



Fuzzy Index Evaluating Image Edge Detection obtained with Ant Colony Optimization

1st Cristina Ticala

Faculty of Sciences
Technical University of Cluj-Napoca
Cluj-Napoca, Romania
0000-0001-8740-8575 


2nd Camelia-M. Pinteia

Faculty of Sciences
Technical University of Cluj-Napoca
Cluj-Napoca, Romania
0000-0003-2338-2454 


3rd Simone A. Ludwig

Department of Computer Science
North Dakota State University
Fargo, USA
0000-0002-8419-0192 


4th Mara Hajdu-Macelararu

Faculty of Sciences
Technical University of Cluj-Napoca
Cluj-Napoca, Romania
0000-0003-3135-1244 

5th Oliviu Matei

Faculty of Electr. Eng.
Technical University of Cluj-Napoca
Cluj-Napoca, Romania
0000-0003-0319-8318 

6th Petrica C. Pop

Faculty of Sciences
Technical University of Cluj-Napoca
Cluj-Napoca, Romania
0000-0002-0626-9284 

Abstract—Ant Colony Optimization is one of the most used methods applied to complex problems. Image processing is a difficult task in particular when complex images are involved. This paper uses a fuzzy euclidean metric as a distance measure between pixels from two images. This is the first evaluation of Ant Colony Optimization image edge detection in the context of fuzzy index and fuzzy euclidean metrics. The Canny edge detection is considered as the ground truth when evaluating similarities between the considered fuzzy images including medical ones. Experiments were run and successful comparisons were conducted using existing data sets as well as well-known non-fuzzy similarity metrics such as Jaccard's index, Dice's coefficient and the Pratt's Figure of Merit were applied.

Index Terms—Fuzzy, ant colony optimization, image processing, edge detection

I. INTRODUCTION

Edge detection is part of image processing and is one of the essential and important first steps in object identification. Edge detection algorithms rely on the detection of discontinuities within an image. There are several gradient-based edge detection algorithms available that are based on measuring local changes in gray value pixels. Thus, most edge detection techniques compute the first or second order derivative of the intensity function.

Roberts, Prewitt, Sobel, and the Laplacian mask operators have been used to estimate the first and second-order derivatives [1]. However, since these so-called detectors are very sensitive to noise and have poor performance on noisy images, Canny in [2] proposed a method to address this shortcoming. Canny's approach convolves the image with first-order derivatives of the Gaussian filter before applying edge detection. Other approaches from the literature [3]–[5] use linear filtering, local orientation analysis, fitting of analytical models, and local energy.

Furthermore, unlike the approaches presented above, which are said to be noncontextual, another category exists.

The contextual edge detectors detect contours that are of interest as part of a specific computer vision task by including information around an edge such as image topology, perceptual difference in texture, edge continuity, and local image statistics. More details can be found in [6] where a comprehensive review of various approaches to contour detection has been listed.

As seen, gradient-based methods for edge detection have the issue of edge localization for images that have smooth transitions in gray levels because of the ambiguity of the edge structure in such images. Thus, fuzzy-based approaches have been introduced to overcome the challenge. In [7]–[9] different fuzzy systems empowers the dynamic parameter adaptation in metaheuristics, complex nonlinear systems and servo systems.

In [10], a fuzzy-based approach for contrast intensification was applied to detect edges in Xray images, which other researchers expanded on. For example, a fuzzy rule-based system was applied in [11], [12]. Using a 3×3 window-based filtering approach followed by fuzzy reasoning was applied in [13]. Some of the authors' related work with the potential to be further improved are mentioned in [17]–[19].

Another approach used a global contrast intensification and a local fuzzy edge detection phase [21]. The first phase applies a modified Gaussian membership function to represent each pixel in the fuzzy plane optimizing an entropy function. In [22] and [23] other features including wrap-gate carbon nanotube transistors are used for image edge detection.

In [24], an edge detector was proposed based on fuzzy IF-THEN inference rules to model edge continuity. The approach described in [25] is based on fuzzy logic reasoning without determining the threshold value of the pixels in edge detection. This is achieved by segmenting the images into regions using a floating 3×3 binary matrix and the edge pixels are mapped to a range of values which are distinct from each other. An ANFIS edge detector was applied in [26]. 81 rules and four 3×3 masks were used to detect edges in four directions.

This was further advanced in whereby a morphological gradient and fuzzy logic approach was used to detect edges using an interval type-2 fuzzy inference system [27].

Evolutionary algorithms have also been applied to edge detection. For example, a genetic algorithm was applied to texture images [28]. The algorithm worked by selecting the edge regions first before applying the GA to decide on the optimum edge regions.

