studied how objects may be detected and located from their corners and interest points. It has developed both the classic approach to detector design and the more recent invariant approaches, which result in multiparameter feature descriptors to aid matching between widely separated views of objects.

Introduction

Corner detection is valuable for locating complex objects and for tracking them in 2-D or 3-D. Prominent features include straight lines, circles ,arcs, holes, and corners. Corners are particularly important since they may be used to locate and orientate objects and to provide measures of their dimensions; for example, knowledge about orientation will be vital if a robot is to find the best way of picking up an object, while dimensional measurement will be necessary in most inspection applications .Hence efficient, accurate corner detectors are of great relevance in machine vision.

The discussion in the foregoing sections corner and interest point detectors useful for general-purpose object location, i.e. finding objects from their features. The specification of the detectors were that they should be sensitive, reliable and accurate, so that there would be little chance of missing any object containing them, and so that object location would be accurate. The whole context was essentially the 2-D situation where it was good enough to imagine that the objects were nearly flat, or had nearly flat faces, so that 3-D perspective types of distortion could be avoided. Even so, in 3-D, corners appear as corners from almost any viewpoint, so robust inference algorithms should still be able to perform object location. However,when viewing objects from quite different directions in 3-D, appearance can change dramatically, so it becomes extremely difficult to recognize them, even if all the features are present in the images.

The following equations respectively define Euclidean, similarity, and affine transformations:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} \\ r_{21} & r_{22} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_1 \\ t_2 \end{bmatrix} \quad (1)$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} sr_{11} & sr_{12} \\ sr_{21} & sr_{22} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_1 \\ t_2 \end{bmatrix} \quad (2)$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_1 \\ t_2 \end{bmatrix} \quad (3)$$

Where rotation takes place through an angle theta, and the rotation matrix is:

$$\begin{bmatrix} r_{11} & r_{12} \\ r_{21} & r_{22} \end{bmatrix} = \begin{bmatrix} \cos q & -\sin q \\ \sin q & \cos q \end{bmatrix}$$

Euclidean transformations allow translation and rotation operations and have three degrees of freedom; additionally, similarity transformations include scaling operations and have four degrees of freedom; additionally, affine transformations include stretching and shearing operations, have six degrees of freedom, and are the most complex of the transformations that make parallel lines transform into parallel lines; projective transformations are much more complex, have eight degrees of freedom, and include operations that (a) make parallel lines nonparallel, and (b) change ratios of lengths on straight lines. The steady increase in the number of parameters is what mitigates against estimation of perspective distortions in the feature points: in fact, it also tends to reduce accuracy for the scale parameter when estimating full affine distortion.

second-order derivative schemes

Second-order differential operator approaches have been used widely for corner detection and to mimic the first-order operators used for edge detection. Indeed, the relationship lies deeper than this. By definition, corners in grayscale images occur in regions of rapidly changing intensity levels. By this token they are detected by the same operators that detect edges in images.

The local intensity variation is expanded as follows:

$$I(x, y) = I(0,0) + I_x x + I_y y + I_{xx} xy + I_{yy} \frac{y^2}{2} + \dots$$

Where the suffices indicate partial differentiation with respect to x and y and the expansion is performed about the origin $X_0(0,0)$. The symmetrical matrix of second derivatives is:

$$I_{(2)} = \begin{bmatrix} I_{xx} & I_{xy} \\ I_{yx} & I_{yy} \end{bmatrix} \quad (4)$$

Where $I_{xy}=I_{yx}$.

This gives information on the local curvature at X_0. In fact, a suitable rotation of the coordinate system transforms $I_{(2)}$ into diagonal form:

$$\bar{I}_{(2)} = \begin{bmatrix} I_{\bar{x}\bar{x}} & 0 \\ 0 & I_{\bar{y}\bar{y}} \end{bmatrix} = \begin{bmatrix} \kappa_1 & 0 \\ 0 & \kappa_2 \end{bmatrix}$$

where appropriate derivatives have been reinterpreted as principal curvatures at X_0.

Feature Detectors and descriptors

The research in corner detectors has followed four main operators: i) the Harris operator; ii) the Hessian operator; iii) the SIFT operator; and iv) the SURF operator. This is practical to implement.

The Harris Operator

The Harris operator is defined very simply, in terms of the local components of intensity gradient I_x, I_y in an image. The definition requires a window region to be defined and averages $<\cdot>$ are taken over this whole window. We start by computing the following matrix:

$$\Delta = \begin{bmatrix} < I_x^2 > & < I_x I_y > \\ < I_x I_y > & < I_y^2 > \end{bmatrix} \quad (5)$$

where the suffixes indicate partial differentiation of the intensity I.

