



**HAL**  
open science

## Curvature based corner detector for discrete, noisy and multi-scale contours

Bertrand Kerautret, Jacques-Olivier Lachaud, Benoît Naegel

► **To cite this version:**

Bertrand Kerautret, Jacques-Olivier Lachaud, Benoît Naegel. Curvature based corner detector for discrete, noisy and multi-scale contours. *International Journal of Shape Modeling*, 2008, 14 (2), pp.127-145. 10.1142/S0218654308001117 . inria-00437575

**HAL Id: inria-00437575**

**<https://inria.hal.science/inria-00437575>**

Submitted on 28 Mar 2011

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

International Journal of Shape Modeling  
© World Scientific Publishing Company

## Curvature Based Corner Detector for Discrete, Noisy and Multi-scale Contours

B. Kerautret,\*†

*LORIA, Nancy-University - IUT de Saint Dié des Vosges*  
*54506 Vandœuvre-lès-Nancy Cedex*  
*kerautre@loria.fr*  
*<http://www.loria.fr/~kerautre>*

J.-O Lachaud \*

*LAMA, University of Savoie*  
*73376 Le Bourget du Lac*  
*jacques-olivier.lachaud@univ-savoie.fr*  
*<http://www.lama.univ-savoie.fr/~lachaud>*

B. Naegel

*LORIA, Nancy-University - IUT de Saint Dié des Vosges*  
*54506 Vandœuvre-lès-Nancy Cedex*  
*benoit.naegel@loria.fr*  
*<http://www.loria.fr/~naegelbe>*

Received (Day Month Year)

Revised (Day Month Year)

Accepted (Day Month Year)

Communicated by (xxxxxxxxxx)

Estimating curvature on digital shapes is known to be a difficult problem even in high resolution images<sup>10,19</sup>. Moreover the presence of noise contributes to the instability of the estimators and limits their use in many computer vision applications like corner detection. Several recent curvature estimators<sup>16,13,15</sup>, which come from the discrete geometry community, can now process damaged data and integrate the amount of noise in their analysis. In this paper, we propose a comparative evaluation of these estimators, testing their accuracy, efficiency, and robustness with respect to several type of degradations. We further compare the best one with the visual curvature proposed by Liu *et al.*<sup>14</sup>, a recently published method from the computer vision community. We finally propose a novel corner detector, which is based on curvature estimation, and we provide a comprehensive set of experiments to compare it with many other classical corner detectors. Our study shows that this corner detector has most of the time a better behavior than the others, while requiring only one parameter to take into account the

\*This work was partially funded by the ANR project GeoDIB, n° ANR-06-BLAN-0225.

†Bertrand Kerautret was partially funded by a BQR project of Nancy University

2 *B. Kerautret, J.-O Lachaud, B. Naegel*

noise level. It is also promising for multi-scale shape description.

*Keywords:* curvature estimators, corner detection, multi-scale analysis, dominant point extraction.

## 1. Introduction

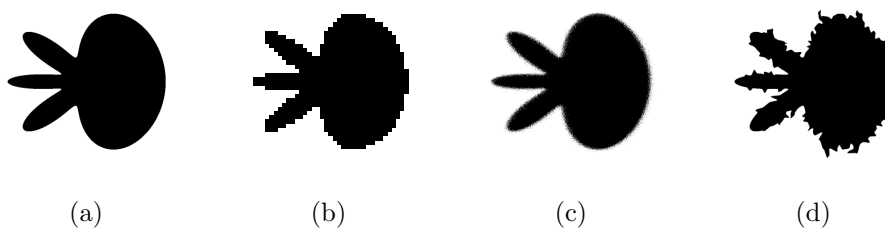


Fig. 1. Example of damaged data: (a) original shape, (b) low digitization artefacts, (c) pepper and salt noise or quantification effects, (d) damaged data due to imperfect segmentation and noise removing image processing.

Curvature estimation is an important task for shape analysis, matching or segmentation. This geometric characteristic has indeed many nice properties for these applications: it is invariant under isometries; it allows to discriminate between straight, smoothly curved, and corner parts; all feature or dominant points are defined from curvature. Its main default is to be sensitive to noise and small perturbations. The correct, precise, and robust estimation of curvature along a shape boundary description is thus a prerequisite for many applications in image analysis.

As noted by several authors (see for instance Utcke<sup>19</sup>), it is already difficult to get a precise estimation of curvature along digital contours that are perfect digitizations of regular shapes. Furthermore, real applications tend to add perturbations to the original data: low resolution digitization, pepper and salt noise, quantification noise, several scales of analysis. It is for instance very difficult to compute the correct curvature for the perturbed images (b,c,d) of Fig. 1. This is why the range of applications of curvature estimators is for now rather limited in the image analysis and computer vision community.

Recently, four new methods were proposed to compute curvature (or pseudo curvature) along the boundary of noisy discrete data. First Nguyen *et al.*<sup>16</sup> define a new curvature estimator (called NDC estimator) by extending the estimator of osculating circles (called CC estimator) proposed by Coeurjolly *et al.*<sup>6</sup> by using blurred segments. By this way, the estimation of osculating circles had a better behavior on noisy contours and was also meaningful for a non connected (but ordered) set of points. Then a second approach (called GMC estimator), proposed by Kerautret and Lachaud<sup>13</sup>, suggests to minimize curvature while satisfying geometric

constraints derived from tangent directions computed on the discrete contour. The robustness to noise of this approach was given also by blurred segments but used in a different manner. The third curvature estimator (called BCC estimator), proposed at the same time by Malgouyres *et al.*<sup>15</sup>, suggests to use discrete binomial convolution to obtain a convergent estimator adapted to noisy data. From these three new estimators we can add the new notion of multi-scale visual curvature (called VC estimator) introduced by Liu *et al.*<sup>14</sup>. This new definition takes into account both digitization artefacts and noise in data with a scale parameter.

In this work, we propose to evaluate experimentally the accuracy, efficiency, and robustness of the three first estimators by using different test contours and by measuring the precision of the estimation with several grid sizes. The visual curvature is also compared with the GMC estimation and with the expected curvature. After comparing the accuracy of curvature fields obtained with low, intermediate, and fine resolutions, we measure the robustness of these estimators on noisy contours. The GMC estimator<sup>13</sup> appears the most stable and robust. Considering its qualities, we then define a new corner detector based on the GMC curvature estimation. This detector is compared with other standard corner detectors and with the recently-published corner detector based on digital multi-scale visual curvature.

The paper is organized as follows. In the following section, after introducing briefly some notions on digital blurred segments, we give a short description of the four curvature estimators. Then the experimental comparison of these estimators is given in Section 3. In Section 4, we propose a new corner detector based on GMC curvature estimation. We then present a comparative evaluation with several other corner detectors, and eventually show its potential for multi-scale contour representation.

## 2. Discrete Curvature Estimators

In this section, we recall briefly the definitions of five curvature estimators, which have all been recently published and which are all able to analyze digital and damaged shape data. For the sake of completeness, we begin by describing the notion of digital blurred segments, which is used by the two first estimators.

Blurred segments were introduced by Debled-Rennesson *et al.*<sup>7</sup>. Note that a comparable concept and algorithm was proposed independently by Buzer<sup>2</sup>. A *blurred segment* is a piece of thick digital straight line, whose *width* is the vertical diameter of its convex hull. From this definition the authors proposed an algorithm for the recognition of blurred segments of width  $\nu$ . Note that the two following curvature estimators use a variant of the original definition of blurred segment, which is not restricted to the hypothesis that points are added with increasing  $x$  coordinate (or  $y$  coordinate). Blurred segments are illustrated in Fig. 2 and Fig. 3(b).

