Gated Local Adaptive Binarization using Supervised Learning

Javier Fumanal-Idocin[1], Juan Uriarte[1], Borja de la Osa[1], Francesco Bardozzo[2], Javier Fernández[1] and Humberto Bustince[1]

[1]*Estadística, Informática, Matemáticas, Public University of Navarre*
[2]*Neuronelab, DISA-MIS, University DegliStudy di Salerno*

Abstract

Image thresholding is one of the most popular problems in image processing. However, changes in lightning and contrast in an image can cause trouble for the existing algorithms that use a global threshold for all the image. A solution for this problem is the adaptive thresholding, in which an image can have different thresholds for different parts of the image. Yet, the problem of choosing the most suitable threshold for each region of the image is still open. In this paper we present the Gated Local Adaptive Binarization algorithm, in which we choose the most appropriate threshold for each region of the image using a logistic regression. Our results show that this algorithm can effectively learn the most appropriate threshold in each situation, and beats other adaptive binarization solutions for a standard dataset in the literature.

Keywords

Fuzzy logic, Image Thresholding, Image Processing, Aggregation functions

1. Introduction

Image processing ins one of the most important research topics in the computer science areas [1, 2, 3]. Many problems have been studied in this area, like classification [4, 5, 6] and segmentation of different objects in an image [7]. One of the most researched topics in image processing is image thresholding [8, 9], also called image binarization, which consists of discriminating the objects in an image from the background.

The most popular binarization algorithm is the Otsu algorithm [10], and many other popular algorithms have been proposed [11, 12, 13]. All of these algorithms work by establishing a global threshold for the whole image. However, this strategy results in poor performance when there are changes in the lightning and contrast of the image. In that case, the same threshold cannot adapt itself to the different conditions in the image.

Adaptive thresholding was proposed in [14] as a mean to solve this problem, by choosing a different threshold for the different parts of the image. This algorithm works by precomputing

WILF 2021: International Workshop on Fuzzy Logic and Applications
javier.fumanal@unavarra.es (J. Fumanal-Idocin); juan@losuriarte.es (J. Uriarte); borjajose.delaosa@unavarra.es (B. d. l. Osa); fbardozzo@unisa.it (F. Bardozzo); fcojavier.fernandez@unavara.es (J. Fernández); bustince@unavara.es (H. Bustince)
https://fuminides.netlify.app/ (J. Fumanal-Idocin)
0000-0002-0644-1355 (J. Fumanal-Idocin); 0000-0003-0199-6623 (F. Bardozzo)

the integral image from the original one, and sliding a 3×3 window through all integral image, where each window will use a different threshold according to its own characteristics. The Adaptive thresholding was further developed in [15]. In that work, the authors focused on the possible improvements using aggregation functions, and proposed a new generalization of the Sugeno integral. Indeed, aggregation functions [16] have been successfully used in many decision making problems [17, 18], brain computer interface classification tasks [19, 20], community detection [21] and other image processing tasks [22, 23, 24, 25].

However, there were some limits in the improvements of the algorithm proposed in [15], as the fusion processes are limited by the quality of the data to fuse with the tested integrals. In this work we propose a new algorithm to perform dynamic thresholding, the Gated Local Adaptive Binarization (GLAB), that uses supervised learning to improve the results obtained by other adaptive algorithms, based on a "gated" fusion process [26]. We also present a series of extensions to the FLAT algorithm using other aggregation functions to compare to our newly developed GLAB.

The rest of this paper goes as follows: Section 2 describes the algorithm proposed in [15], the different aggregation functions used to extend it, and the proposed GLAB. Section 3 describes our experiments and illustrates the results obtained. Finally, Section 4 details our conclusions for this work and future lines of research.

2. Methods

In this section we discuss the Fuzzy Local Adaptive Thresholding (FLAT) algorithm, and the proposed Enhanced Local Adaptive Thresholding.

2.1. Fuzzy Local Adaptive Thresholding

The FLAT algorithm was proposed by Bardozzo et al. in [15] to improve the results of the local Adaptive thresholding algorithm proposed in [14] using a new generalization of the Sugeno integral. The FLAT algorithm consist of computing the fuzzy integral image of the original image, and then perform the Adaptive binarization on the computed integral image.

