Accurate L-corner Measurement using USEF Functions and Evolutionary Algorithms

Gustavo Olague[1], Benjamín Hernández[2], and Enrique Dunn[1]

[1] Departamento de Ciencias de la Computación, División de Física Aplicada,
Centro de Investigación Científica y de Estudios Superiores de Ensenada,
Km. 107 carretera Tijuana-Ensenada, 22860, Ensenada, B.C., México
{olague, edunn}@cicese.mx
http://cienciascomp.cicese.mx/Pagina-Olague.htm
[2] Instituto de Astronomía, Ensenada
Universidad Nacional Autónoma de México
Observatorio Astronómico Nacional
km. 103 Carretera Tijuana-Ensenada, Ensenada, B.C., México
{benja}@astrosen.unam.mx

Abstract. Corner feature extraction is studied in this paper as a global optimization problem. We propose a new parametric corner modeling based on a Unit Step Edge Function (USEF) that defines a straight line edge. This USEF function is a distribution function, which models the optical and physical characteristics present in digital photogrammetric systems. We search model parameters characterizing completely single gray-value structures by means of least squares fit of the model to the observed image intensities. As the identification results relies on the initial parameter values and as usual with non-linear cost functions in general we cannot guarantee to find the global minimum. Hence, we introduce an evolutionary algorithm using an affine transformation in order to estimate the model parameters. This transformation encapsulates within a single algebraic form the two main operations, mutation and crossover, of an evolutionary algorithm. Experimental results show the superiority of our L-corner model applying several levels of noise with respect to simplex and simulated annealing.

1 Introduction

Photogrammetry is the science, and art, of determining the size and shape of objects as a consequence of analyzing images recorded on film or electronic media. Close-range photogrammetry as well as computer vision relies on image processing techniques in order to obtain the information required for tasks devoted to perceiving, sensing and measuring the world around a machine vision system. Corners and contours are recognized as a basic characteristic in machine vision, see [1], [8], [2], [3], [5], [13], [14], [16], [17], [22], [23], [27]. High-accurate corner detection is a complex process due to several factors: 1) the attitude, position and orientation, of the camera with respect to the object, 2) the interior orientation of the camera, 3) the fluctuations on illumination and 4) the camera optics,

see Olague and Mohr [20]. Several approaches to the problem of detecting feature points have been reported in the literature in the last few years. They can be broadly divided into three groups. The first one consists of boundary based approaches like those proposed by Tsai et al. [25], Medioni and Yasumoto [15], and Sohn et al. [24]. Approaches in the second group involve those based on computing geometric properties directly from the gray-level images, see Beaudet [4], Dreschler and Nagel [7], Kitchen and Rosenfeld [12] and finally Wang and Brady [26]. The third group is composed by approaches based on parametric models, see Deriche and Giraudon [5], Rohr [22] and Olague and Hernández [19].

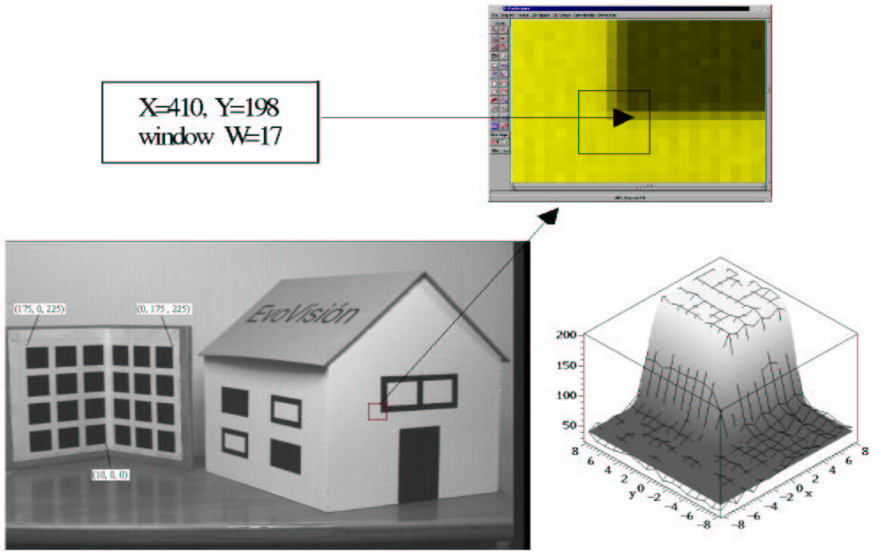

Fig. 1. A zoom of one L-corner of our EvoVisión house is shown as well as its final three dimensional model.

