A Novel Local Thresholding Technique in PSO for Detecting Continuous Edges in Noisy Images

Mahdi Setayesh[1], Mengjie Zhang[1] and Mark Johnston[2]
Evolutionary Computation Research Group
[1] School of Engineering and Computer Science
[2] School of Mathematics, Statistics and Operations Research
Victoria University of Wellington, PO Box 600, Wellington 6140, New Zealand
{Mahdi.Setayesh, Mengjie.Zhang}@ecs.vuw.ac.nz, Mark.Johnston@msor.vuw.ac.nz

***Abstract*—In this paper, a novel local thresholding technique is proposed for a particle swarm optimisation (PSO)-based algorithm to detect edges with greater continuity. The new technique is based on the Sauvola-Pietkinen method which is often used for binarising the illuminated document images but normally cannot be applied to edge magnitude images. This method is equipped by an integral imaging technique for more efficiency and adopted into the PSO-based algorithm to detect edges in grey level noisy images. We compare the performance of the new algorithm with our previous PSO-based edge detector utilising Otsu's method which is commonly used as a thresholding technique in edge detection. Experimental results show that the PSO-based algorithm utilising the new local thresholding technique performs better than the one that uses Otsu's method.**

I. Introduction

The continuity of the edges recognised by an edge detection algorithm is very important in some applications in noisy environments such as medical imaging [1] and remote sensing [2]. Many edge detection algorithms have been proposed to detect edges in noisy images such as Gaussian-based [3], statistical-based [4], curve fitting-based [5], soft computing-based [6] and scale space-based [7] edge detectors. The performance of these algorithms depends on the various environmental conditions such as illumination and noise, and most of them produce broken edges in illuminated and noisy images [8] [2].

Particle Swarm Optimisation (PSO) was introduced by Kennedy and Eberhart in 1995 [9] and it has been successfully applied to solve global optimisation problems. To compensate for broken edges in noisy images, we previously proposed a PSO-based algorithm to detect edges in noisy images and compensate for broken edges and examined its performance in the noisy binary images containing simple shapes [10]. The performance was satisfactory in such images but it could not operate well on grey level images. We then developed a new encoding scheme and a fitness function and compared its performance on real images corrupted by two different types of noise (Gaussian and impulsive) [11]. We later compared its localisation accuracy with a modified version of the Canny algorithm as a Gaussian filter-based edge detector equipped with a static hysteresis thresholding technique [12] to detect continuous edges and applying a non-maxima suppression (NMS) technique to detect thin edges [13]. The accuracy of the PSO-based algorithm was higher than Canny but lower than the robust rank-order (RRO) as a statistical-based edge detector in most cases especially in the images corrupted by impulsive noise. We enlarged the size of the area evolved by the PSO-based algorithm and considered almost all possible edge patterns on a real continuous edge to increase the localisation accuracy of the algorithm [6]. In these papers, we demonstrated that the PSO-based algorithm can work better than Canny and RRO while Canny produced many speckles and broken edges in noisy images and RRO recognised edges more thickly than the others. However Canny and RRO operated better than PSO in a few cases. Finally, in [14], we equipped the PSO-based algorithm with an adaptive method by use of Otsu's method [15] to estimate one of its parameters. For a fair comparison, we used a dynamic hysteresis thresholding proposed in [16] in order to have better connected edges and an adaptive filter size proposed in [17] in order to overcome noise for the Canny algorithm. We also set the edge-height parameter of the RRO detector to the value which gave a highest localisation accuracy. The results showed that the PSO-based algorithm generally outperforms RRO and Canny in noisy images but there were still broken edges in the noisy image with illuminated areas.

Although Otsu's method is better than other state-of-the-art global binarisation methods [18], it cannot work well in illuminated images [19]. Many local binarisation methods have been proposed to solve this problem by extracting local features for each pixel. Sauvola and Peitikain proposed an adaptive local binarisation method for document images [20]. Even though the performance of this method is higher than other global and local binarisation methods for the illuminated noisy images [19], it is not applicable for binarisation of edge magnitude images resulted by edge detection algorithms. The main goal of this paper is to introduce a novel local thresholding technique by use of the Sauvola-Pietkinen method for the PSO-based edge detector in illuminated noisy images.

The rest of this paper is organised as follows. Section II provides background information on the PSO-based edge detector and two major categories of thresholding techniques. The new local thresholding technique will be introduced in Section III. Sections IV and V presents discussion on experimental results followed by concluding remarks in Section VI.

