Fuzzy Edge Detector Using Entropy Optimization

Madasu Hanmandlu
Dept. of Electrical Engineering I.I.T. Delhi
New Delhi 110016, India.
mhmandlu@ee.iitd.ernet.in

John See
Faculty of Engineering
Multimedia University, Cyberjaya
Selangor D.E. 63100, Malaysia
johnsee@mmu.edu.my

Shantaram Vasikarla
IT Department, American InterContinental University
Los Angeles, CA 90066, USA.
svasikarla@la.aiuniv.edu

Abstract: *This paper proposes a fuzzy-based approach to edge detection in gray-level images. The proposed fuzzy edge detector involves two phases – global contrast intensification and local fuzzy edge detection. In the first phase, a modified Gaussian membership function is chosen to represent each pixel in the fuzzy plane. A global contrast intensification operator, containing three parameters, viz., intensification parameter t, fuzzifier* f_h *and the crossover point* x_c*, is used to enhance the image. The entropy function is optimized to obtain the parameters* f_h *and* x_c *using the gradient descent function before applying the local edge operator in the second phase. The local edge operator is a generalized Gaussian function containing two exponential parameters, α and β. These parameters are obtained by the similar entropy optimization method. By using the proposed technique, a marked visible improvement in the important edges is observed on various test images over common edge detectors.*

Keywords *- Edge detector, fuzzy image processing, image enhancement, entropy, contrast intensification operator, fuzzifier, crossover point, Gaussian membership function*

1. Introduction

In many computer vision and image processing applications, edge detectors are important tools of contour feature extraction. The separation of a scene image into object and background, by tracing the edge between them, is an important step in image interpretation. Therefore, precise edge detection is required for numerous image analysis, evaluation and recognition techniques. In the past, a lot of research has been done in the area of image segmentation in various applications using edge detection.

The underlying idea of most edge detection techniques is by the computation of a local first or second derivative operator, followed by some regularization technique to reduce the effects of noise. Earlier edge detection methods, such as Sobel, Prewitt and Roberts' operator used local gradient method to detect edges along a specified direction. The lack of noise control resulted in their poor performance on blurred or noisy images.

Canny [1] proposed a method to counter noise problems, wherein the image is convolved with the first-order derivatives of Gaussian filter for smoothing in the local gradient direction followed by edge detection by thresholding. Marr and Hildreth [2] proposed an algorithm that finds edges at the zero-crossings of the image Laplacian. Non-linear filtering techniques for edge detection also saw much advancement through the SUSAN method [3], which works by associating a small area of neighboring pixels with similar brightness to each center pixel.

More recently, techniques have been proposed that characterize edge detection as a fuzzy reasoning problem. Fuzzy logic by the local approach has been used in [4] for morphological edge extraction method. Ho *et al.* [5] used both global and local image information for fuzzy categorization and classification based on edges.

In this paper, we have proposed a fuzzy-based approach to edge detection that uses both global and local image information. Firstly, we used a modified Gaussian membership function to represent each pixel in the fuzzy domain. After which, a global contrast intensification operator is used to enhance the image by adjusting its parameters. In this process, pixels having more edginess will be enhanced while that with the lesser will be decreased. The optimization of the entropy function by gradient descent function produces new optimized parameters of contrast enhancement. The second phase involves the edge detection process with local image information by a local fuzzy mask, similar to the one suggested in [4, 5]. The last step is a simple thresholding method based on experimental observations.

2. Global Contrast Intensification

2.1 Fuzzy image representation

In the representation of a spatial domain image in the fuzzy domain, a gray tone image X of dimension M x N, and L levels, can be considered as an array of fuzzy singleton sets;

$$X = \{(\mu_{mn}, x_{mn});\ m=1,\ldots,M;\ n=1,\ldots,N;\} \quad (1)$$

where each pixel is characterized by the intensity value x_{mn} and its grade of possessing some membership $\mu_{mn}(0 \le \mu_{mn} \le 1)$, relative to some brightness level in the range [0, L-1].