The paper is organized as follows. A first section is devoted to the presentation of our L-corner model. Then we will present the process of modeling L-corners as an optimization problem. Apart from the problem of determining a model that fits the data, there exists the problem of estimating the accuracy of the parameters. Indeed, our approach provides these magnitudes. A novel evolutionary representation is introduced using the concepts of affine transformation. Finally, experimental results are presented in order to show the superiority of our affine evolutionary algorithm against simulated annealing and downhill simplex.

2 L-Corner Model and χ^2 *Estimator*

Modeling image intensities $(f(x_i, y_i), x_i, y_i)$ of an L-corner to a model that is a linear combination of nonlinear functions of (x_i, y_i) is a subject known as modeling of data, see Figure 1. An L-corner is built of two *Unit Step Edge Functions* (USEF) as follows:

Definition 1 (Unit Step Edge Function). *Let the image coordinates and the set of unknown model parameters be denoted by $I = (x, y)$ and $P = (p_1, \ldots, p_n)$ respectively. The unit step edge function is represented as follows:*

$$U_x(I, P_x) = \pm \frac{1}{\sigma_1 \sqrt{2\pi}} \int_0^x e^{\frac{-(t - y \cdot \tan(\theta_1) - \mu_1)^2}{2\sigma_1^2}} dt + \frac{1}{2} \tag{1}$$

where the image coordinates range from $[-m, m]$. The central point μ_1 designs the position x of the line that crosses along the y-axis. μ_1 ranges from $[-m, m]$. The rotation θ_1 is made clockwise about the (positive) y-axis. θ_1 designs the orientation of the edge model to be fitted to the image within the range $-\frac{\pi}{2} \leq \theta_1 \leq \frac{\pi}{2}$. Finally a scaling factor σ_1 that characterizes the amount of blur introduced by the discretization process needs to be taken into account. σ_1 ranges from $[0, m]$.

The unit edge function describes a distribution function that increases steadily from 0 to 1 with respect to the x-axis. The unit step edge function $U_y(I, P_y)$ with respect to the y-axis is represented in a similar way where all intervals of the variables remain the same. $U_y(I, P_y)$ can be evaluated numerically using the Gaussian error function as follows:

$$U_y(I, P_y) = -\frac{1}{2} erf\left(\frac{\sqrt{2}(-y + x \cdot \tan(\theta_2) + \mu_2)}{2\sigma_2^2}\right) + \frac{1}{2} \tag{2}$$

In order to obtain an L-corner model we multiply both USEF as follows:

$$M_L'(x, y, \boldsymbol{P}) = U_x(I, P_x) \cdot U_y(I, P_y) \tag{3}$$

The parameters σ_1, ϑ_1, μ_1, σ_2, ϑ_2, μ_2, A y B, represents the physical and geometrical contours of an L-corner.

Corner's localization was obtained by fitting our parametric model to the image intensities. Estimates for the model parameters $\boldsymbol{P} = (p_1, \ldots, p_n) \in R^n$ are found by minimizing the squared differences between the (nonlinear) model function and the considered gray values:

$$Q = \chi^2 = F(\boldsymbol{P}) = \sum_{i=1}^{m} \sum_{j=1}^{m} [I(u_i, v_j) - M_L'(x_i, y_j, \boldsymbol{P})]^2 \tag{4}$$

The intensities and the function values of the model in the considered image area are $I(u_i, v_j)$ and $M_L'(x_i, y_j, \boldsymbol{P})$ respectively. Previous approaches used by Rohr (1992) applied the method of Powell utilizing only function values or used

the method of Levenberg-Marquardt (Press et al. 1998) incorporating partial derivatives of the model function in order to reduce the computation time. However, a drawback presented on these approaches is that the identification result relies on the initial parameter values and as usual with nonlinear cost functions in general we cannot guarantee to find the global minimum. This problem is overcome in this work using an evolutionary algorithm.

3 Modeling L-corners as an Optimization Problem

Evolutionary algorithms has raised as a rich paradigm for global optimization. Previous methodologies as the *Down Hill Simplex Method* and *Simulated Annealing* are well known techniques for multidimensional optimization [21]. This section is devoted to our affine evolutionary algorithm for global optimization. Currently, evolutionary algorithms for numerical optimization use real code parameters for which a set of special transformations has been developed. In real coding implementation, each chromosome is encoded as a vector of real numbers of the same length. Several crossover operators have been introduced under the name of arithmetical operators. The arithmetical operators are built by borrowing the concept of linear combination of vectors from the area of convex sets theory. Generally, crossover produces an offspring, which is calculated from the weighted average of two vectors $\boldsymbol{y}_1$ y $\boldsymbol{y}_2$ as follows:

