

Quadrilateral Detection Using Genetic Algorithms

Victor Ayala Ramirez, Sergio A. Mota Gutierrez, and Raul E. Sánchez Yanez

Universidad de Guanajuato, División de Ingenierías Campus Irapuato-Salamanca,
Carr. Salamanca-Valle de Santiago Km. 3.5+1.8, Comunidad Palo Blanco, 36700,
Salamanca, Mexico

{ayalav,sanchezy}@ugto.mx, samota@laviria.org

Abstract. An approach based on the use of genetic algorithms to detect quadrilateral shapes in images is presented in this paper. The proposed approach finds the best sets of four edge points that are the vertices of quadrilateral shapes in the image. The proposed method uses the evidence provided by the image resulting of the application of an edge detection operator to the input image. Individuals having the best fitness scores are those that are supported by the edge evidence as being the vertices of a quadrilateral present in the input image. We use a sharing operator to avoid detecting similar quadrilaterals. This procedure is used to detect multiple quadrilaterals in a single run of our algorithm. Our method can handle perspective distortion and Gaussian noise corruption on the quadrilaterals to be detected. We have fulfilled tests to validate our approach on synthetic, noise-corrupted and real world images. Tests are both quantitative and qualitative. The proposed approach has shown also to be fast for real-time quadrilateral detection.

Keywords. Genetic algorithms, quadrilateral detection, shape recognition.

Detección de cuadriláteros usando algoritmos genéticos

Resumen. En este artículo se presenta un enfoque para la detección de formas cuadriláteras en imágenes usando algoritmos genéticos. El enfoque propuesto encuentra los mejores conjuntos de cuatro puntos de borde que son vértices de cuadriláteros presentes en la imagen. El método propuesto usa la evidencia proporcionada por la imagen resultante de la aplicación de un operador de detección de bordes a la imagen de entrada. Los individuos con mejor valor de adecuación son aquellos que representan a los vértices de cuadriláteros presentes en la imagen. A fin de evitar la detección de cuadriláteros similares entre sí, se usa una función de sharing. Esto permite detectar múltiples cuadriláteros en una sola ejecución del algoritmo. Nuestro método puede manejar la presencia de distorsión perspectiva y de ruido Gaussiano aditivo en

los cuadriláteros por ser detectados. Se presentan pruebas para validar nuestro enfoque sobre imágenes sintéticas, imágenes corrompidas por ruido e imágenes reales. Las pruebas son tanto cuantitativas como cualitativas e incluyen también la detección de cuadriláteros en imágenes dibujadas a mano. El enfoque propuesto muestra también ser rápido para la detección de cuadriláteros.

Palabras clave. Algoritmos genéticos, detección de cuadriláteros, reconocimiento de formas.

1 Introduction

Shape recognition is a fundamental issue in visual pattern recognition systems. We need to search for objects or regions of interest in images in order to extract high-level information from the scene under analysis. A number of methods have been proposed using different approaches for shape description and matching. Recent literature on shape matching shows probabilistic approaches (for example, [26]); categorical shape descriptions (as in the works by [20] and [20]); size functions (for example, in the work of [5]); distance functions for curves (similar to the approach in [14]) and appearance-based templates (where [16] is an example). Some other approaches include the integration of several cues in order to improve efficiency. In some cases, the integration of structural descriptions and geometrical/statistical properties is explored (for example, see [13]) and in some other works (like [17]), texture and color information is fused into geometrical shape information in order to describe object shape.

In particular, we can have a prior knowledge of specific shapes that could appear in a given scene. Then it is important to develop methods for specific shapes in order to detect them efficiently.

Specifically, we need to be capable of detecting specific shapes under noise and distortion (as in the work by [24]). Quadrilateral shapes are one of the most frequently found shapes in man-made environments. They arise in scenarios where there are doors, windows or even posters or some other artifact put on the walls. Quadrilateral shapes are also useful because of their geometric and topological properties in the discretization of volumetric and planar models for use in Computer Aided Design (CAD) software [4]. In robotic tasks, doors and windows can be used as localization landmarks because they can be associated with a known low-uncertainty position. Detecting and recognizing them will enable the robot to lower the uncertainty bounds in its self-estimated position [12, 6, 9, 10]. Another aspect of robotic navigation where a precise detection of quadrilateral shapes is useful is related to the unique identification of a place by detecting artificial landmarks throughout the robot scenario [2]. As we have said before, many of the man-made artifacts have a rectangular shape. These objects present a quadrilateral appearance when they are seen from a given perspective by a camera. We can then identify a place by combining information about the useful landmarks present on it [25]. Industrial robots also handle quadrilateral shapes in manufacturing environments [18].

Having methods to accurately detect quadrilateral position in an image will augment the complexity of the tasks this kind of systems could perform. In another application environment, autonomous vehicles are expected to share the existing infrastructure for vehicle navigation with human drivers. In such environment, quadrilateral shapes arise in the form of traffic signs [21, 8] or even in the road appearance from the camera view of a vision system mounted on the vehicle. Genetic algorithms (GAs) have been used previously in shape recognition tasks like detecting circles [1], partial polygonal shapes [15, 23], digital curves [27] and non-analytic objects [19]. They have shown to be a good alternative for geometric pattern search when a good fitness function is used for the optimization task. In particular, [20] points out noisy edge information and background clutter as the main challenges to recognize shapes in real images.