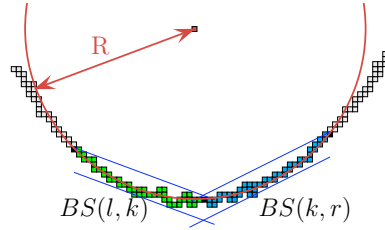
4 *B. Kerautret, J.-O Lachaud, B. Naegel*


Fig. 2. Illustration of curvature estimation using NDC estimator. The blue lines represent the two blurred segments  $BS(l, k)$  and  $BS(k, r)$  of width 5.

### 2.1. Estimator based on osculating circles (CC and NDC)

The curvature estimator proposed by Nguyen and Debled-Rennesson <sup>7</sup> follows the same concept as the estimator proposed by Coeurjolly *et al.* <sup>6</sup> (called CC estimator). The latter is based on the estimation of osculating circles. More precisely, by denoting  $C$  a discrete curve,  $C_{i,j}$  the sequence of points going from  $C_i$  to  $C_j$  and  $BS_\nu(i, j)$  the predicate “ $C_{i,j}$  is a blurred segment of width  $\nu$ ”. They consider the points  $C_r$  and  $C_l$  defined such that:  $BS_\nu(l, k) \wedge \neg BS_\nu(l-1, k) \wedge BS_\nu(k, r) \wedge \neg BS_\nu(k, r+1)$ . With this definition, they determine the curvature of width  $\nu$  at the point  $C_k$  from the radius of the circle passing through  $C_k$ ,  $C_l$  and  $C_r$ . By noting  $s_1 = \|\overrightarrow{C_k C_r}\|$ ,  $s_2 = \|\overrightarrow{C_k C_l}\|$  and  $s_3 = \|\overrightarrow{C_l C_r}\|$ , the authors give the following expression of radius  $R_\nu$  of the circle associated to the point  $C_k$ :

$$R_\nu(C_k) = \frac{s_1 s_2 s_3}{\sqrt{(s_1 + s_2 + s_3)(s_1 - s_2 + s_3)(s_1 + s_2 - s_3)(s_2 + s_3 - s_1)}}$$

If the vectors  $\overrightarrow{C_k C_r}$  and  $\overrightarrow{C_k C_l}$  are not collinear then the curvature of width  $\nu$  can be determined with  $\frac{\text{sign}(\det(\overrightarrow{C_k C_r}, \overrightarrow{C_k C_l}))}{R_\nu(C_k)}$ . Otherwise the curvature is set to 0.

### 2.2. Global Minimization Curvature estimator (GMC)

The estimator proposed by Kerautret and Lachaud is based on two main ideas <sup>13</sup>. The first one is to take into account all the possible shapes that have the same digitization defined by the initial contour and to select the most probable contour. This selection is done with a constrained global optimization which minimizes the squared curvature among admissible shapes. From this idea, we can expect a precise result even with a low contour resolution. The second idea is to obtain an estimator that is not sensitive to noise or to non perfect digitization process by determining the constraints with blurred segments.

For the purpose of minimizing curvature, the tangential cover <sup>8</sup> is computed on the discrete contour and the minimal and maximal possible tangent values are defined for each pixel based on the directions of surrounding maximal segments. Fig. 3(a) illustrates tangential cover and the bounds on the tangent directions is shown in Fig. 4(a). In the space of tangent directions (as in Fig. 4(b)), the shape

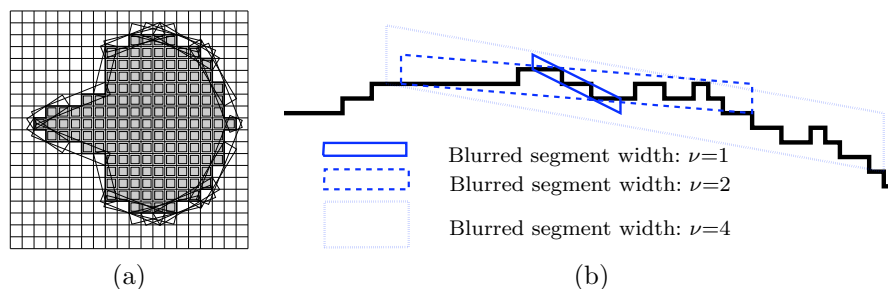


Fig. 3. Illustration of tangential cover (a) and blurred segments with different widths (b).

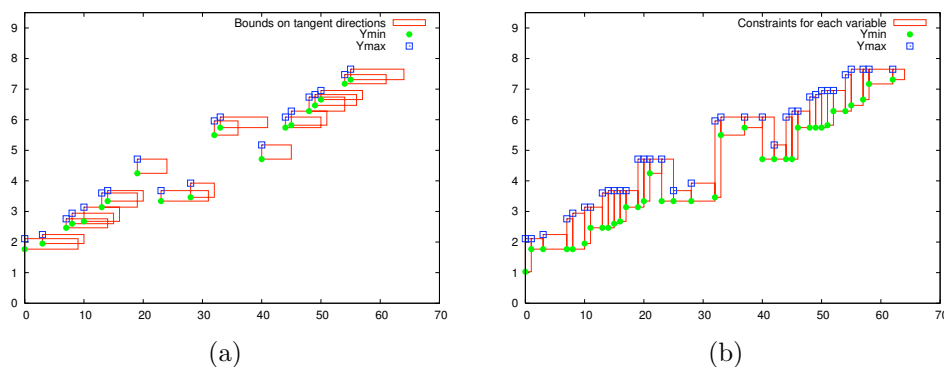


Fig. 4. Illustration of bounds defined from maximal segments (a) (extracted from Fig. 3(a)). The plot (b) shows the constraints defined on each variable.

geometry is represented by a piecewise linear curve which must stay within the constraint boxes. Minimizing the curvature consists in moving each point  $(x_i, y_i)$  of this curve along the  $y$  axis and between the bounds defined by the boxes, such that  $(x_i, y_i)$  is as close as possible to the line joining  $(x_{i-1}, y_{i-1})$  to  $(x_{i+1}, y_{i+1})$ . The global minimization method is a standard relaxation process. The estimated curvature is then simply the slope of this curve at each point.

For the robustness to noise and non perfect digitization process, the discrete maximal segments were replaced by the discrete maximal blurred segments previously described. Note that the definition of the minimal and maximal slopes of the blurred segments is notably different from the non blurred case (see <sup>13</sup> for more details).

### 2.3. Binomial Convolution Curvature Estimator (BCC)

Malgouyres *et al.* proposed an algorithm to estimate derivatives with binomial convolutions <sup>15</sup> which is claimed to be multigrid convergent. They define the operator  $\Psi_K$  which modifies the function  $F : \mathbb{Z} \rightarrow \mathbb{Z}$  by the convolution with kernel

6 *B. Kerautret, J.-O Lachaud, B. Naegel*

$K : \mathbb{Z} \rightarrow \mathbb{Z}$ . For instance the backward finite difference is given by  $\Psi_\delta F$  where the kernel  $\delta$  is defined by:

$$\delta(a) = \begin{cases} 1 & \text{if } a = 0 \\ -1 & \text{if } a = 1 \\ 0 & \text{otherwise} \end{cases}$$

Then the authors give the smoothing kernel defined as:

$$H_n(a) = \begin{cases} \binom{n}{a + \frac{n}{2}} & \text{if } n \text{ is even and } a \in \{-\frac{n}{2}, \dots, \frac{n}{2}\} \\ \binom{n}{a + \frac{n+1}{2}} & \text{if } n \text{ is odd and } a \in \{-\frac{n+1}{2}, \dots, \frac{n-1}{2}\} \\ 0 & \text{otherwise.} \end{cases}$$

Finally the derivative kernel  $D_n$  is given by  $D_n = \delta * H_n$  and the derivative estimator is  $\frac{1}{2^n} \Psi_{D_n} F$ . An interesting point of the method is that higher order derivatives have similar expressions:

$$D_n^2 = \delta * \delta * D_n$$

And for the curvature we compute:

$$\frac{D_n^2(x) * D_n(y) - D_n^2(y) * D_n(x)}{D_n(x)^2 + D_n(y)^2}$$

The value of  $n$  is defined by  $n = \lfloor h^{2(\alpha-3)/3} \rfloor$ , where  $h$  represents the grid size and  $\alpha \in ]0, 1]$  is an additional parameter which can be associated to the amount of noise. More precisely, a value close to 0 allows more noise than a value close to 1.