$$\begin{aligned} \boldsymbol{y}_1' &= \lambda_1 \boldsymbol{y}_1 + \lambda_2 \boldsymbol{y}_2 \\ \boldsymbol{y}_2' &= \lambda_2 \boldsymbol{y}_1 + \lambda_1 \boldsymbol{y}_2 \end{aligned} \tag{5}$$

if the multipliers are restricted to:

$$\lambda_1 + \lambda_2 = 1, \quad \lambda_1 > 0, \quad \lambda_2 > 0$$

the weighted form is known as convex combination. If the non-negativity condition on the multipliers is dropped, the combination is known as affine combination. Finally, if the multipliers are simply required to be in real space, the combination is known as a linear combination [9]. Another operator is known under the name of dynamic mutation, also called non-uniform mutation, introduced by Janikow and Michalewicz [11]. Dynamic mutation is designed for fine-tuning capabilities aimed at achieving high precision. Given a parent $\boldsymbol{y}$, if the element y_k of it is selected for mutation, the resulting offspring is $\boldsymbol{y} = [y_1, \ldots, y_k', \ldots, y_n]$, where y_k' is randomly selected from the following two possibilities:

$$\begin{aligned} y_k' &= y_k + \Delta(t, y_k^U - y_k) \ , \ or \\ y_k' &= y_k - \Delta(t, y_k - y_k^L) \end{aligned}$$

where

$$\Delta(t, y) = yr \left(1 - \tfrac{t}{T}\right)^b \tag{6}$$

The function $\Delta(t, y)$ returns a value in the range $[0, y]$ such that the value approaches 0 as t increases. This property causes the operator to search the space

uniformly initially, when t is small, and very locally at later stages. t is the generation number, b is a parameter determining the degree of non-uniformity and r is a random number between $[0,1]$. It is possible for the operator to generate an offspring, which is not feasible. In such a case, we can reduce the value of the random number r.

3.1 A Novel Evolutionary Representation

The operations of crossover and mutation can be encapsulated into a single complex transformation as follows. In order to handle affine geometry algebraically, we have to characterize the line i by an invariant equation, and we shall suppose that this equation is $y_0 = 0$. Since, the points of i are now regarded as ideal points, no point with $y_0 = 0$ is actual, and this means that we can represent the actual points of the affine plane by pairs of non-homogeneous coordinates $Y = (Y_1, Y_2)$, where

$$Y_1 = \frac{y_1}{y_0}\ ,\ Y_2 = \frac{y_2}{y_0}$$

The allowable representations $\mathcal{R}_A$ of the affine plane are those representations $\mathcal{R}$ of $\hat{S}_2$ in which the line i has the equation $y_0 = 0$; and this leads at once to the following theorem:

Theorem 1. *If $\mathcal{R}_A$ is any allowable representation of the affine plane, then the whole class $(_A)$ of allowable representations consists of all those representations, which can be derived from $\mathcal{R}_A$ by applying a transformation of the form*

$$\begin{aligned} Y_1' &= b_{11}Y_1 + b_{12}Y_2 + C_1 \\ Y_2' &= b_{21}Y_1 + b_{22}Y_2 + C_2 \end{aligned}$$

where the coefficients are arbitrary real numbers, subject to the condition $|b_{rs}| \neq 0$.

Using Theorem 1, it is possible to transform the n variables of two solutions into a new pair of solutions, according to the following transformation:

$$\begin{pmatrix} Y_{1_1}' & Y_{1_2}' & \dots & Y_{1_n}' \\ Y_{2_1}' & Y_{2_2}' & \dots & Y_{2_n}' \end{pmatrix} = \left[\begin{array}{cc} b_{11} & b_{12} \\ \underbrace{b_{21}}_{crossover} & \underbrace{b_{22}}_{} \end{array} \begin{array}{c} C_1 \\ \underbrace{C_2}_{mutation} \end{array} \right]_n \begin{pmatrix} Y_{1_1} & Y_{1_2} & \dots & Y_{1_n} \\ Y_{2_1} & Y_{2_2} & \dots & Y_{2_n} \\ 1 & 1 & \dots & 1 \end{pmatrix} \tag{7}$$

Equation 7, can be expanded to the whole population. The advantages of this representation are:

1. Standardized treatment of all transformations.
2. Complex transformations are composed from single transformations by means of matrix multiplication.
3. An n dimensional point can be transformed by applying a set of n transformations.
4. Simple inversion of the transformation by matrix inversion.
5. Extremely fast, hardware supported matrix operations in high-power graphic workstations.