II. Background

A. Particle Swarm Optimisation

PSO was inspired by the social behaviour of animals such as flocking of birds and schooling of fish. PSO as a population-based optimisation method has been successfully applied to solve global optimisation problems. The population contains m particles which move in an n-dimensional search space according to the PSO velocity and position equations ((1) and (2)).

$$\vec{X}_i(t+1) = \vec{X}_i(t) + \vec{V}_i(t+1). \quad (1)$$

Here, $\vec{X}_i(t)$ and $\vec{V}_i(t)$ are the position and velocity of the i^{th} particle at time t. The velocity is often influenced by three components: current motion influence, particle memory influence, and swarm influence:

$$\begin{aligned} \vec{V}_{i,j}(t+1) &= w\vec{V}_{i,j}(t) + C_1 r_{1,j}(\vec{X}_{pbest_i,j} - \vec{X}_{i,j}(t)) \quad (2) \\ &\quad + C_2 r_{2,j}(\vec{X}_{leader,j} - \vec{X}_{i,j}(t)) \end{aligned}$$

where w is the inertia weight to control the learning factor from the previous movement; C_1 and C_2 are the particle history and swarm influence factors respectively; $r_{1,j}$ and $r_{2,j}$ are uniform random variables between 0 and 1; $\vec{X}_{pbest}$ represents the personal best position of each particle in history; and finally $\vec{X}_{leader}$ is the position of the particle which is defined by a neighbourhood topology and leads other particles in order to explore better regions of the search space.

B. PSO-based Edge Detection Algorithm

Detection of continuous edges in noisy images is the main goal of the our previously developed PSO-based algorithm. In order to reduce broken edges and examine almost all possible edge patterns to increase localisation accuracy of the algorithm, we proposed an encoding scheme for the particles where each particle in the population represents the global structure of a continuous edge [6]. This edge divides an area of an image into two regions: light and dark (see Figure 1(b)), such that it maximises **interset** distance between the average pixel intensities of two regions and minimises **intraset** distances within both regions.

The encoding scheme representing a continuous edge has three components: the offset of the middle pixel (pixel C in Figure 1(b)) of the consecutive pixels on the continuous edge ($\langle o_1, o_2 \rangle$) and two sets of movement direction sequences from the pixel ($\langle m_1, m_2, \ldots, m_L \rangle$ and $\langle m_{L+1}, m_{L+2}, \ldots, m_{2L} \rangle$). The values of the two offsets (o_1 and o_2) are integers ranging from 0 to $SqrSize - 1$ and m_i ranging from 0 to 7. Here, $SqrSize$ is the size of the area which pixel C can be located in. Let m_i show the movement direction from a pixel to one of the eight possible adjacent pixels in its neighbourhood along the continuous edge (see Figure 1(a)). For example, Figure 1(c) shows the particle encoding for the continuous edge in Figure 1(b) where $L = 5$.

At the first step of the algorithm, the intraset and inter-set distances for each single pixel in eight different edge directions (see Figure 1(d)) on an image are calculated by

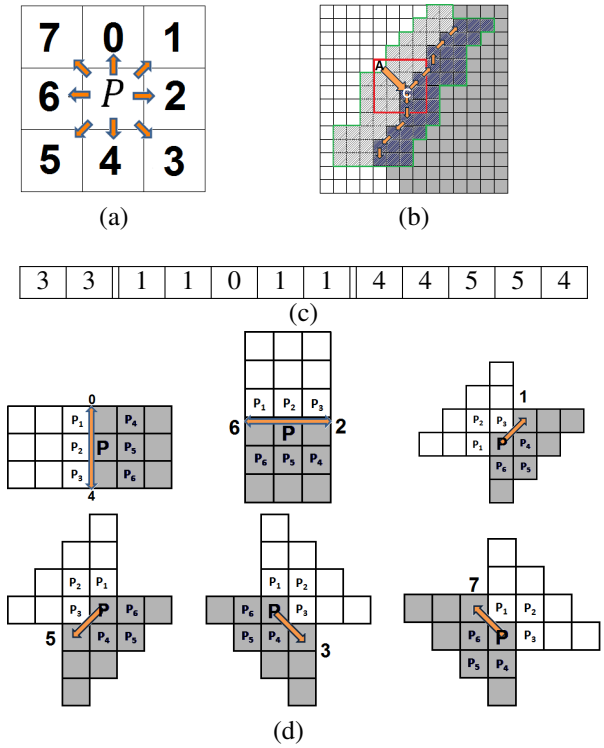


Fig. 1. The particle encoding scheme [6]: (a) eight movement directions from a pixel P; (b) an example of a curve with two regions; (c) the particle representing the curve with $max = 10$; (d) eight moving ways from pixel P to its neighbours.

