Accuracy in Image Measure

Pascal Brand, Roger Mohr

Appeared pp 218–228, in SPIE, VIDEOMETRICS III
31 October – 4 November '94, Boston, U.S.A

Accuracy in Image Measure

Pascal Brand Roger Mohr

LIFIA – INRIA Rhônes-Alpes
46 avenue Félix Viallet
38031 Grenoble Cedex FRANCE
e-mail: Pascal.Brand@imag.fr , Roger.Mohr@imag.fr

ABSTRACT

The aim of this paper is to show how image points can be extracted accurately. We will restrict our search to specific points identified by corners, which are stable given a sequence. Our approach makes use of a model–based corner detector. It matches a part of the image containing a corner against a predefined corner model. Once the fitting is accomplished, the position of the corner in the image can be deduced by the knowledge of the corner position in the image.

The validity of our approach has been proven with 4 independent tests. It is shown that the accuracy which can be achieved is 1/10th of a pixel.

Keywords : extraction of corners, model-based algorithm, accurate location

1 INTRODUCTION

Many applications, such as car crash inspection[1] or medical imaging, require very accurate image measurements. In this context the paper deals with the specific case of an image corner detector. Generally, it is hard to obtain accuracy in image measurements because of different source of noise :

- Electronic noise : data acquisition can never be the same even under perfectly identical lighting and viewing conditions (line-jitter, image digitalisation instability).
- Optical noise : the pin-hole model is usually used to model the camera, but it represents only an approximation of a real camera. The latter, in fact, is much more complex, and its modeling depends heavily on the used lenses.
- Software noise : it is very complex to extract correctly all the features of an image, and to achieve accuracy.

New technologies allow to correct for the first two sources of error. Numerical frame grabbers coupled with numerical cameras have recently appeared which can guarantee a line-jitter of 0 pixel. Optical distortions can be corrected using appropriate software or optical adjustment.

However, the largest errors in image data acquisition come from the feature extraction process. Algorithms which need point correspondences, such as camera calibration, 3D-reconstruction or epipolar geometry computation can be very sensitive to the noise introduced by the point localisation errors. The precise extraction of points in the image has always been treated in specific cases, such as

- the accurate detection of a target.[2]
- the adaptive template matching based on correlation.[7,9]
- the sub-pixel corner detectors.

This paper will follow a similar principle by analysing the case of a precise corner detector. The first sub-pixel corner detectors used gaussian or laplacian filters.[5] Recent research has shown the efficiency of model-based corner detector algorithms.[4,13,11] These methods are more accurate, but they require an initial position of the corner. The method presented in this paper follows a similar approach. It is an adaptation of Deriche's approach.[4]

The paper is organised as follows: in the next section, we will present an algorithm to detect corners accurately; in Section 3, four independent experiments illustrate how to test the accuracy of the corner detection; concluding remarks will close the paper.

2 THE METHOD

We present a model-based algorithm. The aim of such an algorithm is to match a part of an image which contains a corner against a corner model. This is achieved by searching the best transformation which minimises the criterion

$$\chi^2 = \sum_{x_i, y_i} (I_c(x_i, y_i) - I_m(x'_i, y'_i))^2$$

where

- $I_c(x_i, y_i)$ is the intensity of the pixel (x_i, y_i) in the image. All the pixels (x_i, y_i) which are taken into account belong to a template centered on the initial approximate position of the corner.
- $I_m(x'_i, y'_i)$ is the intensity of the pixel (x'_i, y'_i) in the model. A pixel (x'_i, y'_i) in the model corresponds to a pixel (x_i, y_i) in the image. These coordinates are obtained by applying a skew, a rotation and a translation to the set of points (x_i, y_i) which belong to the image corner. These parameters (rotation, skew and translation) are unknown.

