



Image Thresholding Improved by Global Optimization Methods

Felipe Balabanian, Eduardo Sant'Ana da Silva & Helio Pedrini

To cite this article: Felipe Balabanian, Eduardo Sant'Ana da Silva & Helio Pedrini (2017) Image Thresholding Improved by Global Optimization Methods, Applied Artificial Intelligence, 31:3, 197-208, DOI: [10.1080/08839514.2017.1300050](https://doi.org/10.1080/08839514.2017.1300050)

To link to this article: <https://doi.org/10.1080/08839514.2017.1300050>



Published online: 10 Apr 2017.



Submit your article to this journal [↗](#)



Article views: 659



View related articles [↗](#)



View Crossmark data [↗](#)



Citing articles: 2 View citing articles [↗](#)



Image Thresholding Improved by Global Optimization Methods

Felipe Balabanian, Eduardo Sant'Ana da Silva, and Helio Pedrini 

Institute of Computing, University of Campinas, Campinas, SP, Brazil

ABSTRACT

Image thresholding is a common segmentation technique with applications in various fields, such as computer vision, pattern recognition, microscopy, remote sensing, and biology. The selection of threshold values for segmenting pixels into foreground and background regions is usually based on subjective assumptions or user judgments under empirical rules or manually determined. This work describes and evaluates six effective threshold selection strategies for image segmentation based on global optimization methods: genetic algorithms, particle swarm, simulated annealing, and pattern search. Experiments are conducted on several images to demonstrate the effectiveness of the proposed methodology.

Introduction

Image segmentation (Bhanu and Lee 2012; Gonzalez and Woods 2010; Rosenfeld 2013; Schwartz and Pedrini 2006; Silva et al. 2008) plays an important role in several image processing and computer vision tasks. Thresholding is one of the simplest techniques for extracting objects in an image from its background. Thresholding techniques (Ali, Ahn, and Pant 2014; Beauchemin 2013; Sezgin 2004; Singla and Patra 2015) can be classified into bilevel and multilevel categories. In bilevel thresholding (Zou et al. 2012; Zou, Liu, and Zhang 2013), a threshold value is determined to segment the image into two regions, one corresponding to the object in the image and another representing the background. In multilevel thresholding (Ali, Ahn, and Pant 2014; Sarkar, Das, and Chaudhuri 2015) more than one threshold value is determined to segment the image into background and objects present in the image.

Several image thresholding approaches (Dirami et al. 2013; Kapur, Sahoo, and Wong 1985; Otsu 1979; Weszka, Nagel, and Rosenfeld 1974; Ye et al. 2012) have been proposed in the literature. Many of them (Otsu 1979; Weszka, Nagel, and Rosenfeld 1974) are based on shape information of the histogram to select threshold values. In ideal cases, the threshold can be selected at the bottom of a valley between two peaks representing objects and background, respectively. However, it is often difficult to detect valleys precisely in real

images. Other methods attempt to optimize an objective function, such as entropy maximization (Kapur, Sahoo, and Wong 1985), Bayesian error minimization (Ye and Danielsson 1988), Bayesian error minimization (Ye and Danielsson 1988), and intra-class variance minimization (Otsu 1979).

One of the main difficulties related to the use of complex segmentation algorithms is that they have several parameters that are usually defined empirically by users, which affects the quality of segmentation and can vary significantly for different problems.

Since the threshold values defined in a segmentation process can considerably influence the detection and location of objects present in the image, it is important to have a robust method for selecting the best thresholds. In this paper, the parameters used in a number of different image thresholding techniques are automatically defined through four global optimization methods (Floudas 2013; Horst and Tuy 2013).

The main contributions of the work include the definition of an effective fitness function to evaluate the candidate solutions based on an image quality measure and the automatic selection of appropriate image thresholds through optimization approaches.

The remainder of the paper is organized as follows. “Background” section briefly describes concepts and works related to image thresholding techniques. The proposed method for selecting the best parameters in a number of thresholding methods based on optimization methods is presented in “Proposed methodology” section. Experimental results are described and discussed in “Experimental results” section. “Conclusions” section presents the conclusions and directions for future work.