Our objective is then to detect quadrilateral shapes in an image acquired by a general purpose vision system by using a genetic algorithm. The detection task includes determining the four vertices of the quadrilateral shape in image coordinates. In order to evaluate the quantitative performance of our algorithm, we will present tests of the application of our system to synthetic images where a known ground truth is available. We have also run qualitative tests on real images to show the usefulness of the proposed approach. [15] assumes an ideal feature extractor that has already detected line features in images under test. In contrast, our method does not take this step by granted. We use a simple edge detection method, like the Prewitt operator, for example, in order to show that our algorithm is not dependent on an elaborate edge operator. We overcome the shape detection problems by considering an approach based on the implicit encoding of structural information in the edge features of an image. We encode four of the points in the edge image as the chromosomes of the individuals of a genetic algorithm population. Quadrilateral shapes are then detected by evolving the population of this genetic algorithm. The fitness function of our GA measures how well the individuals in the population represent a quadrilateral shape actually present in the image. Best candidates are favored as potential solutions of our problem. New solutions are then bred from the current best individuals until an optimization criterion is satisfied. In summary, we hypothesize that any set of four points actually present in the edge image could be the set of vertices of a quadrilateral shapes in the image. We assign them to an individual in our genetic algorithm. The fitness function evaluates then how well the above hypothesis is supported. To do this, we analyze the edge image evidence for the image under test. As fitness scores converge toward an optimum value, individuals encode the best quadrilateral shape in the input image. We use a sharing approach in order to be able to detect multiple quadrilaterals in a single run of our algorithm. As a result of this approach, background clutter will be ignored because the corresponding edge points will have bad fitness

scores except when the cluttering objects have some quadrilateral-like aspect.

Concerning noisy edge points, they will only result in a lower fitness value for the quadrilaterals to which they belong. As we can expect that they are equally distributed through all the objects in the imaged scene, the fitness landscape will not be modified. That results in the correct detection of the quadrilaterals present in the image even if they show a lower fitness score than in a corresponding noise-free image.

The rest of this paper is organized as follows. In Section 2, we present the details of the proposed approach and its implementation. The experiments, results and their analysis are presented in Section 3. We have designed a performance test that presents synthetic projective distorted quadrilaterals according to a distortion metric as input images to our algorithm. For these synthetic images, we have available ground truth data for the quadrilateral vertices. A second test degrades synthetic quadrilateral boundaries by adding a Gaussian edge location noise before presenting them to our method. We show also the results of applying our algorithm to real images with single and multiple quadrilaterals. Finally, Section 4 gives a summary of our findings and outlines future work.

2 Proposed Approach for Quadrilateral Detection

Genetic algorithms, proposed originally by [7], aim at producing a computational individual which is the best fitted to solve a specific problem. Genetic algorithms have been extensively used to solve optimization problems. The purpose of the system presented here is to detect quadrilateral shapes in an image. The system will output a set of parameters that uniquely recognize them in the image. Our proposal is to use a genetic algorithm for this purpose. We formulate the elements of this algorithm in what follows.

2.1 Individual Representation

Let us consider the edge space as the set of edge points obtained by the application of an edge operator to the original image. Our system uses

the edge space as the basis of the search space of the genetic algorithm. For any input image, an edge extraction step generates an edge vector E :

$$E = \{E_1, E_2, \dots, E_N\} \\ = \{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\} \quad (1)$$

with N being the number of edge points detected in the input image and (x_i, y_i) being the (x, y) image coordinates of the i -th edge point.

The output of a first derivative edge detection method (for example, a Prewitt edge operator) is thresholded to obtain the points to be included in the edge vector E . The use of more elaborated edge detectors is not critical for our approach. The system generates feasible solutions by choosing four vertices in the edge image of the scene for each of the individuals in the GA population. Each solution is encoded using the corresponding four vector indexes in the edge vector. So each individual I is encoded as a 4-tuple of indexes to the edge vector E according to the expression $I = \{i_1, i_2, i_3, i_4\}$ where $i_j \in \{1, \dots, N\}$ is the index to the corresponding vertex of the candidate solution.

In order to avoid individuals corresponding to degenerate quadrilaterals, we have put some additional constraints in the set of vertices to be used in candidate solutions. A colinearity check avoids triads of points resting near a single straight line. Points in the vertices set are also ordered by a relative slope procedure to guarantee convexity of the candidate shape.

2.2 Fitness Function

For each computational individual I , the GA hypothesis is the presence of a quadrilateral shape having as vertices the indexed points encoded in I . We verify the existence of the quadrilateral shape in the edge image by determining if the boundary lines of the quadrilateral shape actually exist. Let us call l_1, l_2, l_3, l_4 the four perimeter lines delimiting the quadrilateral defined by an individual I . For each line l , we test the existence of a set of M equidistant points along the line segment defined by each pair of consecutive vertices encoded in

the individual. The presence of each line is quantified by the following function:

$$R(l) = \sum_{k=0}^M \frac{1}{1 + d_k^2} \quad (2)$$

where d_k is the Manhattan distance among the actually present closer edge point and the predicted position of the edge point in the line segment. If the predicted points are found in the predicted position they contribute more to the fitness score than any other diverging from that location.

Fusion of the existence of the four lines $R(l_k)$, $k \in \{1, 2, 3, 4\}$ into the fitness score $f(l)$ of the individual l is done by using the following equation:

$$f(l) = \prod_{k=1}^4 R(l_k) \quad (3)$$

In this way, we favor individuals that represent quadrilateral shapes with greater existence evidence in the edge space. Ideal individuals will have then a fitness score of 1. Nevertheless, we can accept individuals as approximate solutions to the quadrilateral detection problem if their fitness score is near 1. We can stop the evolution process either by defining a fitness threshold according to the intended application or by setting a limit for the number of generations to be evolved. In the first case, we can also take into account problems in the input space like noise corruption or perspective distortion. In the second case, we can set a bound for the convergence time for the proposed approach in real time applications.