The model (see Figure 1) that we use for a corner is an adaptation of Deriche's model. Our corner model consists of 3 regions: P_1 and P_2 are two regions having a homogeneous grey level, whereas the third region P_3 is a gaussian blur of the first 2 regions. In this model the corner is at position (x'_0, y'_0), which is known and fixed. The pixel intensity $I_m(x, y)$ in the model can be written as:

$$I_m(x, y) = \begin{cases} A \text{ if } (x, y) \in P_1 \\ B \text{ if } (x, y) \in P_2 \\ \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} g(\alpha) g(\beta) I'(x - \alpha, y - \beta)\ d\alpha\ d\beta \quad \text{if } (x, y) \in P_3 \end{cases}$$

where

$$I'(x, y) = \begin{cases} A \text{ if } x \geq x'_0 \text{ and } y \leq y'_0 \\ B \text{ otherwise} \end{cases}$$

$g(\alpha)$ is the gaussian function

$$g(\alpha) = \frac{1}{\sqrt{\pi\sigma^2}} \exp^{-\alpha^2/\sigma^2}$$

We have implemented a model which is suitable for L-corners (corners made of only 2 edges). In this model the 2 edges are perpendicular. This does not prevent the algorithm from detecting corners with 2 non-perpendicular edges. In fact the minimization can cater for a skew deformation of the model.

The corner position is estimated from the parameters which define the best fit between the image and the model. The optimized parameters are the rotation, the skew, the translation, and the two homogeneous grey levels A and B. The value of the blur σ is fixed. The optimization of χ^2 is performed using the Levenberg–Marquardt algorithm.[12]

Figure 2 shows two zoomed views of the corner detection. For each corner, the figure shows the approximation of the edges employed to detect the corner, and the final location that best fits the corner model.

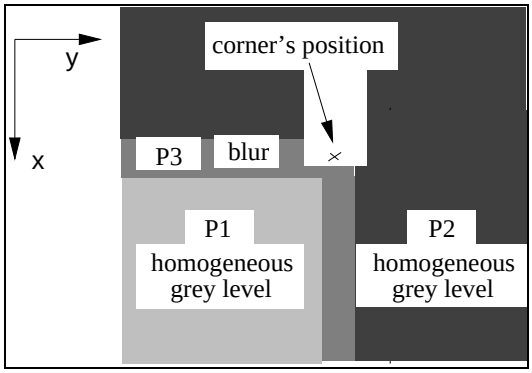

Figure 1: The model which is used for the accurate detection of the corner

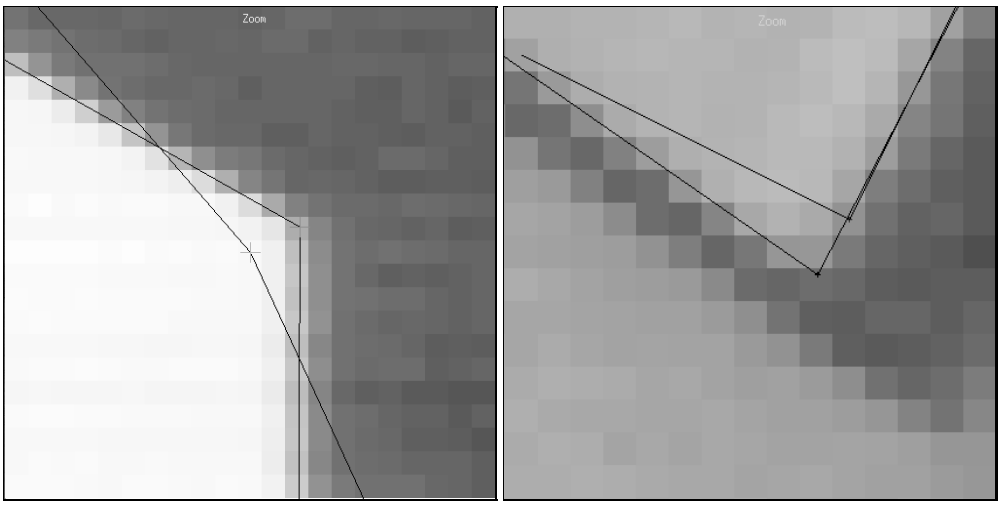

Figure 2: Zoomed views on the location of corners (first approximation and final estimation)

3 EXPERIMENTS

In order to test the accuracy which can be reached with the presented corner extraction, we have performed the 4 following experiments :

1. alignment of points, using the fact that alignment is preserved under perspective projection.
2. comparison between a real 3D-reconstruction and different simulated ones, for different values of noise in the 2D-data.
3. quantification of the dependency of the quality of the extracted points on the epipolar geometry.
4. analysis of the stability of the cross-ratios computed with the extracted corners.