Background

The main purpose of thresholding (Beauchemin 2013; Dirami et al. 2013; Gonzalez and Woods 2010; Ye et al. 2012) is to extract pixels belonging to objects from an image background. Image thresholding techniques are commonly categorized into global and local approaches (Glasbey 1993; Trier and Jain 1995). In global thresholding, a single threshold value T is used for all the image pixels, which is suitable for cases where the pixel values corresponding to object components and background are consistent along the entire image. In local thresholding, a different threshold T is adaptively selected for each pixel according to local image characteristics, which is more suitable to accommodate non-uniform lighting conditions in the image.

Global thresholding

The simplest thresholding is to select an intensity value as a threshold, such that the pixel values below this threshold become 0 (black) and the values

above this threshold become 1 (white). If T is the global threshold of the original image $f(x, y)$, then

$$g(x, y) = \begin{cases} 1, & \text{if } f(x, y) \geq T \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

A widely used global thresholding approach was developed by Otsu (Otsu 1979), which automatically clusters the image pixels into two classes, background and foreground. The method minimizes the intra-class variance (the variance within the class) or, conversely, maximizes the inter-class variance (the variance between the classes).

Considering $[0, L - 1]$ the range of intensity levels, the intra-class variance can be defined as the weighted sum of the variances of each class

$$\sigma_{\text{within}}^2(T) = n_B(T) \sigma_B^2(T) + n_F(T) \sigma_F^2(T) \quad (2)$$

where $n_B(T) = \sum_{i=0}^{T-1} p(i)$ and $n_F(T) = \sum_{i=T}^{L-1} p(i)$. Value $\sigma_B^2(T)$ is the variance of the pixels in the background (below the threshold), whereas $\sigma_F^2(T)$ is the variance of the pixels in the foreground (above the threshold).

The inter-class variance can be defined as

$$\sigma_{\text{between}}^2(T) = n_B(T) n_F(T) [\mu_B(T) - \mu_F(T)]^2 \quad (3)$$

where $\mu_B(T) = \sum_{i=0}^{T-1} ip(i)/n_B(T)$ and $\mu_F(T) = \sum_{i=T}^{L-1} ip(i)/n_F(T)$

Local thresholding

Local thresholding techniques are designed to overcome the limitations found in the global thresholding due to changes in lighting conditions or presence of local shadows.

Bernsen's method (Bernsen 1986) calculates the local threshold value based on the mean value of the minimum and maximum intensities of pixels within a window, defined as

$$T(x, y) = (z_{\min} + z_{\max})/2 \quad (4)$$

where z_{\min} and z_{\max} are the minimum and maximum intensity values, respectively, within an $n \times n$ region centered at (x, y) .

Niblack's method (Niblack 1986) calculates the threshold value for each pixel (x, y) based on the local mean and standard deviation, expressed as

$$T(x, y) = \mu(x, y) + k \sigma(x, y) \quad (5)$$

where $\mu(x, y)$ and $\sigma(x, y)$ are local mean and standard deviation, respectively, within an $n \times n$ region of pixel (x, y) . The region size should be small enough to preserve local details of the image, but also large enough to suppress noise.

The value k is used to adjust the fraction of total pixels that belong to foreground object.

Sauvola and Pietaksinen's method (Sauvola and Pietaksinen 2000) calculates the threshold value for each pixel (x, y) as

$$T(x, y) = \mu(x, y) \left[1 + k \left(\frac{\sigma(x, y)}{R} - 1 \right) \right] \quad (6)$$

where $\mu(x, y)$ and $\sigma(x, y)$ are local mean and standard deviation, respectively, within an $n \times n$ region of pixel (x, y) . Sauvola and Pietaksinen suggested the values $k = 0.5$ and $R = 128$.

Phansalskar, More, and Sabale's method (Phansalskar, More, and Sabale 2011) is variation to Sauvola and Pietaksinen's method (Sauvola and Pietaksinen 2000) to deal with low-contrast images. The local threshold is calculated as

$$T = \mu(x, y) \left[1 + p \exp(-q\mu(x, y)) + k \left(\frac{\sigma(x, y)}{R} - 1 \right) \right] \quad (7)$$

where local mean and standard deviation, respectively, within an $n \times n$ region of pixel (x, y) . They suggested the values $k = 0.25$, $R = 0.5$, $p = 2$, and $q = 10$.