#### 2.4. *Visual Curvature (VC)*

Liu *et al.*<sup>14</sup> have recently proposed a definition of curvature which is suited to regular, polygonal, and digital planar curves. Robustness to noise is achieved through a scale parameter. The pseudo-curvature estimator is then thresholded to detect corners. As we will show on experiments, their corner detector seem to be the most reliable at the moment.

A number  $N$  of directions is given by the user (the authors suggest 128): the bigger is  $N$ , the more precise can be the estimator. The contour is seen as a parametric curve  $(x(s), y(s))$ . Then  $N$  different height functions  $H_{\alpha_i}$  are generated by rotating the parametric curve  $(x(s), y(s))$  along  $N$  directions between 0 and  $\pi$ . The profile along the  $y$ -axis gives an height function  $H_{\alpha_i}$  for each direction. For a given neighborhood  $S(v)$  of size  $\Delta S$  around the point of curvilinear abscissa  $v$ , the *visual curvature* at  $v$  is defined as

$$K_{N, \Delta S}(v) = \pi \frac{\sum_{i=0}^{N-1} \#[H_{\alpha_i}(S(v))]}{N \Delta S},$$

where  $\#[H_{\alpha_i}(S(v))]$  represents the number of local extreme points of the height function  $H_{\alpha_i}$  in the neighborhood  $S(v)$ .

This definition of curvature was shown to tend toward the standard absolute curvature for regular curves, when  $N$  tends toward infinity and  $\Delta S$  toward 0. Unfortunately, no convergence speed is given. Furthermore, this definition is useless both for digital contours and for noisy shapes, since they have plenty of extreme points (the steps) in every direction.

The authors therefore add a *scale measure* to any extreme point  $v$  along a direction: it is the minimum height of the two peaks surrounding the point  $v$  divided by the diameter of the shape along this direction. A peak is the maximal height variation of the function before it reaches again the same height as  $v$ . A strong extreme point has then a scale measure close to 1 (such as any point on the convex hull) while weak extreme points have low scale measure (close to 0). The *multiscale visual curvature* of scale  $\lambda$  is then

$$K_{N,\Delta S}^\lambda(v) = \pi \frac{\sum_{i=0}^{N-1} \#[H_{\alpha_i}^\lambda(S(v))]}{N\Delta S},$$

where  $\#[H_{\alpha_i}^\lambda(S(v))]$  represents the number of local extreme points of the height function  $H_{\alpha_i}$  in the neighborhood  $S(v)$ , such that their scale measure is no smaller than  $\lambda$ .

This definition is more suited to digital or noisy contours, where small extreme points can be removed by the scale parameter. Although the given value is a qualitative estimation of curvature, it is not a correct nor precise quantitative estimator of curvature. In order to better detect and localize corners, Liu *et al.* have further proposed to cut the input contour into pieces of  $T$  points, and then to gather all the visual curvature of each piece onto its point with maximal visual curvature. This allows that several significant curvature points, which are close to each other, are considered as one corner. This latter definition is called the *digital visual curvature*.

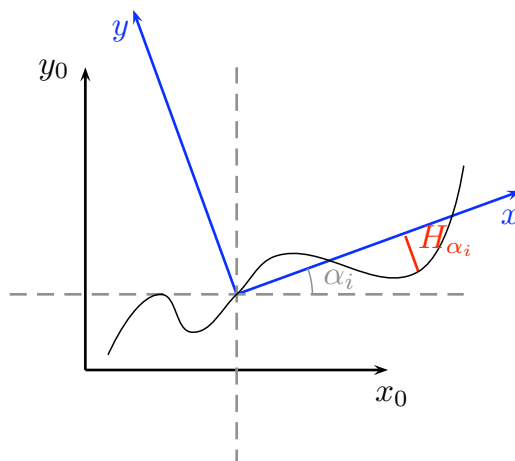
To sum up, the visual curvature requires four parameters: the number  $N$  of directions, the size  $\Delta S$  of the neighborhood, the scale  $\lambda$ , the number of points  $T$  for gathering curvature.

### 3. Experimental Comparisons

In this section, we first compare experimentally the three estimators GMC, CC-NDC and BCC, both on perfectly digitized data and on noisy shapes. Since the notion of visual curvature does not represent rigorously the same type of information and is not directly suited to digital contours, we present it in the last part of this section and we give only a comparative evaluation on noisy contours.

The objective is to measure the accuracy of the previously described estimators and to give execution times. For this purpose several data sets were generated with different grid sizes ( $h$ ), for coarse ( $h = 1$ ), medium ( $h = 0.1$ ), and fine resolution ( $h = 0.01$ ). Fig. 6 illustrates the two shapes defined with a digitization grid step equal to 1 (image “flower” (a) and image “polygon” (b)). We choose these test shapes to analyze the performances of the estimators both on smooth shapes with varying curvature and quick turns, and on polygonal shapes with corners.



8 *B. Kerautret, J.-O Lachaud, B. Naegel*Fig. 5. Illustration of function  $H_{\alpha_i}$ 

The three estimators have been applied on the previous data. For the BCC estimator, the parameter  $\alpha$  was set to 1 and the value of  $n$  was thus set to  $h^{-4/3}$ . For the other estimators, no parameters were used. From the resulting curvature graphs of Fig. 6, it can be seen that, for both the flower and the polygon, the BCC estimator shows numerous oscillations with coarse grid sizes ( $h=1$  and  $h=0.1$ ). But the finer the grid size is, the more the values of curvature of the BCC estimator are stable. It appears that it is not the case with the CC estimators. Indeed the oscillations appear to increase as the grid size gets finer. For the GMC estimator we can see that it is very stable since no oscillations are visible. Other error measures and execution timings were performed on these shapes and they may be found in a previous work <sup>12</sup>.

In order to measure the stability of the curvature estimators on noisy shapes, noise was added according to the model proposed by Kanungo <sup>11</sup>. Fig. 7 shows such noise addition (images (a) and (b)). From these results, we can see that the GMC estimator is very stable compared to the NDC estimator. Even with width 2 the results are stable if we ignore a small part with negative curvature (image (f)). In the same way, the results of the BCC estimators appear satisfying although a large size  $n$  was needed to obtain stable values. The fact that there are no oscillation in the case of the GMC constitutes a real advantage to design a simple and efficient curvature based-algorithm.

