

RESEARCH ARTICLE

# Fuzzy Index to Evaluate Edge Detection in Digital Images

Felicitas Perez-Ornelas<sup>1</sup>, Olivia Mendoza<sup>1\*</sup>, Patricia Melin<sup>2</sup>, Juan R. Castro<sup>1</sup>, Antonio Rodriguez-Diaz<sup>1</sup>, Oscar Castillo<sup>2</sup>

**1** School of Engineering, Autonomous University of Baja California, Tijuana, Baja California, Mexico, **2** Division of Graduate Studies, Tijuana Institute of Technology, Tijuana, Baja, California, Mexico

\* [omendoza@uabc.edu.mx](mailto:omendoza@uabc.edu.mx)

## Abstract

In literature, we can find different metrics to evaluate the detected edges in digital images, like Pratt's figure of merit (FOM), Jaccard's index (JI) and Dice's coefficient (DC). These metrics compare two images, the first one is the reference edges image, and the second one is the detected edges image. It is important to mention that all existing metrics must binarize images before their evaluation. Binarization step causes information to be lost because an incomplete image is being evaluated. In this paper, we propose a fuzzy index (FI) for edge evaluation that does not use a binarization step. In order to process all detected edges, images are represented in their fuzzy form and all calculations are made with fuzzy sets operators and fuzzy Euclidean distance between both images. Our proposed index is compared to the most used metrics using synthetic images, with good results.



CrossMark  
click for updates

## OPEN ACCESS

**Citation:** Perez-Ornelas F, Mendoza O, Melin P, Castro JR, Rodriguez-Diaz A, Castillo O (2015) Fuzzy Index to Evaluate Edge Detection in Digital Images. PLoS ONE 10(6): e0131161. doi:10.1371/journal.pone.0131161

**Editor:** Guy J-P. Schumann, University California Los Angeles, UNITED STATES

**Received:** December 11, 2014

**Accepted:** May 29, 2015

**Published:** June 26, 2015

**Copyright:** © 2015 Perez-Ornelas et al. This is an open access article distributed under the terms of the [Creative Commons Attribution License](https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

**Data Availability Statement:** All relevant data are within the paper and its Supporting Information files.

**Funding:** This work was supported by Consejo Nacional de Ciencia y Tecnología (CONACYT). [www.conacyt.mx](http://www.conacyt.mx).

**Competing Interests:** The authors have declared that no competing interests exist.

## Introduction

In recent years, a wide variety of image taking devices has been developed. Thus, an advance in image processing techniques has become a major interest for many researchers. Although much has been accomplished, there is still a lot to do on this field.

Image processing methods are complex, and widely used in a multitude of areas such as medicine, military, geographical, just to name a few [1], [2], [3].

Usually, images are processed for two reasons, first, for information extraction and second to improve their quality. To accomplish this, we need to pre-process these images; using techniques that help us obtain a more suitable image for the required application [4], [5], [6], [7].

There are several methods for noise elimination and edge detection [1], [2], [6], [8]. The processes mentioned above are considered to be opposite of each other because edge detection emphasizes the changes in the image tones while noise elimination minimizes these changes [7], [9].

Edges detection is one of the most widely used methods for image pre-processing since it is much faster than processing whole images from the start. In this way, total time execution is reduced dramatically.

Some of these methods are Sobel, Prewitt, Roberts, Morphological Gradient, Canny and others [2], [6],[8], [9], [10]. Some variants of the traditional methods include the use of neural networks [3], [7], genetic algorithms or hybrid systems [11], [12], [13], [14], [15], [16]. Edge detection can improve the results of different image processing systems. But the main question remains, how can we choose the best edge detector for a given problem?.

There are different metrics for edges evaluation [17], [18], [19], [20], [21]; all of them are based on finding the similarity between two images. The first image corresponds to the detected edges, and the second image is the reference image or ground truth (GT), which is considered the image with the ideal edges.

All existing metrics include a previous step consisting on image binarization, but much information is lost and, as a result incomplete images will be evaluated. We show a color synthetic image in Fig 1(a), a wired synthetic image (GT) in Fig 1(b) and detected edges image in Fig 1(d). Images in Fig 1(b) and 1(d) are not binarized, while images in Fig 1(c) and 1(e) are binarized. The difference between images can be noticed easily, the loss of edges regions in binarized images is evident.

It has been shown that neural network systems for image processing get greater image recognition percentage when images used in training phase are processed with edges detection algorithms, specifically those improved using fuzzy inference systems [21][22].

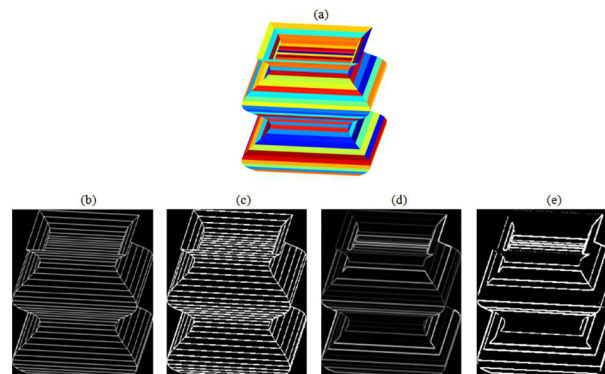
After visual inspection, images from fuzzy edges detectors show more details of original images and more homogeneous background. Better feature extraction allows train neural networks with more relevant attributes and less noise in data sets, improving their performance.

While visual analysis is very useful, it is also qualitative and subjective. For a more objective analysis, it is mandatory to quantify performance. Up to now, we have only found edges detector evaluation methods which need to binarize images before processing, with the disadvantage that the evaluated image will not be the same as the one actually used by the system.

The main contribution of this paper consists in a proposed method to calculate a fuzzy index for the evaluation of the detected edges of an image represented by its fuzzy form (without binarization) [4], [8], [9], [10], [16], [19].

The proposed fuzzy index FI integrates parameters that other metrics use separately. As shown in Table 1.

Since the proposed method was designed for fuzzy images, all the calculations were extended with fuzzy operators. Another significant contribution is the fuzzy extension of Euclidean distance [23] between two images; as part of the method for the calculation of an index to evaluate edge detection.



**Fig 1. Binarization of edges images.** (a) Color image, (b) GT image, (c) Binarized GT, image, (d) Edges image, (e) Binarized edges image.

doi:10.1371/journal.pone.0131161.g001

Table 1. Comparison of metrics for edges detection evaluation.

Metrics	Parameters				Binarize
	Euclidean Distance	False Positive (FP)	True Positive (TP)	False Negative (FN)	
FOM	✓				✓
DC		✓	✓	✓	✓
Jl		✓	✓	✓	✓
FI	✓	✓	✓	✓	

doi:10.1371/journal.pone.0131161.t001

The paper is organized as follows: In a Section 2 we describe some metrics used to evaluate edge detection: The Euclidean distance [23], Pratt's Figure of Merit [20], Jaccard's index and Dice's coefficient [18]. In Section 3 the basic concepts of fuzzy sets operations, like intersection, bounded difference and magnitude of fuzzy set [5],[8] are described. In Section 4, we describe the proposed method, including fuzzy Euclidean distance and the algorithm to compute it.

Finally, we explain how parameters described in Section 3 are integrated, and an algorithm to compute fuzzy index (FI) is proposed. In Section 5, we show experiments and results made with different synthetic images and their comparison with traditional metrics. Section 6 is for our conclusions.

### Metrics for Edges Detection Evaluation

In this section, we describe some existing metrics for edge detection evaluation. These methods are key for designing the proposed FI.