The mean method (Jain, Kasturi, and Schunck 1995) selects the threshold as the mean of the local intensity distribution in an $n \times n$ region. If the intensity of a pixel (x, y) is above the mean of its neighborhood, then the pixel is set to object; otherwise, it is set as background.

In a similar way, the median method (Jain, Kasturi, and Schunck 1995) selects the threshold as the median of the local intensity distribution in an $n \times n$ region. If the intensity of a pixel (x, y) is above the median of its neighborhood, then the pixel is set to object; otherwise, it is set as background.

Proposed methodology

To find the best parameter values for the image thresholding problem, we model a fitness function to guide the selection of candidate solutions. The Structural Similarity Index (SSIM) (Wang et al. 2004) is a quality measure calculated between two images \mathbf{f} and \mathbf{g} is expressed as

$$\text{SSIM}(\mathbf{f}, \mathbf{g}) = \frac{(2\mu_f\mu_g + C_1)(2\sigma_{fg} + C_2)}{(\mu_f^2 + \mu_g^2 + C_1)(\sigma_f^2 + \sigma_g^2 + C_2)} \quad (8)$$

where μ_f is the mean of \mathbf{f} , μ_g is the mean of \mathbf{g} , σ_f^2 is the variance of \mathbf{f} , σ_g^2 is the variance of \mathbf{g} , and σ_{xy} is the covariance of \mathbf{f} and \mathbf{g} . The SSIM measure is in the range of $[0..1]$, such that closer it is to 1, the more similar the images \mathbf{f} and \mathbf{g} are.

The fitness function is defined as

$$F^* = \arg \max_{F \in \Omega} \text{SSIM}(\mathbf{f}, \mathbf{g}) \quad (9)$$

where Ω is the set of all possible solutions. Therefore, our goal is to maximize the quality measure computed between an input image \mathbf{f} and an estimated image \mathbf{g} .

In our work, four global optimization approaches (Arora 2015; Chong and Zak 2013) are used to generate the estimated image \mathbf{g} by automatically selecting the best parameter values employed in the image thresholding techniques described in the previous section.

Genetic Algorithms (GAs) (Holland 1975; Koza 1992) are heuristic search algorithms based on evolutionary strategies of natural selection, which can be used to generate solutions to search and optimization problems. First, an initial population is created with a number of individuals. Each individual corresponds to a possible solution, that is, it represents a set of possible values for the thresholding parameters. Each gene represents a specific parameter encoded in a bit string. The parameters values for the initial set of solutions are randomly generated. A solution (individual) is evaluated through the fitness function [Equation (8)], where the value resulting from the segmentation is compared to a reference. During the optimization process, the best solutions are selected and new solutions are created from them (reproduction). For each generation, new individuals are created via crossover to compose the next generation. The mutation operator helps prevent local optimum. The evolutionary process continues until a predefined number of generations is reached, such that the segmentation parameters (genes) of the best solution found in the last generation correspond to the final result.

The Particle Swarm Optimization (PSO) (Clerc 2010; Kennedy and Eberhart 1995) algorithm starts with a random population (called swarm) of candidate solutions (called particles), each one having the parameters to be optimized. At each iteration, every particle adjusts its velocity vector and the best known position of the swarm is updated according to the fitness function [Equation (8)]. The algorithm stores and progressively replaces the best parameters of each particle, as well as the particle that best fits the parameters. The process continues until a predefined number of iterations is performed.

The Simulated Annealing (SA) (Brooks and Morgan 1995; Dowsland and Thompson 2012) method models the physical process of heating a material and then slowly decreasing the temperature to reduce defects, thus minimizing the system energy. The algorithm starts with a randomly chosen initial solution. At each iteration, a new point is randomly generated. The distance of the new point from the current point is based on a probability distribution with a scale proportional to the temperature. The algorithm accepts not only new points that produce a better solution, but also points that generate a worse solution. This reheating mechanism helps prevent local optimum, such that other possible solutions can be explored globally. As the algorithm proceeds, a cooling schedule (Kirkpatrick, Gelatt, and Vecchi 1983) is selected to systematically decrease the temperature, reducing the extent of its search to converge to an optimum.