2.2 Histogram-based fuzzy membership function

Fuzzy property can be expressed in terms of continuous function called as membership function. Here, we use a modified Gaussian membership function as suggested in [6], being a simpler transformation function that contains only one fuzzifier f_h, and is given as

$$\mu_{mn} = G(x_{mn}) = e^{[-(x_{\max} - x_{mn})^2 / 2 f_h{}^2]} \quad (2)$$

where G(x_{mn}) is a Gaussian function, and x_{max}, x_{mn} are the maximum and (m,n)th gray values respectively.

A fuzzy histogram is used to obtain the frequency of occurrence of membership functions of gray levels in the fuzzy image. Thus,

$$X = \bigcup \{\mu(x), p(x)\} = \{\mu_{mn} / x_{mn}\}; m = 1,..., M; n = 1,2,..., N \quad (3)$$

where $\mu(x)$ is the membership of pixel with intensity value of x, and $p(x)$ is the number of occurrences of the intensity value x, in image X. The distribution of $p(x)$ is normalized such that

$$\sum_{x=0}^{L-1} p(x) = 1. \quad (4)$$

Here, we propose a histogram-based membership function to represent pixels of the spatial domain in the fuzzy domain by histogram fuzzification function as

$$\mu(k) = e^{[-(x_{\max} - k)^2 / 2 f_h{}^2]} \quad (5)$$

where k is a certain gray value in the range [0, L-1], and the fuzzifier parameter, f_h can be determined as

$$f_h{}^2 = \frac{\sum_{k=0}^{L-1} (x_{\max} - k)^4 p(k)}{2 \sum_{k=0}^{L-1} (x_{\max} - k)^2 p(k)} \quad (6)$$

where p(k) stands for the frequency of occurrence of k in histogram X.

In the fuzzy plane, a contrast-enhanced image is low perception (dark), $\mu \in [0, 0.5]$ or high perception (bright), $\mu \in [0.5, 1]$ values. This leaves pixels near $\mu = 0.5$ having the highest ambiguity and do not belong to either perception class. They can be treated as pixels describing the fuzzy boundary. Thus, a range of values near 0.5 is considered to contain edges.

2.3 Contrast intensification function

Poor contrast of degraded images is usually enhanced by common methods such as histogram equalization stretching and gray level transformations. As image degradation is nonlinear in nature, we use the nonlinear new contrast intensification function introduced in [6], $NINT[\mu(k)]$ with 3 tunable parameters, viz., intensification operator t, fuzzifier f_h and the crossover point x_c, defined as

$$\mu'(k) = NINT[\mu(k)] = \frac{1}{1 + \exp[-t(\mu(k) - x_c)]} \quad (7)$$

where t controls the shape of the sigmoid function and the initial value of x_c is taken as 0.5.

Parameters f_h and x_c are adjusted through $\mu(k)$ while the intensification operator t will be fixed instead to control the level of contrast enhancement in the image. The image information remains unchanged when the value of t is approximately 5.

2.4 Entropy optimization of parameters x_c and f_h

Different types of measures are reported to assess the image quality, which is difficult to be quantified. In this fuzzy-based approach, the entropy of a fuzzy set is a functional to measure the degree of fuzziness of a fuzzy set, giving the value of indefiniteness of an image. Entropy E [7], which is based on Shannon's function S_e, is defined by

$$E = \frac{1}{\ln 2} \sum_{k=0}^{L-1} S_e p(k) \quad (8)$$

where

$$S_e(\mu'(k)) = -\mu'(k) \ln \mu'(k) - (1 - \mu'(k)) \ln(1 - \mu'(k)) \quad (0 \le \mu' \le 1) \quad (9)$$

is Shannon's entropy function [8]. Thereafter, the fuzzy entropy function can be rewritten as

$$E(\mu'(k)) = \frac{1}{\ln 2} \sum_{k=0}^{L-1} -[\mu'(k) \ln \mu'(k) + (1 - \mu'(k)) \ln(1 - \mu'(k))] p(k) \quad (10)$$

Since E provides the basis on which the information can be quantified, we use to optimize the parameters x_c and f_h. The intensification operator t is fixed and will not be optimized. We propose an entropy optimization method with the pre-set initial values of x_c and f_h.