3 Experimental Results

In order to validate our approach, we have performed both quantitative and qualitative tests. In one hand, we have tested our method by using quadrilaterals under a perspective transformation to estimate the detection accuracy. We have also recorded the processing time on all these tests. A third test was used to evaluate how well the proposal approach handles quadrilaterals corrupted by Gaussian noise. In the other hand,

we have also performed qualitative tests to show how well the quadrilateral detection can operate both on images having multiple quadrilaterals, real images and hand-drawn images. We proceed to show our results and main findings.

3.1 Distorted Quadrilateral Detection

We have tested our system on quadrilaterals presenting a perspective transformation in the image. In order to execute a quantitative test, we have used quadrilateral shapes with a known distortion factor. The distortion metric β was defined as proposed by [Lee and Lo, 1994]. Let us consider the four triangles T_1, T_2, T_3 and T_4 in Figure 1 composing the quadrilateral shape on it. A distortion metric α_i , $0 \leq \alpha_i \leq 1$, is associated to each triangle T_i , with α_1 computed as follows for the triangle T_1 :

$$\alpha_1 = \frac{2\sqrt{3} \cdot \|OA \times OB\|}{|OA|^2 + |OB|^2 + |AB|^2} \quad (4)$$

with the \times operator representing the cross product of two vectors. $\| \cdot \|$ is then the area of the parallelogram defined by the cross product operation. If we sort the distortion metrics set α_i in increasing order, i. e.:

$$\alpha_1 \leq \alpha_2 \leq \alpha_3 \leq \alpha_4 \quad (5)$$

then, the distortion metric β for the quadrilateral is defined as:

$$\beta = \frac{\alpha_1 \cdot \alpha_2}{\alpha_3 \cdot \alpha_4} \quad (6)$$

Using this metric, on quadrilaterals like the one shown in Fig. 1, $\beta = 1$ is obtained for a perfect square and $\beta = 0$ is got for an infinitely distorted parallelogram. We have used without loss of generality a quadrilateral that is symmetric about the vertical axis and with two sides of constant length. In this case, the variation of factor β results only in the variation of the position of the fourth vertex in the quadrilateral shape.

We have generated synthetically quadrilateral shapes having $1.0 \geq \beta \geq 0.05$ in intervals of 0.05. Figure 2 shows the geometric appearance of these shapes. We have used these images to validate how well the proposed approach handles distorted quadrilaterals. Tests have been performed on the distorted image set. For each image, we have run 100 tests. Our results are summarized in Table 1. In this table, the error ϵ is taken as the maximum deviation of the vertices found by the algorithm against the ground truth corresponding vertices. The Euclidean distance between the ground truth vertex and the actual vertex detected by the algorithm is used as the error measure. It can be seen that the proposed method detects the quadrilateral with no error if $\beta \geq 0.20$. This interval covers the operational range we could expect in quadrilateral detection tasks needed in applications like those cited in Section 1. If we deal with larger distortions ($\beta < 0.20$), the quadrilateral is not well detected because some of the candidates near the global optima will not satisfy the colinearity check. In such a situation, the proposed algorithm will spend more time to detect correctly a quadrilateral shape.

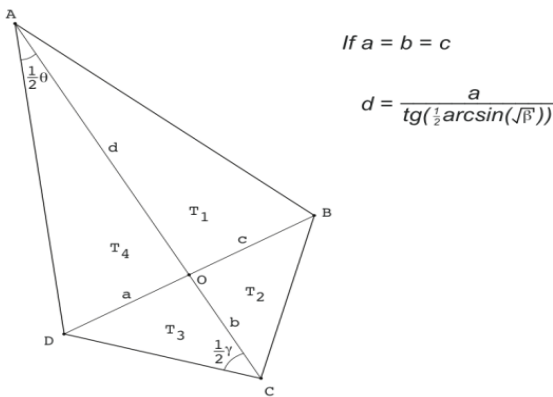


Fig. 1. Relationships between the quadrilateral sides and the distortion metric β

3.2 Processing Time Performance

Our test platform was implemented in C language, on a PC using a Pentium 4 processor running at 3.0 GHz using 1 GB memory. For all the tests in Section 3.1, we have computed the elapsed time T in milliseconds to find the solution of the quadrilateral detection problem. Our results

are also shown in Table 1. As we can see in this table, the main trend is the increasing average processing time as the distortion metrics β diminishes. We can detect quadrilaterals with a distortion factor up to $\beta = 0.50$ with processing rates higher than 20 Hz. This enables our method to be considered for applications where real time performance is needed.

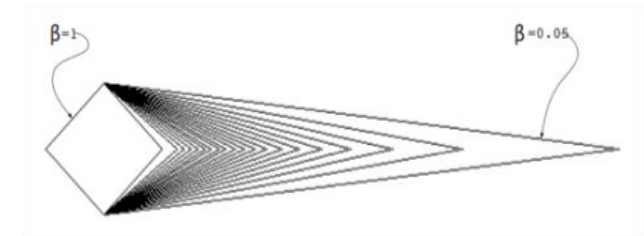


Fig. 2. Synthetic quadrilaterals generated using a set of distortion metrics β values

3.3 Detection of Gaussian Noise-Corrupted Quadrilaterals Subsections

In order to validate the proposed method with respect to input noise, like the one arising for real sensor measurements or any artifact produced by a previous processing step, we have evaluated the algorithm performance when corrupted images are used as input. Let us consider the quadrilateral shape with vertices $V_i = (x_i, y_i)$, $i \in \{1, 2, 3, 4\}$. We build a noisy version by adding Gaussian noise with zero mean and varying deviation. That is, if $y = mx + b$ is the equation of the line joining two consecutive vertices of a quadrilateral, the corresponding noisy segment is defined as:

$$y = mx + b + G(\mu, S) \tag{7}$$

where $G(\mu, S)$ is a Gaussian noise of mean μ and standard deviation S . We have taken $\mu = 0$ in all our tests. In Figure 3, we show a noisy quadrilateral constructed by repeating the procedure described above over the four boundary lines of the quadrilateral under study.