Pattern Search (PS) (Hooke and Jeeves 1961) is a numerical optimization approach that does not require derivatives (gradient) of the fitness function to be optimized. A pattern is a finite set of points (vectors) used for the algorithm to determine which points to search at each iteration. Therefore, the method searches a set of points around the current point to find that one where the value of the fitness function generates a better result than the value at the current point. The algorithm stops when a predefined number of iterations is performed, the variation in the fitness function from one iteration to the next is below a tolerance, or the distance between the point found at one successful poll and the point found at the next one is below a tolerance.

Experimental results

The algorithms proposed in our work were implemented in MATLAB under Linux operating system. Experiments conducted on several images demonstrated the effectiveness of the image thresholding techniques improved by the optimization strategies. Due to space limitations, results for only few images are reported here.

Table 1 reports some results obtained with the optimization methods for selecting the parameters for image thresholding. Values for the computed SSIM measure, percentage of foreground pixels (black), and estimated parameters are shown for each image thresholding algorithm. The size n for the neighborhood region used in the algorithms ranged from 1 to 30.

For the GA, the population size N_{GA} was defined as 50, the crossover probability P_C was 0.8, the mutation probability P_M was 0.05, and the number of generations G was 10. For the PSO, the population size N_{PSO} was defined as 50 and the inertia range was set from 0.1 to 1.1. In the SA algorithm, the initial temperature was set to 100. In the PS algorithm, the initial mesh size was set to 1.0 to perform a local search in a neighborhood around the mesh points, whereas the tolerance on variable was defined as 0.000001 to interrupt the iterations if both the change in position and the mesh size are less than such tolerance.

Images generated with the application of the best estimated parameters are shown in Figure 1. It is possible to observe that the fitness function was able to produce adequate qualitative segmentation results. Moreover, the SSIM measures achieved through the image thresholding methods are higher when compared to those obtained with traditional global thresholding Otsu's method (Otsu 1979).

It is worth mentioning that the PS optimization method, as it can be observed from Table 1, provided the best qualitative results in relation to the other three methods.

