Improved Entropic Edge-Detection[1]

Juan Francisco Gómez Lopera
Dpt. Física Aplicada. Universidad de Granada
Campus Fuente Nueva. 18071 Granada. Spain
jfgomez@ugr.es

Naima Ilhami
Dpt. Física Aplicada. Universidad de Granada
Campus Fuente Nueva. 18071 Granada. Spain
naima@ugr.es

Pedro Luis Luque Escamilla
Dpt. Ingeniería Mecánica y Minera. Universidad de Jaén
C/ Alfonso X El Sabio 28. 23700 Linares. Jaén. Spain
peter@ujaen.es

José Martínez Aroza
Dept. Matemática Aplicada. Universidad de Granada
Av. Fuentenueva s/n. 18071 Granada. Spain
jmaroza@ugr.es

Ramón Román Roldán
Dpt. Física Aplicada. Universidad de Granada
Campus Fuente Nueva. 18071 Granada. Spain
rramon@ugr.es

Abstract

Using the Jensen-Shannon divergence of grey level histograms obtained by sliding a double window over an image, an edge-detector is presented. A new technique for linking unconnected edge points, based on estimated directions, is also described. The method is suitable for textured images and has proved to be fairly robust in the presence of impulsive noise or spurious patterns. An application to images of Diesel spray injection is shown, that may be used to improve combustion in Diesel motors.

1. Introduction

Segmentation can play a major role in image processing, and is used to divide the original image into a set of disjoint parts by means of internal characteristics such as grey scale and texture [1,2]. It usually gives as output a binary image in which the edge points have been highlighted. Segmentation is a mandatory step in many image analysis procedures, and the quality of segmentation is a crucial characteristic [3]. Quality is assessed in two different ways: accuracy of edge detection, and level of connectivity in continuous edges. If some edges are still unconnected after edge detection, then an additional *edge linking* process is required [4,5]. The au-

[1] This work has been partially supported by grant MAR97-0464-C04-02 of Spanish Government

thors present a new method for detection, thinning and linking of edges; this is a development of their earlier edge-detection method and is based on the Jensen-Shannon divergence between histograms provided by a sliding window over an image [6].

The method proposed here is successfully applied to images of Diesel spray injection, which are affected by noise and spurious spatial patterns. The analysis of this kind of images may be used to achieve the combustion in automobiles [7,8,9].

2. Image segmentation by Jensen-Shannon divergence

Jensen-Shannon divergence, as proposed by Lin [10], has proved to be a powerful tool in digital images segmentation [6,11]. It is a measure of the inverse cohesion of a set of probability distributions having the same number of possible realisations:

$$JS_\pi(P_1, P_2, \ldots\ldots, P_r) = H\left(\sum \pi_i P_i\right) - \sum \pi_i H(P_i)$$

where:

- P_1, P_2,..,P_r are discrete probability distributions, such that $P_i = \{P_{i,j} \mid j = 1,\ldots,N\}, i = 1,\ldots,r$.
- $\pi = \{\pi_1, \pi_2, \ldots, \pi_r \mid \pi_i > 0, \sum \pi_i = 1\}$ are the distribution weights for the set of probability distributions. In the present application it has been chosen $\pi_i = 1/r$.
- $H(P_i) = -\sum_j P_{ij} \log P_{ij}$ is the Shannon entropy for distribution P_i.

Divergence grows as differences between its arguments (the probability distributions involved) increase. The application of divergence to edge detection is based on a structured two-step procedure, as follows:

Step 1: calculation of the divergence matrix

Let us consider a window made up of two identical subwindows and sliding down over a straight edge between two different textures (Fig.1). It has been shown [11] that in such conditions, the Jensen-Shannon divergence between the normalised histograms of the subwindows reaches its maximum value when each subwindow lies completely within one texture (central case in Fig. 1).

If the window-to-edge direction is not perpendicular, the JS maximum reaches lower values that may make it undetectable: it may be close to zero or to the base value for a given texture. This then means trying several window orientations for each pixel. Only four orientations are, however, technically possible: vertical, horizontal, and two diagonals. In this study, the maximum of these four was taken as the final divergence value to be assigned to each pixel. A matrix of real numbers — the *divergence matrix*[2] — was then built, and pixels displaying high values became edge-point candidate.

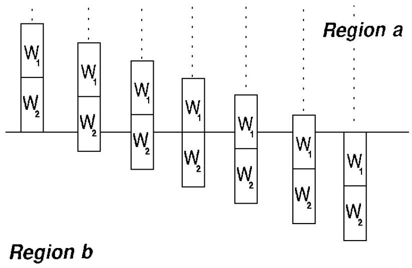

Figure 1: A window sliding across a perfect edge.

Step 2: obtaining the edge points.