All these experiments have been performed with the same images. Figure 3 shows a calibration plane which contains $N_p = 92$ corners. This frame was chosen because it contains L-corners easy to detect (this is an advantage because the noise is reduced), and the 3D-positions of the corners are exactly known. Moreover, all the corners lie on a plane, and they are equally spaced.

Since 3D-reconstruction and epipolar geometry require volumetric data (all the 3D-points should not be coplanar), three different images have been taken with the calibration plane placed in three parallel positions. The set-up is shown in Figure 4. For each plane, the corners are extracted. We have taken a sequence of images of the calibration grid in 3 different positions. Each view V_i of the sequence contains $N = 3N_p = 276$ points $p_j^{(i)}$. For each 2D-point $p_j^{(i)}$, we know the position of the 3D-points P_j which is projected on $p_j^{(i)}$. Figure 3 shows the 3 views of the first plane.

For the experiments an "Imaging Technology Incorporated Series 150" frame grabber was used. This frame grabber provides a 512×512 pixel image. The images have been taken using a PULNIX 6-EX camera (CCD technology) with a KINOPTICS lens ($12.5mm$ focal length). The size of the seen object is $19.8 \times 12.6 \times 6.0$ cm^3.

Since we deal with well-focused images, we can use a small value for the blur in the model. In our experiments the value of σ was set to 0.5. The used model size is equal to 21×21 pixel. The initial approximate location of a corner is supplied by the bundles (intersections of 2D-edges) produced by the software package FEAGRO[8] (FEAture GROuping).

The next four subsection will describe the experiments in more detail.

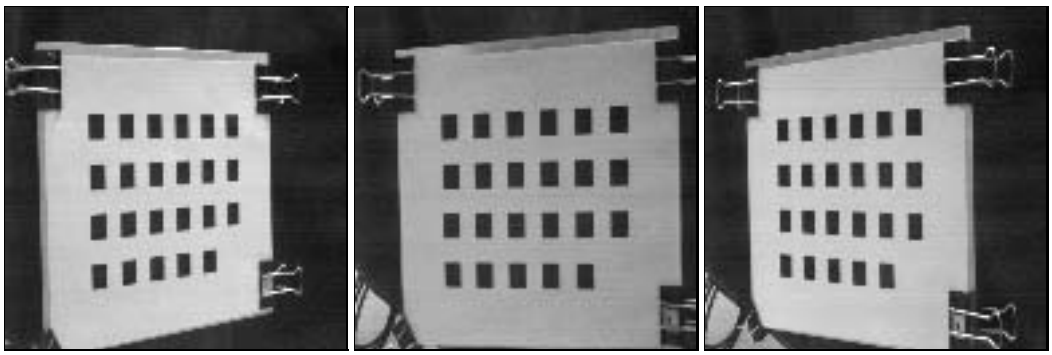

Figure 3: The 3 views of the first plane of the calibration grid.