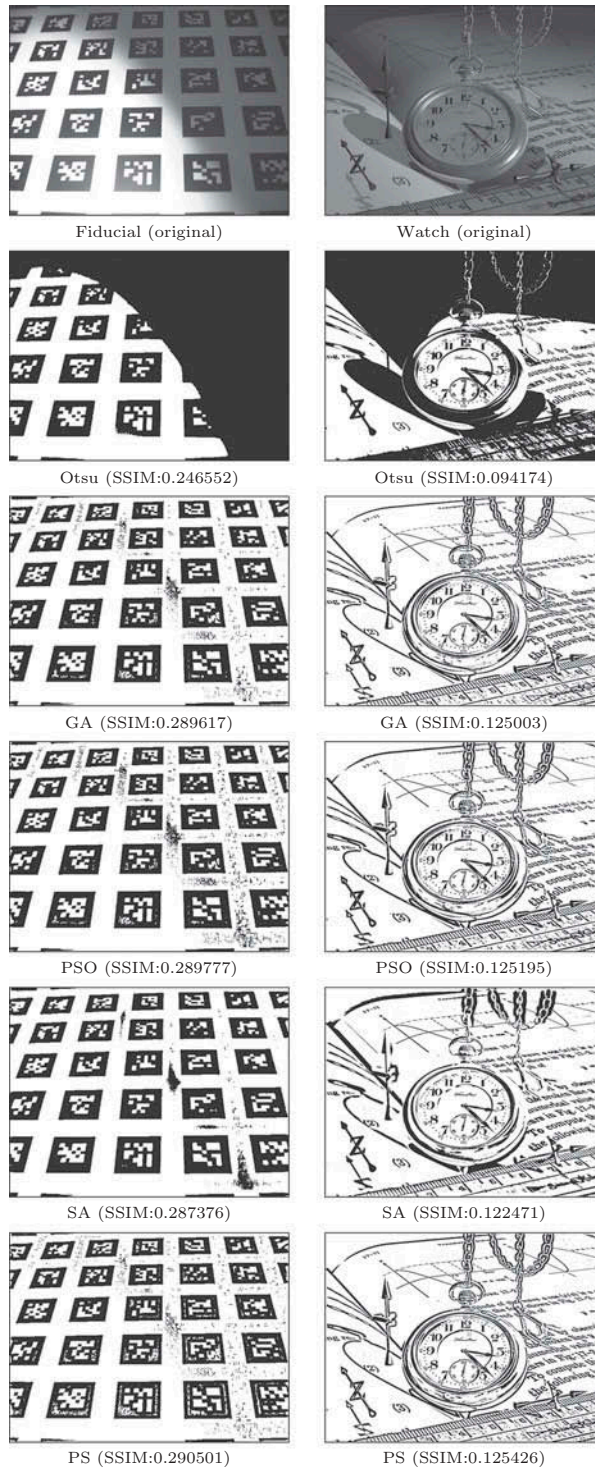


Figure 1. Images resulting from the application of the best estimated parameters to each thresholding method. The values of SSIM are provided for each image by taking the original one as reference.



Table 1. Comparison of the results obtained with four different optimization methods.

Image	Method	Result	GA	PSO	SA	PS
Fiducial (640 × 840 pixels)	Bernsen (1986)	SSIM	0.283080	0.283050	0.283050	0.283050
		Foreground (%)	35.91	35.87	35.87	35.87
		N	22	21	22	22
	Niblack (1986)	SSIM	0.288407	0.288657	0.287688	0.288657
		Foreground (%)	38.89	38.98	36.11	38.98
		n	11	10	14	10
	Mean (Jain, Kasturi, and Schunck, 1995)	k	-0.000522	-0.0224538	-0.0671209	-0.022460
		SSIM	0.154173	0.288516	0.288516	0.288516
		Foreground (%)	51.31	39.55	39.55	39.55
	Median (Jain, Kasturi, and Schunck, 1995)	n	7	10	10	10
		SSIM	0.090706	0.092082	0.092082	0.092082
		Foreground (%)	56.24	55.83	55.83	55.83
	Sauvola and Pietaksinen (2000)	n	29	30	30	30
		SSIM	0.289609	0.286833	0.287376	0.290410
		Foreground (%)	36.96	34.91	35.07	32.87
	Phansalskar, More, and Sabale (2011)	n	10	18	16	7
		k	0.00406674	0.0142321	0.0125171	0.014707
		R	-33.904	101.883	86.4602	16385
	Phansalskar, More, and Sabale (2011)	SSIM	0.289617	0.289777	0.274469	0.290501
		Foreground (%)	34.27	35.20	23.46	32.98
		n	9	9	7	7
	Phansalskar, More, and Sabale (2011)	k	0.0160981	0.0103334	0.178103	-0.000289
		R	49.8369	88.3678	-169.032	1.00513
		p	5.15045	-693.918	16.7733	-0.013382
	Phansalskar, More, and Sabale (2011)	q	28.2679	553.543	133.475	0.000457

(Continued)

Table 1. (Continued).

Image	Method	Result	GA	PSO	SA	PS
Watch (768 × 1024 pixels)	Bernsen (1986)	SSIM	0.103089	0.103081	0.103089	0.103089
		Foreground (%)	39.06	39.02	39.06	39.06
Niblack (1986)		n	17	16	17	17
		SSIM	0.111335	0.111426	0.111416	0.111465
		Foreground (%)	20.41	20.97	20.71	21.00
		n	24	23	22	25
Mean (Jain, Kasturi, and Schunck, 1995)		k	-0.388632	-0.370083	-0.3779	-0.371017
		SSIM	0.101979	0.106767	0.106767	0.106767
		Foreground (%)	41.54	38.80	38.80	38.80
		n	8	10	10	10
Median (Jain, Kasturi, and Schunck, 1995)		SSIM	0.0726361	0.0724491	0.0726361	0.0726361
		Foreground (%)	54.85	55.40	54.85	54.85
		n	30	29	30	30
		SSIM	0.125003	0.125006	0.122471	0.125091
Sauvola and Pietaksinen (2000)		Foreground (%)	20.23	19.90	17.12	19.31
		n	6	7	4	6
		k	0.054655	0.0566141	0.0632998	0.0530624
		R	57.9847	646.569	88.9129	984.746
Phansalskar, More, and Sabale (2011)		SSIM	0.12487	0.125195	0.118852	0.125426
		Foreground (%)	19.51	19.97	19.84	19.88
		n	6	7	16	6
		k	0.0780287	0.0624377	0.0406067	0.00976562
		R	41.3679	121.909	-10.3443	0.986542
		p	9.16471	9603.67	-53.8454	-0.03578
		q	24.3086	1102.03	189.716	-0.00683975