We have to decide which pixels from the divergence matrix are edge pixels. Thresholding of the divergence matrix [6,11] is not always useful, since maximum divergence values depend on the composition of adjacent textures and will thus vary according to texture. Thus it would seem more appropriate to use a local criterion. Accordingly, each edge-pixel candidate is the centre of an odd-length monodimensional window, for each of the four possible orientations. For that pixel to be qualified as an edge pixel it has to satisfy, in at a least one direction, that

$$\left|JS_{central} - JS_j\right| \leq T_d \qquad (1)$$

for any other pixel j in that particular monodimensional window, where T_d is a user-defined parameter. In other words, the pixels marked as edge points are local maxima of the divergence matrix, in at least one of the four basic directions. Detection results will depend directly on the parameter T_d.

However, small fluctuations, often due to noise in the original image or to texture regularity, may introduce a great number of false maxima, although these are usually fairly low. Thus repeteadly applying a mean filter before looking for the maxima smoothes the divergence matrix out.

Selection of a local maximum is, in sense, a thinning procedure since just one pixel will usually be detected as an edge point within the neighbourhood, as determined by the size of the window. In fact, rarely more than one pixel would share the same maximum divergence. Nevertheless, a thinning process is achieved for being sure of it.

[2] The use of window (rectangular in the present study) imposes an exclusion frame comprising pixels near the outside borders of the image and whose divergence cannot be calculated.

3. Edge prolongation

The two steps described above enable us to extract the edge pixels from an image. It is not always possible, however, to establish a good compromise between the quality of the binary image obtained and the desired connectivity of the edge pixels. One reason for edge connection to be difficult might be the presence of noise in the original image.

In order to deal with these problems, a third step may be added: edge pixel prolongation. It attempts to join the various sets of edge pixels, using information from the divergence matrix associated with the image, together with knowledge of the direction in which maximum divergence is produced.

In fact it is possible to label each pixel with an estimation of the edge direction as it passes through that pixel. This direction is written as $\alpha \in [0,\pi)$ and is estimated from the JS_1, JS_2, JS_3 and JS_4 divergences calculated from the four windows, which are oriented according to the angle formed by the window axis (the line dividing the two subwindows) and the horizontal. These angles are $\{0, \pi/4, \pi/2, 3\pi/4\}$, respectively. The problem is solved by individually analysing each of the intervals $[0, \pi/4]$, $[\pi/4, \pi/2]$, $[\pi/2, 3\pi/4]$ and $[3\pi/4, \pi)$. For each interval we calculate, by least squares, a second-degree polynomial in α which attempts to find the best fit for the values of JS_1 (α=0), JS_2 ($\alpha=\pi/4$), JS_3 ($\alpha=\pi/2$) and JS_4 ($\alpha=3\pi/4$). Continuity and derivability are required for each interval bound, bearing in mind that the orientations α=0 and $\alpha=\pi$ are identical in this study. Finally, an estimated expression for divergence as a function of α is obtained for each interval.

Yet the problem that really interests us is quite the opposite: given $\{JS_1,JS_2,JS_3,JS_4\}$, which direction produces the maximum JS? It is thus a matter of calculating for each interval the value of α that maximises JS. Results obtained from the foregoing polynomial are shown below, and may be grouped by differences in divergence, hence:

1) if $(JS_1 - JS_3 \geq 0)$ and $(JS_2 - JS_4 \geq 0)$ $\quad \alpha = \pi \dfrac{JS_2 - JS_4}{4\left[(JS_1 - JS_3) - (JS_2 - JS_4)\right]} \in [0, \pi/4]$

2) if $(JS_1 - JS_3 \geq 0)$ and $(JS_2 - JS_4 \leq 0)$ $\quad \alpha = \pi \dfrac{4\ (JS_1 - JS_3) - 3(JS_2 - JS_4)}{4\left[(JS_1 - JS_3) - (JS_2 - JS_4)\right]} \in [3\pi/4, \pi)$

3) if $(JS_1 - JS_3 \leq 0)$ and $(JS_1 - JS_3 \geq 0)$ $\quad \alpha = \pi \dfrac{2\ (JS_1 - JS_3) - (JS_2 - JS_4)}{4\left[(JS_1 - JS_3) - (JS_2 - JS_4)\right]} \in [\pi/4, \pi/2]$

4) if $(JS_1 - JS_3 \leq 0)$ and $(JS_1 - JS_3 \leq 0)$ $\quad \alpha = \pi \dfrac{2\left(JS_1 - JS_3\right) + 3\ (JS_2 - JS_4)}{4\left[(JS_1 - JS_3) - (JS_2 - JS_4)\right]} \in [\pi/2, 3\pi/4]$

It has been shown empirically that the error between the estimated value of α and its actual value is never greater than 0.004. It was impossible to perform the theoretical calculation due to its difficulty. However, the exact value can be obtained in cases where the edge is oriented in any direction of the type $\alpha = k\pi/8$, k={0,1,...,7}.