Another approach combined a fuzzy heuristics with PSO (Particle Swarm Optimization) [29]. Furthermore, ACO (Ant Colony Optimization) and adaptive threshold is reported on in [30]. ACO is used to determine the well-connected edge-map by guiding the ant movement using a local variation of intensity values. Other ACO with fuzzy approaches are reported in [14]–[16].

The current research structure and methods used in correlation with existing work are further enumerated:

1. **Ant Colony Optimization (ACO) for Image Edge Detection** is presented and further used to obtain feasible edges of given images as in [31]–[33].
2. **Image post-processing** is done with the use of a Denoise Convolutional Neural Network (DnCNN) followed by the fuzzy method and computing the Fuzzy Index [36].
3. **Comparison** between Canny, as the ground truth method and ACO fuzzy edge results, with fuzzy and non-fuzzy operators as Pratt’s Figure of merit (FOM) [34], Jaccard’s index (JI) and Dice’s coefficient (DC) on the tested image is done.

Section II includes a preview of the Ant Colony Optimization used for edge image detection. A description of the fuzzy model follows. In Section III, the similarity results are included that are based on a case study by applying ant algorithm and Canny edge detection with fuzzy approaches followed by a comparison with other metrics, analysis, and final discussion.

II. PRELIMINARIES

A. Ant Colony Optimization for Edge Detection

Nowadays, solving complex problems is a must; heuristics, including bio-inspired ones show their efficiency. Ant Colony Optimization (ACO) is one of the methods with well known results. The artificial agents, the artificial ants in ACO’s case, search for the best path in a particular space using “pheromones”; the pheromone trails is considered the “memory” list of agents; memory which is updated during the search. The solution quality is based on the pheromone value. The best solution uses the previous pheromone information while the ants (e.g. K) “travel” throughout the most promising regions of the search space, e.g., the space \mathcal{X} with $M_1 \times M_2$ nodes.

ACO edge detection [31] uses artificial ants searching for the best pixels in a 2D image and build a pheromone matrix. At first, all ants are randomly placed on the image. After a specified number of construction steps, N , constructing and updating the pheromone matrix, finally the edges of the image are detected.

The pheromone matrix, $\tau(0)$, is initialized with the constant τ_{init} . At each n -th construction step, a random ant will search for the best solution for L times, while moving from pixel i to j with the (1) probability.

The parameters used are: τ_{IJ} , the pheromone value of (I, J) , η_{IJ} , the heuristic value of (I, J) , α , β as weighting factors, and Ω_I , the neighborhood of node I .

$$p_{IJ}^n = \frac{(\tau_{IJ}(n-1))^\alpha (\eta_{IJ})^\beta}{\sum_{J \in \Omega_I} (\tau_{IJ}(n-1))^\alpha (\eta_{IJ})^\beta}, \quad J \in \Omega_I, \quad (1)$$

The connectivity domain, $\Omega_{l,m}$ from (1), is the 8-connectivity domain with the eight closest neighbors of I_{ij} .

The normalization factor $Z = \sum_{i=1}^{M_1} \sum_{j=1}^{M_2} V_c(I_{i,j})$, where $I_{i,j}$ is the intensity value of the pixel at position (i, j) in the image, and $V_c(I_{i,j})$ is a function processing $cI_{i,j}$ the clique [37]; the heuristic value of the pixel (i, j) is $\eta = \frac{1}{Z} \cdot V_c(I_{i,j})$ [31].

$V_c(I_{i,j})$ depends on the variation of intensity values on $cI_{i,j}$, see (2).

$$V_c(I_{i,j}) = f(|I_{i-2,j-1} - I_{i+2,j+1}| + |I_{i-2,j+1} - I_{i+2,j-1}| + |I_{i-1,j-2} - I_{i+1,j+2}| + |I_{i-1,j-1} - I_{i+1,j+1}| + |I_{i-1,j} - I_{i+1,j}| + |I_{i-1,j+1} - I_{i+1,j-1}| + |I_{i-1,j+2} - I_{i+1,j-2}| + |I_{i,j-1} - I_{i,j+1}|). \quad (2)$$

The operator \sin is used (3), where λ adjusts the shape of the \sin operator as in [31], [37].

$$f(x) = \begin{cases} \sin\left(\frac{\pi x}{2\lambda}\right), & 0 \leq x \leq \lambda; \\ 0, & \text{otherwise.} \end{cases} \quad (3)$$

The local pheromone matrix update (4) after each construction step enhances exploitation of the search space. Notations: ρ represents the pheromone evaporation rate, and Δ_{IJ} the quantity of pheromone on edge (I, J) .

$$\tau_{IJ}(n) = \begin{cases} \tau_{IJ}n - 1 \cdot (1 - \rho) + \rho \cdot \Delta_{IJ}, & \text{if } (I, J) \text{ is on} \\ & \text{the best tour,} \\ \tau_{IJ}(n - 1), & \text{otherwise.} \end{cases} \quad (4)$$

The global pheromone update (5) enhances exploration within the search space; ψ represents the pheromone decay coefficient.

$$\tau(n) = (1 - \psi) \cdot \tau(n - 1) + \psi \cdot \tau(0), \quad (5)$$

A threshold T is applied to the pheromone matrix $\tau(n)$ [38], and the best solution of the problem is obtained, which is the edge of the image. ACO convergence, finding the optimal solution, is possible just in some cases, for example for some specific parameters or particular instances.