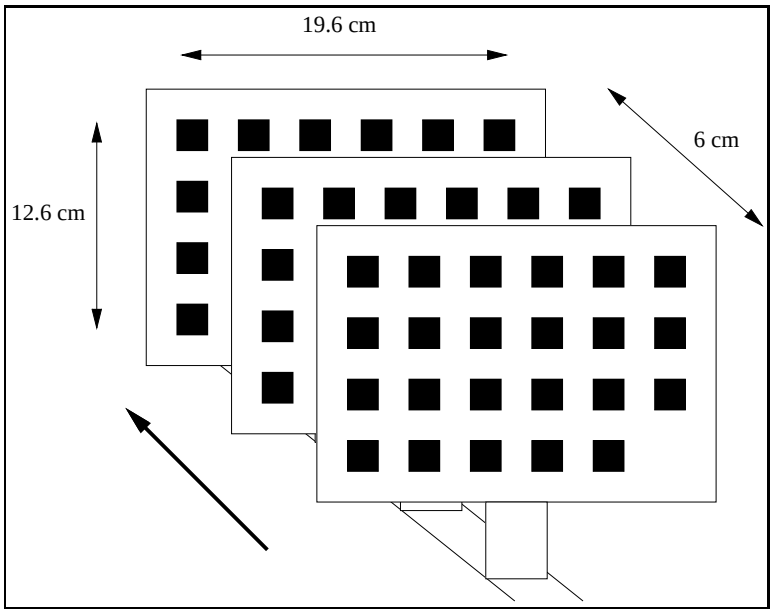

Figure 4: Motion of the calibration plane to get from a planar object a 3D object

3.1 Alignment

The alignment test makes use of the 3D-collinearity of the corners in the calibration grid and exploits the property that alignments are preserved under perspective projection (we use the pin-hole model for the camera). So, for each set $\{(x_i, y_i)\}$ of 2D-points corresponding to aligned 3D-points, we fit, in a least square sense, a line defined by $y = ax + b$, optimizing the following criterion

$$\chi^2 = \sum_i (y_i - ax_i - b)^2$$

Then, we compute the mean and the standard deviation of the distance between the fitted lines and the points. The results, for horizontal and vertical alignments, are displayed in Table 1. Such table illustrates that the corner extraction was computed up to a sub-pixel accuracy. If there is no bias in the extraction (the errors are all independent), the obtained accuracy is equal to $\sqrt{2\ \text{mean}^2} = 0.11$ pixel.

	Mean	Standard Deviation
Vertical	0.07	0.056
Horizontal	0.08	0.069

Table 1: Quality alignment

3.2 3D-Reconstruction

The second experiment consists of the 3D-reconstruction of the corners. Qualitatively, a good 3D reconstruction of the points can be obtained when corners have been accurately detected. The quality of the 3D-reconstruction can be easily obtained from the knowledge of the location of the 3D-corners (because we are observing a calibration grid).

A standard method for 3D-reconstruction is based on 2 steps, namely calibration and triangulation. In the calibration stage the perspective matrices are computed using all the 3D-points. Triangulation is then applied using the computed transformation. The major drawback of this method is that it needs the knowledge of many 3D-points. Besides its stability depends on the number of points.

This is the main reason why we used the method described in the paper of Boufama *et al.*[3] Using correspondences of points in a sequence and the 3D-positions of only 5 points P_j, this method computes the perspective matrix for each view and the 3D-positions of all the points.

Once the reconstruction is done, we have measured its quality, defined by the mean distance between the estimated 3D-positions and their real 3D-positions. The drawn statistics have a mean distance of $0.020cm$, and a standard deviation of $0.010cm$. Therefore the reconstruction have a precision of $1/1000th$ of the size of the scene object.

A quantitative value of the extraction accuracy cannot be deduced from these results. A simulation of the 3D-reconstruction has been performed as follow

> We assume we have the 3D-coordinates of a set of points viewed from 3 vantage points. The 3 perspective matrices, which set the relation between the scene and the camera, are given. The coordinates of the 2D-projection of the 3D-set are perturbed by a white gaussian noise. We then apply Boufama's method[3] to reconstruct the scene using the perturbed 2D-points. Finally we draw some statistics on the results.

In order to compare the errors obtained with the simulated data and in the real case, we have chosen a configuration (3D-points and perspective matrices) very close to the real one. The 3D points coincide with the corner of the calibration grid, and the perspective matrices have been obtained using the Faugeras-Toscani calibration.

The results of the simulation are summarized in the graph of Figure 5a. It displays for different values of noise (ranging from 0.02 to 0.50 pixel) the mean and the standard deviation of the error of the reconstruction. The graph is not smooth. This is because of the dependency of the euclidean reconstruction on the errors of the extraction of the 5 points used as a projective base. Looking at the graph in Figure 5a, for the value $0.020cm$ of the mean, the corresponding noise ranges between 0.09 and 0.20 pixels.

3.3 Epipolar geometry

The epipolar geometry between 2 images is considered as a projective calibration. It represents the relation between 2 images, and is represented by a 3×3 matrix of rank 2 : the fundamental matrix F. Given 2 correspondent points u_1 and u_2 between 2 images, we have $u_2^t.F.u_1 = 0$, i.e. u_2 lies on the epipolar line $F.u_1$.