2.2. Computing the fuzzy integral image

To compute the Fuzzy Integral Image, F_A, we first compute the integral image S, using the formula:

$$S(x, y) = p(x, y) + S(x, y - 1) + S(x - 1, y) - S(x - 1, y - 1) \quad (1)$$

where p is the original image, and with convention $S(0, \cdot) = 0$ and $S(\cdot, 0) = 0$. Then, we compute the F_A as follows:

$$F_A(x, y) = A(S(x, y), S(x, y - 1), S(x - 1, y), S(x - 1, y - 1)) \quad (2)$$

where A is an aggregation function. The best result obtained in [15] was obtained using the following Sugeno-like integral:

$$A = \sum_{i=1}^{n} \left(x_{\sigma(i)} \cdot \mu(E_i) \right) \tag{3}$$

where x_σ is a permutation of x such that $x_{\sigma_i} < x_{\sigma_i+1}$ for $i \in \{1, \ldots, |x|\}$, $E_i = \{(i), \ldots, (n)\}$ and $\mu(E_i)$ is a fuzzy measure [27] that follows the expression:

$$\mu(E_i) = \frac{|E_i|}{n} \tag{4}$$

2.3. Computing adaptive binarization

We compute the threshold for each window of size $a_i \times a_j$, usually a 3×3. For each of these windows we do as follows:

1. Compute the area of the window:

$$p_{area} = a_i \times a_j \tag{5}$$

2. Compute the area in the fuzzy integral image:

$$p_s = F_A[y_1, x_1] - F_A[y_0, x_1] - F_a[y_1, x_0] + F_A[y_0, x_0] \tag{6}$$

3. Compute the threshold using the ratio:

$$threshold = \frac{p_s}{p_{area}} \tag{7}$$

where x_0, x_1, y_0, y_1 are the corners of the window. So, for each pixel we compute the corresponding threshold using the $a_1 \times a_2$ window where that pixel is the center, cropped in the case of the borders of the image.

2.4. Gated Local Adaptive Thresholding

The Gated Local Adaptive Thresholding (GLAB) is a modification of the FLAT algorithm in which the final threshold is computed using a logistic regression, using the values in the $a_1 \times a_2$ window as an input vector, I, we compute the resulting threshold using the expression:

$$threshold = \sigma(WI + b) \tag{8}$$

where W is the weight vector and b the bias to learn respectively, and σ is the logistic function.

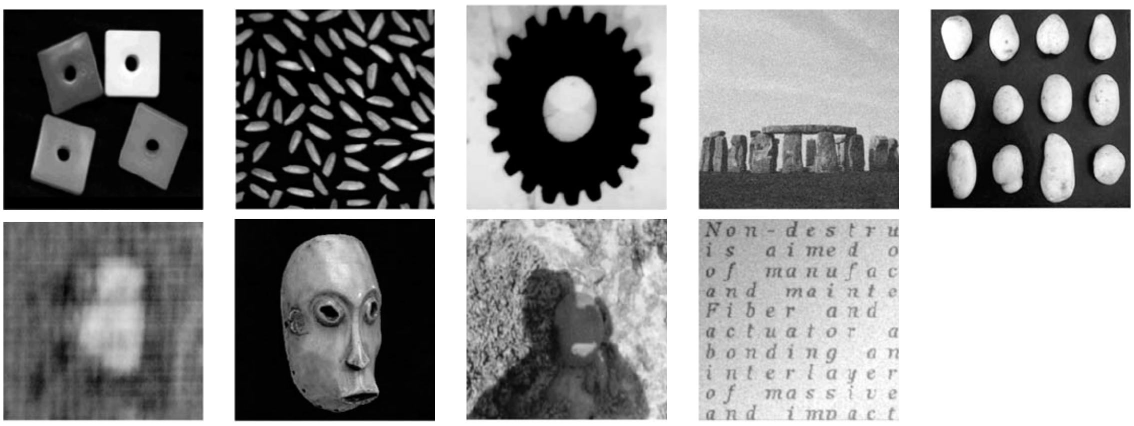

Figure 1: The dataset of 9 images used for this experimentation.

2.5. Evaluation Metrics

As a evaluation metric, we have used the F_1 score, comparing the obtained thresholding solution with the ground truth label for each image. The F_1 score is computed using the following formulas, using the concepts of precision and recall:

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive} \tag{9}$$

$$Recall = \frac{TruePositive}{TruePositive + FalseNegative} \tag{10}$$

$$F_1 = 2\frac{precision * recall}{precision + recall} \tag{11}$$

3. Experimentation

In this work we have taken the image thresholding dataset taken from [28], that consists of 9 different images in grayscale with ground-truth labels for each pixels. We show the images in Figure 1.

We studied the effect of different global threshold in the FLAT algorithm, in order to study the relevance of this parameter in the FLAT algorithm, and then, we studied the performance of the FLAT and GLAB algorithm for the images displayed in Figure 1.

3.1. Studying different thresholds in Fuzzy Local Adaptive Thresholding

First, we studied how setting different fixed threshold could impact the performance of the FLAT algorithm. This is contrary to the local nature of the FLAT algorithm, but is representative of the performance of the integral image-based thresholding for the global image, and can give us an intuition of the expected changes in performance when changing the threshold value.