B. Fuzzy approaches

This paper uses fuzzy values with triangular membership functions (6), provided in [36].

Further, the fuzzy Euclidean distance matrix FD will be used. FD is computed between two images, an initial image with the pixel set N_i , here the image obtained with Canny, and the image with detected edges, N_d , the image obtained with ACO.

$$\eta_{A(x)} = \frac{x}{\max(A)}. \quad (6)$$

1) *Similarity metrics*: Furthermore, several metrics are employed, including the Pratt's Figure of merit (FOM) [34], Jaccard's index (JI) and Dice's coefficient (DC) [35] with emphasis on the Fuzzy index FI (Section II-B3).

$$FOM = \frac{1}{\max(N_i, N_d)} \sum_{i=1}^{N_d} \frac{1}{1 + \alpha d_i^2}. \quad (7)$$

FOM uses the Euclidean distance d_i^2 [39] to compare two images where the image of reference edges N_i , is called Ground truth (GT), and the second one with the detected edges image N_d ; α it is a scaling factor to penalize detected edges.

The FOM, JI and DC metrics are using binary data before evaluating images. JI and DC are based on rendering images with sets. The detected images are represented by the Results Set and reference images are represented by the Truth Set.

The Results Set includes: False Positive (FP) (the pixels marked as edges when they are not), and True Positive (TP) (true edges), see Figure 1. The Truth Set includes pixels marked as not edges, but they are edges: True Positive (TP) and False Negative (FN). The evaluation uses values between 0, no similarity, and 1, when the images are identical.

$$JI = \frac{TP}{FP + TP + FN}. \quad (8)$$

$$DC = 2 \frac{TP}{(FP + TP) + (TP + FN)}. \quad (9)$$

2) *Fuzzy Euclidean distance*: An improved explicit Fuzzy Euclidean FD , distance matrix computation is used [36].

- **Input**: the Fuzzy images with $N_d = \{d_{11}, d_{12}, d_{13}, \dots, d_{mn}\}$, and $N_i = \{i_{11}, i_{12}, i_{13}, \dots, i_{mn}\}$.
- **Output**: the Fuzzy Euclidean distance matrix $FD(N_i, N_d)$.

- For each pixel in $N_i(x_1, y_1)$ Do find and use similarities:

a) *Find similarities*:

- Compute the difference between the pixel $N_i(x_1, y_1)$ and all $N_d(x_2, y_2)$ pixels and find the most similar pixels to $N_i(x_1, y_1)$ in N_d with the minimum $diff(x_1, y_1)$ value.

$$\min diff(x_1, y_1) = \min_{N_d} |N_i(x_1, y_1) - N_d(x_2, y_2)|$$

b) *Use similarities*: For each most similar pixel in $N_d(x_2, y_2)$ do

- Compute the Euclidean distance between $N_i(x_1, y_1)$ and $N_d(x_2, y_2)$ and choose D with the nearest most similar pixel, with the minimum D value.

$$\min D = \min_{N_d} \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2};$$

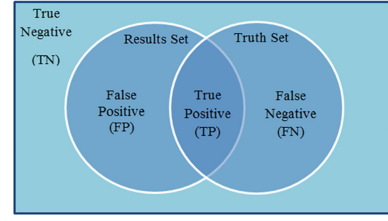


Fig. 1. Image Edge Detection: Results Set includes the detected image and the Truth Set includes the reference image; the intersection of the Results Set and Truth Set result is the True Positive (TP) pixels from identified edges [36].

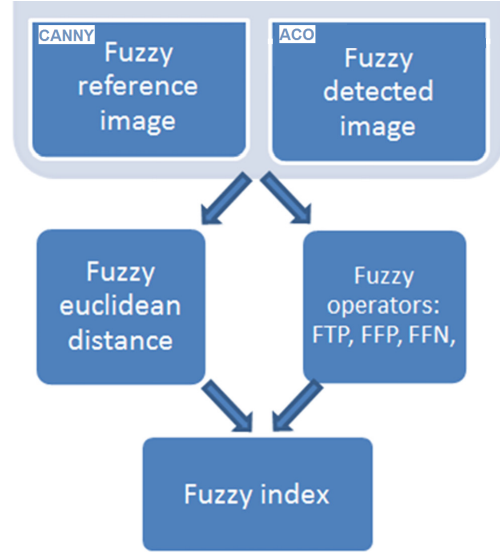


Fig. 2. Computing the fuzzy index flow [36] for the particular Canny reference image Canny and detected image with ACO.