#### Euclidean distance between two images

The Euclidean distance between two points, as shown in Eq 1, is commonly used to find the similarity between two images. If one of the images is assigned as reference of correctly detected edges, the Euclidean distance can be a quality measure of the detected edges of another image.

$$Ed(A_{x_1,y_1}, B_{x_2,y_2}) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \tag{1}$$

Given two images A and B, the Euclidean distance between pixels can be calculated as follows:

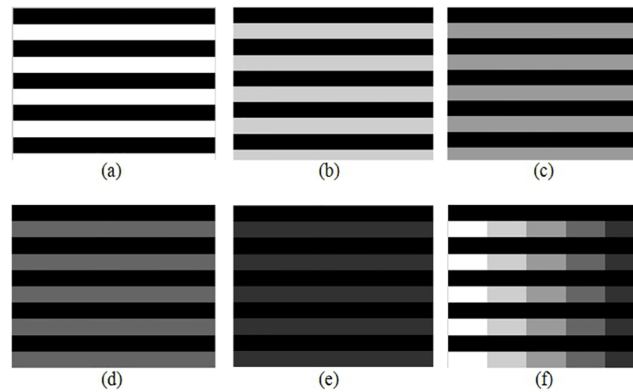
- Fig 2(a) represent image A and Fig 2(b) represent image B.
- The image in Fig 2(c) shows the Euclidean distance from pixel  $a_{1,1}$  to all other pixels on B.
- A value of 0 (zero) corresponds to the Euclidean distance from pixel  $a_{1,1}$  to pixel  $b_{1,1}$  and increases as pixels move farther.
- The image in Fig 2(d) shows Euclidean distances from  $a_{3,3}$  to all other pixels on B.

#### Pratt's figure of merit (FOM)

One of the most used metrics is the Pratt's Figure of merit Eq 2 [20]. Abdou and Pratt proposed this metric in 1978 [17].

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

It uses Euclidean distance  $d_i^2$  [23] to compare two images, the first is the image of reference



**Fig 2. Euclidean distance matrices between images A and B.** (a) Image A, (b) Image B, (c) Euclidean distance matrix between  $a_{1,1}$  and  $b_{ij}$ , (d) Euclidean distance matrix between  $a_{3,3}$  and  $b_{ij}$ .

doi:10.1371/journal.pone.0131161.g002

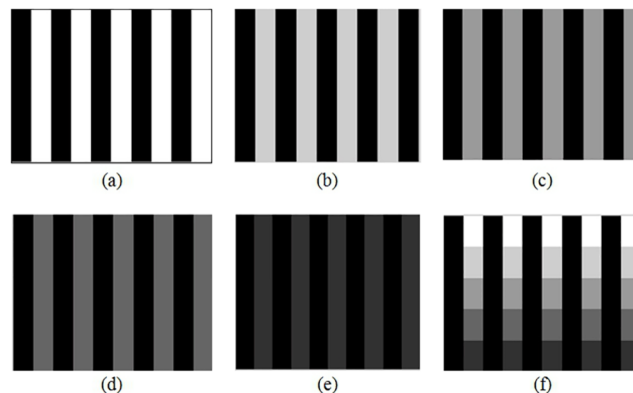
edges  $N_i$ , which is also called Ground truth (GT) and the second is the detected edges image  $N_d$ . It multiplies a scale factor  $\alpha$  to the Euclidean distance calculated between the two images to penalize detected edges, this factor can vary or not even be used, and then it normalizes values and makes the sum of all calculations. Finally, it is multiplied by the inverse of the maximum amount of edge pixels between the two compared images.

It is noteworthy that these metrics binarize data before evaluating images; this means that evaluation is made over images that have lost information. These metrics return values between 0 and 1, where 0 would mean that there were no similarities found between detected image and reference image, and 1 meaning that great similarity was found, in other words, all pixels found in one image edges are detected in the same position as the other.

### Jaccard’s index (JI) and Dice’s coefficient (DC)

We also considered Jaccard’s index and Dice’s coefficient metrics [18], as shown below, which are based on rendering images with sets. Detected images are represented by set *Results Set* and reference images are represented by set *Truth Set* as is shown in Fig 3.

In *Results Set* we have: False Positive (FP) or “false alarms” (these are pixels marked as edges when they are not), and True Positive (TP) (which are truly edges); in *Truth Set* we have True Positive (TP) and False Negative (FN) (these are pixels marked as not edges, but they really



**Fig 3. Results Set and Truth Set.**

doi:10.1371/journal.pone.0131161.g003

are). These metrics are considered in the evaluation as false alarms (FP) and not detected edges (FN). The evaluation uses values between 0 and 1, where 0 means that there is no similarity between the images and 1 that the images are the same, but it also binarizes information before any evaluation is made.

Jaccard's index (JI) and Dice's coefficient (DC) use sets to represent images, as shown in Eqs 3 and 4 respectively. With the help of some set operations we are able to measure any detected edges, and these measures are some of the most applied.

$$JI = \frac{TP}{FP + TP + FN} \tag{3}$$

$$DC = 2 \left( \frac{TP}{(FP + TP) + (TP + FN)} \right) \tag{4}$$

### Fuzzy sets

Today we can find a variety of areas where fuzzy logic is applied [24], [25], [26], [27], [28], [29], [30], [31]. For example, control is one of the areas where it is most used, simulation, prediction, optimization, information systems, such as databases, pattern recognition [32], [33], [34], [35], [36], [37] computer vision, etc. Fuzzy logic is an alternative to traditional logic, which assigns a degree of membership to evaluate things such as human reasoning.

Fuzzy logic started in 1965 by Lotfi Zadeh, Professor at the University of Berkeley, mainly applied in control systems and complex industrial processes, working with information that is not accurate or with high inaccuracy [38].

There are concepts that do not have clear boundaries, this is why these fuzzy sets are associated to a linguistic value. Its membership function is defined in the range between 0 and 1, as shown in Eq 5.

A fuzzy set has been defined as shown in Eq 6, where  $\mu_A$  is the membership function of variable  $x$  in the universe of discourse  $U$ . The membership function is the essence of fuzzy sets and operations between fuzzy sets are based on it.

$$\mu_A : x \rightarrow [0, 1] \tag{5}$$

$$A = \{(x, \mu_A(x)) | x \in U\} \tag{6}$$

### Fuzzy sets operators

There are different operations between fuzzy sets, such as union, intersection, complement, etc. In this paper we use intersection, bounded difference and fuzzy scalar cardinality, these are explained below.

**Intersection of fuzzy sets.** Given sets A and B, intersection C is defined as Eq 7. For fuzzy sets Zadeh introduced fuzzy intersection defined in Eq 8 as the search for minimum of two fuzzy sets [38].

$$C = \{x | x \in A \text{ and } x \in B\} \tag{7}$$

$$\mu_{A \cap B} = \min\{\mu_A(x), \mu_B(x)\} \tag{8}$$

**Bounded difference of fuzzy sets.** Given sets A and B, bounded difference C is defined as shown in Eq 9. Fuzzy bounded difference between A and B is defined in Eq 10 as the maximum of 0 and the difference between each membership values of A and B. This operator is not commutative, then the bounded difference between B and A is defined in Eq 11.

$$C = (A \theta B) \forall x \in \cup \tag{9}$$

$$\mu_{A\theta B} = \max\{0, \{\mu_A(x) - \mu_B(x)\}\} \tag{10}$$

$$\mu_{B\theta A} = \max\{0, \{\mu_B(x) - \mu_A(x)\}\} \tag{11}$$

**Magnitude of fuzzy sets.** The magnitude of a fuzzy set can be calculated in different ways, one of which is the scalar cardinality defined as the sum of all the fuzzy values of each of the elements of the fuzzy set represented by Eq 12.

$$|A| = \sum_{x \in X} \mu_A(x) \tag{12}$$

### The Proposed Method

In the proposed method we introduce the calculation of the following fuzzy operations between a fuzzy reference image of edges and a fuzzy detected image: The fuzzy Euclidean distance (FD), the fuzzy true positive (FTP), the fuzzy false positive (FFP) and the fuzzy false negative (FFN).

The method integrates all of them in a single operator named fuzzy index (FI). A diagram of the process is shown in Fig 4.