the equations proposed in [6]. These values, in conjunction with the NMS factor (proposed in [6]), are used to estimate the edge magnitude of each single pixel ($TotalEdgeMag_m$) in edge direction m. Most edge detection algorithms utilise thresholding techniques to identify edges after calculation of edge magnitudes. A sigmoid function as equation (3) is used to scale the total edge magnitude of each pixel between 0 and 1 in order to estimate the possibility score $PScore$ of the pixel (P) lying on an edge in direction m [14]:

$$PScore_m(P) = \frac{1}{1 + e^{-\frac{3.317}{TH}(TotalEdgeMag_m(P) - 0.6229TH)}} \quad (3)$$

where TH is a threshold value between 0 and 1 which is estimated by Otsu's method [15]. In order to estimate TH, the local maximum of the edge magnitude of each pixel, $LocalEdgeMag(P)$, is first calculated by equation (4) proposed in [14]. The result of applying this equation is an edge magnitude image which is used as the input to Otsu's method in order to estimate the value of parameter TH in Equation (3).

$$LocalEdgeMag(P) = \max_{i=1}^{8}(TotalEdgeMag_i(P)) \quad (4)$$

To assess each particle at each generation of the PSO algorithm, the total sum of possibility score ($PScore$) of the pixels along the the continuous edge represented by the

particle, in conjunction with its curvature cost factor proposed in [6], is used to formulate the fitness function of the PSO algorithm.

C. Thresholding Techniques

Thresholding techniques in edge detection can be categorised into two main groups: (a) global thresholding techniques, which apply the features extracted from the whole of an image; and (b) local thresholding techniques, which use local features to choose a threshold value in order to identify edges.

Most thresholding techniques operate as image binarisation methods in the field of image segmentation. In these techniques, edge magnitude images are used as inputs to binarise. Global binarisation techniques such as the methods proposed in [15] and [18] use global features extracted from the whole image to estimate a single threshold value for the image binarisation. Local thresholding techniques such as [21] and [20] utilise local features extracted from a small area around each pixel. The comparison of local binarisation methods experienced in [19] shows that the Sauvola-Pietkinen method proposed in [20] operates better than other types of local and global binarisation methods especially in document images with illuminated areas. This method considers a grey scale document image as an array in which $I_{P'_{x,y}} \in [0, 255]$ is the intensity of pixel P' at position (x, y). In this method, the local threshold $TH_{P'_{x,y}}$ is estimated using the local mean $m_{P'_{x,y}}$ and standard deviation $s_{P'_{x,y}}$ of the pixel intensities in a $WS \times WS$ window centred around pixel $P'_{x,y}$:

$$TH_{P'_{x,y}} = m_{P'_{x,y}} \left[1 + k\left(\frac{s_{P'_{x,y}}}{R} - 1\right)\right] \quad (5)$$

Here, R is the maximum value of the standard deviation whose value for a 8-bit grey level image is 128; k is a real parameter ranging from 0 to 1. This parameter controls the threshold value for pixel $P'_{x,y}$ such that the lower the value of k, the higher the threshold value from the local mean $m_{P'_{x,y}}$.

III. New Local Thresholding Technique

According to the knowledge of authors, the Sauvola-Pietkinen method has never been applied for the binarisation of edge magnitude images. It is because of the different nature of document images and edge magnitude images. Document images are always positive and their background is white whereas edge magnitude images are negative and their background is black. To solve this problem, we first invert the local edge magnitude ($LocalEdgeMag(P_{x,y})$) and calculate its inverse as $1 - LocalEdgeMag(P_{x,y})$, and then scale the result in range $[0, 255]$ by multiplied by 255. Let $I_{P'_{x,y}} = (1 - LocalEdgeMag(P_{x,y})) \times 255$. Therefore, $m_{P'_{x,y}} = (1 - m_{P_{x,y}}) \times 255$ and $s_{P'_{x,y}} = 255 \times s_{P_{x,y}}$ where $m_{P_{x,y}}$ and $s_{P_{x,y}}$ are the local features at pixel $P_{x,y}$ in the edge magnitude image. Thus,

$$TH_{P_{x,y}} = (1 - m_{P_{x,y}}) \left[1 + k\left(\frac{255 s_{P_{x,y}}}{R} - 1\right)\right] \quad (6)$$

where $TH_{P_{x,y}}$ is in the range between 0 and 1. We use the threshold value estimated by equation (6) in equation (3) to calculate the possibility score of each pixel. In this equation, the local mean and standard deviation are used to adapt the value of the threshold according to the magnitude of the edges inside of the local neighbourhood of each pixel. In the case of the neighbourhoods with high edge magnitudes, the threshold $TH_{P_{x,y}}$ is almost equal to $m_{P_{x,y}}$ and in the case of the neighbourhoods with low edge magnitudes, the threshold value is less than the local mean in order to relatively increase the possibility score of the weak edges in these regions.