- Final computation steps:

- the maximum Euclidean distance between two elements in D with $\max D = \sqrt{(m-1)^2 + (n-1)^2}$, where m and n are the matrix image dimensions;
- the Fuzzy Euclidean distance matrix with (10).

$$FD(N_i, N_j) = \frac{D(i, j)}{\max D}; \quad (10)$$

3) *Fuzzy Index*: The Fuzzy Index (FI) [36] computing method (Figure 2) follows.

- **Input**: the Fuzzy images with $N_d = \{d_{11}, d_{12}, d_{13}, \dots, d_{mn}\}$, $N_i = \{i_{11}, i_{12}, i_{13}, \dots, i_{mn}\}$ and the fuzzy Euclidean distance matrix $FD_{mn}(N_i, N_d)$.
- **Output**: the Fuzzy index FI .

- The scalar cardinality of images sets N_i and N_d follows.

$$|N_d| = \sum_{0 \leq i \leq m, 1 \leq j \leq n} N_d(i, j); |N_i| = \sum_{0 \leq i \leq m, 1 \leq j \leq n} N_i(i, j);$$

- The Fuzzy True Positive FTP , the fuzzy intersection of N_i and N_d , and its scalar cardinality, $|FTP|$ are computed with: $FTP = \min\{N_i(i, j), N_d(i, j)\}$;

$$|FTP| = \sum_{0 \leq i \leq m, 1 \leq j \leq n} FTP(i, j);$$

- The Fuzzy False Positive FFP bounded difference between N_i and N_d , and its scalar cardinality, $|FFP|$ are computed with: $FFP = \max\{0, N_i(i, j) - N_d(i, j)\}$;

$$|FFP| = \sum_{0 \leq i \leq m, 1 \leq j \leq n} FFP(i, j);$$

- The Fuzzy False Negative FFN bounded difference between N_d and N_i , and its scalar cardinality, $|FFN|$ are computed with: $FFN = \max\{0, N_d(i, j) - N_i(i, j)\}$;

$$|FFN| = \sum_{0 \leq i \leq m, 1 \leq j \leq n} FFN(i, j);$$

- The Fuzzy Index FI is computed as in (11).

$$FI = \frac{1}{(m \cdot n \cdot \max\{|N_i|, |N_d|\})} \cdot (|FTP| \cdot \sum_{1 \leq i \leq m, 1 \leq j \leq n} \frac{1}{1 + FD(i, j)} - |FFP| - |FFN|). \quad (11)$$

III. EXPERIMENTS AND RESULTS

A. Data and Software

The data tests used are several images with 128×128 , 200×200 resolution and larger images, with 256×256 resolution. The medical image *Brain CT* is free of copyright, and could be provided for free by request for scientific reasons.

In order to compare our medical images we use the Sphere from [36], an 200×200 resolution sphere generated with Matlab; therefore, a conversion of the *Brain CT* image was made for a 200×200 resolution using Cubic interpolation. The sphere image was scaled to cope with the resolution of *Brain CT* images, to 128×128 , and 256×256 , respectively. Canny and ACO was used to test the sphere images for all resolutions included.

The ACO edge detection was coded and tested with Matlab using an AMD Ryzen 5 2500U, 2GHz processor; the software is based on an improved version [31] of the Image Edge Detection using ACO [40]. The ACO running time was around 4500 seconds. The Canny software used is from [41].

Details including the data set used for the case studies and representation of results is available at Github https://github.com/cristina-ticala/Fuzzy_image_edge_detection_ACO.

B. Parameters

ACO parameters [31], [37] are further presented. The *constant* values include: the number of ants considered is the same as the resolution, 128, 200 and 256, respectively for the larger images; each ant makes 300 movements in each of the $L = 100$ steps; for the image with 128×128 resolution, a total of 38400 iterations during each of the 100 steps; for the larger images 100 steps and 900 movements with a total of 230400 iterations for each image are used.