### The fuzzy Euclidean distance

Given two images A and B, the fuzzy Euclidean distance between each pixel  $a_{i,j}$  in A and all the pixels  $b_{i,j}$  in B can be calculated with Eq 1 as described in section 2.1.

Once the Euclidean distance matrix D was calculated, the fuzzy Euclidean distance matrix FD can be obtained with some fuzzy membership function. In our method, we propose a triangular membership function as shown in Fig 5.

Once the Euclidean distance matrix D was obtained, the fuzzy Euclidean distance matrix FD can be calculated with some fuzzy membership function. In our method we propose a triangular membership function.

### Fuzzy synthetic images

Images can be stored in matrices where each pixel is represented with values in the interval [0,255], as we can see in Figs 6 and 7. Particularly in binarized edges images, components with values of 1 represent pixels classified as edges, and components with values of 0 represent non edge pixels.

For fuzzy images approach, in edges images, all pixels are classified as edges with different membership values.

The fuzzy values using the triangular membership functions in Eq 13 according to the degrees of membership rightfully assigned are shown in Fig 8.

$$\mu_A(x) = \frac{x}{\max(A)} \tag{13}$$

0	0	0	0	0	0	0	0	0	0
1	1	0.8	0.8	0.6	0.6	0.4	0.4	0.2	0.2
0	0	0	0	0	0	0	0	0	0
1	1	0.8	0.8	0.6	0.6	0.4	0.4	0.2	0.2
0	0	0	0	0	0	0	0	0	0
1	1	0.8	0.8	0.6	0.6	0.4	0.4	0.2	0.2
0	0	0	0	0	0	0	0	0	0
1	1	0.8	0.8	0.6	0.6	0.4	0.4	0.2	0.2
0	0	0	0	0	0	0	0	0	0
1	1	0.8	0.8	0.6	0.6	0.4	0.4	0.2	0.2

Fig 4. Diagram for the index calculation.

doi:10.1371/journal.pone.0131161.g004

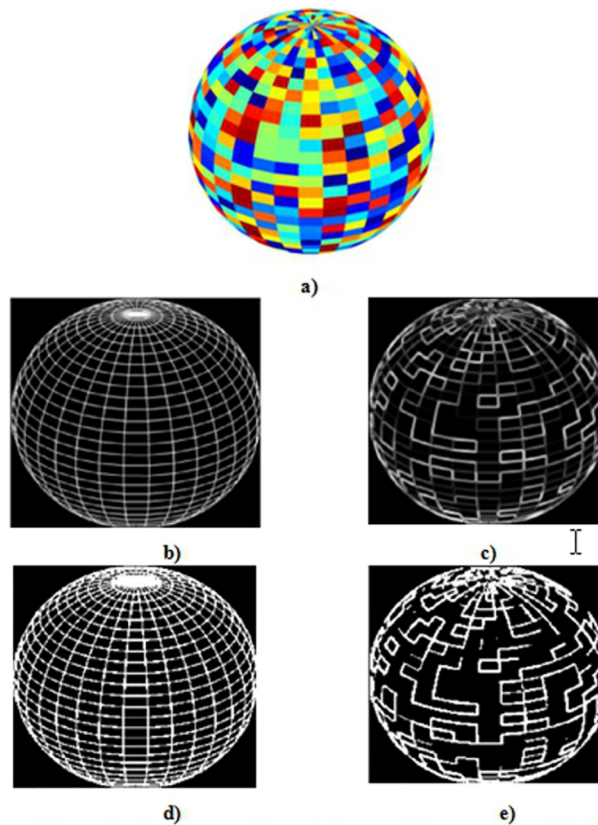
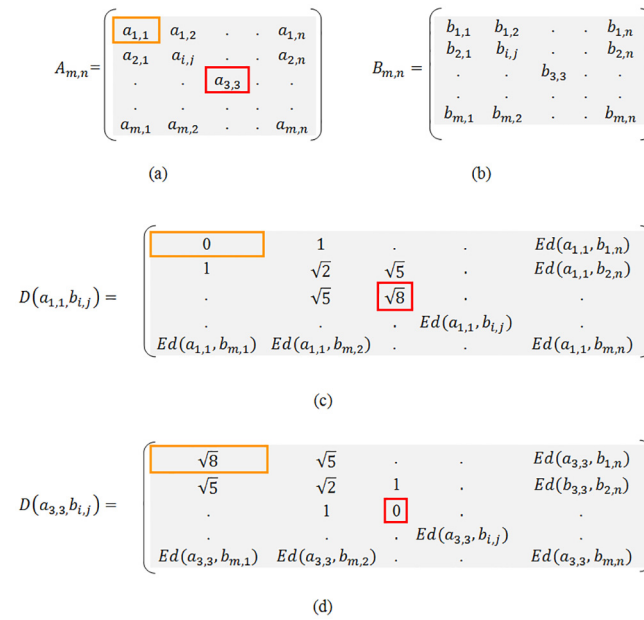


Fig 5. Euclidean distance matrices between images A and B. (a) Image A, (b) Image B, (c) Fuzzy Euclidean distance matrix between  $a_{11}$  and  $b_{1j}$ , (d) Fuzzy Euclidean distance matrix between  $a_{33}$  and  $b_{1j}$ .

doi:10.1371/journal.pone.0131161.g005



**Fig 6. Image in gray map.**

doi:10.1371/journal.pone.0131161.g006

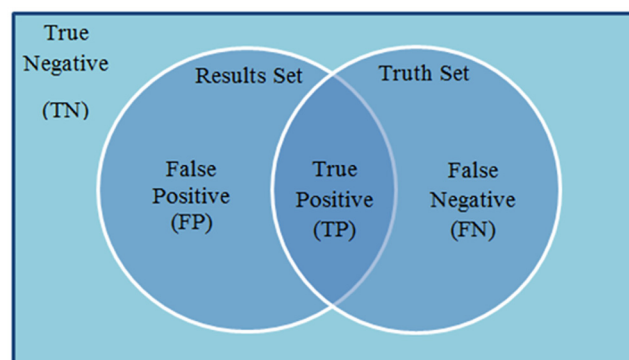
In order to test that our proposed method finds the correct index for a given pair of images, synthetic images were generated as shown in the Figs 9,10 and 11 in order to control the amount of edges, and also to manipulate gray tones of pixels and to show their differences.

Each image was a 10x10 matrix with values in the interval [0,1], where 0 indicates an absence of color or black and 1 means white, e.g. image in Fig 10(f) can be generated with matrix shown in Fig 12.

### Fuzzy sets operators for fuzzy images

Fuzzy sets operators between two fuzzy images can be defined according to the theory of fuzzy sets and the definitions in section 2.3.

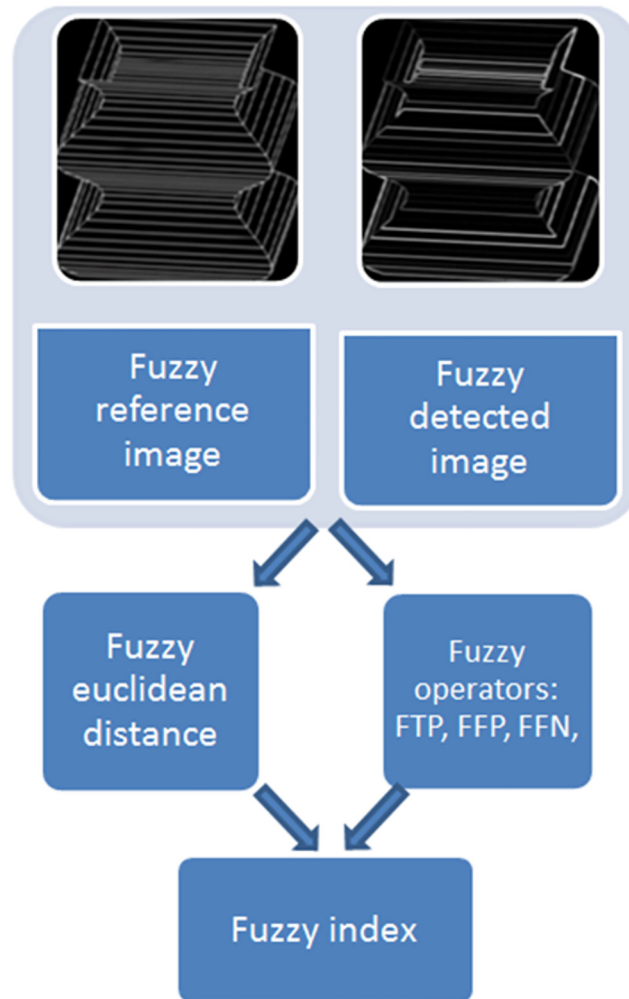
**Fuzzy true positive (FTP) between two fuzzy images.** The set of true positive pixels between fuzzy images A and B is defined as the intersection of A and B. Then, FTP can be calculated using Eq 7.