We have built ten random variations of noisy quadrilaterals for the parameter set generated by the Cartesian product $S \times \beta = \{0.25, 0.5, 1.0, 2.0, 4.0\} \times \{1.0, 0.5, 0.25\}$.

Table 1. Results of the quantitative performance evaluation for the detection of quadrilaterals having different distortion factor β

β	ϵ_{avg} (pixels)	ϵ_{max} (pixels)	σ_{ϵ} (pixels)	T_{avg} (ms)	T_{max} (ms)
1.00	0.00	0.00	0.00	20.7	60.0
0.95	0.00	0.00	0.00	21.3	60.0
0.90	0.00	0.00	0.00	24.3	110.0
0.85	0.00	0.00	0.00	27.3	120.0
0.80	0.00	0.00	0.00	28.7	220.0
0.75	0.00	0.00	0.00	29.3	100.0
0.70	0.00	0.00	0.00	32.2	129.9
0.65	0.00	0.00	0.00	33.3	140.0
0.60	0.00	0.00	0.00	40.7	170.0
0.55	0.00	0.00	0.00	45.3	150.0
0.50	0.00	0.00	0.00	48.0	140.0
0.45	0.00	0.00	0.00	68.7	200.0
0.40	0.00	0.00	0.00	78.3	225.0
0.35	0.00	0.00	0.00	82.0	230.0
0.30	0.00	0.00	0.00	89.8	137.0
0.25	0.00	0.00	0.00	131.0	570.0
0.20	0.00	0.00	0.00	142.3	550.4
0.15	2.07	17.26	4.72	286.1	1820.7
0.10	3.21	23.35	5.72	487.8	1852.2
0.05	11.29	23.19	7.52	559.0	2658.9

We have applied the proposed method 50 times for each image in the variation set of the known ground truth quadrilateral. The error ϵ is measured as in the perspective distortion test. Table 2 shows main findings of our experiments. We can see that quadrilateral detection accuracy is degraded less than two pixels for Gaussian noise variances (S_2) up to 1. As a result, we can expect our method to behave steadily under the presence of Gaussian noise. That is a common situation in real world applications.

3.4 Quadrilateral Discrimination Tests

The proposed approach can handle input images where there are multiple shapes on it (including only one quadrilateral). We show a typical

example for this condition in Fig. 4a. Our algorithm has detected the quadrilateral on the image (noted by a darker print in Fig. 4b) with no error in 50 test runs. The average time for the detection has been 0.59 seconds and in the worst case, the processing time has been 2.51 s.

3.5 Simultaneous Detection of Multiple Quadrilaterals

We can detect multiple quadrilaterals in two ways: i) A sequential execution of the proposed algorithms after removing the already detected quadrilateral shapes in the edge image and, ii) a GA sharing based approach to detect multiple quadrilaterals in a single algorithm run. In the second case, the sharing operator avoids keeping

individuals that are very similar among them as part of the population. That increases the computational cost of the algorithm, but this increase is lower than the overall time resulting

from the iterated execution of the single quadrilateral detection approach already explained.



Fig. 3. An example of a Gaussian noise corrupted quadrilateral shape having $\alpha = 50$, $\beta = 0.25$, $\mu = 0$ and $S = 4$

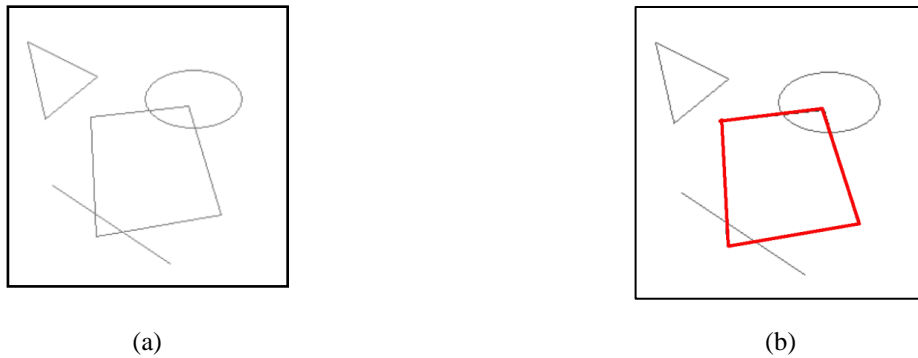


Fig. 4. Quadrilateral discrimination test. (a) Input image. (b) Result of the quadrilateral detection process

3.6 Quadrilateral Detection on Real Images

Here we show how the proposed algorithm is applied to real images and the results obtained from it. When we deal with images originated by any image source, we add a preprocessing step consisting of a simple edge detector (for example, the Prewitt operator.) We have no ground truth data about the real images used in our tests, but we present qualitative results for them. In all images, we show only the quadrilaterals that present a score higher to a given threshold. Figures 5(a),(d),(g),(j) show four typical input images. The corresponding edge maps are presented in Figures 5(b),(e),(h),(k). The detected

quadrilaterals overlaid on the original image are shown correspondingly in Figures 5(c),(f),(i),(l). We can observe in these images that the most salient quadrilaterals in the image are correctly detected by our approach. However, if the quadrilaterals have a small area they are rejected by our approach. If the quadrilateral shapes have an incomplete or deformed boundary below the fitness threshold, then our algorithm does not detect them.