Given this result, the value of the angle α may be estimated, knowing the JS values for each of the four window orientation. Two real numbers may thus be assigned to each point in the image: the maximum divergence value and the value of the direction producing the maximum. This set of real numbers can be stored in a matrix, called *direction matrix*[3].

The information contained in the direction matrix may be used for extracting edge-pixels from those which have not been marked since they did not satisfy the condition (1), but were close to doing so. Not all the pixels in the image are candidates for filling the gaps, only those classified as *end points*. The definition of end point in [12] includes several variants that may influence the result of the prolongation process. The present paper uses the definition of end point as that having one or two joint marked pixels in a 3×3 centered window. For a pixel to be selected as a continuation of the edge, it must satisfy two conditions:

1. Its associated divergence must be reasonably high, such that

$$JS_{end} - JS_{neighbour} \leq \tau_d$$

where τ_d is a user defined parameter, and has at first no relation with the parameter T_d of the step 2.

2. The mean of the edge direction at the end point and that of the neighbouring pixel under examination must not differ by more than a specified amount from the direction of the physical line joining them.

$$\left| Dir_{(end,\ \mathrm{neighbour})} - (\mathrm{Dir}_{\mathrm{end}} + Dir_{neighbour})/2 \right| \leq \tau_\theta$$

where τ_θ is a user defined parameter.

The two foregoing conditions are used in an attempt to select as edge pixels those lying next to end points and whose divergence and direction are sufficiently close to those of the end point to be extended. It should be borne in mind that when a new pixel is marked as an edge, other adjacent pixels may then become end points, so the algorithm must foresee this event in order to continue the procedure.

4. Linking edges

The two steps above, maximums detection and prolongation, may have spurious effects. Although the method is very robust against noise [11] some pixels can satisfy condition (1) because of statistical fluctuations. Then the output of the present algorithm could present some groups of individual pixels as edge ones.

On the other hand, the prolongation procedure tries to

---

3 As in the case of the divergence matrix (see footnote 2), the use of a rectangular sliding window imposes an exclusion frame.

connect the end points of an edge to those adjacent pixels of highest divergence and with the appropriate direction. But it may occur that these adjacent pixels all have similar characteristics, so the algorithm may join one to another and reach the initial end point at last, giving a loop as a result. This prevent the edge to be closed.

In order to deal with these problems, a final step is performed. It is an iterative process that consists in four phases. First of all, detect and delete all groups of isolated pixels; then it detects and cut the loops of a certain user-dependent size. Finally it tries to link again the end points in a more forced way than in Section 3. If the linking is definitively not possible, all not closed lines are removed. The iterations end when no more loops and isolated pixels are found.

The detection and deletion of groups of individual pixels is made locally. A window of certain user-defined size is centred on each edge pixel: if no other edge pixel is found, it is considered as isolated and then it is eliminated. The forced linking is made now by connecting each end point to the closer one inside an user-defined interval. This interval has semicircle size and is oriented in the direction of the edge. If no other edge-pixel is detected in that interval, then the corresponding line will not be closed, so it is removed until an edge-crossing is found.

5 Results

The present algorithm has been proven with Diesel spray injection images, in order to show the behaviour of the edge detector in an industrial, non-ideal, application. The need for minimising fuel use and emissions (noise and contaminants) in automotive motors requires an exhaustive investigation of injection systems. It is possible to make Diesel spray combustion models from the knowledge of some macroscopic parameters such as aperture angle, penetration, or volume. These values may be measured automatically from images, such as Fig 2. However, this kind of images are affected by noise and spatial patterns due to CCD characteristics, vibrations... So, the edge-detection methods used are not very simple, because they need hard preprocessing. In addition, output images may require some postprocessing [7,8,9]. The detector used in this application is based on gradient operators or thresholding [7,8,9]. Here we have used Canny filter and a threshold method [9] that makes use of the mean and standard deviation of the two distributions (background and spray) found in the histogram of the image.