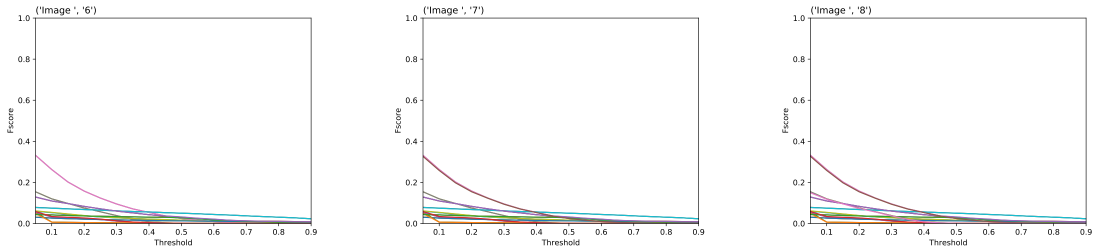

Figure 2: Performance for different FLAT executions using different threshold values for different aggreagtion functions.

Algorithm	Agg.	Img. 1	Img. 2	Img. 3	Img. 4	Img. 5	Img. 6	Img. 7	Img. 8	Img. 9	Img. 10	Average
FLAT	Sugeno	0.56	0.58	0.93	0.77	0.55	0.28	0.53	0.77	0.92	0.89	0.68
	Choquet	0.82	0.89	0.98	0.78	0.76	0.29	0.68	0.79	0.93	0.89	0.78
	C_{F_1,F_2}	0.56	0.58	0.93	0.77	0.55	0.28	0.53	0.77	0.93	0.89	0.68
	F-Sugeno	0.56	0.58	0.93	0.77	0.55	0.28	0.53	0.77	0.93	0.89	0.68
	CF	0.81	0.89	0.98	0.78	0.76	0.29	0.66	0.79	0.93	0.89	0.78
GLAB	Sugeno	0.53	0.52	0.67	0.77	0.55	0.29	0.53	0.77	0.93	0.89	0.66
	Choquet	0.98	0.88	0.97	0.98	0.96	0.90	0.97	0.95	0.93	0.89	**0.95**
	C_{F_1,F_2}	0.53	0.52	0.67	0.77	0.55	0.29	0.53	0.77	0.93	0.89	0.66
	F-Sugeno	0.53	0.52	0.67	0.77	0.55	0.29	0.53	0.77	0.93	0.89	0.66
	CF	0.97	0.87	0.96	0.98	0.95	0.28	0.96	0.95	0.93	0.89	0.87

Table 1
Results for all the images in the dataset and the average performance for the FLAT and GLAB algorithm, using different aggregation functions to construct the Fuzzy Integral Image.

Some of the results of this study are illustrated in Figure 2. We can determine that there is an evident impact in the chosen threshold for each image, and that for the different combinations of aggregations and images tested, the optimal threshold seem to vary a lot, which is an indication of the suitability of the GLAB to optimize the threshold for each one.

3.2. Results of Gated Local Adaptive Thresholding

To train and evaluate the performance of the GLAB we first computed the fuzzy integral image of each of the original images. Then, we divided each image and the corresponding fuzzy integral image in non-overlapping windows of 3×3. Finally, we split the 80% of the images to train the model and the rest to evaluate the performance of the GLAB.

In Table 1 we show the results for the GLAB and FLAT algorithms for the evaluation windows corresponding to each image. We found results to be much higher than those obtained using the FLAT, and that the best case was using the Choquet integral to construct the fuzzy integral image, and then using the GLAB to perform the binarization.

4. Conclusions and Future Lines

In this work we have presented the Gated version of the FLAT algorithm, the GLAB. The GLAB computes the local threshold from each image using a logistic regression that learns the most appropriate threshold for each region of the image. We found the results using GLAB to be superior to the FLAT algorithm, and the GLAB constructing the fuzzy integral image using the Choquet integral.

Future research shall study the use of further aggregation functions, and to study the use of the GLAB algorithm in a Convolutional Neural Network.

References

[1] M. M. Petrou, C. Petrou, Image processing: the fundamentals, John Wiley & Sons, 2010.

[2] E. R. Dougherty, Digital image processing methods, CRC Press, 2020.

[3] Q. Chen, J. Xu, V. Koltun, Fast image processing with fully-convolutional networks, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 2497–2506.

[4] J. Zhang, Y. Xie, Q. Wu, Y. Xia, Medical image classification using synergic deep learning, Medical image analysis 54 (2019) 10–19.

[5] D. Ciregan, U. Meier, J. Schmidhuber, Multi-column deep neural networks for image classification, in: 2012 IEEE conference on computer vision and pattern recognition, IEEE, 2012, pp. 3642–3649.