The estimation of the epipolar geometry can be achieved by 8 known correspondent image points.[6] Such estimate can be improved using all the corner correspondences present in the calibration grid. Only the epipolar geometry between image 1 and 3 has been computed to increase the stability of the process. It is well-known in fact that the closer are the images the more unstable is the epipolar geometry. The epipolar geometry has been obtained using all the corners. We have then drawn some statistics on the distance d between corners $p_j^{(3)}$ of the third image and their corresponding epipolar line $F.p_j^{(1)}$:

$$\text{Mean} = \bar{D} = \frac{1}{N}\sum_{j=1}^{N} d\left(p_j^{(3)}, F.p_j^{(1)}\right) = 0.11$$

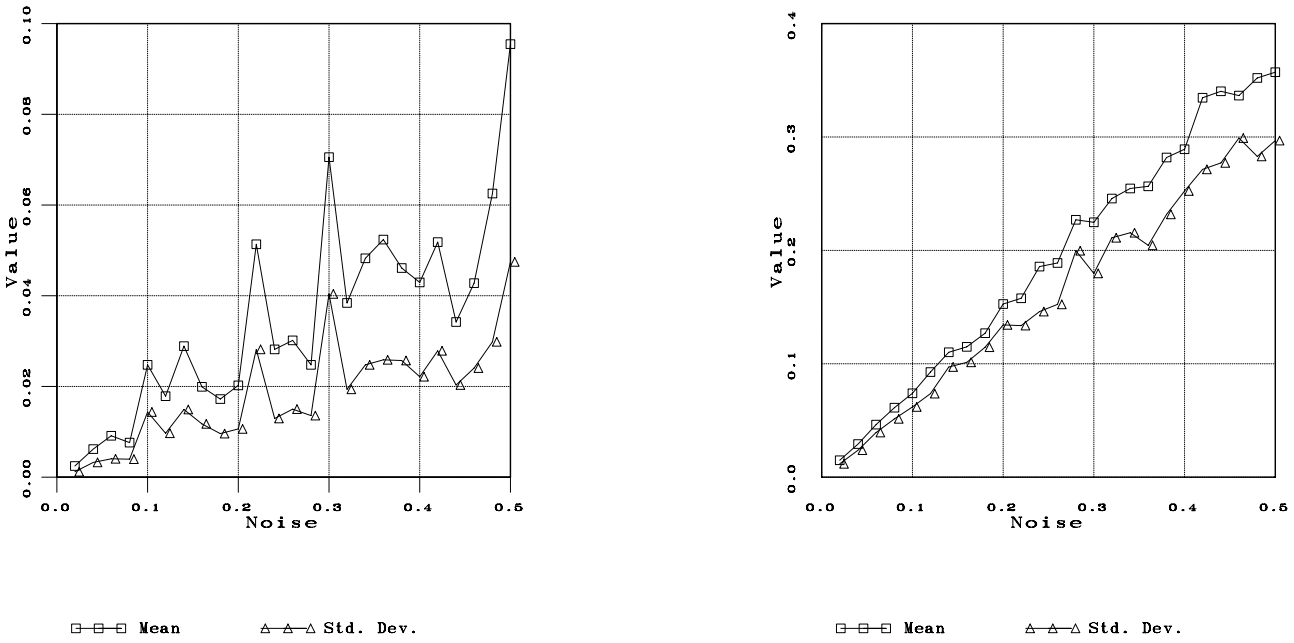

Figure 5: Results on the accuracy with simulated data. The X-axis represents the standard deviation of the added gaussian noise, and the Y-axis represents the value of the mean or the standard deviation of the errors. Graph a) shows the error on the 3D-reconstruction (from 0.0 to 0.10 cm) , and graph b) shows the error on the epipolar geometry (from 0.0 to 0.4 pixel)

$$\text{Standard deviation} = \sqrt{\frac{1}{N-1}\sum_{j=1}^{N}\left(d\left(p_j^{(3)}, F.p_j^{(1)}\right) - \bar{D}\right)^2} = 0.090$$

Such statistics are insufficient to derive an accuracy measure of the extracted corners. The method used to build the epipolar geometry is a minimisation of

$$\chi = \sum_{j=1}^{N}\left(d^2\left(p_j^{(3)}, F.p_j^{(1)}\right) + d^2\left(p_j^{(1)}, F^t.p_j^{(3)}\right)\right)$$

with the constraint $\text{rank}(F) = 2$. This involves a non-linear optimisation which is unstable in the presence of noise.