A. Different Degrees of Integral Images

In order to calculate $TH_{P_{x,y}}$, the local mean and standard deviation should be computed for each pixel. We use the concept of integral image which is an intermediate representation for an image in order to compute rectangle features in computer vision using an efficient way [22]. We generalise this concept to the different integral degrees of an image and show integral degree i (Ω^i) of an edge magnitude image as equation (7):

$$\Omega^i(x, y) = \sum_{i=0}^{x} \sum_{j=0}^{y} LocalEdgeMag^i(P_{x,y}) \quad (7)$$

The integral image can be calculated in an efficient manner by equation (8).

$$\begin{aligned} \Omega^i(x, y) \quad = \quad & \Omega^i(x-1, y) + \Omega^i(x, y-1) - \Omega^i(x-1, y-1) \\ & + LocalEdgeMag^i(P_{x,y}) \end{aligned} \quad (8)$$

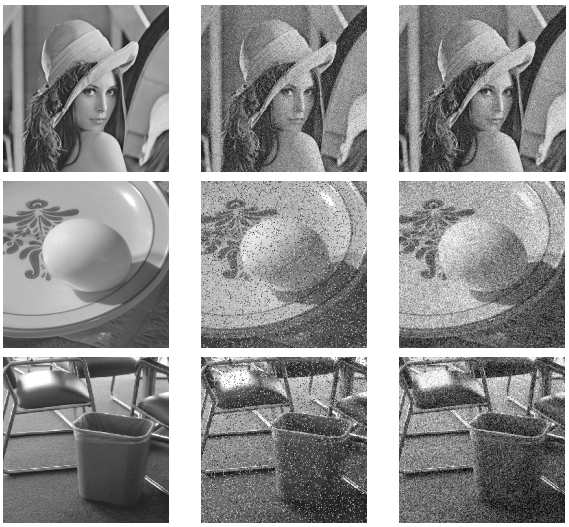
B. Calculation of Local Features using Different Degrees of Integral Images

The local mean and standard deviation can be easily calculated as equations (9) and (10) once the integral images are computed.

$$\begin{aligned} m_{P_{x,y}} = \quad & (\Omega^1(x + (WS-1)/2, y + (WS-1)/2) + \\ & \Omega^1(x - (WS-1)/2, y - (WS-1)/2) - \\ & \Omega^1(x - (WS-1)/2, y + (WS-1)/2) - \\ & \Omega^1(x + (WS-1)/2, y - (WS-1)/2))/WS^2 \end{aligned} \quad (9)$$

Since the variance of a variable is always equal to the expectation of the square of the variable minus the square of the mean of the variable,

$$\begin{aligned} s^2_{P_{x,y}} \quad = \quad & (\Omega^2(x + (WS-1)/2, y + (WS-1)/2) + \\ & \Omega^2(x - (WS-1)/2, y - (WS-1)/2) - \\ & \Omega^2(x - (WS-1)/2, y + (WS-1)/2) - \\ & \Omega^2(x + (WS-1)/2, y - (WS-1)/2))/WS^2 \\ & - m^2_{P_{x,y}} \end{aligned} \quad (10)$$



(a) (b) (c)

Fig. 2. Example images for subjective comparison. (a) Original images Lena, egg, and rubbish bin; (b) images with *impulssive noise* (noise probability=0.1); (c) images with *Gaussian noise* (PSNR=16dB).

IV. Experimental Design

Although the proposed local thresholding technique can be applied to any edge detector which can estimate edge magnitudes of the pixels of an image such as different order derivatives [23] [24], we apply this technique to the PSO-based edge detector which is more insensitive to noise than other edge detectors such as Canny and RRO. To examine the performance of the new local thresholding technique, we compare our previous PSO-based algorithm (PSO1) [14], utilising the sigmoid function with parameter TH estimated by Otsu's method, with the improved PSO-based algorithm (PSO2) utilising the same function with parameter $TH_{P_{x,y}}$ estimated by the proposed method. This section provides the details on the two image sets, objective performance measure, and parameter settings, which are used in the experiments.