For each image thresholding, values for the computed SSIM measure, percentage of foreground pixels (black), and evaluated parameters are shown.

Conclusions

The selection of threshold values for image segmentation is typically performed manually based on empirical assumptions; however, this process may significantly affect the quality of the final result.

This work presented and discussed the application of four global optimization methods (GAs, particle swarm, SA, and PS) to six image thresholding techniques.

The automatic threshold values estimated by the algorithms, according to a fitness function based on image quality, were able to generate highly promising results. The proposed method can be used as a preprocessing step in other tasks, such as object location and image interpretation.

As directions for future work, we intend to investigate the improvement of other image processing techniques, such as filtering, registration, and compression.

Funding

The authors are thankful to the São Paulo Research Foundation (FAPESP grant #2015/12228-1) and the Brazilian National Council for Scientific and Technological Development (CNPq grant #305169/2015-7) for their financial support.

ORCID

Helio Pedrini  <http://orcid.org/0000-0003-0125-630X>

References

- Ali, M., C. W. Ahn, and M. Pant. 2014. Multi-level image thresholding by synergetic differential evolution. *Applied Soft Computing* 17 (April):1–11. doi:10.1016/j.asoc.2013.11.018.
- Arora, R. K. 2015. *Optimization: Algorithms and applications*. Chapman and Hall/CRC, Taylor & Francis Group, CRC Press Boca Raton, FL, USA.
- Beauchemin, M. 2013. Image thresholding based on semivariance. *Pattern Recognition Letters* 34 (5):456–62. doi:10.1016/j.patrec.2012.11.017.
- Bernsen, J. 1986. Dynamic thresholding of grey-level images. Proceedings of the 6th International Conference on Pattern Recognition, Berlin, Germany, 1251–55, October.
- Bhanu, B., and S. Lee. 2012. *Genetic learning for adaptive image segmentation*, Vol. 287. Springer Science & Business Media, Kluwer Academic Publishers New York, NY, USA.
- Brooks, S. P., and B. J. T. Morgan. 1995. Optimization using simulated annealing. *Journal of the Royal Statistical Society, Series D (The Statistician)* 44 (2):241–57.
- Chong, E. K. P., and S. H. Zak. 2013. *An introduction to optimization*. Hoboken, New Jersey, USA: John Wiley & Sons.
- Clerc, M. 2010. *Particle swarm optimization*. Newport Beach, CA, USA: John Wiley & Sons.
- Dirami, A., K. Hammouche, M. Diaf, and P. Siarry. 2013. Fast multilevel thresholding for image segmentation through a multiphase level set method. *Signal Processing* 93 (1):139–53. doi:10.1016/j.sigpro.2012.07.010.
- Dowland, K. A., and J. M. Thompson. 2012. Simulated annealing. In *Handbook of natural computing*, Rozenberg, Grzegorz, Bäck, Thomas, Kok, Joost N., Eds., 1623–55. Heidelberg Dordrecht, London, New York: Springer.