3.7 Hand-Drawn Quadrilateral Detection

Another interesting application for the proposed method could be as a module of a hand-drawn sketch recognition system. To investigate how

well the proposed approach works on hand-drawn images, we have built a test set of images containing quadrilaterals drawn by human users. Our goal is to determine qualitatively if the detected quadrilateral has good resemblance with the hand-drawn images. Two examples of the input images and the results of the quadrilateral

detection method on these images are shown in Figures 6(a), 6(c) and 6(b), 6(d) respectively. It can be noticed that the proposed approach gets a close estimation of the hand-drawn shape. It can also be seen that our system can discriminate the quadrilateral shapes from others even in hand-drawn input images.

Table 2. Quadrilateral detection performance under Gaussian noise data corruption

S	β	ϵ_{avg} (pixels)	ϵ_{max} (pixels)	σ_{ϵ} (pixels)
0.25	1.00	0.00	0.00	0.00
0.25	0.50	0.00	0.00	0.00
0.25	0.25	0.00	0.00	0.00
0.50	1.00	0.00	0.00	0.00
0.50	0.50	0.00	0.00	0.00
0.50	0.25	0.00	0.00	0.00
1.00	1.00	0.00	0.00	0.00
1.00	0.50	0.00	0.00	0.00
1.00	0.25	1.77	2.41	1.08
2.00	1.00	2.83	2.83	0.00
2.00	0.50	5.62	6.16	1.00
2.00	0.25	8.08	9.88	1.29
4.00	1.00	5.80	17.70	4.89
4.00	0.50	13.27	21.91	5.47
4.00	0.25	9.99	10.16	0.07

3.8 Qualitative Performance Comparison to Other Approaches

Here we try to explain why our method overperforms other approaches in literature. For example, Hough transform-based quadrilateral detection requires the extraction of all the straight lines in the image. In most of cases, this detection procedure is performed on the edge image. So, there is the need of a good edge detector, most of works use a Canny edge detector because it results in an image where the edge map is composed of closed contours. Nevertheless, most of lines are duplicated because there are two

edges in any line segment. A non-maximal suppression step is then needed to associate corresponding lines.

In contrast, our approach is not dependent on an elaborate edge detector, the search process of the edge points that are near the predicted points in the quadrilateral boundary counts as a tolerance region for the edge location. If there are two close line segments, the GA-based approach will find what are the parameters of the line segment that minimizes the accumulated Manhattan distance between the actual edge points and the predicted locations of the points belonging to the line segment that is been verified to be present in the image.