**Fig 7. Pixels matrix in gray map between 0 and 255.**

doi:10.1371/journal.pone.0131161.g007





**Fig 8. Pixels matrix after fuzzification with a triangular membership function**

doi:10.1371/journal.pone.0131161.g008

**Fuzzy false positive (FFP) between two fuzzy images.** The set of false positive pixels between fuzzy images  $A$  and  $B$  is defined as the bounded difference between  $A$  and  $B$ . Then,  $FFP$  can be calculated using Eq 9.

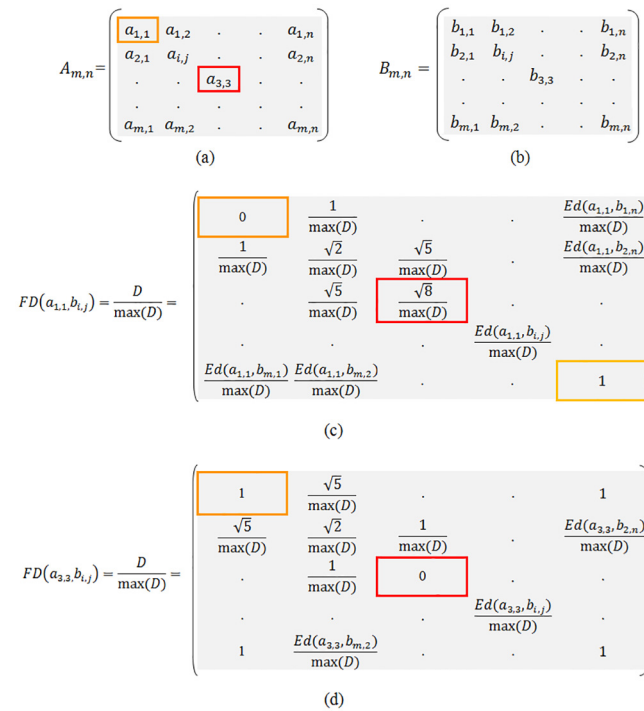
**Fuzzy false negative (FFN) between two fuzzy images.** The set of false negative pixels between fuzzy images  $A$  and  $B$  is defined as the bounded difference between  $B$  and  $A$ . Then,  $FFN$  can be calculated using Eq 10.

**Magnitude of a fuzzy image.** The magnitude of a fuzzy image can be defined as the scalar cardinality  $|A|$  of a fuzzy image using Eq 11.

### Calculation of the fuzzy index

Synthetic fuzzy images used in our proposal are matrices calculated by Eqs 14 and 15. The fuzzification of the images were made with triangular membership functions as described in Eq 13.

$$N_d = \{d_{11}, d_{12}, d_{13}, \dots, d_{mn}\} = \mu_{DI}(x) = \frac{x}{\max(DI)} \quad (14)$$

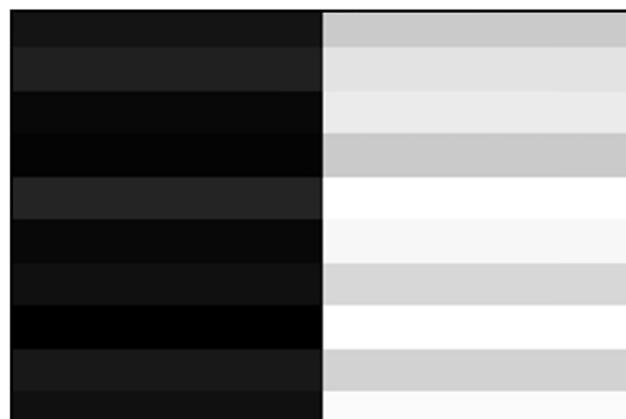


**Fig 9. Synthetic grid images.** (a) Synthetic image with values 0 and 1, (b) Synthetic image with values 0 and 0.8, (c) Synthetic image with values 0 and 0.6, (d) Synthetic image with values 0 and 0.4, (e) Synthetic image with values 0 and 0.2, (f) Synthetic image with values 0, 0.2, 0.4, 0.6, 0.8 and 1.

doi:10.1371/journal.pone.0131161.g009

$$N_i = \{i_{11}, i_{12}, i_{13}, \dots, i_{mn}\} = \mu_{II}(x) = \frac{x}{\max(II)} \quad (15)$$

Where  $DI$  are the edges detected image,  $II$  are the reference image (ground truth),  $N_d$  are the fuzzy detected image and  $N_i$  are the fuzzy reference edge image (Ground Truth or GT).



**Fig 10. Synthetic horizontal images.** (a) Synthetic image. (a) Synthetic image with values 0 and 1, (b) Synthetic image with values 0 and 0.8, (c) Synthetic image with values 0 and 0.6, (d) Synthetic image with values 0 and 0.4, (e) Synthetic image with values 0 and 0.2, (f) Synthetic image with values 0, 0.2, 0.4, 0.6, 0.8 and 1.

doi:10.1371/journal.pone.0131161.g010

30	30	30	30	30	200	200	200	200	200
40	40	40	40	40	220	220	220	220	220
20	20	20	20	20	230	230	230	230	230
15	15	15	15	15	200	200	200	200	200
45	45	45	45	45	250	250	250	250	250
20	20	20	20	20	240	240	240	240	240
28	28	28	28	28	210	210	210	210	210
10	10	10	10	10	250	250	250	250	250
36	36	36	36	36	205	205	205	205	205
25	25	25	25	25	245	245	245	245	245

**Fig 11. Synthetic vertical images.** (a) Synthetic image with values 0 and 1, (b) Synthetic image with values 0 and 0.8, (c) Synthetic image with values 0 and 0.6, (d) Synthetic image with values 0 and 0.4, (e) Synthetic image with values 0 and 0.2, (f) Synthetic image with values 0, 0.2, 0.4, 0.6, 0.8 and 1.

doi:10.1371/journal.pone.0131161.g011

In our test, we used synthetic images stored in matrices  $N_i$  and  $N_d$  represented in their fuzzy form. Since images were already fuzzy, we started calculating fuzzy Euclidean distance. Then, we integrate FTP, FFP, FFN and Euclidean distance (FD) for the most similar pixels to obtain FI using Eq 16.

$$FI = \frac{1}{m \cdot n \cdot \max(|N_i|, |N_d|)} \left( |FTP| \sum_{\substack{1 \leq i \leq m \\ 1 \leq j \leq n}} \left( \frac{1}{1 + FD_{ij}} \right) - |FFP| - |FFN| \right) \quad (16)$$

0.12	0.12	0.12	0.12	0.12	0.78	0.78	0.78	0.78	0.78
0.16	0.16	0.16	0.16	0.16	0.86	0.86	0.86	0.86	0.86
0.08	0.08	0.08	0.08	0.08	0.90	0.90	0.90	0.90	0.90
0.06	0.06	0.06	0.06	0.06	0.78	0.78	0.78	0.78	0.78
0.18	0.18	0.18	0.18	0.18	0.98	0.98	0.98	0.98	0.98
0.08	0.08	0.08	0.08	0.08	0.94	0.94	0.94	0.94	0.94
0.11	0.11	0.11	0.11	0.11	0.82	0.82	0.82	0.82	0.82
0.04	0.04	0.04	0.04	0.04	0.98	0.98	0.98	0.98	0.98
0.14	0.14	0.14	0.14	0.14	0.80	0.80	0.80	0.80	0.80
0.10	0.10	0.10	0.10	0.10	0.96	0.96	0.96	0.96	0.96

**Fig 12. Example for the matrix to generate the Fig 10(f).**

doi:10.1371/journal.pone.0131161.g012

### Algorithms

The algorithm for calculating the FI is divided into two phases: one that calculated the fuzzy Euclidean distance matrix FD and the second that calculated the FI. Then algorithms are shown in pseudo code as follows.