In order to avoid this problem, we have simulated the computation of the epipolar geometry with noisy 2D-points. The data is the same as for the simulated 3D-reconstruction. The results of this simulation are shown in Figure 5b. For different values of a white gaussian noise, ranging from 0.02 to 0.50 pixel (X-axis), on the 2D-points, we have computed the fundamental matrix F, and then derived the mean and the standard deviation (Y-axis) of the distance $d(p_j^{(3)}, F.p_j^{(1)})$ for each corner $p_j^{(1)}$ of the first image.

The graph of the error of the epipolar geometry (Figure 5) is more stable than the one obtained for the simulated 3D-reconstruction because all errors are equally taken into account. Similar values to the real case are obtained when the noise was equal to 0.13 pixel.

3.4 Cross-ratios

3.4.1 Non-existence of a bias

We prove that, by means of the invariance of the cross-ratios results, the extraction of corners is unbiased. The experiments will show that the extracted corner is correctly positioned in the image. The cross-ratio of four aligned points is a basic invariant of projective geometry.[14] It can be expressed by

$$K(a,b,c,d) = \frac{a-c}{a-d} \div \frac{b-c}{b-d}$$

where a, b, c and d are the 1D-coordinates of four aligned points. In our case, we will compute some cross-ratios in the image close to the theoretical cross-ratio $K(0,3,2,1) = 4$ because such cross-ratios appear frequently in the calibration plane we used.

Let's separate these cross-ratios in 2 sets : a set composed of 4 aligned points lying on 2 squares, and a second set with corners lying on 3 squares (Figure 6). If a bias exists, then the value of the cross-ratios computed with the 2 different configurations shall not be equal.

For each set, we have computed the mean and the standard deviation of the cross-ratio distribution. The drawn statistics are very similar as shown in Table 2. This allows us to say that the corner extraction is unbiased.

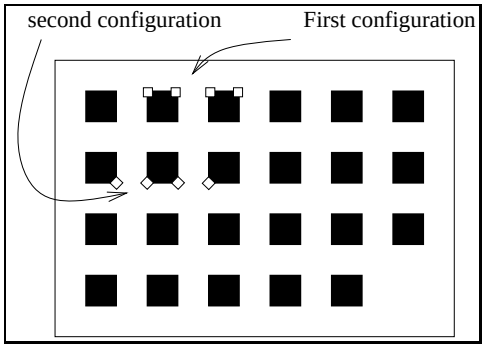

Figure 6: Example of an item of a cross-ratio composed by four aligned points lying on 2 squares (first set) and on 3 squares (second set)

Image	Configuration of the corner	Number of cross-ratios	Mean	Standard deviation	Maximum Error
Image 1	Set 1	90	4.031	0.050	0.145
	Set 2	72	3.971	0.046	0.133
Image 2	Set 1	90	3.990	0.041	0.134
	Set 2	72	4.009	0.039	0.092
Image 3	Set 1	90	4.022	0.053	0.157
	Set 2	72	3.982	0.052	0.157

Table 2: Stability, for each image, of cross-ratios composed by four aligned points lying on 2 squares (first set) or on 3 squares (second set)

3.4.2 Accuracy

In this subsection, we will compute the accuracy of the corner extraction, using cross-ratios properties. First of all, we show that cross-ratio values can be approximated by a gaussian approximation. To prove that, we have computed for each view of the calibration plane in the three different positions the mean and the standard deviation of the cross-ratio values. Results are displayed in Table 3. Each drawn distribution was compared with a gaussian distribution using the Kolmogorov-Smirnov similarity test.[12] This test requires the mean and the standard deviation of the population, and returns a measure of how probable is that the test is positive. Table 3 shows that probability measures are always greater than 50%, and in the majority of the cases, it is equal to 90%. In conclusion we can assume that the cross-ratio distribution is gaussian. Figure 7 displays the histogram of a distribution of the cross-ratios, with the gaussian distribution.