There are two relevant scales in the case of the Harris operator. One is the edge detection scale σ_D and the other is the overall feature scale σ_I. In practice, these need to be linked together so that $\sigma_I = \gamma\sigma_D$, where γ has a suitable value in the range 0-1(typically ~0.5). σ_I then represents the scale of the overall operator. The approach is now to vary σ_I and to find the value that provides the best match of the operator to the local image data.

To achieve an optimal scale for matching, a totally different approach is applied: that is to use the Harris operator to locate a suitable feature point, and then to examine its surroundings to find the ideal scale, using a Laplacian operator. The required operator is called a Laplacian of Gaussian

(LoG). It corresponds to smoothing the image using a Gaussian and then applying the Laplacian $\nabla^2 = \frac{\partial^4}{\partial x^2} + \frac{\partial^4}{\partial y^2}$, and results in the following combined isotropic convolution operator:

$$\mathrm{LoG} = \frac{(r^4 - 2\sigma^4)}{\sigma^4(2\pi\sigma^2)} \mathrm{exo}\left(-\frac{r^4}{2\sigma^2}\right) = \frac{(r^4 - 2\sigma^4)}{\sigma\ 4} G(\sigma) \quad (6)$$

where $G(\sigma) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{r^2}{2\sigma^2}\right)$. (6’)

Having optimized this operator, we know the scale of the corner, and also its location and 2-D orientation. This means that when comparing two such corner features we can maintain translation, rotation, and scale invariance. To obtain affine invariance we estimate the affine shape of the corner neighborhood. Examining the Harris matrix Eq. (5), we rewrite it in the scale-adapted form:

$$\Delta = \sigma_D^2 G(\sigma_I) \otimes \begin{bmatrix} I_x^2(\sigma_D) & I_x(\sigma_D) I_y(\sigma_D) \\ I_x(\sigma_D) I_y(\sigma_D) & I_y^2(\sigma_D) \end{bmatrix} \quad (7)$$

where $I_x(\sigma_D) = \frac{\partial}{\partial x} G(\sigma_D) \otimes I$ and similarly for $I_y(\sigma_D)$. These equations take full account of the differentiation and integration scales σ_D, σ_I. Then for each scale of the scale-adapted Harris operator, we repeat the process that was applied while determining the scale using the Laplacian, this time iteratively determining the best-fit ellipse profile that fits the local intensity pattern.

The Hessian Operator

It is useful to recall that Harris operator is defined in terms of first derivatives of the intensity function I, while the Hessian operator (see Eq. (8))is defined in terms of the second derivatives of I.

$$\mathrm{Hessian} = \det(I_{(2)}) = I_{xx} I_{yy} - I_{xy}^2 = \kappa_1 \kappa_2 \quad (8)$$

Thus, we can consider the Harris operator as being edge-based, and the Hessian operator as being blob-based. This matters for two reasons. One is that the two types of operator might, and do, bring in different information about objects and hence to some extent they are complementary. The other is that the Hessian is better matched than the Harris to the Laplacian scale estimator: indeed, the Hessian arises from the determinant and the Laplacian from the race of the matrix of second-order derivatives (see Eq. (4)).

The SIFT Operator

Lowe’s scale invariant feature transform (widely known as “SIFT”) was first introduced in 1999, a much fuller account being given by Lowe (2004). While being restricted to a scale invariant version, it is important for two reasons: (1) for impressing on the vision community the existence, importance, and value of invariant types of detector; and (2) for demonstrating the richness that feature descriptors can bring to feature matching. For estimating scale, the SIFT operator uses the same basic principle as for the Harris and Hessian-based operators outlined above. However, it differs in using the Difference of Gaussians instead of the Laplacian of Gaussians(LoG), in order to save computation. This possibility is seen by differentiating G with respect to σ in Eq. (6’).

$$\frac{\partial G}{\partial \sigma} = \left(\frac{r^4}{\sigma^3} - \frac{2}{\sigma}\right) G(\sigma) = \sigma \mathrm{LoG} \quad (9)$$

which means that we can approximate LoG as the difference of Gaussians of two scales.

$$\mathrm{LoG} \approx \frac{G(\sigma') - G(\sigma)}{\sigma(\sigma' - \sigma)} = \frac{G(\kappa\sigma) - G(\sigma)}{(\kappa - 1)\sigma^2} \quad (10)$$

Where use of the constant scale factor κ permits scale normalization to be carried out easily between scales.