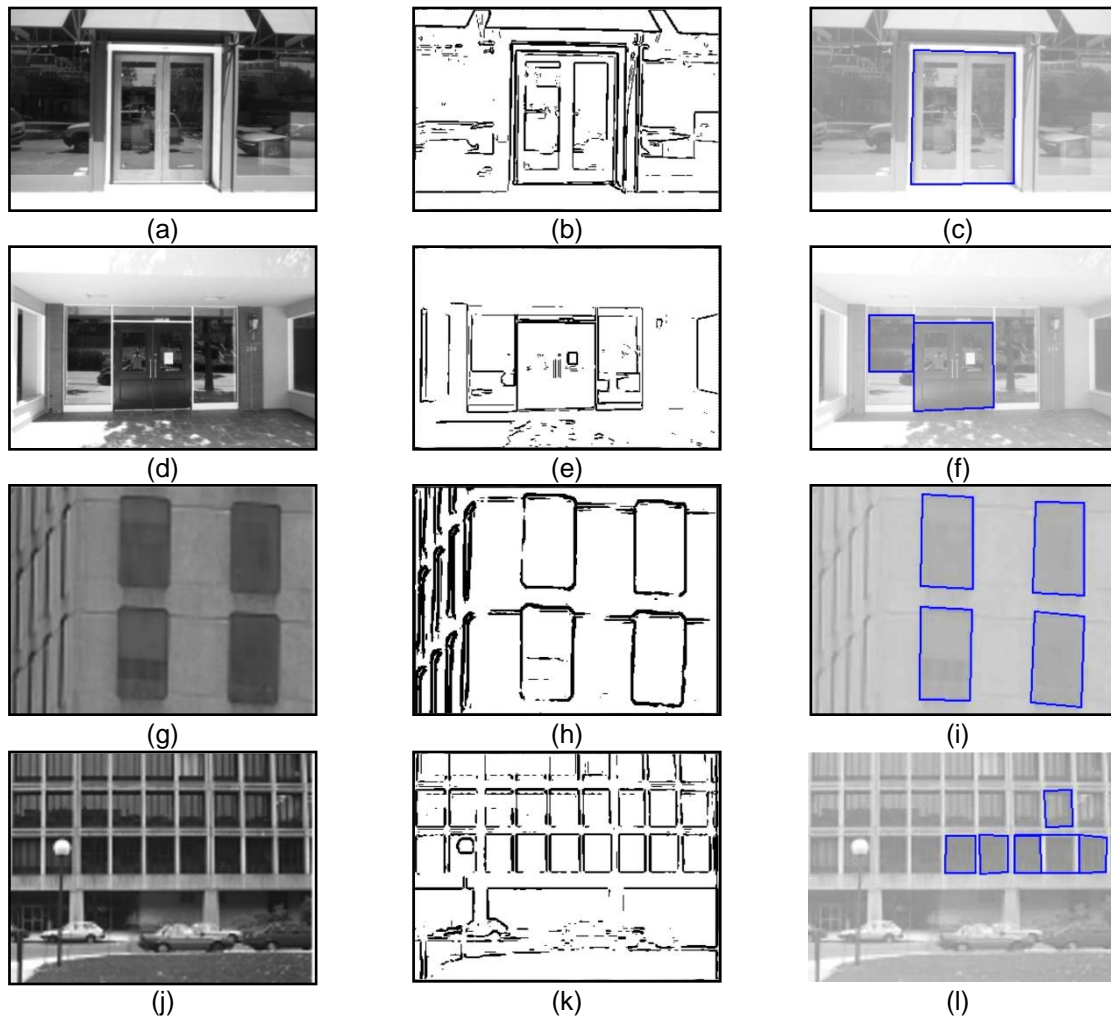


Fig. 5. Typical performance of the proposed approach on real images. First column (a),(d),(g), (j) shows the input images. Second one (b),(e),(h),(k) presents the edge maps obtained by applying a Prewitt edge operator. Third column (c), (f), (i), (l) shows the results of the quadrilateral detection as an overlay of the original image

Hough transform-based methods also need a second step to group all the line segments that form quadrilateral shapes. This implies to find the intersections between all the possible pairs of line segments and to check if four of them are located in a quadrilateral arrangement. This step is not explicitly performed on the proposed approach because the quadrilateral corners are taken from the edge points actually present in the edge map of the image. Figure 7 depicts this in a visual

way. If we use instead the probabilistic Hough transform, the resulting problem is that there are fragmented line segments that are difficult to arrange. Both Hough line detection methods have been performed using the implementation provided by the OpenCV library [3]. If the quadrilaterals are composed of noisy lines, the detection task is difficult to be achieved. The edge discontinuity problem will not let us to chain local line segments into the line segments that

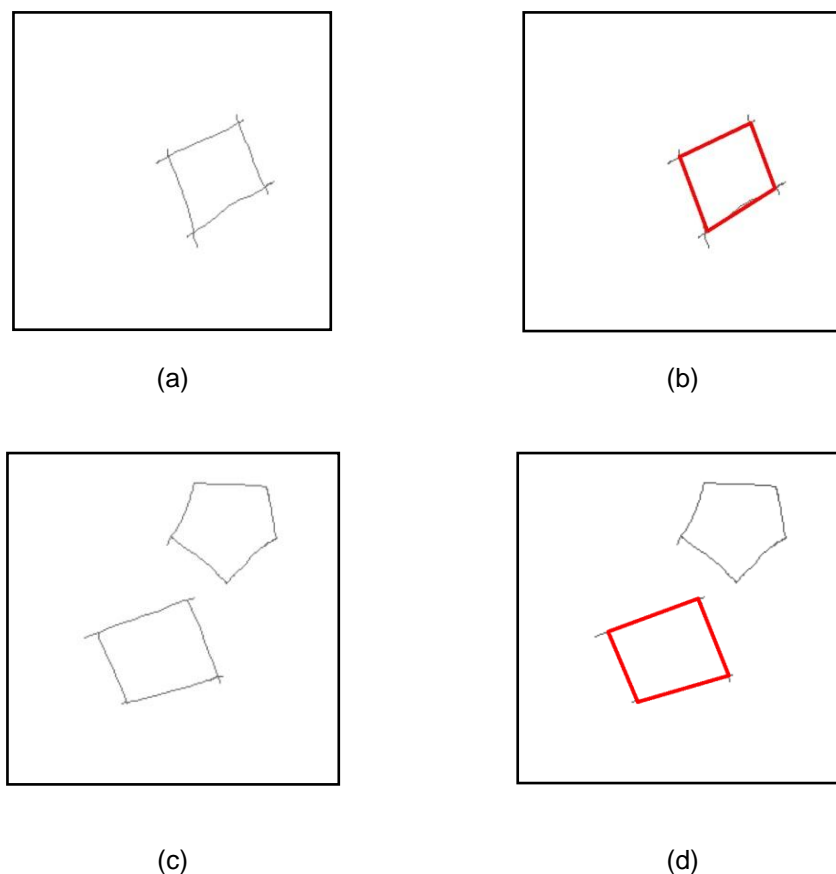


Fig. 6. Typical examples of hand-drawn quadrilateral detection. (a)(c) Original hand drawn images. (b)(d) Results of the corresponding quadrilateral detection process

represent the entire boundary lines of quadrilaterals that are present in the input image. These artifacts in line detection using Hough transform based methods are depicted in Figure 8. We can observe that these methods are not capable of detecting lines if there are significant discontinuities in the input image. No line has been detected when Figure 8(a) has been used as input to a Hough-transform based detection method (Figure 8(b)) and to a Probabilistic Hough transform (Figure 8(c)). The proposed approach has no problems to detect the quadrilateral shape, even if there is a small location error for the farthest corner.

4 Conclusions and Perspectives

We have proposed a novel method for quadrilateral shape detection in images. The proposed approach searches the corresponding edge image by using a genetic algorithm. The initial population of this algorithm comprises individuals encoding random sets of edge points actually present in the edge image. The fitness function of the GA filters the population according to how well the edge information supports the hypothesis of the quadrilateral shape encoded by the GA individual.