The derivatives of E with respect to x_c and f_h are obtained as:

$$\frac{\partial E}{\partial \mathrm{x_c}} = \frac{\partial E}{\partial \mu'(k)} \frac{\partial \mu'(k)}{\partial \mathrm{x_c}} = \frac{1}{\ln 2} \sum_{k=0}^{L-1} [t^2(\mu(k) - \mathrm{x_c}) g(\mu')] p(k) \tag{11}$$

$$\frac{\partial E}{\partial f_h} = \frac{\partial E}{\partial \mu'(k)} \frac{\partial \mu'(k)}{\partial \mu(k)} \frac{\partial \mu(k)}{\partial f_h} = \frac{1}{\ln 2} \sum_{k=0}^{L-1} [\frac{t^2 \mu(k)(\mu(k) - \mathrm{x_c})(\mathrm{x_{max}} - k)^2 g(\mu')}{f_h^{\,3}}] p(k) \tag{12}$$

where $g(\mu')$ is denoted as

$$g(\mu') = \mu'(k)(1 - \mu'(k)) = \frac{e^{-t(\mu(k) - x_c)}}{[1 + e^{-t(\mu(k) - x_c)}]^2} \tag{13}$$

These derivatives are used in the learning of the parameters x_c and f_h recursively by the gradient descent technique:

$$x_{c,new} = x_{c,old} - \varepsilon_x \frac{\partial E}{\partial x_c} \tag{14}$$

$$f_{h,new} = f_{h,old} - \varepsilon_f \frac{\partial E}{\partial f_h} \tag{15}$$

where ε_x and ε_f are learning factors or learning rates for both parameters x_c and f_h respectively. If x_c and f_h diverge or converge too quickly, the value of ε_x and ε_f have to be altered respectively in order that the convergence of these values is ensured.

We note that the optimization of x_c moves in both decreasing positive and negative search directions. The nearest optimization point of the both is taken as $x_{c,new}$.

3. Local edge detection

3.1 Local edge detector mask

Redefining contrast intensification function, *NINT*(.) in terms of (m,n)th pixel,

$$\mu'(m,n) = NINT[\mu_{mn}] = \frac{1}{1 + \exp[-t(\mu_{mn} - x_c)]} \tag{16}$$

A fuzzy parameter-based new Gaussian-type edge detector is proposed as

$$\eta(m,n) = e^{-\sum_i \sum_j [\mu'(m+i,n+j) - \mu'(m,n)]^{\alpha} \big/ 2(f_h)^{\beta}} \tag{17}$$

where $i, j \in [-(w-1)/2, (w-1)/2]$, and w x w is the size of the edge detector mask. $\mu'(m,n)$ is the membership value of central pixel of the mask at location (m,n) and $\eta(m,n)$ is the output edge pixel replacing the previous central pixel. The fuzzifier f_h is earlier optimized using equation (15).

Parameters α and β are adjustable and are pre-selected by experiments. As the mask is a generalized Gaussian function, different values of α and β would yield different functions, i.e. selecting $\alpha = \beta = 1$ would produce an exponential mask, while $\alpha = \beta = 2$ would yield a normal Gaussian mask.

The operation performed by the mask is a nonlinear mapping process and the output pixel value $\eta(m,n) \in [-\infty,\infty]$, though in general, the value of $\eta(m,n)$ lies in [0,1].

3.2 Entropy optimization of parameters α and β

At the local window, optimization is also required to fine-tune parameters α and β, as the final edge output depends very much on the values of these two parameters.

Taking into consideration that the edge mask is applied locally and does not involve the entire image, the entropy function is taken as

$$E(\eta(m,n)) = -[\eta(m,n)\ln\eta(m,n) + (1-\eta(m,n))\ln(1-\eta(m,n))] \tag{18}$$

where the global membership value, $\mu(k)$ is now replaced by the local edge pixel $\eta(m,n)$.