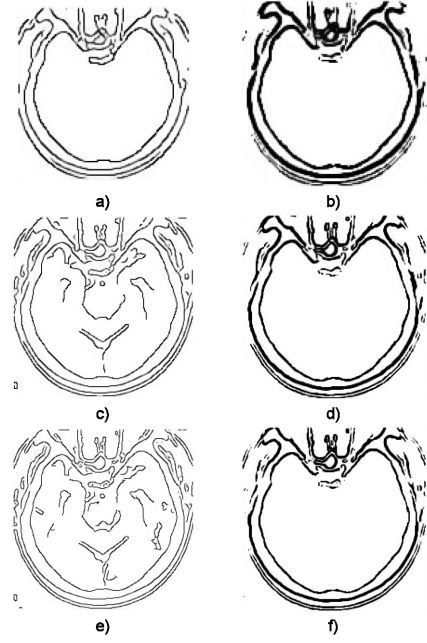


Fig. 3. The comparative image Brain CT with 128×128 resolution results obtained with Canny (a) and ACO on Fuzzy image (b); 200×200 resolution results obtained with: Canny (c) and ACO on Fuzzy image (d) and 256×256 resolution results obtained with: Canny (e) and ACO on Fuzzy image (f).

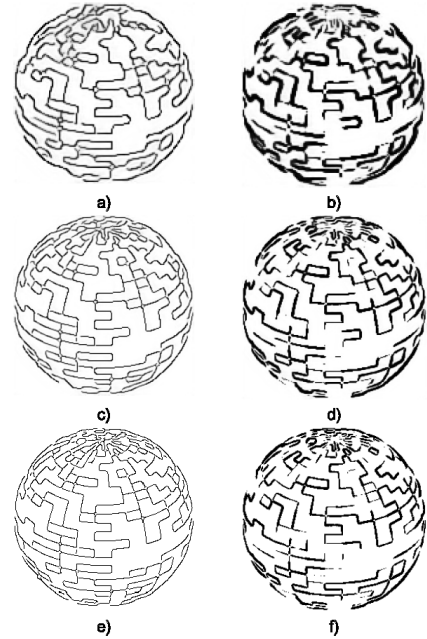


Fig. 4. The comparative state-of-art image sphere with 128×128 resolution results obtained with Canny (a) and ACO on Fuzzy image (b); 200×200 resolution results obtained with: Canny (c) and ACO on Fuzzy image (d) and 256×256 resolution results obtained with: Canny (e) and ACO on Fuzzy image (f), based on the Sphere Dataset [36].

The pheromone matrix is initialized with $\tau_{init} = 0.0001$. Other constant values, tested on previous work [31]–[33] are: the weighting factors of the pheromone information, $\alpha = 1$ and heuristic information, $\beta = 0.01$ (1); the evaporation rate, $\rho = 0.1$ (4) and the pheromone decay coefficient, $\psi = 0.001$ (5). Other parameters are the adjusting factor, $\lambda = 10$ (3), the tolerance $\varepsilon = 0.1$ within the decision process of the proposed method; the stopping criteria is provided by the maximal number of steps L .

The initial matrix of the *Brain CT* edge image matrix of Canny and ACO for the first 10×10 elements, in the binary format have all the pixels white, the exceptions are two black pixels for Canny: $a_{10,8}$ and $a_{10,9}$. The fuzzy edge image matrix for both ACO and Canny are included in Figure 5.

C. Post-Processing

The processing with Denoise Convolutional Neural Network (DnCNN) [42] of edges detected with Canny and ACO is made. Furthermore, the fuzzy triangular membership functions (6), provided in [36] is applied. We adapted the sphere image to the *Brain CT* image, solved with Ant Colony Optimization and Canny in our previous works [31]–[33].

D. Results Comparison and Discussions

Tables I-II show the non-fuzzy and fuzzy comparative results within Case Studies, 128×128 and 256×256 resolution: I Sphere and III Sphere, respectively. The 200×200 resolution is considered within the Case Study: II Sphere in the Tables I-II with the fuzzy and non-fuzzy results.

Based on the techniques used for image edge detection, similarity values differ. From our research, for the *Brain CT* image, when compared with the non fuzzy operators: the Pratt’s Figure of merit, Jaccard’s index and Dice’s coefficient, the number of the fuzzy true positives is less than the true positives, and there are more fuzzy false positives than false positives and also, more fuzzy false negatives than false negatives, the same as in [36]. The corresponding results of the non-fuzzy approach are shown in Table III.

When compared with the non-fuzzy index, as Dice, JI, FOM, where binary files with 0s and 1s, the Fuzzy index (FI) implies images with gray tone, some pixels are not dismissed through a threshold and the final results are improved. Table II shows that Fuzzy index (FI) and Jaccard’s index (JI) have the lowest values. Furthermore, FI results could be improved if the processing of the images is further improved; here a processing with Denoise Convolutional Neural Network (DnCNN) [42] of edges detected with Canny and ACO was used to obtain the 256 grey values for the further fuzzy processing.