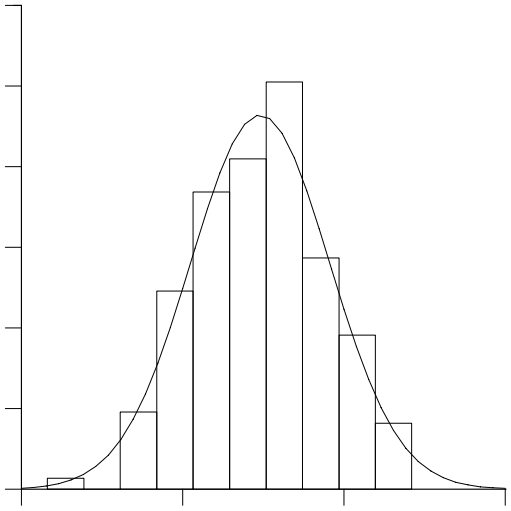

Figure 7: histogram of the cross-ratio distribution for the second image. The density function of the gaussian is superimposed

Image number	Plane number	Number of cross-ratios	Mean $\bar{K}$	Standard Deviation σ_K	Max error	Distance λ in 2D	Accuracy σ	Probability to have a Gaussian
Image 1	1	54	4.004	0.055	0.138	20.12	0.12	0.891
	2	54	4.005	0.053	0.145	19.74	0.12	0.964
	3	54	4.006	0.064	0.145	19.36	0.14	0.997
	All	162	4.005	0.057	0.145	19.74	0.13	–
Image 2	1	54	3.998	0.043	0.092	25.10	0.12	0.818
	2	54	3.999	0.041	0.073	24.16	0.11	0.697
	3	54	3.997	0.040	0.134	23.30	0.10	0.919
	All	162	3.998	0.041	0.134	24.19	0.11	–
Image 3	1	54	4.003	0.042	0.100	19.40	0.09	0.935
	2	54	4.005	0.066	0.155	19.11	0.14	0.551
	3	54	4.004	0.058	0.157	18.83	0.12	0.743
	All	162	4.004	0.056	0.157	19.11	0.12	–

Table 3: Distribution statistics of the value of cross-ratios. The theoretical value is 4.

In the case of cross-ratios equal to $K(0, 3, 2, 1) = 4$, it can be proved that K can be linearized.[10] Using a

first-order Taylor expansion, we have the relation between the standard deviation of the cross-ratio computed in the image, and the standard deviation of the precision of the extraction of the corners :

$$\sigma_k^2 = J_K^t \cdot \Lambda \cdot J_k$$

where

- σ_k is the standard deviation of the cross–ratio K. For each calibration plane view, the cross-ratios of the ith quadruplet of aligned points is equal to K_i, and so

$$\sigma_k^2 = \frac{1}{n-1}\sum_{i=1}^{n}(K_i - \bar{K})^2 \quad \text{with mean} \quad \bar{K} = \frac{1}{n}\sum_{i=1}^{n} K_i$$

- J_K is the jacobian vector of K :

$$J_K = \left(\frac{(b-d)(c-d)}{(a-d)^2(b-c)} \quad \frac{(a-c)(d-c)}{(a-d)(b-c)^2} \quad \frac{(b-d)(a-c)}{(a-d)(b-c)^2} \quad \frac{(a-c)(b-a)}{(a-d)^2(b-c)} \right)^t$$

$$\text{and so} \quad J_k(\lambda a, \lambda b, \lambda c, \lambda d) = \frac{1}{\lambda} J_k(a,b,c,d) \quad \text{and} \quad ||J_k(0,3,2,1)||^2 = 80$$

- Λ is the covariance matrix of the 4 points which form the cross–ratio. Since there is no bias in the corner detection (proved in the previous subsection), we can write $\Lambda = \sigma^2 \cdot I_4$, where σ represents the standard deviation of the corner detection and I_4 is the 4×4 identity matrix.