[6] I. Goodfellow, Y. Bengio, A. Courville, Y. Bengio, Deep learning, volume 1, MIT press Cambridge, 2016.

[7] O. Ronneberger, P. Fischer, T. Brox, U-net: Convolutional networks for biomedical image segmentation, in: International Conference on Medical image computing and computer-assisted intervention, Springer, 2015, pp. 234–241.

[8] M. Sezgin, B. Sankur, Survey over image thresholding techniques and quantitative performance evaluation, Journal of Electronic imaging 13 (2004) 146–165.

[9] T. Y. Goh, S. N. Basah, H. Yazid, M. J. A. Safar, F. S. A. Saad, Performance analysis of image thresholding: Otsu technique, Measurement 114 (2018) 298–307.

[10] P. L. Rosin, E. Ioannidis, Evaluation of global image thresholding for change detection, Pattern recognition letters 24 (2003) 2345–2356.

[11] H. Bustince, E. Barrenechea, M. Pagola, Image thresholding using restricted equivalence functions and maximizing the measures of similarity, Fuzzy Sets and Systems 158 (2007) 496–516.

[12] M. P. De Albuquerque, I. A. Esquef, A. G. Mello, Image thresholding using tsallis entropy, Pattern Recognition Letters 25 (2004) 1059–1065.

[13] L.-K. Huang, M.-J. J. Wang, Image thresholding by minimizing the measures of fuzziness, Pattern recognition 28 (1995) 41–51.

[14] D. Bradley, G. Roth, Adaptive thresholding using the integral image, Journal of graphics tools 12 (2007) 13–21.

[15] F. Bardozzo, B. De La Osa, L. Horanská, J. Fumanal-Idocin, M. delli Priscoli, L. Troiano,

R. Tagliaferri, J. Fernandez, H. Bustince, Sugeno integral generalization applied to improve adaptive image binarization, Information Fusion 68 (2021) 37–45.

[16] G. Beliakov, H. B. Sola, T. C. Sánchez, A practical guide to averaging functions, Springer, 2016.

[17] M. Papčo, I. Rodríguez-Martínez, J. Fumanal-Idocin, A. H. Altalhi, H. Bustince, A fusion method for multi-valued data, Information Fusion 71 (2021) 1–10.

[18] Z. Takáč, J. Fernandez, J. Fumanal, C. Marco-Detchart, I. Couso, G. Dimuro, H. Santos, H. Bustince, Distances between interval-valued fuzzy sets taking into account the width of the intervals, in: 2019 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), IEEE, 2019, pp. 1–6.

[19] J. Fumanal-Idocin, Y.-K. Wang, C.-T. Lin, J. Fernández, J. A. Sanz, H. Bustince, Motor-imagery-based brain-computer interface using signal derivation and aggregation functions, IEEE Transactions on Cybernetics (2021).

[20] J. Fumanal-Idocin, Z. Takac, J. Fernandez, J. A. Sanz, H. Goyena, C.-T. Lin, Y. Wang, H. Bustince, Interval-valued aggregation functions based on moderate deviations applied to motor-imagery-based brain computer interface, IEEE Transactions on Fuzzy Systems (2021) 1–1.

[21] J. Fumanal-Idocin, A. Alonso-Betanzos, O. Cordón, H. Bustince, M. Minárová, Community detection and social network analysis based on the italian wars of the 15th century, Future Generation Computer Systems 113 (2020) 25–40.

[22] C. Lopez-Molina, H. Bustince, J. Fernández, P. Couto, B. De Baets, A gravitational approach to edge detection based on triangular norms, Pattern Recognition 43 (2010) 3730–3741.

[23] M. Delić, L. Nedović, E. Pap, Extended power-based aggregation of distance functions and application in image segmentation, Information sciences 494 (2019) 155–173.

[24] H. Jégou, A. Zisserman, Triangulation embedding and democratic aggregation for image search, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2014, pp. 3310–3317.

[25] D. Paternain, J. Fernández, H. Bustince, R. Mesiar, G. Beliakov, Construction of image reduction operators using averaging aggregation functions, Fuzzy Sets and Systems 261 (2015) 87–111.

[26] C.-Y. Lee, P. Gallagher, Z. Tu, Generalizing pooling functions in cnns: Mixed, gated, and tree, IEEE transactions on pattern analysis and machine intelligence 40 (2017) 863–875.

[27] M. Sugeno, Fuzzy measure and fuzzy integral, Transactions of the Society of Instrument and Control Engineers 8 (1972) 218–226.

[28] B. Pekala, U. Bentkowska, D. Kosior, Z. Takáč, A. Castillo, M. Sesma-Sara, J. Fernandez, J. Lafuente, H. Bustince, Interval-valued equivalence measures respecting uncertainty in image processing, International Journal of Intelligent Systems 36 (2021) 2767–2796.