**Algorithm to compute the fuzzy Euclidean distance FD of the most similar pixels between  $N_i$  and  $N_d$ .** This algorithm takes the images  $N_i$  and  $N_d$  to calculate the fuzzy Euclidean distance of each pixel on  $N_i$  to the most similar pixel on  $N_d$ .

**Input:** Fuzzy images  $N_d = \{d_{11}, d_{12}, d_{13}, \dots, d_{mm}\}$  and  $N_i = \{i_{11}, i_{12}, i_{13}, \dots, i_{mm}\}$

**Output:** Fuzzy Euclidean distance matrix  $FD(N_i, N_d)$

1. For each pixel in  $N_i(x_1, y_1)$  do steps 2 to 4
2. Calculate the difference between the pixel  $N_i(x_1, y_1)$  and all pixels on  $N_d$  with

$$dif(x_1, y_1) = |N_i(x_1, y_1) - N_d(i, j)|$$

3. Found the most similar pixels to  $N_i(x_1, y_1)$  in  $N_d$  as those with the minimum value in  $dif(x_1, y_1)$
4. For each most similar pixel in  $N_d(x_2, y_2)$  do steps 5 and 6
5. Calculate the Euclidean distance between  $N_i(x_1, y_1)$  and  $N_d(x_2, y_2)$  with

$$D(x_1, y_1) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

6. Replace  $D(x_1, y_1)$  if a nearest most similar pixel is founded
7. Calculate the maximum Euclidean distance between two elements in D with

$$maxD = \sqrt{(m - 1)^2 + (n - 1)^2}$$

8. Calculate the Fuzzy Euclidean distance matrix with

$$FD(N_i, N_d) = \left( \frac{D(i, j)}{maxD} \right)$$

**Algorithm to compute the proposed fuzzy index.** This algorithm takes the images  $N_i$  and  $N_d$  and  $FD$  to calculate the fuzzy index  $FI$

**Input:** Fuzzy images  $N_d = \{d_{11}, d_{12}, d_{13}, \dots, d_{mm}\}$ ,  $N_i = \{i_{11}, i_{12}, i_{13}, \dots, i_{mm}\}$ , and the fuzzy Euclidean distance matrix  $FD_{m,n}(N_i, N_d)$ .

**Output:** Fuzzy index  $FI$ .

1. Calculate the scalar cardinality  $N_i$  and  $N_d$  with

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

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

2. Calculate  $FTP$  as the fuzzy intersection of  $N_i$  and  $N_d$  with

$$FTP = \min(N_i(i, j), N_d(i, j))$$

3. Calculate the scalar cardinality of  $FTP$  with

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

4. Calculate  $FFP$  as the bounded difference between  $N_i$  and  $N_d$  with

$$FFP = \max\{0, \{N_i(i, j) - N_d(i, j)\}\}$$

5. Calculate the scalar cardinality of  $FFP$  with

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

6. Calculate  $FFN$  as the bounded difference between  $N_d$  and  $N_i$

$$FFN = \max\{0, \{N_d(i, j) - N_i(i, j)\}\}$$

7. Calculate the scalar cardinality of  $FFN$

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

8. Calculate fuzzy index  $FI$  with

$$FI = \frac{1}{m \cdot n \cdot \max(|N_i|, |N_d|)} \left( |FTP| \sum_{\substack{1 \leq i \leq m \\ 1 \leq j \leq n}} \left( \frac{1}{1 + FD_{i,j}} \right) - |FFP| - |FFN| \right)$$