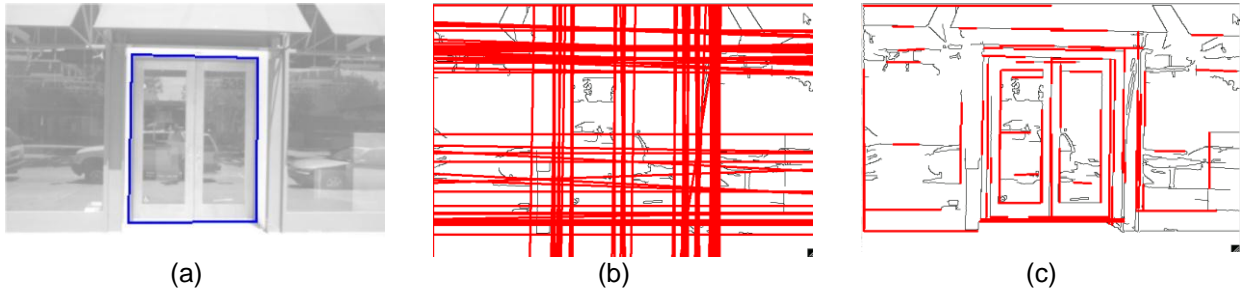


Fig. 7. Comparison of the results of the quadrilateral detection procedure on a real image. (a) Results of using the proposed approach. (b) Lines detected by a generic Hough transform. (c) Line segments detected using the Probabilistic Hough transform

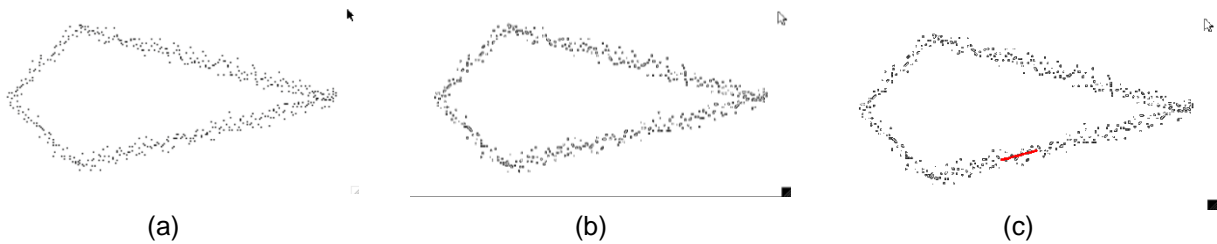


Fig. 8. Comparison of the results of the quadrilateral detection procedure over an image exhibiting a Gaussian distortion in the quadrilateral boundary. (a) Input image (same as Figure 3). (b) Lines detected by a generic Hough transform. (c) Line segments detected using the Probabilistic Hough transform

As a result of the artificial evolution process, best individuals converge towards the quadrilateral shapes present in the input image. We can stop evolution either by setting a minimal fitness score threshold to be reached by the best individual, either by fixing a number of epochs to be used as a maximum. That depends on the intended application and let us to accept approximate quadrilateral shapes up to an arbitrary degree of accuracy. We have validated our approach by evaluating its performance on quantitative and qualitative tests.

We have shown that the proposed method can handle with no error shapes having a distortion factor β ranging from 0.20 to 1.0. Our method can detect quadrilaterals with a distortion factor $\beta > 0.4$ at processing rates higher than 7 Hz. Other tests have demonstrated the validity of our approach to handle Gaussian noise corrupted shapes. This enables us to claim that our method is useful for problems where input data comes from a real sensor or any artifact produced in a previous processing step. We have also shown in a qualitative manner that our method can handle

multiple quadrilaterals on real input images and that can approximate quadrilateral hand drawn shapes. That could be useful in landmark detection tasks for robotics application and in hand-drawn sketch interpretation, respectively. As future work, we will extend the approach to detect quadrilateral shaped texture patches. We will also work towards detection of arbitrary polygonal shapes on images. Another application in perspective is to use the algorithm in a robot localization system given that it requires real time performance.

Acknowledgements

This work has been partially funded by the Fondos Mixtos Conacyt-Concyteq project "Herramientas mecatrónicas para la implementación de entornos virtuales" Project No. GTO- 2005-C04-18605. The work of Mota-Gutierrez is supported by Mexico's Conacyt scholarship grant No. 253676/213766.