A. Image Sets

We use two different image sets in the experiments. The images in the first set are clean and easily accessible through the South Florida University database [25]. Since the main goal of the algorithm is the detection of continuous edges in noisy and illuminated images, we first chose three images from this database with illuminated areas which are commonly used as benchmarks for edge detection: Lena, egg and rubbish bin (see Figure 2). Then these images are corrupted by two different types of noise: impulsive and Gaussian (see the images in Figure 2 in columns (b) and (c)). In these noisy images, the probability of the impulsive noise is 0.1 and the peak-signal to noise ratio (PSNR) is $16dB$ for the Gaussian noise . We will use them for a subjective (qualitative) comparison of PSO1 and PSO2.

The second image set includes the images which were used for an objective (quantitative) comparison. This image set contains four images (Saturn, multi-cube, wall and road) which have been provided by the University of Cordoba (Spain) and their ground truth edge maps are available from [26]. In addition to those four images, we added two images (rubbish-bin and egg) from South Florida University database to the second image set. The ground truth images of these two images are available to download from [27]. The size of each image of the second image set is 256×256 pixels and the resolution of each is 8 bits per pixel. These images along with their ground truth images are shown in Figure 3. We also add two different types of noise in different noise levels to these images in order to investigate the performance of the new algorithm in noisy images. For the images corrupted by the impulsive noise, the noise probability ranges from 0.1 to 0.5 with a step size of 0.05. For the Gaussian noise, PSNR ranges from 0 to $22dB$ with a step size of $1dB$.

B. Quantitative Performance Measure

Pratt's Figure of Merit (PFOM) as a quantitative measure is used to compare the performance of the new algorithm [28]. This measure is defined by Equation (11):

$$R_{PFOM} = \frac{1}{max(I_I, I_A)} \sum_{i=1}^{I_A} \frac{1}{1 + \beta d(i)^2} \qquad (11)$$

where I_I and I_A are the number of ideal and actual edge pixels in the ground truth and the edge map images generated by an edge detector. The function $d(i)$ is the distance between the pixel i in the generated edge map and the closest ideal edge pixel in the ground truth; and the parameter β is a constant scale factor whose typical value is $\frac{1}{9}$.

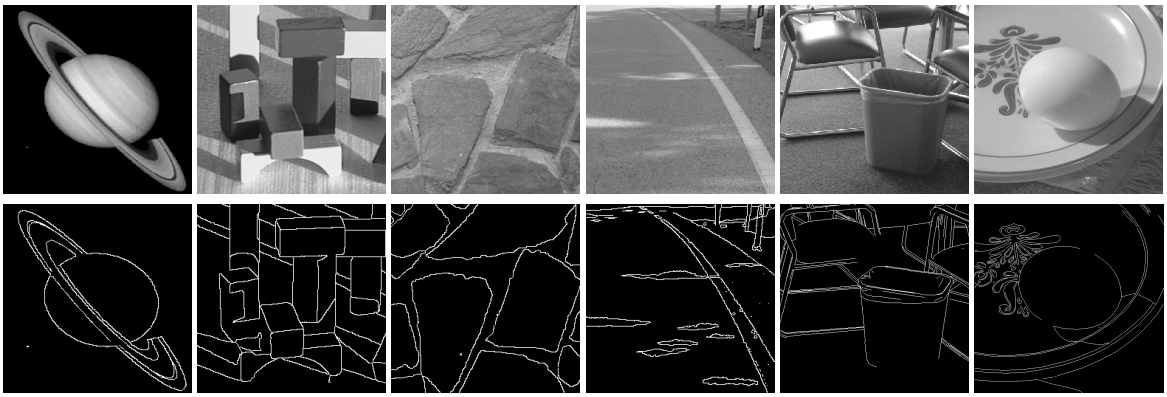
C. Parameter Settings

In PSO1 and PSO2, the population size is 50 and the maximum number of iterations was set at 200 according to the chosen particle length. We used the values $w = 0.7298$, $c_1 = 1.4962$, $c_2 = 1.4962$ for the parameters in Equation (2) [29]. The minimum length of a continuous edge, $2L + 1$ was set at 21 and $SqrSize$ at 6 [14]. The parameters of the novel thresholding techniques, WS and k, were respectively set at 21 and 0.05 [19]. The experiments in [19] showed that a small value of k like 0.05 gives better results with fewer broken edges.