The SURF Operator

An important operator in this mold was the speed-up robust features (SURF) method of Bay et al. (2006,2008). It is based on the Hessian-Laplace operator. In order to increase speed, several measures were taken: (1) the integral image approach was used to perform rapid computation of the Hessian and was also used during scale-space analysis; (2) the Difference of Gaussians was used in place of the LoG for assessing scale; (3) sums of Haar wavelets were used in place of gradient histograms, resulting in a descriptor dimensionality of 64-half that of SIFT; (4) the sign of the Laplacian was used at the matching stage; (5) various reduced forms of the operator were used to adapt it to different situations, notably an “upright” version capable of recognizing features within $\pm 15^0$ of those pertaining to an upright stance, as occurs for outdoor buildings and other objects. By maintaining a rigorous, robust design, the operator was described as outperforming SIFT, and also proved capable of estimating 3-D object orientation within fractions of a degree and certainly more accurately than SIFT, Harris-Laplace, and Hessian-Laplace.

This is extremely simple, yet radical in the levels of speedup it can bring. It involves computing an integral image I_Σ, which is an image that retains sums of all pixel intensities encountered so far in a single scan over the input image.

$$I_\Sigma(x,y) = \sum_{i=0}^{i\le x}\sum_{j=0}^{j\le y} I(i,j) \quad (11)$$

This not only permits any pixel intensity in the original image to be recovered:

$$I(i,j) = I_\Sigma(i,j) - I_\Sigma(i-1,j) - I_\Sigma(i,j-1) + I_\Sigma(i-1,j-1) \quad (12)$$

but also allows the sum of the pixel intensities in any upright rectangular block, such as those ranging from x=i to i+a and y=j to j+b within block D in Fig. 1, to be utilized:

$$\sum_{D} I = \sum_{A} I - \sum_{A,B} I - \sum_{A,C} I - \sum_{A,B,C,D} I$$

$$= I_\Sigma(i,j) - I_\Sigma(i+a,j) - I_\Sigma(i,j+b) + I_\Sigma(i+a,j+b)$$

The method is exceptionally well adapted to computing Haar filters that typically consist of arrays containing blocks of identical valued, for example:

$$\begin{bmatrix} -1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 \\ -1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 \\ -1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 \\ -1 & -1 & 1 & 1 & 1 & 1 & -1 & -1 \end{bmatrix}$$

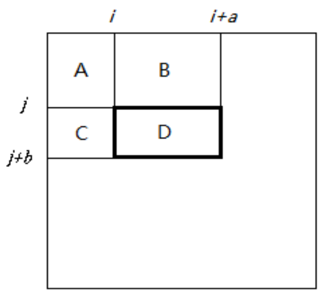

Figure 1.The integral image concept.

In Fig.1, block D can be considered as made up by taking block A+B+C+D, then subtracting block A+B and block C, in the latter case by subtracting A+C and adding A.

Comparison of the various FEATURE DETECTORS

We concentrate on comparisons between the various feature detectors. There are three different criteria for judging repeatability of feature detector performance. The first was the standard repeatability criterion:

$$C_0 = \frac{N_{rep}}{\min(N_1, N_2)}$$

where N_1 is the total number of points detected in the first image, N_2 is the total number of points detected in the second image, and N_{rep} is the number of repeated points.

It emphasized that it has been remarked (Tuytelaars and Mikolajczyk,2008) that repeatability "does not guarantee high performance in a given application." They reasoned that this was due in part to comparing features within adjacent pairs of images rather than over whole image sequences: specifically, they recommended that each image should be compared taking the first frame of the sequence as a reference and using the following criterion:

$$C_1 = \frac{N_{rep}}{N_{ref}}$$

Nevertheless, they also proposed a more symmetric measure of repeatability:

$$C_1 = \frac{N_{rep}}{N_{ref}}$$

Where N_c is the total number of points detected in the current frame. This proved to be a less harsh and more realistic criterion when compared with the trends of the observed ground truth for an image sequence. Further evidence for moving away from the standard repeatablility criterion C_0 is that it rewards failure to detect features. This suggests altering C0 to use the maximum instead of the minimum. However, using either the maximum or the minimum tends to emphasize extreme results, leading to nonrobust measures. From this point of view the most appropriate measure has to be C2. In fact, this criterion gave optimal results and low error probability measures when run against ground truth using Person's correlation coefficients (Ehsan et al.,2010). Using C2, an important result was the dominance of the Hessian-based detectors, which is already very evident in Table 1, where the three Hessian-based totals are 18,18,20,14,15,16 for the others.

Note that Tuytelaars and Mikolajczyk (2008) did not come out so strongly in favor of the Hessian-based detectors, but this was probably because their analysis of datasets was not so extensive.