4 Experimental results

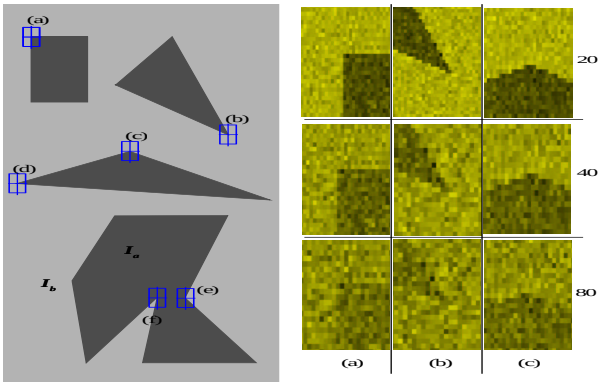

Fig. 2. On the right structures (a), (b) and (c) of the test synthetic image with Gaussian noise scaled by $\lambda = 20, 40, 80$.

In order to show the robustness of each algorithm and the stability of our L-corner model, we applied Gaussian noise of zero mean and unit variance. This noise is scaled by a constant λ in order to produce perturbations on the synthetic test image. This noise is known as *additive* noise [6]. We decide to use a synthetic image in order to know precisely the location of the corners. The signal-to-noise ratio (SNR) is computed in decibels (DB). Figure 2, shows the structures (a), (b) and (c) with the Gaussian noise scaled by the factors $\lambda = 20, 40, 80$. Table 1, presents final results of corners (a) $\rightarrow$ (f) of Figure 2, using three optimization strategies without noise. It is important to remember that for $SNR \rightarrow 1$ the error on the noisy image is approximately equal to the amount of signal of the synthetic image. If $SNR \leq 1$ the noise is bigger than the original signal and if $SNR \gg 1$ the original synthetic image and the synthetic image with noise are equivalents, the error tends to zero. The test of our L-corner detector considering three optimization algorithms was built as follows:

1. The *Down Hill Simplex*, *Simulated Annealing* and *Evolutionary Algorithm* were applied to the structures (a), (b) and (c) of Figure 2. Those structures show three different corners: straight angle corner, acute angle corner and obtuse angle corner respectively.
2. The size of the window is 13×13 pixels, centered around the pixel:
   (a) $(u_0, v_0) = (87, 87)$.
   (b) $(u_0, v_0) = (707, 353)$.
   (c) $(u_0, v_0) = (396, 397)$.
3. The control parameters of each algorithm are:
   – *Down Hill Simplex*. Maximal number of movements of the simplex $N = 4500$.

Down Hill simplex

Corner	Initial Pixel		Corner Point		Aperture	s_{U_1}	s_{U_2}	χ^2
	u_0	v_0	u_e	v_e	α^o			
(a)	89	89	87.404142	87.709506	89.76333	+1	-1	1.13215e-15
(b)	706	353	707.028574	353.831573	35.7994	-1	+1	5.44007e-16
(c)	398	397	396.825889	397.466375	149.9239	-1	-1	1.31891e-09
(d)	44	484	40.985292	484.451324	19.1055	+1	+1	4.22482e+04
(e)	570	795	570.573303	792.934090	97.4398	+1	+1	1.22339e-11
(f)	482	795	483.081960	792.272648	37.2723	-1	-1	6.41950e-23

	σ_1	σ_2	μ_1	μ_2	ϑ_1^o	ϑ_2^o	A	B
(a)	6.04357e-02	5.20620e-02	-1.60603e+00	1.33941e+00	-8.74457e-01	6.37787e-01	103.00	76.00
(b)	2.34760e-02	9.25585e-03	4.76497e-01	-4.44535e-01	-3.35800e+01	-2.06206e+01	103.00	76.00
(c)	2.34221e-03	1.72416e-02	-2.75132e+00	-1.82255e-01	-7.35272e+01	1.36033e+01	103.00	76.00
(d)	6.29735e-02	8.53333e-03	-1.25144e+00	-7.01737e-01	7.56428e+01	-4.74830e+00	102.54	76.46
(e)	1.15833e-02	8.79562e-04	-6.67335e-01	2.51901e+00	3.09704e+01	-3.84093e+01	103.00	76.00
(f)	1.33997e-02	4.80644e-03	3.97708e-01	1.86938e+00	1.41271e+01	3.86009e+01	103.00	76.00