References

1. **Ayala-Ramirez, V., Garcia-Capulin, C.H., Perez-Garcia, A. & Sanchez-Yanez, R.E. (2006)**. Circle detection on images using genetic algorithms. *Pattern Recognition Letters* 27(6), 652–657.
2. **Ayala, V., Hayet, J.B., Lerasle, F. & Devy, M. (2000)**. Visual localization of a mobile robot in indoor environments using planar landmarks. *2000 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2000)*, Takamatsu, Japan, 1, 275–280.
3. **Bradski, G. & Kaehler, A. (2008)**. *Learning OpenCV: Computer Vision with the Open CV Library*, CA., USA: O'Reilly.
4. **Canann, S. A., Tristano, J.R. & Staten, M.L. (1998)**. An approach to combined laplacian and optimization-based smoothing for triangular, quadrilateral, and quaddominant meshes. *7th International Meshing Roundtable*, Michigan, U.S.A, 479–494.
5. **Cerri, A., Biasotti, S. & Gorgi, D. (2007)**. K-dimensional size functions for shape descriptions and comparison. *14th International Conference on Image Analysis and processing (ICIAP2007)*, Modena, Italy, 795–800.
6. **Fraundorfer, F., Ober, S. & Bischof, H. (2004)**. Natural, salient image patches for robot localization. *17th International Conference on Pattern Recognition (ICPR'04)*, Cambridge UK, 4, 881–884.
7. **Holland, J.H. (1975)**. *Adaptation in Natural and Artificial Systems*. Ann Arbor, MI: University of Michigan Press.
8. **Keller, C.G., Sprunk, C., Bahlmann, C., Giebel, J. & Baratoff, G. (2008)** Real-time recognition of U.S. speed signs. *2008 IEEE Intelligent Vehicles Symposium*, Eindhoven, The Netherlands, 518–523.
9. **Ko, B.C. & Nam, J.Y. (2006)**. Automatic object-of-interest segmentation from natural images. *18th International Conference on Pattern Recognition (ICPR'06)*, Hong Kong, China, 4, 45–48.
10. **Lazebnik, S., Schmid, C. & Ponce, C. (2006)**. Beyond bags of features: spatial pyramid matching for recognizing natural scene categories. *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, New York, USA, 2, 2169–2178.
11. **Lee, C.K. & Lo, S.H. (1994)**. A new scheme for the generation of a graded quadrilateral mesh. *Computers and Structures*, 52(5), 847–857.
12. **Murillo, A.C., Kosecka, J., Guerrero, J.J. & Sagues, C. (2008)**. Visual door detection integrating appearance and shape cues. *Robotics and Autonomous Systems*, 56(6), 512–521.
13. **Nishida, H. (1996)**. Shape recognition by integrating structural descriptions and geometrical/statistical transforms. *Computer Vision and Image Understanding*, 64(2), 248–262.
14. **Orrite, C. & Herrero, J.E. (2004)**. Shape matching of partially occluded curves invariant under projective transformation. *Computer Vision and Image Understanding*, 93(1), 34–64.
15. **Ozcan, E. & Mohan, C.K. (1997)**. Partial shape matching using genetic algorithms. *Pattern Recognition Letters*, 18(10), 987–992.
16. **Paragios, N., Rousson, M. & Ramesh, V. (2003)**. Non-rigid registration using distance functions. *Computer Vision and Image Understanding*, 89(2-3), 142–165.
17. **Prasad, B.G., Biswas, K.K. & Gupta, S.K. (2004)**. Region-based image retrieval using integrated color, shape and location index. *Computer Vision and Image Understanding* 94(1-3), 193–233.
18. **Sanchez, A.J. & Martinez, J.M. (2000)**. Robot-arm pick and place behavior programming system using visual perception. *15th International Conference on Pattern Recognition*, Barcelona, Spain, 4, 507–510.
19. **Ser, P.K., Choy, C.S.T. & Siu, W.C. (1999)**. Genetic algorithm for the extraction of nonanalytic objects from multiple dimensional parameter space. *Computer Vision and Image Understanding*, 73(1), 1–13.
20. **Shokoufandeh, A., Bretzner, L., Macrini, D., Demirci, M.F., Jonsson, C. & Dickinson, S. (2006)**. The representation and matching of categorical shape. *Computer Vision and Image Understanding*, 103(2), 139–154.
21. **Shotton, J., Blake, A. & Cipolla, R. (2008)**. Multiscale categorical object recognition using contour fragments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(7), 1270–1281.
22. **Silapachote, P., Weinman, J., Hanson, A., Weiss, R. & Mattar, M.A. (2005)**. Automatic sign detection and recognition in natural scenes. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, California, USA, 3, 27.
23. **Sun, Y.N. & Huang, S.C. (2000)**. Genetic algorithms for error-bounded polygonal approximation. *International Journal of Pattern Recognition and Artificial Intelligence*, 14(3), 297–314.

24. **Super, B.J. (2002).** Fast retrieval of isolated visual shapes. *Computer Vision and Image Understanding*, 85(1), 1–21.
25. **Torralba, A., Murphy, K.P., Freeman, W.T. & Rubin, M.A. (2003).** Context-based vision system for place and object recognition. *Ninth IEEE International Conference on Computer Vision (ICCV'03)*, Nice, France, 273–280.
26. **Tu, Z., Zheng, S. & Yuille, A. (2008).** Shape matching and registration by data-driven EM. *Computer Vision and Image Understanding*, 109(3), 290–304.
27. **Yin, P.Y. (2000).** A tabu search approach to polygonal approximation of digital curves. *International Journal of Pattern Recognition and Artificial Intelligence*, 14(2), 243–255.

research interests include color and texture analysis for computer vision tasks, and the computational intelligence applications in feature extraction and decision making.

Article received on 12/03/2010; accepted 05/03/2011.



Victor Ayala Ramirez got a Ph.D. degree on Industrial Informatics from the Université Paul Sabatier in Toulouse, France in 2000. He is with the Universidad de Guanajuato DICIS at Salamanca, Mexico. His research interest is the study and development of robotic and computer vision systems that use soft computing techniques.



Sergio A. Mota Gutiérrez received BEng degree in 2007 and MEng degree in 2010 from the University of Guanajuato for works on areas related to shape recognition and human-machine interaction under the supervision of Professor Victor Ayala Ramirez. Currently He is pursuing PhD degree at the Mathematics

(CIMAT) under the direction of Professor Salvador Ruiz Correa.



Raúl E. Sánchez Yáñez is a Doctor of Science (Optics), concluding his studies at the Centro de Investigaciones en Óptica (CIO), Leon, Mexico, in 2002. He is also a Master of Electrical Engineering and BEng in Electronics, both degrees were obtained from the Universidad de

Guanajuato, where he has been a full time professor since 2003 at the Engineering Division of the Campus Irapuato-Salamanca (formerly FIMEE). His main