V. Results and Discussion

A. Subjective/Qualitative Comparison

The resulting images are shown in Figures 4 and 5 after applying PSO1 and PSO2 in the first set (Figure 2) corrupted by impulsive and Gaussian noises respectively. As can be seen in Figure 4, the edges detected by PSO2 were improved on Lena's hat especially on the top which is an illuminated area and they are more connected than the edges recognised by PSO1. For the egg image, PSO2 could operate better than PSO1 around the egg. However, there are still broken edges in this area. This problem may be solved by increasing parameter L. Although there are several false edges and still broken



(a) (b) (c) (d) (e) (f)

Fig. 3. Example images for objective comparision. (a)–(d) four real image from the UCO university and their manual ground truth images [26], (e) and (f) two real images from the South Florida University and their ground truth images [27].

edges, PSO2 improved the edges in the rubbish-bin image particularly on its bottom-right corner, .

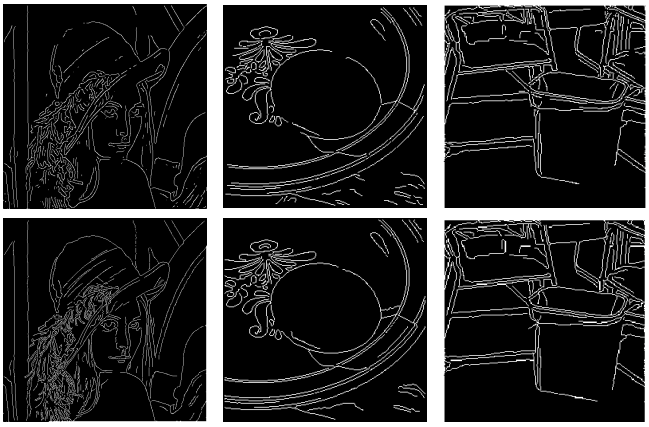


Fig. 4. Subjective results of edge detection produced by PSO1 (top row) and PSO2 (bottom row) on the three images corrupted by *impulsive noise* (noise probability=0.1).

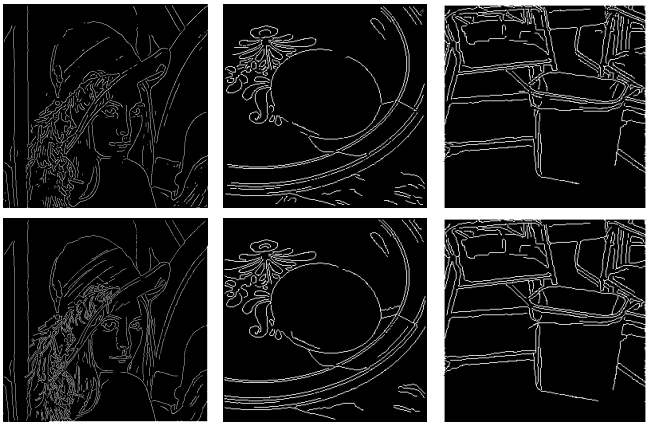


Fig. 5. Subjective results of edge detection produced by PSO1 (top row) and PSO2 (bottom row) on the three images with *Gaussian noise* (PSNR=16dB).