**Table 2. Results of the comparison of  $N_i$  and  $N_d$  of Fig 6 with non fuzzy sets and fuzzy sets operators.**

#	$N_i$	$N_d$	%TP	%FN	%FP	%FTP	%FFN	%FFP
1	<a href="#">Fig 10(a)</a>	<a href="#">Fig 10(a)</a>	100.00	0.00	0.00	<b>100.00</b>	<b>0.00</b>	<b>0.00</b>
2	<a href="#">Fig 10(a)</a>	<a href="#">Fig 10(b)</a>	100.00	0.00	0.00	<b>83.33</b>	<b>0.00</b>	<b>16.67</b>
3	<a href="#">Fig 10(a)</a>	<a href="#">Fig 10(c)</a>	100.00	0.00	0.00	<b>62.50</b>	<b>0.00</b>	<b>37.50</b>
4	<a href="#">Fig 10(a)</a>	<a href="#">Fig 10(d)</a>	100.00	0.00	0.00	<b>41.67</b>	<b>0.00</b>	<b>58.33</b>
5	<a href="#">Fig 10(a)</a>	<a href="#">Fig 10(e)</a>	100.00	0.00	0.00	<b>20.83</b>	<b>0.00</b>	<b>79.17</b>
6	<a href="#">Fig 10(a)</a>	<a href="#">Fig 10(f)</a>	80.00	20.00	0.00	<b>61.67</b>	<b>0.83</b>	<b>38.33</b>
7	<a href="#">Fig 10(b)</a>	<a href="#">Fig 10(a)</a>	100.00	0.00	0.00	<b>83.33</b>	<b>16.67</b>	<b>0.00</b>
8	<a href="#">Fig 10(b)</a>	<a href="#">Fig 10(b)</a>	100.00	0.00	0.00	<b>100.00</b>	<b>0.00</b>	<b>0.00</b>
9	<a href="#">Fig 10(b)</a>	<a href="#">Fig 10(c)</a>	100.00	0.00	0.00	<b>75.00</b>	<b>0.00</b>	<b>25.00</b>
10	<a href="#">Fig 10(b)</a>	<a href="#">Fig 10(d)</a>	100.00	0.00	0.00	<b>50.00</b>	<b>0.00</b>	<b>50.00</b>
11	<a href="#">Fig 10(b)</a>	<a href="#">Fig 10(e)</a>	100.00	0.00	0.00	<b>25.00</b>	<b>0.00</b>	<b>75.00</b>
12	<a href="#">Fig 10(b)</a>	<a href="#">Fig 10(f)</a>	80.00	20.00	0.00	<b>70.00</b>	<b>5.00</b>	<b>30.00</b>
13	<a href="#">Fig 10(c)</a>	<a href="#">Fig 10(a)</a>	100.00	0.00	0.00	<b>62.50</b>	<b>37.50</b>	<b>0.00</b>
14	<a href="#">Fig 10(c)</a>	<a href="#">Fig 10(b)</a>	100.00	0.00	0.00	<b>75.00</b>	<b>25.00</b>	<b>0.00</b>
15	<a href="#">Fig 10(c)</a>	<a href="#">Fig 10(c)</a>	100.00	0.00	0.00	<b>100.00</b>	<b>0.00</b>	<b>0.00</b>
16	<a href="#">Fig 10(c)</a>	<a href="#">Fig 10(d)</a>	100.00	0.00	0.00	<b>66.67</b>	<b>0.00</b>	<b>33.33</b>
17	<a href="#">Fig 10(c)</a>	<a href="#">Fig 10(e)</a>	100.00	0.00	0.00	<b>33.33</b>	<b>0.00</b>	<b>66.67</b>
18	<a href="#">Fig 10(c)</a>	<a href="#">Fig 10(f)</a>	80.00	20.00	0.00	<b>80.00</b>	<b>20.00</b>	<b>20.00</b>
19	<a href="#">Fig 10(d)</a>	<a href="#">Fig 10(a)</a>	100.00	0.00	0.00	<b>41.67</b>	<b>58.33</b>	<b>0.00</b>
20	<a href="#">Fig 10(d)</a>	<a href="#">Fig 10(b)</a>	100.00	0.00	0.00	<b>50.00</b>	<b>50.00</b>	<b>0.00</b>
21	<a href="#">Fig 10(d)</a>	<a href="#">Fig 10(c)</a>	100.00	0.00	0.00	<b>66.67</b>	<b>33.33</b>	<b>0.00</b>
22	<a href="#">Fig 10(d)</a>	<a href="#">Fig 10(d)</a>	100.00	0.00	0.00	<b>100.00</b>	<b>0.00</b>	<b>0.00</b>
23	<a href="#">Fig 10(d)</a>	<a href="#">Fig 10(e)</a>	100.00	0.00	0.00	<b>50.00</b>	<b>0.00</b>	<b>50.00</b>
24	<a href="#">Fig 10(d)</a>	<a href="#">Fig 10(f)</a>	80.00	20.00	0.00	<b>60.00</b>	<b>40.00</b>	<b>6.67</b>
25	<a href="#">Fig 10(e)</a>	<a href="#">Fig 10(a)</a>	100.00	0.00	0.00	<b>20.83</b>	<b>79.17</b>	<b>0.00</b>
26	<a href="#">Fig 10(e)</a>	<a href="#">Fig 10(b)</a>	100.00	0.00	0.00	<b>25.00</b>	<b>75.00</b>	<b>0.00</b>
27	<a href="#">Fig 10(e)</a>	<a href="#">Fig 10(c)</a>	100.00	0.00	0.00	<b>33.33</b>	<b>66.67</b>	<b>0.00</b>
28	<a href="#">Fig 10(e)</a>	<a href="#">Fig 10(d)</a>	100.00	0.00	0.00	<b>50.00</b>	<b>50.00</b>	<b>0.00</b>
29	<a href="#">Fig 10(e)</a>	<a href="#">Fig 10(e)</a>	100.00	0.00	0.00	<b>100.00</b>	<b>0.00</b>	<b>0.00</b>
30	<a href="#">Fig 10(e)</a>	<a href="#">Fig 10(f)</a>	80.00	20.00	0.00	<b>33.33</b>	<b>66.67</b>	<b>0.00</b>
31	<a href="#">Fig 10(f)</a>	<a href="#">Fig 10(a)</a>	80.00	0.00	20.00	<b>61.67</b>	<b>38.33</b>	<b>0.83</b>
32	<a href="#">Fig 10(f)</a>	<a href="#">Fig 10(b)</a>	80.00	0.00	20.00	<b>70.00</b>	<b>30.00</b>	<b>5.00</b>
33	<a href="#">Fig 10(f)</a>	<a href="#">Fig 10(c)</a>	80.00	0.00	20.00	<b>80.00</b>	<b>20.00</b>	<b>20.00</b>
34	<a href="#">Fig 10(f)</a>	<a href="#">Fig 10(d)</a>	80.00	0.00	20.00	<b>60.00</b>	<b>6.67</b>	<b>40.00</b>
35	<a href="#">Fig 10(f)</a>	<a href="#">Fig 10(e)</a>	80.00	0.00	20.00	<b>33.33</b>	<b>0.00</b>	<b>66.67</b>
36	<a href="#">Fig 10(f)</a>	<a href="#">Fig 10(f)</a>	100.00	0.00	0.00	<b>100.00</b>	<b>0.00</b>	<b>0.00</b>

doi:10.1371/journal.pone.0131161.t002

## Experiments and Results

In this section we compare the results of the calculation of the fuzzy sets operators proposed in this paper, with non fuzzy sets operators. The calculations were made with all the images shown in Figs 9, 10 and 11, and all of them yield similar results. In Table 2, are shown the results of fuzzy and non fuzzy sets operators for some comparisons between images in Fig 10 (S1 Dataset).

The first observation was the value for all operators when two identical images were compared. For all operators, the similarity between the two images is 100% as expected. When we observe images in Fig 10(a) and 10(f), it is possible to note great differences. In this comparison

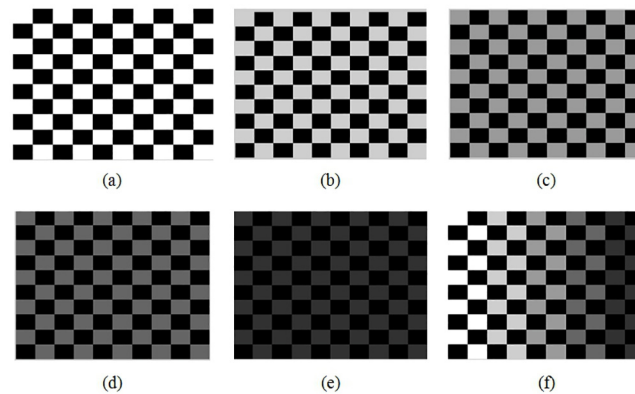
**Table 3. Results of the comparison of the images  $N_i$  and  $N_d$  on Fig 6 with the proposed index FI and non fuzzy metrics.**