The derivatives of E with respect to α and β are obtained as:

$$\frac{\partial E}{\partial \alpha} = \frac{\partial E}{\partial \eta(m,n)} \frac{\partial \eta(m,n)}{\partial \alpha} = \frac{\eta \sum_i \sum_j K^{\alpha} \ln K}{2(f_h)^{\beta}} \ln\left\{\frac{\eta}{1-\eta}\right\} \tag{19}$$

$$\frac{\partial E}{\partial \beta} = \frac{\partial E}{\partial \eta(m,n)} \frac{\partial \eta(m,n)}{\partial \beta} = \frac{\eta \ln f_h \sum_i \sum_j K^{\alpha}}{2(f_h)^{\beta}} \ln\left\{\frac{\eta}{1-\eta}\right\} \tag{20}$$

where $K = [\mu'(m+i, n+j) - \mu'(x,y)]$

These derivatives are used in the learning of the parameters x_c and f_h recursively by the gradient descent technique:

$$\alpha_{new} = \alpha_{old} - \varepsilon_\alpha \frac{\partial E}{\partial \alpha} \tag{21}$$

$$\beta_{new} = \beta_{old} - \varepsilon_\beta \frac{\partial E}{\partial \beta} \tag{22}$$

where ε_α and ε_β are learning factors for both parameters α and β respectively. Similarly, if α and β diverge or converge too quickly, the value of ε_α and ε_β have to be altered respectively to ensure stability.

Since the optimization formulae might be burdensome, we may not use all points *(m,n)* on the image. We proposed using only the maximum and minimum intensity points or a selection of points to represent different intensity ranges.

Some conditions and assumptions are needed to monitor the convergence of these values and prevent optimization process from yielding local minima or maxima. The following are the selection criteria and feasible range of values for α and β:

- $\alpha_{new} \geq 1$ and $\alpha_{new} \leq \alpha_{old}+0.2$
- $\beta_{new} \geq \beta_{old}/2$ and $\beta_{new} \leq \alpha_{old}$

If the value of α and β converges outside the above range of values, optimization can be discarded, and the old values, α_{old} and β_{old}, are used.

3.3 Removal of strong edges and impulse noise

However, when strong edge and impulse noise are encountered, $\eta(m,n)$ will have either

- very large values of $\eta(m,n) > 1$; or
- very small values of $\eta(m,n) < 0$

Thus, the AND operation is taken to avoid such situations, so that the membership is within [0,1], that is

$$\eta'(m,n) = \min[\eta(m,n)] \approx 1 \text{; when } \eta(m,n) > 1; \text{ AND}$$
$$\eta'(m,n) = \max[\eta(m,n)] \approx 0 \text{; when } \eta(m,n) < 0; \tag{23}$$

3.4 Edge image thresholding

After the edge image is produced through the edge detector, simple thresholding is required to binarize it according to a certain threshold level. An optimum threshold level λ is determined through experiments to be in the range of 0.7 to 0.9, where

$$\eta'(m,n) = \begin{cases} 1, & \lambda \geq 0.7 \to 0.9 \\ 0, & \lambda < 0.7 \to 0.9 \end{cases} \tag{24}$$

4. Proposed algorithm

The steps of the fuzzy-based edge detection algorithm are as follows:

Global level:

1. Calculate the value of the parameter f_h from (6).
2. Calculate the membership function $\mu(k)$ from (5) for $k=1,\ldots,L-1$.
3. Determine $\mu'(k)$ by performing contrast intensification, NINT(.) on all membership values
4. Optimize x_c iteratively until it converges.
5. Optimize f_h (using optimized x_c) iteratively until it converges.

Local level:

6. Optimize α iteratively until it converges.
7. Optimize β iteratively until it converges.
8. Apply the local fuzzy edge detector, $\eta(m,n)$.
9. Remove strong edges and impulse noise to produce filtered edge image, $\eta'(m,n)$.
10. Apply simple edge thresholding according to the selected threshold level, λ.
11. Assess final edge image visually.