Simulated Annealing

Corner	Initial Pixel		Corner Point		Aperture	s_{U_1}	s_{U_2}	χ^2
	u_0	v_0	u_e	v_e	α^o			
(a)	89	89	87.526970	87.437709	90.01414	+1	-1	1.94364e-10
(b)	706	353	707.017039	353.829675	35.7614	-1	+1	3.56136e-13
(c)	398	397	396.797404	397.427744	149.4558	-1	-1	9.75333e-13
(d)	44	484	40.077467	484.514532	16.977	+1	+1	4.22483e+04
(e)	570	795	570.647257	793.111032	97.7988	+1	+1	6.15267e-01
(f)	482	795	483.046760	792.255049	37.3776	-1	-1	1.54081e-13

	σ_1	σ_2	μ_1	μ_2	ϑ_1^o	ϑ_2^o	A	B
(a)	8.69092e-02	7.67896e-02	-1.47251e+00	1.56314e+00	-1.88830e-02	3.30323e-02	103.00	76.00
(b)	5.01805e-03	7.87605e-03	4.71431e-01	-4.47754e-01	-3.35296e+01	-2.07090e+01	103.00	76.00
(c)	2.11139e-02	2.12681e-02	-2.62621e+00	-1.21410e-01	-7.34947e+01	1.40389e+01	103.00	76.00
(d)	3.16852e-07	3.69060e-04	-1.76589e+00	-7.58444e-01	7.65812e+01	-3.55820e+00	102.56	76.46
(e)	9.50347e-10	7.77463e-10	-4.33799e-01	2.38709e+00	2.97825e+01	-3.75813e+01	102.98	76.03
(f)	1.37867e-02	3.61182e-03	3.66855e-01	1.91193e+00	1.39431e+01	3.86793e+01	103.00	76.00

Evolutionary Algorithm

Corner	Initial Pixel		Corner Point		Aperture	s_{U_1}	s_{U_2}	χ^2
	u_0	v_0	u_e	v_e	α^o			
(a)	89	89	87.895717	87.644934	89.3094	+1	-1	9.40007e-06
(b)	706	353	706.807636	353.755519	36.2416	-1	+1	1.24802e-06
(c)	398	397	396.943119	397.390921	148.7181	-1	-1	5.69696e-05
(d)	44	484	41.009408	484.473710	18.53073	+1	+1	4.22483e+04
(e)	570	795	570.411461	792.991919	95.7904	+1	+1	4.53057e-04
(f)	482	795	483.170778	792.099214	37.0405	-1	-1	1.17246e-06

	σ_1	σ_2	μ_1	μ_2	ϑ_1^o	ϑ_2^o	A	B
(a)	1.24043e-02	1.41591e-02	-1.11322e+00	1.34753e+00	6.73623e-02	-7.57916e-01	103.00	76.00
(b)	4.32836e-03	9.65062e-03	3.20867e-01	-4.52084e-01	-3.30288e+01	-2.07296e+01	103.00	76.00
(c)	9.80709e-03	8.63958e-03	-2.33377e+00	-1.17649e-01	-7.30867e+01	1.43686e+01	103.00	76.00
(d)	3.31949e-02	2.23349e-03	-1.10955e+00	-7.08228e-01	7.58993e+01	-4.43003e+00	102.59	76.45
(e)	1.22747e-03	1.74839e-03	-8.23944e-01	2.32158e+00	3.15841e+01	-3.73745e+01	103.00	76.00
(f)	1.13199e-02	5.92920e-03	4.34052e-01	1.97009e+00	1.43014e+01	3.86581e+01	103.00	76.00

Table 1. Exact corner location (u_e, v_e) and parameter values $\boldsymbol{P}$ using the *Down-Hill Simplex*, *Simulated Annealing* and *Evolutionary Algorithm*, considering structures (a) $\rightarrow$ (f) of Figure 2.

- *Simulated Annealing.* Initial temperature $T = 1$, size of the equilibrium state $I = 20$, maximal number of iterations $N = 4500$.
- *Evolutionary Algorithm.* Crossover percentage $pc = 0.80$, mutation percentage $pm = 0.05$, convergence percentage $pf = 0.75$, offspring number in the population $P = 22$, maximal number of generations $N = 2000$ approximately equivalent to 4500 movements.

4. 30 samples for each test were performed.
5. Each noisy window $I_r(i,j)$ was normalized to the intensity values $[0, 255]$ considering real numbers through the following function:

$$I_n(j,i) = \frac{255}{max(I_r(j,i)) - min(I_r(j,i))} I_r(j,i)$$

where $I_n(j,i) \mid i,j = 1,\ldots,2w+1$ is the normalized studied window including Gaussian noise.