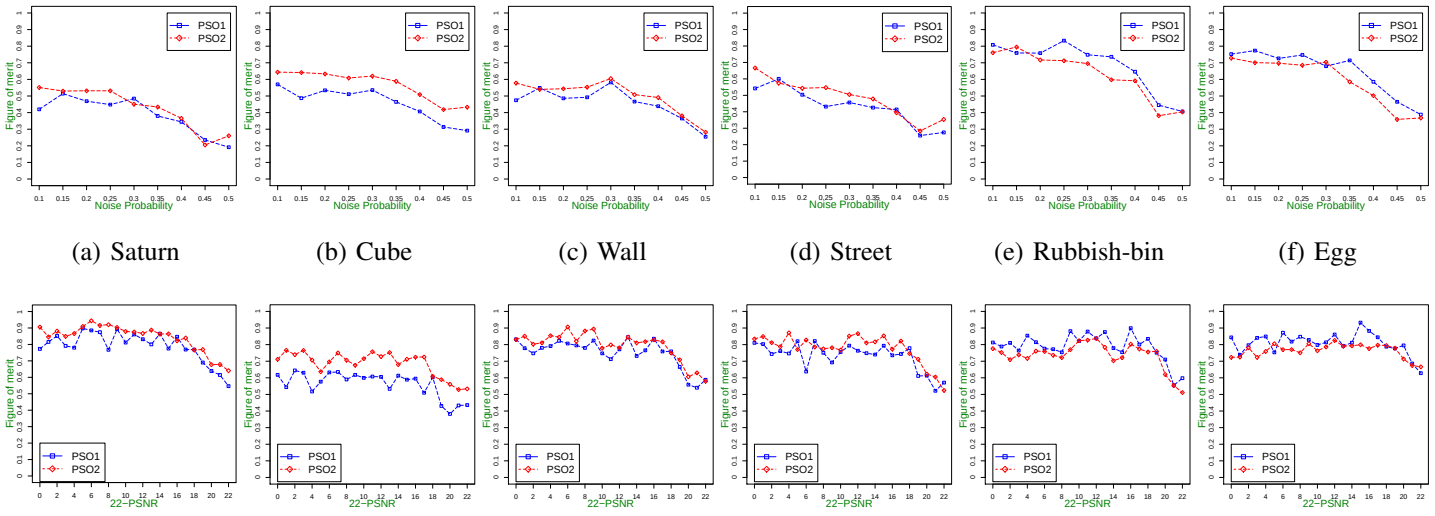
B. Objective/Quantitative Comparison

For an objective comparison of PSO1 and PSO2, the localisation accuracy (PFOM) was calculated from the resulting images after applying both algorithms to the images in the second set (Figure 3) at different noise levels. Figure 6 depicts the graphs in which the average of the resulting PFOM values after 30 runs are plotted versus different noise levels for each image. PSNR ranges from 0 to 22dB with step of $1dB$ and the noise probability ranges from 0.1 to 0.5 with step of 0.05.

The plots in Figure 6 indicate that PSO2 generally performed better than PSO1. Statistical analysis showed that PSO2 has a higher accuracy in 127 cases out of 192. For the first four images, PSO2 could detected the edges more accurately than PSO1 in 115 cases out of 128 (see Figures 6(a), (b), (c), (d), (g), (h), (i) and (j)). For the last two images, its accuracy is lower in 52 cases out of 64 (see Figures 6(e), (f), (k) and (l)). The reason is that there are a few "false positive" edges which were recognised by PSO2 as edges whereas they were actually *incorrectly* labelled as non-edges in their ground truth images. This caused that the real improvements of PSO2 were considered "false". Figures 4 and 5 clearly show that PSO2 can even detect the difficult edges in the the boundary of the egg and rubbish-bin that were not labelled in the ground truth images.

VI. Conclusions

The main goal of this paper was the development of a local thresholding technique for a previously developed PSO-based algorithm to overcome the problem of detection of edges in noisy images with illuminated area. The goal was achieved by borrowing the main idea from the Sauvola-Pietkinen method as a way of binarisation of illuminated document images and adapting this method to the PSO-based algorithm. The experiments showed that the performance of the PSO-based algorithm equipped by the proposed local thresholding technique performs better than the previous PSO-based algorithm in noisy images with illuminated area.



(a) Saturn (b) Cube (c) Wall (d) Street (e) Rubbish-bin (f) Egg

(g) Saturn (h) Cube (i) Wall (j) Street (k) Rubbish-bin (l) Egg

Fig. 6. PFOM vs. noise level for Saturn, cube, wall, street, rubbish-bin and egg images in the second image set. (a)–(f): with different impulsive noise levels (the noise probability ranging from 0.1 to 0.5); and (g)–(l) with different Gaussian noise levels (PSNR ranging from 0 to 22dB).

REFERENCES

[1] M. Gudmundsson, E. El-Kwae, and M. Kabuka, "Edge detection in medical images using a genetic algorithm," *IEEE Transactions on Medical Imaging,*, vol. 17, no. 3, pp. 469–474, 1998.

[2] Y. Jing, J. An, and Z. Liu, "A novel edge detection algorithm based on global minimization active contour model for oil slick infrared aerial image," *IEEE Transactions on Geoscience and Remote Sensing,*, vol. 49, no. 6, pp. 2005–2013, 2011.

[3] M. Basu, "Gaussian-based edge-detection methods: a survey," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 32, no. 3, pp. 252–260, 2002.

[4] D. H. Lim, "Robust edge detection in noisy images," *Comput. Stat. Data Anal.*, vol. 50, no. 3, pp. 803–812, 2006.

[5] G. Chen and Y. Hong Yang, "Edge detection by regularized cubic b-spline fitting," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 25, no. 4, pp. 636–643, 1995.

[6] M. Setayesh, M. Zhang, and M. Johnston, "Detection of continuous, smooth and thin edges in noisy images using constrained particle swarm optimisation," in *GECCO*, 2011, pp. 45–52.

[7] B. Tremblais and B. Augereau, "A fast multiscale edge detection algorithm based on a new edge preserving pde resolution scheme," *International Conference on Pattern Recognition*, vol. 2, pp. 811–814, 2004.