5. Results and Discussions

The fuzzy edge detector algorithm is implemented on 3 common 256x256 pixel gray level test images, i.e. the Lena, Cameraman and Barbara images. Prior to the application of this algorithm, no pre-processing was done on these images.

As the algorithm has two phases – global enhancement phase and local detection phase, we present the results of implementation on these images separately. Here, the Lena image is used for visual analysis.

5.1 Experimental results

The enhancement phase, is essential for creating contrast in the images. Lack of contrast often leads to loss of visible information, i.e. sharp details, texture. Generally, enhancement is not part of many existing edge detectors in the literature. This is the reason they do not fare well when it comes to the retaining of the basic shape of the object. This added feature to the proposed edge detector makes it different from other available edge detectors.

Enhancement is performed by a contrast intensification operator that contains 3 parameters, x_c, f_h, t. Generally after optimization, x_c varies around 0.5, and may reach up to ≈ 0.7 in some cases. The initial value of f_h is found from the approximate formula (6). However, it usually does not change much after optimization, with possible changes of ± 1.0. The value of t is pre-set and by experimentation, values of $5 < t < 7$ allows reasonable amount of contrast enhancement on these images. With these initial values, the entropy optimization yields the final values as shown in Table 1.

Table 1. Entropy and contrast intensification parameter values for the Lena, Cameraman, and Barbara images

Image	Stage*	E	f_h	t preset	x_c
Lena 256x256	I	0.6719	116.5739	5.00	0.5000
	EU	0.5941	120.5549	5.00	0.5000
	EO	0.6354	122.2986	5.00	0.5783
Cameraman 256x256	I	0.5490	120.2502	5.00	0.5000
	EU	0.5641	96.0800	5.00	0.5000
	EO	0.7800	108.7025	5.00	0.6931
Barbara 256x256	I	0.6596	116.7062	5.00	0.5000
	EU	0.6249	104.3196	5.00	0.5000
	EO	0.7069	110.3554	5.00	0.5974

**I: Initial; EU: Enhanced, unoptimized; EO: Enhanced, optimized*

The original and enhanced image of Lena is shown in Fig. 1. As can be seen, the enhanced image appears to be more suitable for the detection of edges as there is sufficient contrast at the weak edges.

Fig. 1. The original (left) and, enhanced and parameter-optimized (right) Lena image with $t = 5$.

In the local phase, edge detection is performed by the proposed edge operator, which contains 2 tunable exponential parameters, α and β. The α gives a measure of edge strength, with higher values yielding prominent edges while discarding weak ones, and vice versa. The β gives the scale of visibility of edges in the image, with lower values of β showing more fine textured detail, and vice versa. It is also found that β should always be less than or equal to α, i.e. $\beta \leq \alpha$.

By experiments, suitable values of α are in the range of 2.5 to 3.2, where values above or below this range will result in thick, lumpy edges. Similarly, β values range from 2 to 3.2, where β below 2 will result in much unwanted texture and speckled edges. With the initial values of α and β fixed, implementation of the edge detection operator yields the new optimized α and β values for the test images, as shown in Table 2. The Lena edge images before and after parameter-optimization are shown in Fig. 2.

Table 2. Initial and new optimized values for parameters α and β for the Lena, Cameraman, and Barbara images using optimized enhancement parameters from Table 1

Image	α (initial)	α (new)	β (initial)	β (new)
Lena ($\lambda = 0.75$) $\varepsilon_\alpha = 0.001$, $\varepsilon_\beta = 0.0001$	2.9500	2.9819	2.4000	2.4000
Cameraman ($\lambda = 0.90$) $\varepsilon_\alpha = 0.1$, $\varepsilon_\beta = 0.1$	2.9000	2.8987	2.5000	2.5126
Barbara ($\lambda = 0.90$) $\varepsilon_\alpha = 0.01$, $\varepsilon_\beta = 0.2$	3.2000	2.7706	2.6000	2.6000

Fig. 2. The initial (left) and parameter-optimized (right) edge image for Lena with $\alpha_{initial}$=2.95, $\beta_{initial}$=2.4, λ = 0.75.