#	$N_i$	$N_d$	Jl	DC	PFOM	FI
1	<a href="#">Fig 10(a)</a>	<a href="#">Fig 10(a)</a>	1.0000	1.0000	1.0000	<b>1.0000</b>
2	<a href="#">Fig 10(a)</a>	<a href="#">Fig 10(b)</a>	1.0000	1.0000	1.0000	<b>0.8317</b>
3	<a href="#">Fig 10(a)</a>	<a href="#">Fig 10(c)</a>	1.0000	1.0000	1.0000	<b>0.6213</b>
4	<a href="#">Fig 10(a)</a>	<a href="#">Fig 10(d)</a>	1.0000	1.0000	1.0000	<b>0.4108</b>
5	<a href="#">Fig 10(a)</a>	<a href="#">Fig 10(e)</a>	1.0000	1.0000	1.0000	<b>0.2004</b>
6	<a href="#">Fig 10(a)</a>	<a href="#">Fig 10(f)</a>	0.8000	0.8889	0.8000	<b>0.5549</b>
7	<a href="#">Fig 10(b)</a>	<a href="#">Fig 10(a)</a>	1.0000	1.0000	1.0000	<b>0.7416</b>
8	<a href="#">Fig 10(b)</a>	<a href="#">Fig 10(b)</a>	1.0000	1.0000	1.0000	<b>1.0000</b>
9	<a href="#">Fig 10(b)</a>	<a href="#">Fig 10(c)</a>	1.0000	1.0000	1.0000	<b>0.7475</b>
10	<a href="#">Fig 10(b)</a>	<a href="#">Fig 10(d)</a>	1.0000	1.0000	1.0000	<b>0.4950</b>
11	<a href="#">Fig 10(b)</a>	<a href="#">Fig 10(e)</a>	1.0000	1.0000	1.0000	<b>0.2425</b>
12	<a href="#">Fig 10(b)</a>	<a href="#">Fig 10(f)</a>	0.8000	0.8889	0.8000	<b>0.6458</b>
13	<a href="#">Fig 10(c)</a>	<a href="#">Fig 10(a)</a>	1.0000	1.0000	1.0000	<b>0.5537</b>
14	<a href="#">Fig 10(c)</a>	<a href="#">Fig 10(b)</a>	1.0000	1.0000	1.0000	<b>0.7475</b>
15	<a href="#">Fig 10(c)</a>	<a href="#">Fig 10(c)</a>	1.0000	1.0000	1.0000	<b>1.0000</b>
16	<a href="#">Fig 10(c)</a>	<a href="#">Fig 10(d)</a>	1.0000	1.0000	1.0000	<b>0.6633</b>
17	<a href="#">Fig 10(c)</a>	<a href="#">Fig 10(e)</a>	1.0000	1.0000	1.0000	<b>0.3267</b>
18	<a href="#">Fig 10(c)</a>	<a href="#">Fig 10(f)</a>	0.8000	0.8889	0.8000	<b>0.7449</b>
19	<a href="#">Fig 10(d)</a>	<a href="#">Fig 10(a)</a>	1.0000	1.0000	1.0000	<b>0.3987</b>
20	<a href="#">Fig 10(d)</a>	<a href="#">Fig 10(b)</a>	1.0000	1.0000	1.0000	<b>0.4950</b>
21	<a href="#">Fig 10(d)</a>	<a href="#">Fig 10(c)</a>	1.0000	1.0000	1.0000	<b>0.6633</b>
22	<a href="#">Fig 10(d)</a>	<a href="#">Fig 10(d)</a>	1.0000	1.0000	1.0000	<b>1.0000</b>
23	<a href="#">Fig 10(d)</a>	<a href="#">Fig 10(e)</a>	1.0000	1.0000	1.0000	<b>0.4950</b>
24	<a href="#">Fig 10(d)</a>	<a href="#">Fig 10(f)</a>	0.8000	0.8889	0.8000	<b>0.5518</b>
25	<a href="#">Fig 10(e)</a>	<a href="#">Fig 10(a)</a>	1.0000	1.0000	1.0000	<b>0.1928</b>
26	<a href="#">Fig 10(e)</a>	<a href="#">Fig 10(b)</a>	1.0000	1.0000	1.0000	<b>0.2334</b>
27	<a href="#">Fig 10(e)</a>	<a href="#">Fig 10(c)</a>	1.0000	1.0000	1.0000	<b>0.3145</b>
28	<a href="#">Fig 10(e)</a>	<a href="#">Fig 10(d)</a>	1.0000	1.0000	1.0000	<b>0.4950</b>
29	<a href="#">Fig 10(e)</a>	<a href="#">Fig 10(e)</a>	1.0000	1.0000	1.0000	<b>1.0000</b>
30	<a href="#">Fig 10(e)</a>	<a href="#">Fig 10(f)</a>	0.8000	0.8889	0.8000	<b>0.2936</b>
31	<a href="#">Fig 10(f)</a>	<a href="#">Fig 10(a)</a>	0.8000	0.8889	0.9592	<b>0.5763</b>
32	<a href="#">Fig 10(f)</a>	<a href="#">Fig 10(b)</a>	0.8000	0.8889	0.9592	<b>0.6914</b>
33	<a href="#">Fig 10(f)</a>	<a href="#">Fig 10(c)</a>	0.8000	0.8889	0.9592	<b>0.7902</b>
34	<a href="#">Fig 10(f)</a>	<a href="#">Fig 10(d)</a>	0.8000	0.8889	0.9592	<b>0.5953</b>
35	<a href="#">Fig 10(f)</a>	<a href="#">Fig 10(e)</a>	0.8000	0.8889	0.9592	<b>0.3267</b>
36	<a href="#">Fig 10(f)</a>	<a href="#">Fig 10(f)</a>	1.0000	1.0000	1.0000	<b>1.0000</b>

doi:10.1371/journal.pone.0131161.t003

results, we can observe that always the fuzzy operators found less true positives, more false positives and more false negatives than the non fuzzy operators. This result means that the non fuzzy operators do not evaluate the image as we see it, because the binarization forces the values to the extremes of the interval [0,1]. The fuzzy operators instead, evaluate the image as we see it, with all the details, because it is not binarized.

Once we observe the results of fuzzy operators above, we compute the FI using algorithms described in section 4.5, and results are shown in Table 3. The indices are in the interval [0,1], where 1 means that images compared are identical.



**Fig 13. Reference and detected edges images of a sphere.** a) Color image. b) Reference edges image without binarization, c) Detected edges image without binarization, d) Reference edges image after binarization, e) Detected edges image after binarization.

doi:10.1371/journal.pone.0131161.g013

Once again we can observe that the binarization of the images made before the calculation of FOM, JI and DC, forces the comparison results. For non fuzzy operators, we can observe 100% in similitude for images evidently different.

Synthetic images can be generated using some computer program, as the sphere shown in Fig 13. This image was generated using the function sphere of Matlab [39], this is an image of 200x200 (S2 Dataset).

The results on Table 4 shown that the non fuzzy indices FOM, JI and DC are identical, because the result is 1. But we can observe the images compared and found many differences. FI index calculates a very low similitude between the compared images, because of the few pixels correctly classified as edges, calculated as FTP. The non fuzzy operator TP instead, calculated a 100% of similitude between the images.

With the fuzzy index proposed, we can compare two fuzzy images with high precision. Then if one of the images is a ground truth of edges image, we can evaluate the correct edges detection in fuzzy images.

### Conclusions

For a first conclusion of great importance, if the evaluation of an image without binarizing is carried out with more information about the image, you can have more pixels to work with.

The edge images without binarization used in classification systems allow us to achieve better recognition rates than the use of binarized images. Then the ability to evaluate the edges images in their fuzzy form can be used to preprocess data sets for optimal results.

Another important point is that we can use the simplest form of fuzzy logic to implement the evolution of images. We can notice in the results obtained that there is a great difference between the estimates obtained with traditional metrics because the FI we can give an index according to their gray tone and not simply dismiss it through a specified threshold.

These results show us that fuzzy logic is a good alternative in image processing and applied in many areas where it is extremely important edge detection.

**Table 4. Results of the comparison of the picture  $N_i$  and  $N_d$ .**

$N_i$	$N_d$	%TP	%FN	%FP	%FTP	%FFN	%FFP	JI	DC	PFOM	FI
<a href="#">Fig 13(b)</a>	<a href="#">Fig 13(c)</a>	44.76	55.23	30.56	15.89	74.09	84.10	0.3428	0.5106	0.6957	<b>0.1588</b>

doi:10.1371/journal.pone.0131161.t004



The proposed fuzzy index is highly recommended, given the results. This fuzzy index includes parameters that other indices do not consider in their calculations, and allows us to compare two images represented as fuzzy sets.

Only synthetic images were used to calibrate proposed index FI, with very good results. Next step is to use it on real images. We are currently optimizing a fuzzy border detector to generate reference images that will help us evaluate border images given by our proposed index.

## Supporting Information

**S1 Dataset. Matrices used for the tests that show Tables 2 and 3.**

(ZIP)

**S2 Dataset. Images used for the tests that show Table 4.**

(ZIP)

## Acknowledgments

We thank the MyDCI program of the Division of Graduate Studies and Research, UABC, and Tijuana Institute of Technology the financial support provided by our sponsor CONACYT.

## Author Contributions

Conceived and designed the experiments: FP OM PM JC AR OC. Performed the experiments: FP OM JC. Analyzed the data: FP OM PM JC AR OC. Contributed reagents/materials/analysis tools: FP OM PM JC AR OC. Wrote the paper: FP OM PM JC AR OC.