[8] A. Jevtic, I. Melgar, and D. Andina, "Ant based edge linking algorithm," in *35th Annual Conference of IEEE Industrial Electronics*, 2009, pp. 3353–3358.

[9] F. Kennedy, R. Eberhart, and Y. Shi, *Swarm Intelligence*. San Francisco, CA: Morgan Kaufmann, 2001.

[10] M. Setayesh, M. Johnston, and M. Zhang, "Edge and corner extraction using particle swarm optimisation," in *AI2010: Advances in Artificial Intelligence*, ser. LNCS. Springer, 2011, vol. 6464, pp. 323–333.

[11] M. Setayesh, M. Zhang, and M. Johnston, "Improving edge detection using particle swarm optimisation," in *Proceedings of the 25th International Conference on Image and Vision Computing, New Zealand*. IEEE Press, 2010.

[12] J. Ferwerda, "Elements of early vision for computer graphics," *IEEE Transactions on Computer Graphics and Applications*, vol. 21, no. 5, pp. 22–33, 2001.

[13] R. Deriche, "Using Canny's criteria to derive a recursively implemented optimal edge detector," *International Journal of Computer Vision*, vol. 1, no. 2, pp. 167–187, 1987.

[14] M. Setayesh, M. Zhang, and M. Johnston, "A novel particle swarm optimisation approach to detecting continuous, thin and smooth edges in noisy images," submitted to an IEEE transactions journal.

[15] N. Otsu, "A threshold selection method for gray level histograms," *IEEE Transaction on Trans. Syst. Man Cybern*, vol. 9, pp. 62–66, 1976.

[16] R. Medina-Carnicer, F. Madrid-Cuevas, A. Carmona-Poyato, and R. Muoz-Salinas, "On candidates selection for hysteresis thresholds in edge detection," *Pattern Recognition*, vol. 42, no. 7, pp. 1284–1296, 2009.

[17] H. Jeong and C. Kim, "Adaptive determination of filter scales for edge detection," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 14, pp. 579–585, 1992.

[18] E. Badekas and N. Papamarkos, "Automatic evaluation of document binarization results," in *Progress in Pattern Recognition, Image Analysis and Applications*, ser. LNCS. Springer, 2005, vol. 3773, pp. 1005–1014.

[19] S. S. Bukhari, F. Shafait, and T. M. Breuel, "Adaptive binarization of unconstrained hand-held camera-captured document images," *Journal of Universal Computer Science*, vol. 15, no. 18, pp. 3343–3363, 2009.

[20] J. Sauvola and M. Pietikinen, "Adaptive document image binarization," *Pattern Recognition*, vol. 33, pp. 225–236, 2000.

[21] R. Rakesh, P. Chaudhuri, and C. Murthy, "Thresholding in edge detection: a statistical approach," *IEEE Transactions on Image Processing*, vol. 13, no. 7, pp. 927–936, 2004.

[22] P. Viola and M. Jones, "Robust real-time face detection," *International Journal of Computer Vision*, vol. 57, pp. 137–154, 2004.

[23] J. Canny, "A computational approach to edge detection," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 679–698, 1986.

[24] K. Ghosh, S. Sarkar, and K. Bhaumik, "A possible mechanism of zero crossing detection using the concept of the extended classical receptive field of retinal ganglion cells," *Biol. Cybern.*, vol. 93, no. 1, pp. 1–5, 2005.

[25] K. Bowyer, C. Kranenburg, and S. Dougherty, "Images from South Florida University for edge detector comparison," available from http://marathon.csee.usf.edu/edge/edge_detection, accessed in 2009.

[26] N. L. Fernández-García, A. Carmona-Poyato, R. Medina-Carnicer, and F. J. Madrid-Cuevas, "Images from automatic generation of consensus ground truth for comparison of edge detection techniques," available from http://www.uco.es/~ma1fegan/investigacion/imagenes/ground-truth.html, accessed in 2009.

[27] K. Bowyer, C. Kranenburg, and S. Dougherty, "Ground truth images from edge detector evaluation using empirical ROC curves," available from http://figment.csee.usf.edu/edge/roc/, accessed in 2011.

[28] W. Pratt, *Digital Image Processing*. Wiley Interscience, 2007.
[29] E. Laskari, K. Parsopoulos, and M. Vrahatis, "Particle swarm optimization for integer programming," in *Proceedings of the 2002 Congress on Evolutionary Computation*. IEEE Press, 2002, pp. 1582–1587.