Curvature estimations with the multiscale visual curvature estimator were performed on the noisy versions of the flower and polygon shapes (see Fig. 8) at two different scales ( $\lambda = 0.5$  and  $\lambda = 0.05$ ). The result on the flower shape shows clearly that the visual curvature cannot be used to extract accurate measures on the parts outside corners. It appears that even near corners and even with larger scale val-

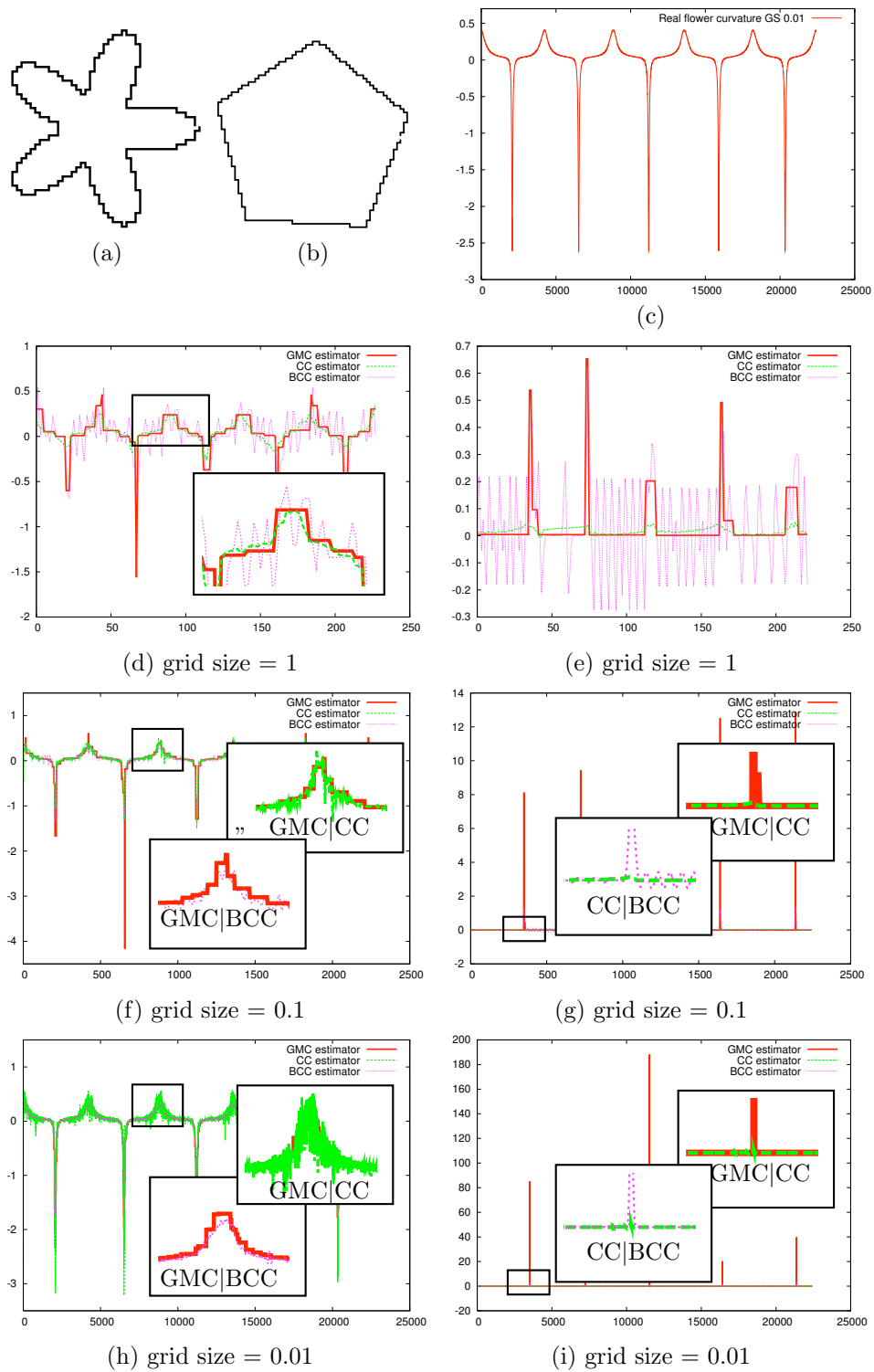


Fig. 6. Comparisons of the CC, GMC and BCC estimators on the flower (a,d,f,h) and on the polygon (b,e,g,i). The graph (c) shows the real curvature of the flower shape.

10 *B. Kerautret, J.-O Lachaud, B. Naegel*

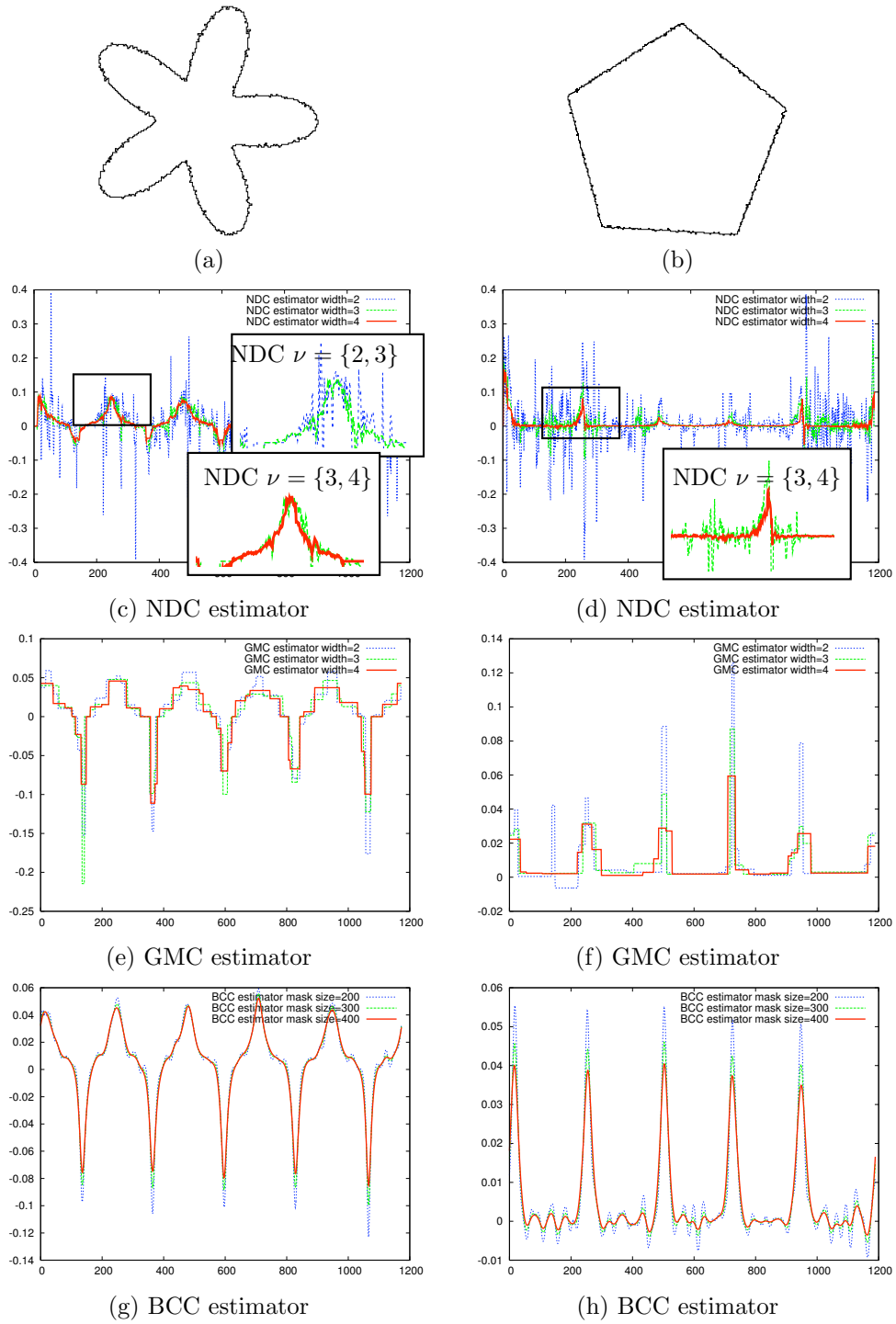


Fig. 7. Curvature estimations on the noisy version of the shapes of Fig.6. Each row shows the result obtained for each estimator with different parameter values related to the amount of noise (width  $\nu$  for GMC and NDC and *mask\_size* for the BCC estimator) .