As the above formula (Eq. (11)) shows, the fuzzy index includes specific parameters, not used in the non-fuzzy indexes when comparing two fuzzy images; one of the images is the ground truth of edges image, the Canny images in our case studies, and the other one the ACO edge image results. Using FI an improved evaluations of the edges detection correctness in fuzzy images is made.

0	0	0.0078	0.0039	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0.0039	0.0118	0	0	0	0	0	0	0
0.0078	0.0039	0	0	0	0	0	0	0
0	0	0.0039	0.0039	0	0	0	0.0039	0
0	0.0118	0.0118	0.0039	0.0039	0.0157	0.0039	0	0
0.0078	0.0039	0.0196	0.0118	0.0196	0.0039	0	0.0157	0.0118
0.0078	0.0118	0.0235	0.0118	0	0	0.0196	0	0
0.0196	0.0118	0.0235	0	0	0.0039	0	0.0431	0.0824
0.0196	0	0.0039	0	0	0	0.1255	0.8941	0.8745

0	0	0.0039	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0.0078	0	0.0039	0	0	0	0	0
0	0	0	0	0	0	0	0	0
0	0	0	0.0039	0	0	0	0.0078	0
0	0.0039	0.0078	0	0	0.0039	0.0118	0.0118	0.0078
0	0	0.0039	0.0039	0	0	0.0118	0.0039	0.0039
0	0	0.0078	0.0039	0	0.0039	0.0039	0.0039	0
0	0	0.0078	0.0039	0.0039	0	0	0	0.0118
0	0	0.0039	0.0039	0.0039	0	0	0	0

Fig. 5. The fuzzy edge image with the first ten pixels matrix representations for Canny (upper matrix) and ACO (downside matrix) results for the *Brain CT* with 128×128 resolution.

TABLE I
RESULTS OF THE NI (CANNY) AND ND (ACO) WITH FUZZY SETS OPERATORS FUZZY INDEX FI (SECTION II-B3).

Case study	Description	N_i Canny	N_d ACO	FI
I Brain CT	128 × 128	a)Figure 3	b)Figure 3	0.8743
II Brain CT	200 × 200	c)Figure 3	d)Figure 3	0.8939
III Brain CT	256 × 256	e)Figure 3	f)Figure 3	0.8996
I Sphere	128 × 128	a)Figure 4	b)Figure 4	0.8222
II Sphere	200 × 200	c)Figure 4	d)Figure 4	0.8707
III Sphere	256 × 256	a)Figure 4	b)Figure 4	0.8891

TABLE II
COMPARISON RESULTS: NI (CANNY) AND ND (ACO) WITH FUZZY & NON FUZZY OPERATORS: PRATT'S FIGURE OF MERIT (PFOM), JACCARD'S INDEX (JI), DICE'S COEFFICIENT (DC) AND FUZZY INDEX FI .

Case study	JI	DC	PFOM	FI
I Brain CT	0.8702	0.9306	0.9892	0.8743
II Brain CT	0.8776	0.9348	0.9911	0.8939
III Brain CT	0.8811	0.9368	0.9909	0.8996
I Sphere	0.8143	0.8976	0.9853	0.8222
II Sphere	0.8589	0.9241	0.9889	0.8707
III Sphere	0.8750	0.9333	0.9901	0.8891

TABLE III
RESULTS: NI (CANNY) AND ND (ACO) WITH FUZZY SETS OPERATORS.

Case study	%TP	%FN	%FP	%FTP	%FFN	%FFP
I Brain CT	97.73	11.18	2.27	80.84	54.96	19.16
II Brain CT	96.10	9.00	3.90	69.77	30.23	49.52
III Brain CT	96.17	8.68	3.83	70.20	29.80	48.49
I Sphere	96.07	15.77	3.93	70.58	29.42	64.95
II Sphere	97.05	11.82	2.95	75.58	24.42	57.11
III Sphere	97.32	10.34	2.68	77.24	22.76	53.43

IV. CONCLUSION

The paper is the first study of image edge detection results obtained with Ant Colony Optimization (ACO) while using the Fuzzy index. Canny edge detection is the ground truth model used to compare the ACO edge image detection results with.

As in [36], for the majority of the studied cases when compared with the non fuzzy operators, the fuzzy operators found more false positives and false negatives but less true positive results.

Based on the obtained index values, the Ant Colony technique has a high quality, and it is comparable with Canny on the studied cases; for some cases it is almost similar: the Pratt figure of merit [43] shows the highest similarity for binary files, with values greater than 0.9853. The studied Fuzzy index shows also a good similarity for complex fuzzy images.

The fuzzy images include more pixels than binary images. As the fuzzy images have more data to work with, it will have the opportunity to increase the edge image recognition rates; therefore, processing images with the fuzzy rules will hopefully lead to better edge image detection.