By visual assessment, the basic shape is retained in all the tested images, with a good balance of detailed and speckle-textured edges. For comparison purpose, we have applied the Canny, and scale-space Gaussian gradient diffusion [9] operators on the test images. Results are shown in Fig. 3.

Now, Figs. 2 and 3 reveal much difference in the appearance of Lena. Canny yields more unnecessary edges thus cluttering the main shape details whereas the proposed detector drops some edges in favor of important edges that make up the shape. The Gaussian gradient diffusion operator causes blurred edges, and mainly accentuates only the strong edges but most valuable details of the face are lost. This distinctive feature of the proposed detector goes a long way as an operator suitable for face recognition applications.

Fig. 3. Canny edge operator (left) and Gaussian gradient diffusion edge operator (right)

5.2 Discussion of results

Though the performance of the proposed fuzzy edge detector excels as a shape and detail detector, it is fraught with some drawbacks. It fails to provide all thinned edges. The presence of thick edges at some locations needs to be addressed by the proper choice of parameters. The weak edges are not eliminated but for some applications, these may be required. This detector has another distinctive feature, i.e. it retains the texture of the original image. This feature can be utilized for the identification of fingerprints, where the ridges may have different intensities. As far as the parameters of edge detection operator are concerned, optimization has to be simplified by devising simple solutions for the estimation of these parameters. Furthermore, by the determination of proper initial values for α and β, optimization work could be lightened. As the success of the edge detection depends on these parameters, we are experimenting on several images to come up with a useful selection guideline.

6. Conclusions

The fuzzy edge detector presented in this paper uses both global (histogram of gray levels) and local (membership function in a window) information. The local information is fuzzified using a modified Gaussian membership function. Using the contrast intensification operator, the image is enhanced to the required level of visual quality by the entropy optimization of parameters f_h and x_c. Then, the local edge detection operator is applied on the enhanced image using parameters α and β, which are again obtained from entropy optimization. Finally, simple edge thresholding is applied to produce the final edge image.

Results show that this edge detector is immensely suitable for applications such as face recognition and fingerprint identification, as it does not distort the shape and is able to retain the important edges unlike the Canny edge detector. Choice of some of the parameters, t, α and β is crucial for the success of this algorithm.

References

[1] Canny, J.F., "A computational approach to edge detection", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 8(6), 1986, 679-698.

[2] Marr, D., and Hildreth, E.C., "Theory of edge detection", *Proc. of the Royal Society of London*, b207, 1980, 187-217.

[3] Smith, S.M., and Brady, J.M., "SUSAN – A new approach to low level image processing", *International Journal of Computer Vision*, 23(1), 1997.

[4] Bloch I., "Fuzzy sets in image processing", *ACM Symposium on Allied Computing, 1994*.

[5] Ho, K.H.L., and Ohnishi, N., "FEDGE – Fuzzy edge detection by fuzzy categorization and classification of edges", *Fuzzy Logic in Artificial Intelligence*, Springer, IJCAI'95 Workshop, Montreal, Canada, 1188:182-196.

[6] M. Hanmandlu, D. Jha, and R. Sharma, "Color image enhancement using fuzzification", Pattern Recognition Letters, Vol. 24(1-3) 2003, pp. 81-87. [3]

[7] Chi, Z.R., Yan, H., Pham, T., *Fuzzy Algorithms: with applications to image processing and pattern recognition*, World Scientific, Singapore, River Edge, N.J., 1996.

[8] Pal N.R. and Pal S.K., "Entropy: A new definition and its applications, *IEEE Trans. Sys., Man, Cybernetics*, SMC-21(5).

[9] ter Haar Romeny, B.M., *Front-End Vision and Multi-Scale Image Analysis*, Kluwer Academic Publishers, 2002.