6. Percentage of stop criteria $ftol = 1 \times 10^{-09}$.

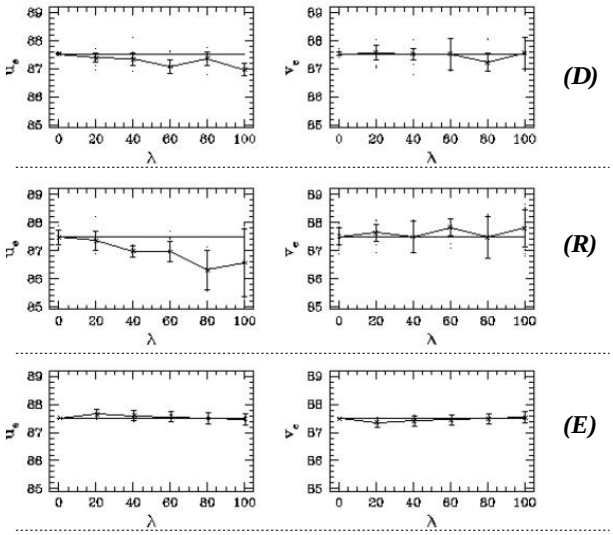

Fig. 3. Behaviour of the *Down Hill Simplex* ***(D)***, *Simulated Annealing* ***(R)*** and *Evolutionary Algorithm* ***(E)*** considering a Gaussian noise scaled by a factor λ over the structure (a) of the synthetic image.

As a result of the test the Figures: 3, 4, 5 were built. These figures show the displacement of the corner position (u_e, v_e) of the structures (a), (b) and (c) respectively, considering that a random Gaussian noise was applied over the test

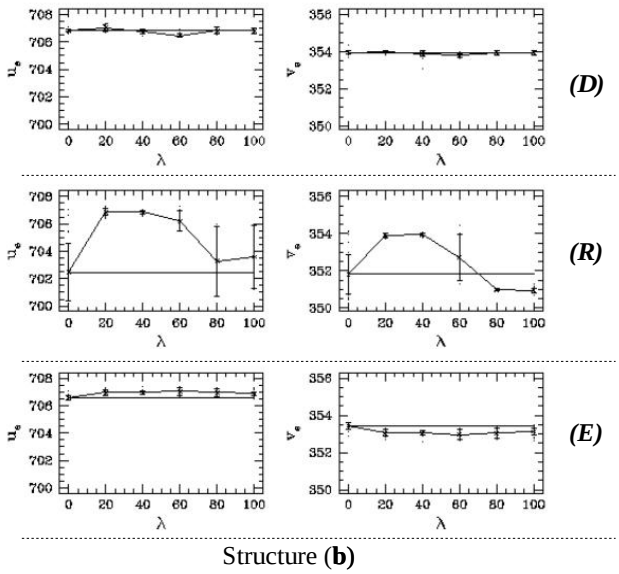

Fig. 4. Behaviour of the *Down Hill Simplex* ***(D)***, *Simulated Annealing* ***(R)*** and *Evolutionary Algorithm* ***(E)*** considering a Gaussian noise scaled by a factor λ over the structure (b) of the synthetic image.

image. Each figure shows the average corner point $(\overline{u_e}, \overline{v_e})$ and its final standard deviation considering 30 samples for each optimization strategy. The charts on the left of each figure represent the u coordinate and those on the right represent the v coordinate of the image coordinate system (u, v). The horizontal straight line denotes the corner position (u_e, v_e) of the free-noise synthetic image $\lambda = 0$. After a careful analysis of these figures we conclude the following:

1. Beyond $\lambda = 80 (SNR \approx 3.5)$, see Figure 3, the contours of the structures (a), (b) and (c) are difficult to be distinguished. However, the random Gaussian noise doesn't blur the borders. Hence, the structure is more or less preserved in shape. Then, it is possible to study the ability that each optimization strategy offers, in order to integrate and reconstruct each structure.
2. In the case of structure (a) the *evolutionary algorithm* presents the best curve of behaviour in presence of noise. Moreover, the maximum standard deviation is obtained for $\lambda = 20$. This value is approximately 0.14 pixels.
3. In the case of structure (b) the *Down Hill Simplex* presents the best curve of behaviour in presence of noise. The maximum standard deviation occurs in $\lambda = 60$ over the u axis. This value is about 0.37 pixels compared to 0.47 pixels obtained by the *evolutionary algorithm* also in $\lambda = 60$. The curve of the *evolutionary algorithm* remains constant around the average value (x_e, y_e) in all cases.