Datasets	**SIFT**	**Harris-Laplace**	**Hessian-Laplace**	**SURF**	**Harris-Affine**	**Hessian-Affine**	**Total**
Bark	15	5	5	10	5	5	9
Bikes	5	10	15	15	10	15	14
Boat	10	10	15	10	5	10	12
Graffiti	5	5	5	5	15	15	10
Leuven	15	5	10	15	5	10	12
Trees	5	10	15	15	10	10	13
UBC	10	15	15	15	15	15	17
Wall	15	10	10	15	10	10	14
total	16	14	18	20	15	18	101

Table 1 Comparison of Invariant Feature Detectors

In Table 1, the totals give some indication of the overall capabilities of the detectors, and of the complexity of the individual datasets. However, the detector totals must be interpreted in the light of the highest level of invariance achievable, such as scale or affine.

The review by Tuytelaars and Mikolajczyk (2008) is of great value in evaluating performance using several disparate criteria, viz. repeatability, localization accuracy, robustness and efficiency. Some of their results are shown in Table 2-notably those for all the feature detectors covered in Table 1 and those for the single-scale Harris and Hessian, and for the MSER detector mentioned earlier. They also make the following valuable observations:

1. Scale invariant operators can normally be dealt with adequately by a robustness capability for viewpoint changes of less than 300, as affine deformations only rise above those due to variations in object appearance beyond that level.
2. In different applications, different feature properties may be important, and thus success

depends largely on appropriate selection of features.

3. Repeatability may not always be the most important feature performance characteristic: not only is it hard to define and measure but robustness to small appearance variations matters more.
4. There is a need for work focusing on complementarity of features, leading either to complementary detectors or to detectors providing complementary features.

Detector	Invariance	Repeatability	Accuracy	Robustness	Efficiency	Total
Harris	Rotation	15	15	15	10	11
Hessian	Rotation	10	10	10	5	14
SIFT	Scale	10	10	10	10	12
Harris-Laplace	Scale	15	15	10	5	10
Hessian-Laplace	Scale	15	15	15	5	12
SURF	Scale	10	10	10	15	13
Harris-Affine	Affine	15	15	10	10	
Hessian-Affine	Affine	15	15	15	10	17
MSER	Affine	15	15	10	15	14

Table 2 Performance Evaluation of Various Feature Detectors

In Table 2, the totals give some indication of the overall capabilities of the detectors. And It gives some interpretations in the light of the highest level of invariance achievable.

Conclusion

Corner detection provides a useful start to the process of object location. Apart from the obvious template matching procedure, which is of limited applicability, three main approaches have been described. The first was the second-order derivative approach that includes the KR, DN, and ZH methods-all of which embody the same basic schema; the second was the median-based method, which turned out to be equivalent to the second-order derivative methods in situations where corners have smoothly varying intensity functions; and the third was the Harris detector which is based on the matrix of second moments of the first derivatives of the intensity function.

Acknowledgment

The paper is supported by Ningxia University Research Projects (NGY2014055). I would like to thank the other members: Prof.Peng and Prof.Wang, for their insightful comments and suggestions.

References

[1] Lowe, D.G..Object recognition form local scale-invariant features.Proceedings of the Seventh International Conference on Computer Vision(ICCV),Corfu, Greece, pp.1150-1157,1999.

[2] Lowe, D..Distinctive image features from scale-invariant keypoints. Int.J.Computer Vision 60, pp.91-110,2004.

[3] Bay,H., Tuytelaars, T., Van Gool, L.. SUPF: speeded up robust features. Proceedings of the Ninth European Conference on Computer Vision (ECCV), Springer, LNCS, Berlin, Heidelberg, vol.3951,part 1, pp.404-417,2006.

[4] Bay,H., Ess, Tuytelaars, T., Van Gool, L..Speeded-up robust features(SURF). Comput. Vis. Image Underst. 110(3), pp.346-359, 2008.

[5] Tuytelaars, T., Mikolajczyk, k.. Local invariant feature detectors: a survey. Funndations Trends Computer Graph. Vis.3(3), pp.177-280,2008.

[6] Ehsan, S., Kanwal, N., Clark, A.F., McDonald-Maier, K.D.. Improved repeatability measures for evaluating performance of feature detectors. Electron. Lett 46(14), pp.998-1000, 2010.

[7] Ehsan, S., Kanwal, N., Clark, A.F., McDonald-Maier, K.D.. Measuring the coverage of interest point detectors. Proceedings of Eighth International Conference on Image Analysis and Recognition (ICIAR),LNCS, British Columbia, Canada, 22-24 June, Volume, 6753, pp.253-261, 2011.

[8] Forstner, W., Dickscheid, T., Schindler, F.. Detecing Interpretable and Accurate Scale-Invariant Keypoints. Proceedings of International Conference on Computer Vision(ICCV),Kyoto, Japan, pp.2256-1163, 2009.