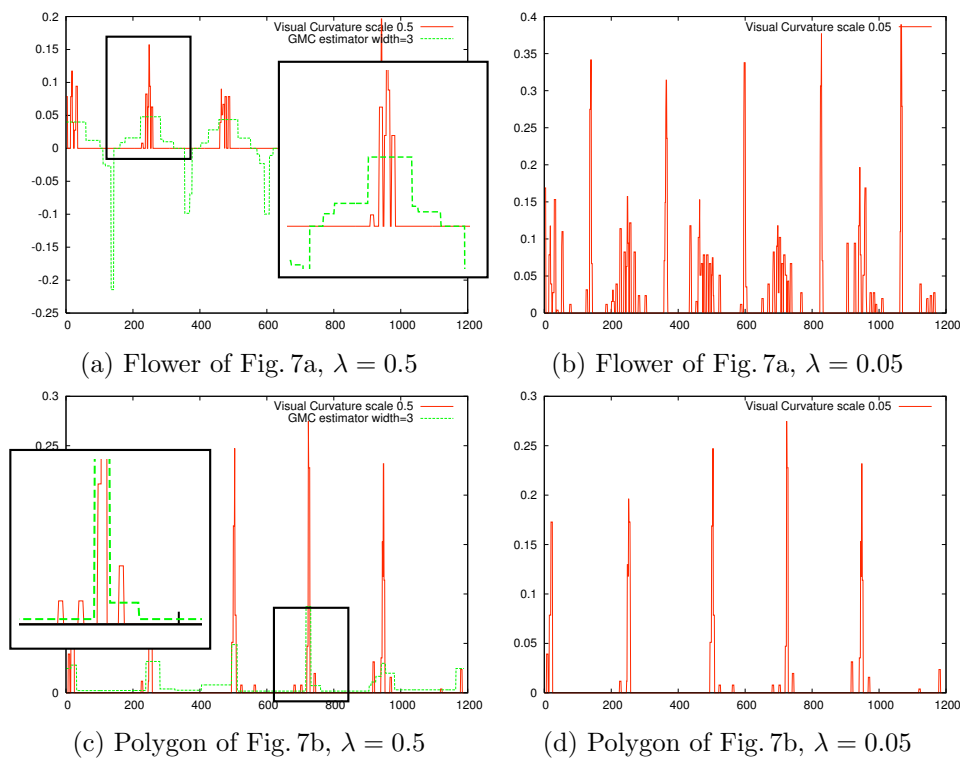


Fig. 8. Illustration of the visual curvature accuracy and comparison with GMC estimator.

ues, there exists some oscillations which are not present with the GMC estimator (Fig. 8).

In the following, we exploit the stability of the GMC estimator in order to implement a robust and fast corner detector.

#### 4. Application to Corner Detection

We introduce an original algorithm to detect corner points, which is based on the GMC estimator introduced in Section 2. In the sequel we refer to this algorithm as GMCCD. This algorithm requires a parameter  $\nu$ , which is the estimated thickness of the noise along the shape boundary, expressed in pixels. The main steps of the algorithm for the detection of convex corner points are:

- (1) for all contour points  $(p_i)_{i \in I}$  compute curvature  $\kappa(p_i)$  with the GMC estimator, setting the width parameter to  $\nu$ .

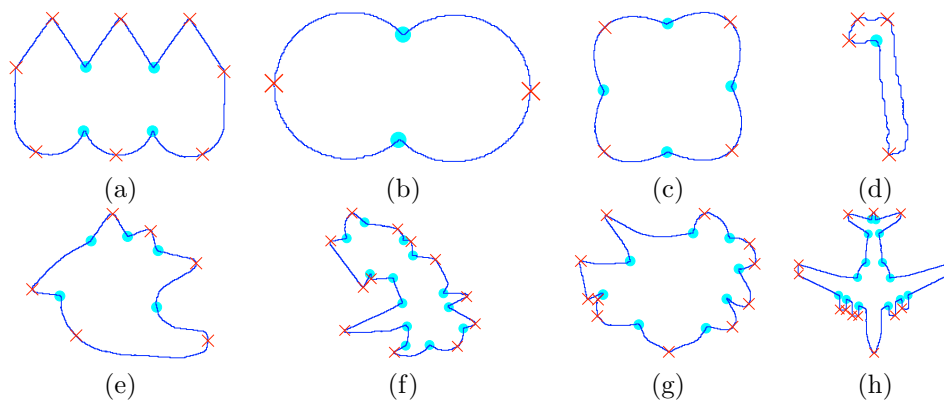
12 *B. Kerautret, J.-O Lachaud, B. Naegel*


Fig. 9. Results of corner detection using GMCCD (with parameter  $\nu = 2$ ). Convex and concave corners are respectively represented by crosses and disks.

(2) detect all the maximal curvature regions defined by sets of consecutive points:

$$R_k = \{(p_i)_{i \in [a,b]} \mid \forall i, \kappa(p_i) = \kappa(p_a) \wedge \kappa(p_{a-1}) < \kappa(p_a) \wedge \kappa(p_{b+1}) < \kappa(p_b) \wedge \kappa(p_a) > 0\}$$

(3) for each region  $R_k$  mark the point  $p_{(a+b)/2}$  as a corner.

(4) (optional) select only corner points with curvature greater than a threshold value  $\kappa_{min}$ .

Note that a quantification of the curvature field was applied in order to simplify the detection of the local minima/maxima (set to  $1e - 3$ ). In the rest of this paper we have not applied the optional step (4) since it adds a new parameter and it could be fixed for each particular application. The concave corner detection is deduced by replacing the maximal by the minimal curvature regions.

The previous algorithm was applied on the standard image collection available on-line<sup>5</sup> (see Fig. 9). All the corner detections were performed using the GMCCD algorithm with parameter  $\nu = 2$ , which is a commonly used value for real data presenting a weak amount of noise. We can see on Fig. 9 that some parts, which are considered as corners, are more local maxima. Note that these points could be easily removed by applying a small threshold on the curvature values of the local maxima. For this standard dataset our results are comparable to previous works, but we can already note that only one parameter is required and it remains unchanged for all data considered as non noisy. This is not the case for example for the corner detectors described below.

The images of Fig. 10 were generated from the previous test images and with the same noise model as mentioned in the previous section. The parameter  $\nu$  was set to 4 according to the noise level. Note also that we are currently working on an automated determination of this parameter. Despite the important amount of

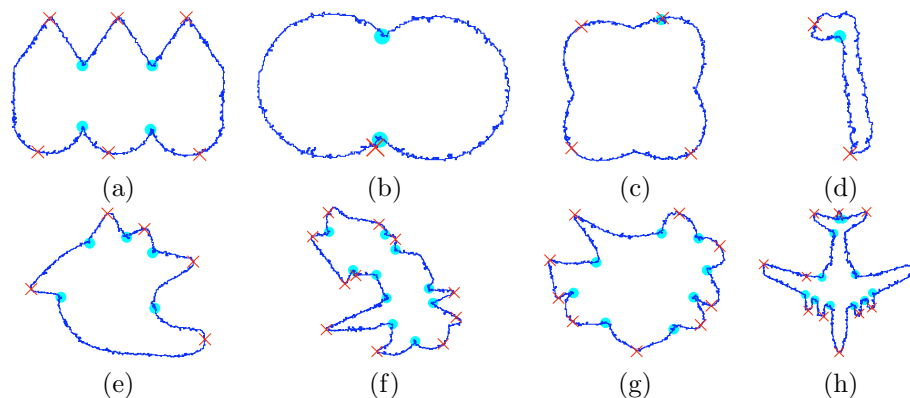


Fig. 10. Results obtained on noisy version of the original image database. The parameter  $\nu$  for the GMCCD algorithm was set to 4 for all images.

noise, the corners were well detected. The only defect can be seen on the left wing of the plane (h), but it disappears when using a width equals to 5. Even with a lot of noise, corners can be detected as illustrated on Fig. 20d ( $\nu = 25$ ). Note that the total execution time was approximately 3 s for the test images of Fig. 9 and Fig. 10.