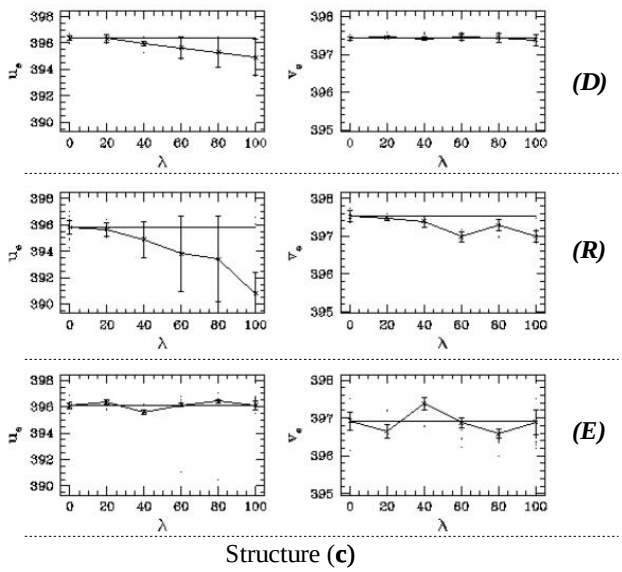

Fig. 5. Behaviour of the *Down Hill Simplex* ***(D)***, *Simulated Annealing* ***(R)*** and *Evolutionary Algorithm* ***(E)*** considering a Gaussian noise scaled by a factor λ over the structure (c) of the synthetic image.

4. In the case of structure (c) the *evolutionary algorithm* presents the best curve in presence of noise over the u axis. While, the *Down Hill Simplex* presents the best curve around the v axis.
5. If we observe the error bars that represents the standard deviation of 30 samples for each test considering the three structures. The *Evolutionary Algorithm* presents the best average for each experiment as follows: 1) The average standard deviation of the *Evolutionary Algorithm* is 0.19, 2) The average standard deviation of the *Down Hill Simplex* is 0.25, 3) The average standard deviation of the *Simulated Annealing* is 0.76 pixels.

As a result, the *Evolutionary Algorithm* is less sensitive to noise. Hence, it can be considered more robust. However, the *Down Hill Simplex* offers similar results for the L-corner studied here. Finally, the *Simulated Annealing* shows the worst behaviour in the presence of Gaussian noise.

5 Summary and Conclusions

Accurate L-corner measurement was obtained with a parametric model $M_L'(x, y, \boldsymbol{P})$ using a global optimization approach. The goal was to obtain the best set of parameters $\boldsymbol{P} = (\sigma_1, \mu_1, \vartheta_1, \sigma_2, \mu_2, \vartheta_2, A, B)$, that fits a window of $2w - 1 \times 2w - 1$ pixels centered around a pixel (u_0, v_0) within a digital image. The optimization

criterion used was the maximum likelihood estimator χ^2 obtained through a multidimensional least squares fitting of the data to the L-corner model. We use the Levenberg Marquardt method as many scientists have done in the past. The Levenberg Marquardt method is considered as a local method. We propose an *Evolutionary Algorithm* using a novel representation that integrates the two main operators into a single affine transformation. As a result, we obtain a local criteria embedded into a global method. Hence, our algorithms are accelerated by the Levenberg Marquardt Method. Concluding, the *evolutionary algorithm* presents the best behaviour against noise. The *Down Hill Simplex* is the simplest strategy to operate because it doesn't requires any control parameter. On the other side, the *Evolutionary Algorithm* has the ability to increase the population size, which increases the probability to find a better result in a smaller number of generations. The other two strategies doesn't have this ability. Finally, the *Evolutionary Algorithm* can be easily parallelized.

Acknowledgments

This research was funded through the LAFMI (Laboratoire Franco-Mexicain d'Informatique) project sponsored by CONACyT-INRIA.