Therefore, if we take cross-ratios made by 4 consecutive points, having the configuration with cross-ratio $K(0,3,2,1) = 4$

$$\sigma_K^2 = \sigma^2 \cdot ||J_k||^2 \qquad \Longrightarrow \qquad \sigma = \sigma_K \frac{\lambda}{\sqrt{80}}$$

where λ is the distance between two consecutive corners. This distance is quite stable for couple of points lying on the same plane. This allows us to take the mean of such distance to compute λ.

Table 3 shows statistics of cross-ratios values for each calibration plane view. This yields to a precision close to 1/10th of a pixel.

4 CONCLUSIONS

We have presented an algorithm to detect corners. The main goal was to obtain a precise location of corners. The used method is a model-based algorithm. The model contains 2 regions of unknown grey level, separated by a gaussian blur. The position of the corner in the image is obtained by optimising a transformation (represented by a skew, a rotation and a translation) which best fits the image with the corner model.

The obtained accuracy has been tested in 4 different ways: quality of the alignment of corners, accuracy of the 3D-reconstruction and of the epipolar geometry and stability of cross-ratios. All the results of the tests are coherent, and lead to a precision close to $1/10th$ of a pixel. Moreover, we have shown that the extraction is unbiased and that the error can be described by a gaussian distribution.

The current technique can be improved. The number of candidate corners can be increased by employing a classical corner detector. The technique can also be extended. We can apply the same principle to any shape model to be able to detect targets.

5 REFERENCES

[1] H. Beyer. Photogrammetric on-line inspection for car crash analysis. In *Proceedings of First Australian Photogrammetric Conference*, Sydney, Australia, 1991.

[2] H.A. Beyer. *Geometric and Radiometric Analysis of a CCD-Camera Based Photogrammetric Close-Range System*. PhD thesis, ETH-Zurich, 1992.

[3] B. Boufama, R. Mohr, and F. Veillon. Euclidean constraints for uncalibrated reconstruction. In *Proceedings of the 4th International Conference on Computer Vision, Berlin, Germany*, pages 466–470, May 1993.

[4] R. Deriche and T. Blaszka. Recovering and characterizing image features using an efficient model based approach. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, New York, USA*, pages 530–535, June 1993.

[5] R. Deriche and G. Giraudon. Accurate corner detection: an analytical study. In *Proceedings of the 3rd International Conference on Computer Vision, Osaka, Japan*, 1990.

[6] O.D. Faugeras, Q.T. Luong, and S.J. Maybank. Camera Self-Calibration: Theory and Experiments. In G. Sandini, editor, *Proceedings of the 2nd European Conference on Computer Vision, Santa Margherita Ligure, Italy*, pages 321–334. Springer-Verlag, May 1992.

[7] A.W. Gruen. Adaptative least squares correlation: a powerful image matching technique. *S. Afr. Journal of Photogrammetry, Remote Sensing and Cartography*, 14(3):175–187, 1985.

[8] R. Horaud, T. Skordas, and F. Veillon. Finding geometric and relationnal structures in an image. In *Proceedings of the 1st European Conference on Computer Vision, Antibes, France*, pages 374–384. Springer-Verlag, April 1990.

[9] R. Mohr, P. Brand, and P. Remagnino. Correlation techniques in adaptive template matching with uncalibrated cameras. In *Vision Geometry III, SPIE's international symposium on photonic sensors & control for commercial applications*, October 1994. to appear.

[10] L. Morin. *Quelques Contributions des Invariants Projectifs à la Vision par Ordinateur*. PhD thesis, Institut National Polytechnique de Grenoble, January 1993.

[11] C.M. Orange and F.C.A Groen. Model based corner detection. In *Proceedings of the Conference on Computer Vision and Pattern Recognition, New York, USA*, pages 690–691, June 1993.

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

[13] K. Rohr. Recognizing corners by fitting parametric models. *International Journal of Computer Vision*, 9(3):213–230, December 1992.

[14] J.G. Semple and G.T. Kneebone. *Algebraic Projective Geometry*. Oxford Science Publication, 1952.