#### 4.1. Comparison with other corner detectors

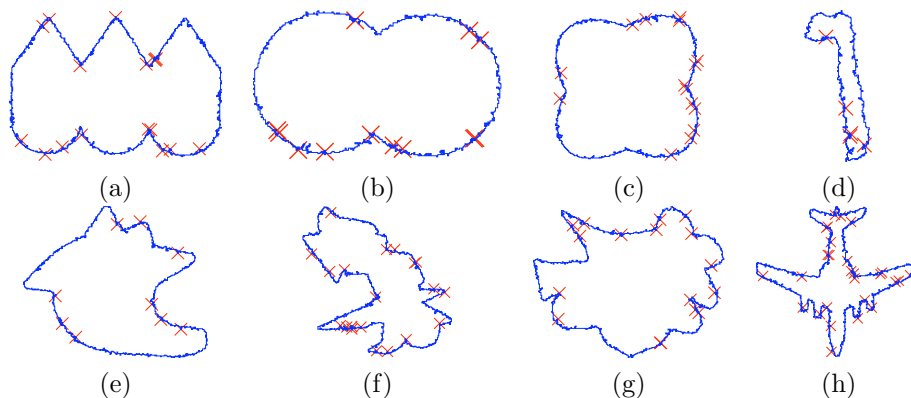
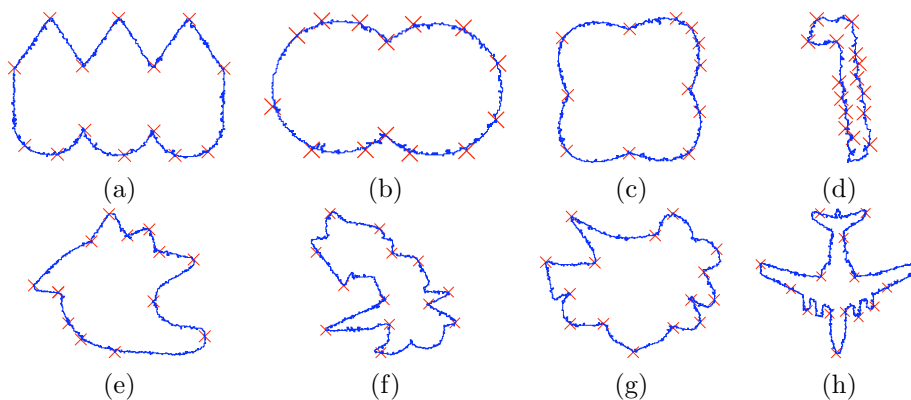
We have compared our method with other corner detectors. Moreover, in order to assess the versatility of our corner detector, experiments were conducted on noisy versions of the classical reference shapes<sup>5</sup>.

In a first experiment, our method was compared with the five methods implemented on the internet<sup>5</sup> (RJ73<sup>17</sup>, RW75<sup>18</sup>, FD77<sup>9</sup>, BT87<sup>1</sup>, IPAN03<sup>4</sup>). A set of parameters, left constant for all the 8 images, was chosen for each of the five methods in order to obtain the best possible results. The results as well as the parameters used in the experiments are described respectively in Fig. 11, Fig. 12, Fig. 13, Fig. 14, and Fig. 15.

The IPAN03, FD77 and BT87 methods are very sensitive to noise and therefore give poor and instable results (see Fig. 11, Fig. 14, Fig. 15). A lot of true corners are missing while false detections are frequent.

The RJ73 and RW75 methods give the best results and appear to be rather robust to noise. Using parameter  $\kappa = 0.05$ , RJ73 enable to detect all true corners, while some false corners are detected. The RW75 method with the parameter  $\kappa = 0.05$  give the best result, with only a few false detections (see Fig. 13b) and a few corners missing (see Fig. 13 (c),(f),(g),(h)). However the RW75 method does not achieve the degree of robustness of the GMCCD method.

In a second experiment, the corner detector based on the visual curvature<sup>14</sup> (which we will refer to as VCCD) was experimented. Similarly to the GMCCD algorithm, a point is detected as corner each time the curvature exceeds a particular

14 *B. Kerautret, J.-O Lachaud, B. Naegel*

 Fig. 11. Result of IPAN (IPAN03) using parameters:  $d_{min} = 7, \alpha_{max} = 30$ .

 Fig. 12. Result of Rosenfeld and Johnston (RJ73) using parameter:  $\kappa = 0.05$ .

threshold  $DK_0$ . In this experiment we have generated a large number of tests in order to obtain the best results. We have used the following list of parameters:  $\Delta S = \{2, 4, 8, 10, 15\}$ ,  $\lambda = \{0.05, 0.1, 0.2, 0.3, 0.4\}$ ,  $T = \{10, 20, 30, 50\}$  with  $N = 300$  and applied a list of threshold  $DK_0$  from 0.3 to 2 by step of 0.2 (1300 tests for each shape). Fig. 16 and Fig. 17 show two sets of results obtained with the same set of parameters for all the test shapes. We can see that the corners are well detected but on the round parts of the shape we can detect some wrong corners (on shapes (b) and (c)) and some corners are often missing (shapes (g) and (c)).

Other comparisons of GMCCD with a recent morphological corner detector <sup>3</sup> can be found in previous work <sup>12</sup>, but the results were not interesting on noisy shapes compared to GMCCD and VCCD.

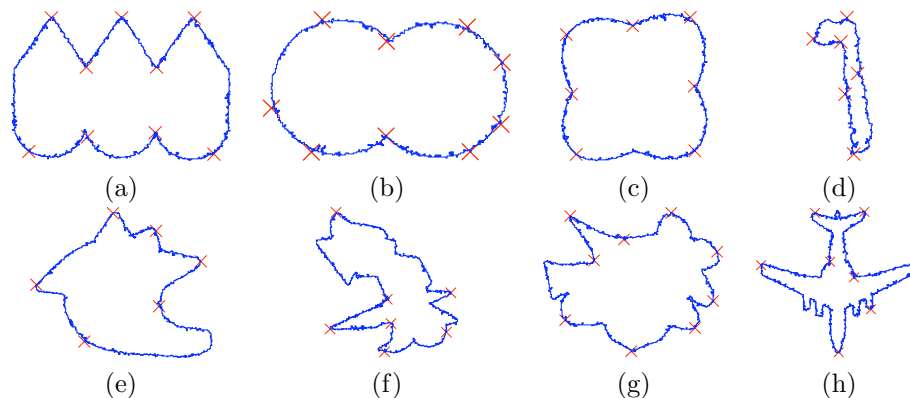


Fig. 13. Result of Rosenfeld and Weszka (RW75) using parameter:  $\kappa = 0.10$ .

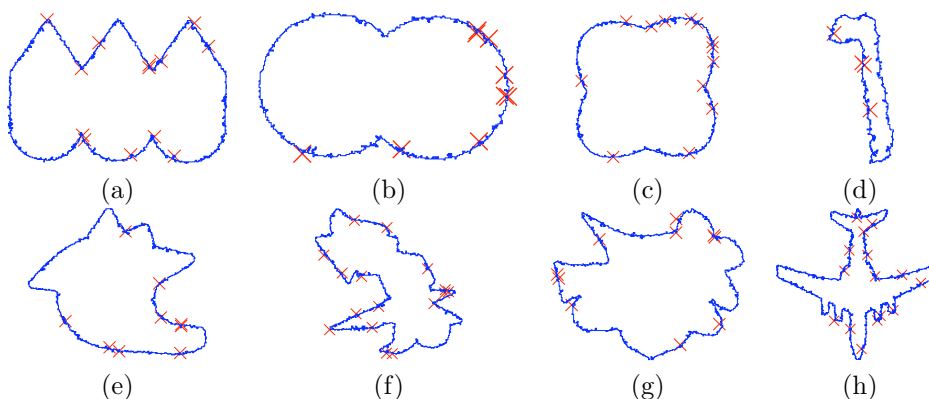


Fig. 14. Result of Freeman and Davis (FD77) using parameters:  $k = 5, S = 1000$ .