References

1. L. Alvarez and F. Morales. "Affine Morphological Multiscale Analysis of Corners and Multiple Junctions". *International Journal of Computer Vision.* 25(2), pp. 95-107, Kluwer Academic Publishers, 1997.
2. J. Canny. "A Computational Approach to Edge Detection". *IEEE Trans. on Pattern Analysis and Machine Intelligence.* Vol. 8, No. 6, November, 1986.
3. S. Baker, S. K. Nayar and H. Murase. "Parametric Feature Detection". *International Journal of Computer Vision*, 27(1), pp. 27-50, Kluwer Academic Publishers, 1998.
4. Beaudet, P. R. "Rotationally Invariant Image Operators." *In Proc. of the International Conference on Pattern Recognition.* 579-583. 1978.
5. R. Deriche and G. Giraudon. "A Computational Approach for Corner and Vertex Detection". *International Journal of Computer Vision*, 10(2), pp. 101-124, Kluwer Academic Publishers, 1993.
6. Dougherty, E. R. Random Processes for Image and Signal Processing. SPIE Optical Engineering Press, and IEEE Press, Inc.1999.
7. Dreschler, L., and Nagel, H. H. " On the Selection of Critical Points and Local Curvature Extrema of Region Boundaries for Interframe Matching." *In Proc. of the International Conference on Pattern Recognition.* 542-544. 1982
8. Ebner, M., and Zell, A. "Evolving a Task Specific Image Operator." *In Evolutionary Image Analysis, Signal Processing and Telecommunications.* LNCS 1596, Poli et al. (Eds.), EvoIASP. 1999.
9. M. Gen and R. Cheng. "Genetic Algorithms and Engineering Design". John Wiley and Sons, Inc. 1997.
10. A. W. Gruen. "Adaptive Least Squares Correlation: A Powerful Image Matching Technique". *S. Afr. Journal of Photogrammetry, Remote Sensing and Cartography.* 14(3), pp. 175-187. 1985.

11. C. Janikow and Z. Michalewicz. "An Experimental Comparison of Binary and Floating Point Representations in Genetic Algorithms". *In Proceedings of the Fourth International Conference on Genetic Algorithms.* pp. 31-36, San Mateo California, USA. 1991.
12. L. Kitchen and A. Rosenfeld. "Gray Level Corner Detection". *Pattern Recognition Letters.* No. 1, pp. 95-102, 1982.
13. T. Lindeberg. "Feature Detection with Automatic Scale Selection". *International Journal of Computer Vision.* 30(2), pp. 79-116, Kluwer Academic Publishers, 1998.
14. D. Marr and E. Hildreth. "Theory of Edge Detection". *Proc. Roy. Soc. London, 207, pp. 187-217*, 1980.
15. Medioni, G., and Yasumoto, Y. " Corner Detection and Curve Representation using cubic B-splines." *Computer Vision, Graphics and Image Processing.* 39, 267-278., 1987.
16. R. Mehrotra and S. Nichani. "Corner Detection". *Pattern Recognition.* Vol. 23, No. 11, pp. 1223-1233, 1990.
17. H. P. Moravec. "Towards automatic visual obstacle avoidance". *In Proceedings of the 5th International Joint Conference on Artificial Intelligence*, pp. 584, Cambridge, Massachusetts, USA. 1977.
18. G. Olague and B. Hernández. "Autonomous Model Based Corner Detection using Evolutionary Algorithms". *In American Society for Photogrammetry and Remote Sensing.* 12 pages. ASPRS Annual Conference 2001
19. G. Olague and B. Hernández. "Flexible Model-based Multi-corner Detector for Accurate Measurements and Recognition". *16th International Conference on Pattern Recognition.* IEEE Computer Society Press. pp. 578-583, Vol. 2, 11-15 August 2002. Québec, Canada.
20. G. Olague and R. Mohr. "Optimal Camera Placement for Accurate Reconstruction". Pattern Recognition, Vol. 35(4), pp. 927-944, 2002.
21. W. H. Press, B. P. Flanery, S. A. Teukolsky and W. T. Vetterling. "Numerical Recipes in C". Cambridge University Press, Second Edition. 1992.
22. K. Rohr. "Recognizing Corners by Fitting Parametric Models". *International Journal of Computer Vision.* 9(3), pp. 213-230, Kluwer Academic Publishers, 1992.
23. P. L. Rosin. "Augmenting Corner Descriptors". *Graphical Models and Image Processing.* Vol. 58, No. 3, May, pp. 286-294, 1996.
24. Sohn, K., Kim, J. H., Alexander, W. E. "A Mean Field Annealing Aproach to Robust Corner Detection." *IEEE Transactions on System, Man, and Cybernetics-Part B* 28, 82-90. 1998.
25. Tsai, D.-M., Hou, H.-T., Su, H.-J. "Boundary-base Corner Detection using Eigenvalues of Covariance Matrices." *Pattern Recognition Letters* 20, 31-40. Elsevier, 1999.
26. Wang, H. and Brady M., "Real-time Corner Detection Algorithms for motion Estimation." *Image and Vision Computing.* 13 (9). 1995.
27. Z. Zheng, H. Wang and E. K. Teoh. "Analysis of Gray Level Corner Detection". *Pattern Recognition Letters.* 20, pp. 149-162, Elsevier, 1999.