REFERENCES

- [1] R. C. Gonzalez and R. E. Woods, "Digital Image Processing", Pearson Education (Prentice-Hall), 3rd ed., 2009.
- [2] J. Canny, A computational approach to edge detection, IEEE Trans. Pattern An. Mach. Intell., vol. 8, no. 6, pp. 679–698, 1986.
- [3] E. C. Hildreth, "The detection of intensity changes by computer and biological vision systems," Computer Vision, Graphics and Image Proc., vol. 22, no. 1, pp. 1–27, 1983.
- [4] R. M. Haralick, "Digital step edges from zero-crossings of second directional derivatives", IEEE Trans. Pattern An. Mach. Intell., vol. 6, no. 1, pp. 58–68, 1984.
- [5] V. S. Nalwa and T. O. Binford, "On detecting edges", IEEE Trans. Pattern An. Mach. Intell., vol. 8, no. 6, pp. 699–714, 1986.
- [6] G. Papari and N. Petkov, "Edge and line oriented contour detection: state of the art," Image Vis. Comput., vol. 29, no. 2-3, pp. 79–103, 2011.
- [7] O. Castillo, P. Melin, E. Ontiveros, C. Peraza, P. Ochoa, F. Valdez, and J. Soria, "A high-speed interval type 2 fuzzy system approach for dynamic parameter adaptation in metaheuristics," Eng. Appl. Artif. Intell., vol. 85, pp. 666–680, 2019.
- [8] T. Chen, A. Babanin, A. Muhammad, B. Chapron, and C. Chen, "Modified evolved bat algorithm of fuzzy optimal control for complex nonlinear systems," Rom. J. Inf. Sci. Technol., vol. 23, pp. 28–40, 2020.
- [9] R. E. Precup, R.C. David, R. C. Roman, E. M. Petriu and A. I. Szedlak-Stinean, "Slime mould algorithm-based tuning of cost-effective fuzzy controllers for servo systems. Int. J. Comput. Intell. Sys., vol. 14, no. 1, pp. 1042–1052, 2021.
- [10] S. K. Pal and R. A. King, "On edge detection of X-ray images using fuzzy sets," IEEE Trans. Pattern An. Mach. Intell., vol. 5, no. 1, pp. 69–77, 1983.
- [11] C. W. Tao, W. E. Thompson, and J. S. Taur, "A fuzzy if-then approach to edge detection," in Proc. FUZZ-IEEE, vol. 2, pp. 1356–1360, San Francisco, Calif, USA, 1993.
- [12] F. Russo and I. G. Ramponi, "Edge extraction by fire operators," in Proc. FUZZ-IEEE, pp. 249–253, Orlando, Florida, USA, 1994.
- [13] F. Russo, "Edge detection in noisy images using fuzzy reasoning," IEEE Trans. Instrum. Meas., vol. 47, no. 5, pp. 1102–1105, 1998.
- [14] Z. Dorrani, H. Farsi, S. Mohamadzadeh, "Image Edge Detection with Fuzzy Ant Colony Optimization Algorithm" Int. J. Eng., vol. 33, no. 12, pp. 2464–2470, 2020.
- [15] Y. Han and P. Shi, "An improved ant colony algorithm for fuzzy clustering in image segmentation," Neurocomputing, vol. 70, no. 4–6, pp. 665–671, 2007.
- [16] O. P. Verma, M. Hanmandlu and A. K. Sultania, "A Novel Fuzzy Ant System for Edge Detection," 2010 IEEE/ACIS 9th Int. Conf. Computer and Inf. Sci., pp. 228–233, 2010.
- [17] C.M. Pinteal et al. "A Fuzzy Approach of Sensitivity for Multiple Colonies on Ant Colony Optimization," Soft Comput. Appl., vol. 634, pp. 87–95, 2016.
- [18] O. Matei, "Defining an ontology for the radiograph images segmentation," In Proc. of the 9th Int. Conf. Dev. Appl. Sys., pp. 266–271, 2008.
- [19] A.N. Marginean et al, "Reliable learning with PDE-based CNNs and dense nets for detecting COVID-19, pneumonia, and tuberculosis from chest X-ray images." Math., vol. 9, pp. 1–20, 2021.
- [20] L. R. Liang and C. G. Looney, "Competitive fuzzy edge detection", Appl. Soft Comput. J., vol. 3, no. 2, pp. 123–137, 2003.
- [21] M. Hanmandlu, J. See, and S. Vasikarla, "Fuzzy edge detector using entropy optimization," in Proc. Int. Conf. Infor. Technol.: Coding Computing, ITCC'04, pp. 665–670, Las Vegas, USA, 2004.
- [22] D. Cavaliere, A. Saggese, S. Senatore, M. Vento and V. Loia, "Empowering UAV scene perception by semantic spatio-temporal features," 2018 IEEE Int. Conf. Environ. Eng., pp. 1–6, 2018.
- [23] A. Bozorgmehr, M. K. Q. Jooq, M. H. Moayeri, K. Navi and N. Bagherzadeh, "A novel digital fuzzy system for image edge detection based on wrap-gate carbon nanotube transistors", Comput. Electr. Eng., vol. 87, 106811, 2020.
- [24] L. Hu, H. D. Cheng, and M. Zhang, A high performance edge detector based on fuzzy inference rules, Inf. Sci., vol. 177, no. 21, pp. 4768–4784, 2007.
- [25] A. Alshennawy and A. Aly, "Edge detection in digital images using fuzzy logic technique", World Academy of Sci., Eng. & Technol., vol. 27, pp. 178–186, 2009.
- [26] L. Zhang, M. Xiao, J. Ma, and H. Song, "Edge detection by adaptive neuro-fuzzy inference system", in Proc. Congress on Image and Signal Processing, CISP'09, Tianjin, China, pp. 1–4, 2009.