#### 4.2. Multiscale comparisons between GMC and VC curvature estimators

Since the two curvature estimators GMC and VC define a multi-scale analysis of the contour, we can compare the contour simplifications obtained from these two estimators. We propose a simple way to simplify the contour by using GMC curvature estimated with a width  $\nu$  and by using only the frontier points associated to the constant curvature areas. Such a contour is denoted by  $G_\nu$ . Figure 18 illustrates the results obtained with our approach. In the same way Liu *et al.* proposed a definition of  $\lambda$ -scale approximation of a contour by removing all points for which curvature vanishes (denoted by  $V_\lambda$ ). This representation is illustrated with different values of  $\lambda$  on Fig. 19. We can remark that the number of points selected by GMC is smaller and more representative than the ones selected by the visual curvature.

To conclude this part on experimentations and comparisons, we applied the two



16 *B. Kerautret, J.-O Lachaud, B. Naegel*

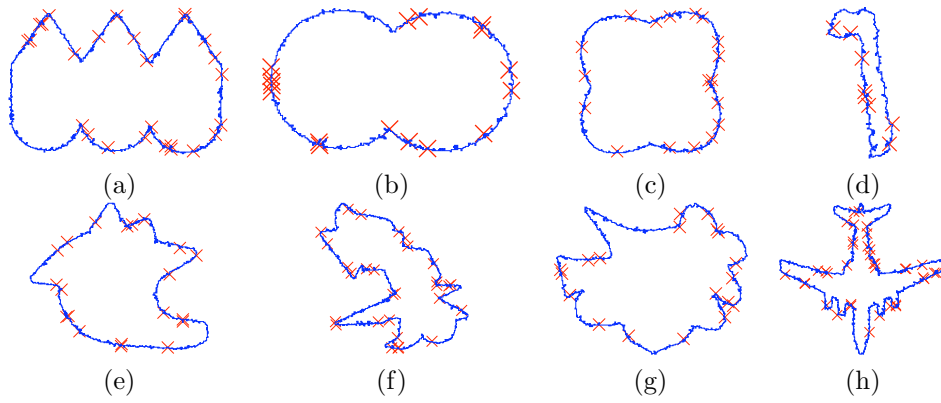


Fig. 15. Result of Beus and Tei (BT87) using parameters:  $k_1 = 3, k_2 = 7, \tau = 0.05, S = 600$ .

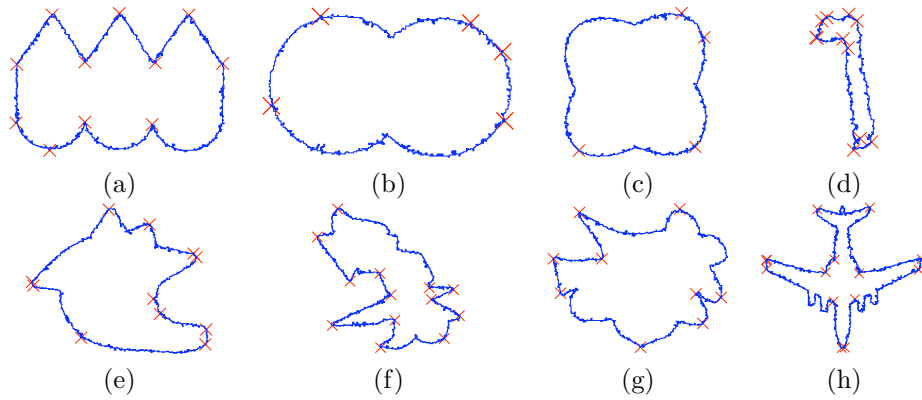


Fig. 16. Results obtained with the visual curvature (VC) with the following parameters:  $\Delta S = 2, \lambda = 0.1, T = 30, N = 300, DK_0 = 0.6$

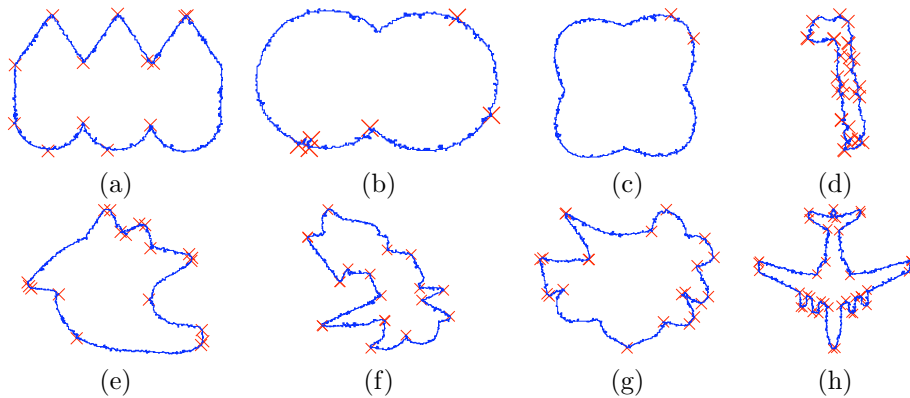


Fig. 17. Results obtained with the visual curvature (VC) with the following parameters:  $\Delta S = 2, \lambda = 0.05, T = 10, N = 300, DK_0 = 0.6$

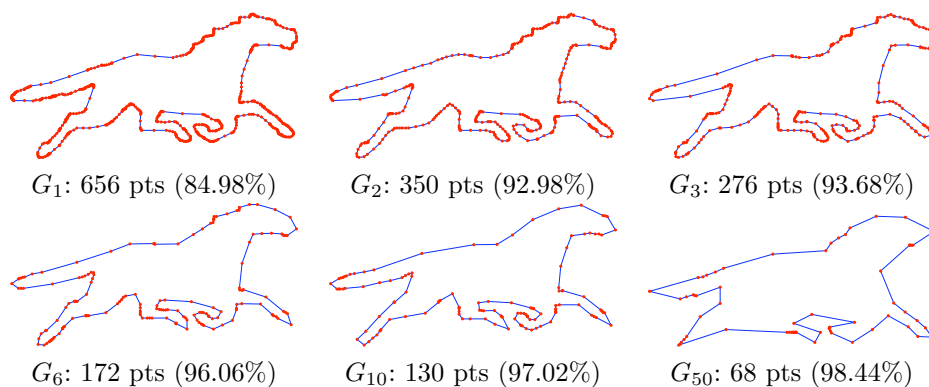


Fig. 18. Multiscale shape representation by using GMC and all changing curvature points.