## References

1. Pratt WK. Digital Image Processing: PIKS Inside: John Wiley & Sons, Inc.; 2001. 758 p.
2. Mendoza O, Melin P, Licea G, editors. A New Method for Edge Detection in Image Processing Using Interval Type-2 Fuzzy Logic. GrC; 2007: IEEE; dblp.
3. Mendoza O, Licea G, Melin P, editors. Modular Neural Networks and Type-2 Fuzzy Logic for Face Recognition. Fuzzy Information Processing Society, 2007 NAFIPS '07 Annual Meeting of the North American; 2007 24–27 June 2007.
4. Mendel JM. A quantitative comparison of interval type-2 and type-1 fuzzy logic systems: First results. FUZZ-IEEE. conf/fuzzIEEE/Mendel10: IEEE; 2010. p. 1–8.
5. Castillo O, Melin P, Pedrycz W, Kacprzyk J. Preface to the special section on new trends on pattern recognition with fuzzy models. Fuzzy Sets and Systems. 2010; 161(1):1–2.
6. Melin P, Gonzalez CI, Castro JR, Mendoza O, Castillo O. Edge-Detection Method for Image Processing Based on Generalized Type-2 Fuzzy Logic. Fuzzy Systems, IEEE Transactions on. 2014; 22(6):1515–25.
7. Lopes NV, Mogadouro do Couto PA, Bustince H, Melo-Pinto P. Automatic Histogram Threshold Using Fuzzy Measures. Image Processing, IEEE Transactions on. 2010; 19(1):199–204.
8. Bustince H, Barrenechea E, Pagola M, Fernandez J. Interval-valued fuzzy sets constructed from matrices: Application to edge detection. Fuzzy Sets and Systems. 2009; 160(13):1819–40.
9. Pagola M, Lopez-Molina C, Fernandez J, Barrenechea E, Bustince H. Interval Type-2 Fuzzy Sets Constructed From Several Membership Functions: Application to the Fuzzy Thresholding Algorithm. Fuzzy Systems, IEEE Transactions on. 2013; 21(2):230–44.
10. Barrenechea E, Bustince H, De Baets B, Lopez-Molina C. Construction of Interval-Valued Fuzzy Relations With Application to the Generation of Fuzzy Edge Images. Fuzzy Systems, IEEE Transactions on. 2011; 19(5):819–30.
11. Galar M, Fernandez J, Beliakov G, Bustince H. Interval-Valued Fuzzy Sets Applied to Stereo Matching of Color Images. Image Processing, IEEE Transactions on. 2011; 20(7):1949–61.

12. Gonzalez CI, Castro JR, Martinez GE, Melin P, Castillo O, editors. A new approach based on generalized type-2 fuzzy logic for edge detection. *IFSA World Congress and NAFIPS Annual Meeting (IFSA/NAFIPS)*, 2013 Joint; 2013 24–28 June 2013.
13. Mendoza O, Melin P, Licea G. Interval type-2 fuzzy logic for edges detection in digital images. *International Journal of Intelligent Systems*. 2009; 24(11):1115–33.
14. Bustince H, Pagola M, Barrenechea E. Construction of fuzzy indices from fuzzy DI-subsethood measures: Application to the global comparison of images. *Information Sciences*. 2007; 177(3):906–29.
15. Gonzalez C, Castro JR, Melin P, Castillo O, editors. An edge detection method based on generalized type-2 fuzzy logic. *Advances in Type-2 Fuzzy Logic Systems (T2FUZZ)*, 2013 IEEE Symposium on; 2013 16–19 April 2013.
16. Dongrui W, Mendel JM. On the Continuity of Type-1 and Interval Type-2 Fuzzy Logic Systems. *Fuzzy Systems, IEEE Transactions on*. 2011; 19(1):179–92.
17. Prieto MS, Allen AR. A similarity metric for edge images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*. 2003; 25(10):1265–73.
18. Shattuck DW, Prasad G, Mirza M, Narr KL, Toga AW. Online resource for validation of brain segmentation methods. *NeuroImage*. 2009; 45(2):431–9. doi: [10.1016/j.neuroimage.2008.10.066](https://doi.org/10.1016/j.neuroimage.2008.10.066) PMID: [19073267](https://pubmed.ncbi.nlm.nih.gov/19073267/)
19. Castro JR, Castillo O, Melin P, editors. An Interval Type-2 Fuzzy Logic Toolbox for Control Applications. *Fuzzy Systems Conference, 2007 FUZZ-IEEE 2007 IEEE International*; 2007 23–26 July 2007.
20. Abdou IE, Pratt WK. Quantitative design and evaluation of enhancement/thresholding edge detectors. *Proceedings of the IEEE*. 1979; 67(5):753–63.
21. Mendoza O, Melin P, Castillo O, editors. Neural networks recognition rate as index to compare the performance of fuzzy edge detectors. *Neural Networks (IJCNN), The 2010 International Joint Conference on*; 2010 18–23 July 2010.
22. Mendoza O, Melin P, Castillo O, Castro J. Comparison of Fuzzy Edge Detectors Based on the Image Recognition Rate as Performance Index Calculated with Neural Networks. In: Melin P, Kacprzyk J, Pedrycz W, editors. *Soft Computing for Recognition Based on Biometrics. Studies in Computational Intelligence*. 312: Springer Berlin Heidelberg; 2010. p. 389–99.
23. Moreno JMD. *Introducción a la topología de los espacios métricos*: Universidad de Cádiz Servicio de Publicaciones; 1998.
24. Mendoza O, Melin P, Castillo O. Interval type-2 fuzzy logic and modular neural networks for face recognition applications. *Applied Soft Computing*. 2009; 9(4):1377–87.
25. Melin P, Mendoza O, Castillo O. Face Recognition With an Improved Interval Type-2 Fuzzy Logic Sugeno Integral and Modular Neural Networks. *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on*. 2011; 41(5):1001–12.
26. Mendoza O, Melin P, Licea G. A hybrid approach for image recognition combining type-2 fuzzy logic, modular neural networks and the Sugeno integral. *Information Sciences*. 2009; 179(13):2078–101.
27. Melin P, Gonzalez C, Gonzalez F, Castillo O, editors. Face recognition using modular neural networks and fuzzy Sugeno integral for response integration. *Neural Networks, 2005 IJCNN '05 Proceedings 2005 IEEE International Joint Conference on*; 2005 31 July–4 Aug. 2005.
28. Melin P. Image Processing and Pattern Recognition with Mamdani Interval Type-2 Fuzzy Inference Systems. In: Trillas E, Bonissone PP, Magdalena L, Kacprzyk J, editors. *Combining Experimentation and Theory. Studies in Fuzziness and Soft Computing*. 271: Springer Berlin Heidelberg; 2012. p. 179–90.
29. Melin P. *Modular Neural Networks and Type-2 Fuzzy Systems for Pattern Recognition*. Berlin: Springer-Verlag Berlin Heidelberg; 2012. 214 p.
30. Mendoza O, Melin P, Licea G, editors. Type-2 Fuzzy Systems for Improving Training Data and Decision Making in Modular Neural Networks for Image Recognition. *Neural Networks, 2007 IJCNN 2007 International Joint Conference on*; 2007 12–17 Aug. 2007.
31. Castro JR, Castillo O, Melin P, Rodríguez-Díaz A. A hybrid learning algorithm for a class of interval type-2 fuzzy neural networks. *Information Sciences*. 2009; 179(13):2175–93.
32. Melin P, Mendoza O, Castillo O. An improved method for edge detection based on interval type-2 fuzzy logic. *Expert Systems with Applications*. 2010; 37(12):8527–35.
33. Melin P, Castillo O. A review on type-2 fuzzy logic applications in clustering, classification and pattern recognition. *Applied Soft Computing*. 2014; 21(0):568–77.
34. Melin P, Castillo O. A review on the applications of type-2 fuzzy logic in classification and pattern recognition. *Expert Systems with Applications*. 2013; 40(13):5413–23.

35. Melin P, Sánchez D, Castillo O. Genetic optimization of modular neural networks with fuzzy response integration for human recognition. *Information Sciences*. 2012; 197(0):1–19.
36. Lopez-Molina C, De Baets B, Bustince H. Quantitative error measures for edge detection. *Pattern Recognition*. 2013; 46(4):1125–39.
37. Bustince H, Barrenechea E, Pagola M, Fernandez J, Sanz J. Comment on: “Image thresholding using type II fuzzy sets”. Importance of this method. *Pattern Recognition*. 2010; 43(9):3188–92.
38. Zadeh LA. Fuzzy sets. *Information and Control*. 1965; 8(3):338–53.
39. MathWorks. Matlab Documentation 2015. Available: <http://www.mathworks.com/help/matlab/graphics.html>.