- [27] P. Melin, O. Mendoza, and O. Castillo, "An improved method for edge detection based on interval type-2 fuzzy logic", *Expert Syst. Appl.*, vol. 37, no. 12, pp. 8527–8535, 2010.
- [28] M. Yoshimura and S. Oe, "Edge detection of texture image using genetic algorithms," in *Proc. SICE Conf.*, pp. 1261–1266, Tokushima, Japan, 1997.
- [29] N. Khalid, M. Manaf, and M. Aziz, "Fusion of fuzzy heuristic and particle swarm optimization as an edge detector", in *Proc. Int. Conf. Inf. Retrieval Knowl. Man.*, pp. 250–254, Shah Alam, Malaysia, 2010.
- [30] O. P. Verma, P. Singhal, S. Garg, and D. S. Chauhan, "Edge detection using adaptive thresholding and ant colony optimization", in *Proc. World Congress Inf. Commun. Technol., WICT '11*, pp. 313–318, 2011.
- [31] C. Ticala, I. Zelina, and C-M. Pintea, "Admissible Perturbation of Demicontractive Operators within Ant Algorithms for Medical Images Edge Detection," *Math.*, vol. 8(1040), pp. 1–13, 2020.
- [32] C-M. Pintea, and C. Ticala, "Medical image processing: A brief survey and a new theoretical hybrid ACO model," *Comb. Intell. Methods App.*, 2016, pp. 117–134.
- [33] C. Ticala, C-M. Pintea, and O. Matei, "Sensitive Ant Algorithm for Edge Detection in Medical Images," *Appl. Sci.*, vol. 11(23), pp. 1–10, 11303, 2021.
- [34] I.E. Abdou and W.K. Pratt, "Quantitative design and evaluation of enhancement thresholding edge detectors," *IEEE Proc.*, vol. 67(5), pp. 75363, 1979.
- [35] D.W. Shattuck, G. Prasad, M. Mirza, K.L. Narr, and A.W. Toga, "Online resource for validation of brain segmentation methods," *NeuroImage*, vol. 45(2), 431-439, 2009.
- [36] F. Perez-Ornelas, O. Mendoza, P. Melin, J. R. Castro, A. Rodriguez-Diaz, and O. Castillo, "Fuzzy Index to Evaluate Edge Detection in Digital Images," *Plos One*, vol. 10, no. 6, e0131161, 2015.
- [37] J. Tian, W. Yu, and S. Xie, "An ant colony optimization algorithm for image edge detection," in *Proc. IEEE CEC*, 2008, pp. 751–756.
- [38] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. Syst. Man Cybern.*, vol. 9(1), pp. 62–66, 1979.
- [39] J.M.D. Moreno "Introduccion a la topologia de los espacios metricos", Univ. de Cadiz Servicio de Publicaciones, 1998.
- [40] J. Kanchi-Tian, "Image Edge Detection Using Ant Colony Optimization" v. 1.2.0.0. MATLAB Central File Exchange, 2011.
- [41] Edge Function. MATLAB Central File Exchange. Available online: <https://www.mathworks.com/help/images/ref/edge.html>.
- [42] Denoise image using Deep Neural Network. MATLAB Central File Exchange. Available online: <https://www.mathworks.com/help/images/ref/denoiseimage.html>
- [43] V. Bhadouria (2022). Pratt's Figure of Merit, MATLAB Central File Exchange. Available online: <https://www.mathworks.com/matlabcentral/fileexchange/60473-pratt-s-figure-of-merit>