As may be seen, gradient based detectors fail in the detection of the borders, although noise suppression (5×5 median filter) is achieved. The patterns present in the background makes such an edge-detector to give good results. This is true too for our algorithm, although it is minimised because of the characteristics of the Jensen-Shannon divergence. Thresholding method shows good performance, but

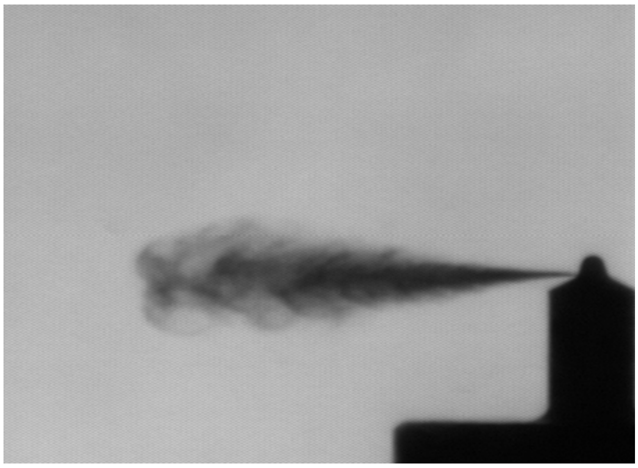

Fig. 2a: original image

Fig. 2b: segmented image (threshold method [9])

Fig. 2c: segmented image with the proposed algorithm.

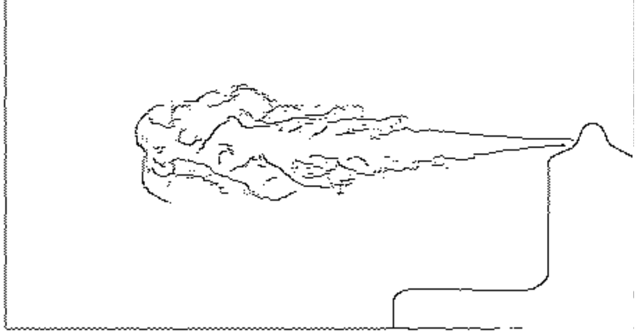

Fig. 2d: segmented image using Canny's method

no edge image is the output (it must be bear in mind that the parameters that are going to measure does not need binary edge image), so the comparison is more difficult. However, it must be said that to obtain the result preprocessing and postprocessing have to be made (suppression of the injector in the image, noise reduction, threshold determination, skeleton [1] detection and selection of the greatest one [9]).

6. Conclusions

This paper has presented a new method for the extraction of edge pixels, using information provided by the Jensen-Shannon divergence of the histograms of two subwindows, which together form a sliding window. The process consists of selecting local divergence maxima as edge pixels. Additionally, a new algorithm is presented for edge linking using the same entropic technique. Results are provided, which show that a combination of the two procedures allows the segmentation of even very noisy and "background-patterned" images to a much higher standard than that of other methods based on the gaussian gradient operator, such as the Canny filter[13], or a thresholding method used in the specialised literature[1][2].

Acknowledgements

We are grateful to Alberto Palomares Chust, from the Universidad Politécnica de Valencia, for his gently offer of the Diesel spray images.

References

[1] Pratt, W. *"Digital Image Processing"* Wiley-Intrescience, 1991.
[2] Gonzalez, R. *"Digital Image Processing"* Addison, 1992.
[3] Zhang, J, Y. *"A survey on evaluation methods for Image Segmentation"* Pattern Recognition **29** 8, 1335-1346, 1996.
[4] Farag, A, A. *"Edge Linking by Sequential Search"* Pattern Recognition **28** 5, 1995.
[5] Sonka, M. *"Image Processing, Analysis and Machine Vision"* Chapman &Hall, 1993.
[6] V. Barranco López, P.L. Luque Escamilla, J. Martínez Aroza, R.R.Román Roldán. "Entropic Texture Edge Detection for Image Segmentation". *Electronics Letters*, **31** 11, 867-869, 1995.
[7] Lin, S.P.; Reitz, R.D. *"Drop and Spray Formation from Liquid Jet"*. Annu. Rev. Fluid Mech. **30**, 85-105, 1998.
[8] Wu, K.J. et al. *"Measurements of the Spray Angle of Atomising Jets"* J. Fluid Eng. WA/FE-10, 1983.
[9] Pastor, J.V.; Correas, D.; Palomares, A. *"Medida de Penetración y Ángulo en Chorros de Inyección Diesel Mediante Técnicas de Procesamiento de Imágenes"* Anales Ingeniería Mecánica **12**, vol 2, pg 336-341. ISSN 0212-5072
[10] Lin, J. *"Divergence Measures based on the Shannon Entropy"*. IEEE Transactions on Information Theory, **37** 1, 145-150, 1991.
[11] Barranco, Lopez, V.; Luque Escamilla, P.; Martinez Aroza, J.; Román Roldán, R. *"Texture Segmentation based on Information-theoretic Edge Detection Method"*. VI Spanish Symposium on Pattern Recognition, 1995.
[12] Lam, L. *"Thinning Methodologies-A Comprehensive Survey"* IEEE Transactions on Pattern Analysis and Machine Intelligence, **14** 9 , 1992.
[13] Canny, J. *"A computational Approach to Edge Detection"* IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-**8** 6, November, 1986.