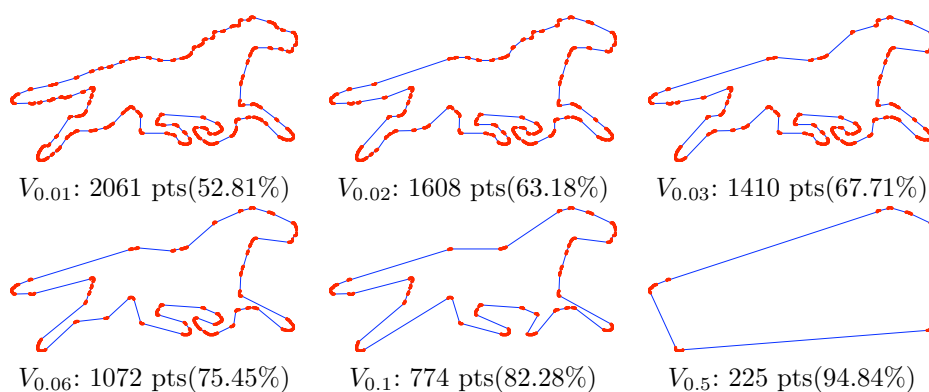


Fig. 19. Shape simplification based on Visual Curvature.

detectors GMCCD and VCCD on the shapes of figure Fig. 1 (see Fig. 20). The corner detection was successful with the GMCCD on all damaged data: coarse resolution, weak pepper and salt noise, and multiscale interpretation. For the VCCD only convex corners were detected and with the noisy shape (f) there appear some false corners on the large round part of the shape. Increasing the scale to  $\lambda = 0.3$  does not remove them. For the execution times, GMCCD appears most of the time faster if we exclude the result with large scale (d,h). Note that the parameter  $N$  was set to 128 as recommended by the authors of VCCD.

## 5. Conclusion

In this paper we have compared several recent discrete curvature estimators suited to the analysis of noisy digital data. From these experimentations the GMC estimator appears the most stable and was selected to define a new curvature-based corner detector. The results obtained demonstrate a great robustness to noise and

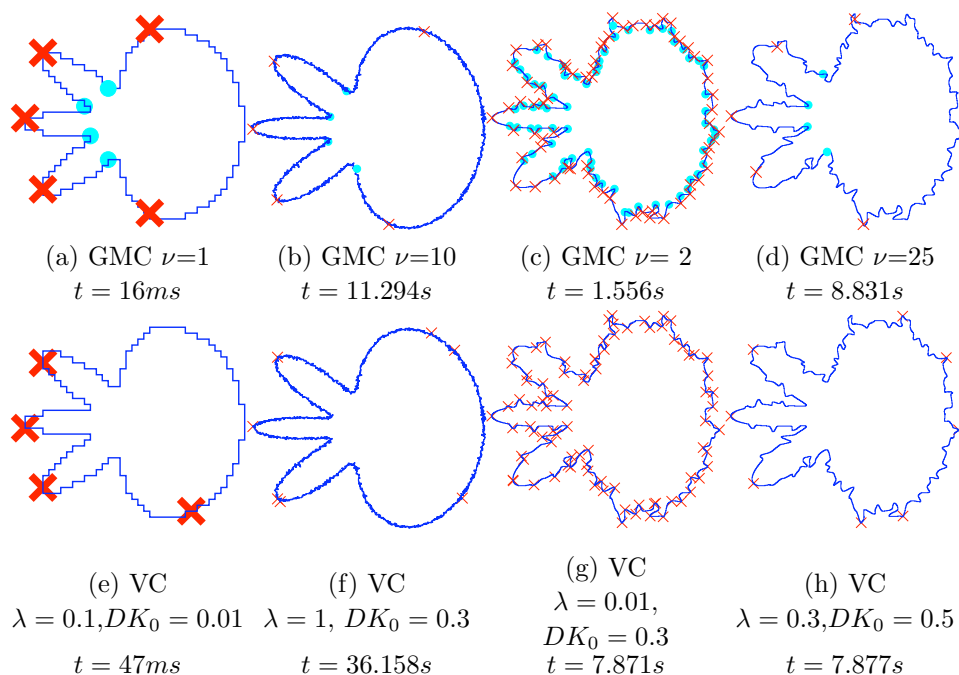
18 *B. Kerautret, J.-O Lachaud, B. Naegel*


Fig. 20. Comparison between GMC and VC based corner detectors on different damaged versions of the shape of Fig. 1. The others parameters for VC,  $\Delta S$ ,  $T$ , and  $N$ , were respectively set to 4, 50 and 128.

they outperform other classical and recent corner detectors. Furthermore our new estimator requires only one parameter to set, which has a simple geometric interpretation related to the noise level along the contour and the scale for which corners are considered.

### Acknowledgments

A preliminary version of this work was presented at the International Symposium on Visual Computing, Las Vegas, NV, 2008, Special Track on Discrete and Computational Geometry and their Applications in Visual Computing<sup>12</sup>.

### References

1. H.L. Beus and S.S.H. Tiu. An improved corner detection algorithm based on chain-coded plane curves. *Pattern Recognition*, 20(3):291 – 296, 1987.
2. L. Buzer. An elementary algorithm for digital line recognition in the general case. In *Proc. of Int. Conf. DGCI*, volume 3429 of *LNCS*, pages 299–310. Springer, 2005.
3. X. Chang, L. Gao, and Y. Li. Corner detection based on morphological disk element. In *Proceedings of the 2007 American Control Conference*, pages 1994–1999. IEEE, 2007.

4. D. Chetverikov. A simple and efficient algorithm for detection of high curvature points in planar curves. In *CAIP'03*, pages 746–753, 2003.
5. D. Chetverikov and Z. Szabo. [http://visual.ipan.sztaki.hu/corner/corner\\\_click.html](http://visual.ipan.sztaki.hu/corner/corner\_click.html). Online, 1999.
6. D. Coeurjolly, S. Miguet, and L. Tougne. Discrete curvature based on osculating circle estimation. In *Proc. Int. workshop Visual Form*, volume 2059 of *LNCS*, pages 303–302. Springer, 2001.
7. I. Debled-Rennesson, F. Feschet, and J Rouyer-Degli. Optimal blurred segments decomposition of noisy shapes in linear times. *Computers and Graphics*, 2006.
8. F. Feschet and L. Tougne. Optimal time computation of the tangent of a discrete curve: Application to the curvature. In *Proc. of the Int Conf DGCI*, volume 1568, pages 31–40. springer, 1999.
9. H. Freeman and L.S. Davis. A corner-finding algorithm for chain-coded curves. *IEEE Transactions on Computers*, C-26(3):297–303, March 1977.
10. S. Hermann and R. Klette. A comparative study on 2d curvature estimators. In *International conference on computing: Theory and applications*, 2007.
11. T. Kanungo. *Document Degradation Models and a Methodology for Degradation Model Validation*. PhD thesis, University of Washington, 1996.
12. B. Kerautret, J.-O Lachaud, and B. Naegel. Comparison of discrete curvature estimators and application to corner detection. In G. Bebis et al., editor, *Proc. 4th International Symposium on Visual Computing (ISVC 08)*, volume 5358 of *LNCS*, pages 710–719. Springer, 2008.
13. B. Kerautret and J.O. Lachaud. Robust estimation of curvature along digital contours with global optimization. In *Proc. of Int Conf DGCI*, volume 4992 of *LNCS*, pages 334–345. Springer, April 2008.
14. H. Liu, L. J. Latecki, and W. Liu. A unified curvature definition for regular, polygonal, and digital planar curves. *Int. J. Comput. Vision*, 80(1):104–124, 2008.
15. R. Malgouyres, F. Brunet, and S. Fourey. Binomial convolutions and derivatives estimations from noisy discretizations. In *Proceedings of the Int Conf on DGCI*, volume 4992 of *LNCS*, pages 370–379. Springer, April 2008.
16. T.P. Nguyen and I. Debled-Rennesson. Curvature estimation in noisy curves. In *CAIP*, volume 4673 of *LNCS*, pages 474–481. Springer, 2007.
17. A. Rosenfeld and E. Johnston. Angle detection on digital curves. *IEEE Transactions on Computers*, C-22(9):940–941, Sept. 1973.
18. A. Rosenfeld and J.S. Weszka. An improved method of angle detection on digital curves. *Computers, IEEE Transactions on*, C-24(9):940–941, Sept. 1975.
19. S. Utcke. Error-bounds on curvature estimation. In *Proc. of Scale Space Methods in Computer Vision (SSMC'2003)*, volume 2695 of *LNCS*, pages 1080–1089. Springer, 2003.