- Floudas, C. A. 2013. *Deterministic global optimization: Theory, methods and applications*, Vol. 37. Springer-Verlag New York, Inc. Secaucus, NJ, USA.
- Glasbey, C. A. 1993. An analysis of histogram-based thresholding algorithms. *CVGIP: Graphical Models and Image Processing* 55 (6):532–37.
- Gonzalez, R., and R. Woods. 2010. *Digital image processing using matlab*. India: McGraw Hill Education.
- Holland, J. H. 1975. *Adaptation in natural and artificial system: An introduction with application to biology, control and artificial intelligence*. Ann Arbor, MI, USA: University of Michigan Press.
- Hooke, R., and T. A. Jeeves. 1961. "Direct Search" solution of numerical and statistical problems. *Journal of the ACM* 8 (2):212–29. doi:[10.1145/321062.321069](https://doi.org/10.1145/321062.321069).
- Horst, R., and H. Tuy. 2013. *Global optimization: Deterministic approaches*. Heidelberg, Berlin: Springer Science & Business Media.
- Jain, R., R. Kasturi, and B. G. Schunck. 1995. *Machine vision*. New York, NY, USA: McGraw-Hill.
- Kapur, J. N., P. K. Sahoo, and A. K. C. Wong. 1985. A new method for gray-level picture thresholding using the entropy of the histogram. *Computer Vision, Graphics, and Image Processing* 29 (3):273–85. doi:[10.1016/0734-189X\(85\)90125-2](https://doi.org/10.1016/0734-189X(85)90125-2).
- Kennedy, J., and R. Eberhart. 1995. Particle swarm optimization. IEEE International Conference on Neural Networks, Perth, Australia, 1942–48.
- Kirkpatrick, S., C. Gelatt, and M. P. Vecchi. 1983. Optimization by simulated annealing. *Science* 220 (4598):671–80. doi:[10.1126/science.220.4598.671](https://doi.org/10.1126/science.220.4598.671).
- Koza, J. R. 1992. *Genetic programming: On the programming of computers by means of natural selection*. Cambridge, MA, USA: MIT press.
- Niblack, W. 1986. *An introduction to digital image processing*. Englewood Cliffs, NJ, USA: Prentice Hall.
- Otsu, N. 1979. A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man and Cybernetics* 9 (1, January):62–66. doi:[10.1109/TSMC.1979.4310076](https://doi.org/10.1109/TSMC.1979.4310076).
- Phansalkar, N., S. More, and A. Sabale. 2011. Adaptive local thresholding for detection of nuclei in diversity stained cytology images. In *Proceedings of the International Conference on Communications and Signal Processing*, Kerala, India, 218–20.
- Rosenfeld, A. 2013. *Multiresolution image processing and analysis*, Vol. 12. Springer-Verlag Berlin Heidelberg New York Tokyo: Springer Science & Business Media.
- Sarkar, S., S. Das, and S. S. Chaudhuri. 2015. A multilevel color image thresholding scheme based on minimum cross entropy and differential evolution. *Pattern Recognition Letters* 54:27–35. doi:[10.1016/j.patrec.2014.11.009](https://doi.org/10.1016/j.patrec.2014.11.009).
- Sauvola, J., and M. Pietaksinen. 2000. Adaptive document image binarization. *Pattern Recognition* 33:225–36. doi:[10.1016/S0031-3203\(99\)00055-2](https://doi.org/10.1016/S0031-3203(99)00055-2).
- Schwartz, W. R., and H. Pedrini. 2006. Textured image segmentation based on spatial dependence using a markov random field model. In *the Proceedings of the IEEE International Conference on Image Processing*. Atlanta, GA, USA: IEEE, 2449–52.
- Sezgin, M. 2004. Survey over image thresholding techniques and quantitative performance evaluation. *Journal of Electronic Imaging* 13 (1):146–68. doi:[10.1117/1.1631315](https://doi.org/10.1117/1.1631315).
- Silva, R. D., R. Minetto, W. R. Schwartz, and H. Pedrini. 2008. Satellite image segmentation using wavelet transforms based on color and texture features. In *the Proceedings of the International Symposium on Visual Computing*, Las Vegas, NV, USA: Springer, 113–22.
- Singla, A., and S. Patra. 2015. A context sensitive thresholding technique for automatic image segmentation. In *Computational intelligence in data mining*, eds. L. C. Jain, H. S. Behera, J.

- K. Mandal, and D. P. Mohapatra, Volume 32 of *Smart Innovation, Systems and Technologies*, 19–25. New Delhi, India: Springer.
- Trier, B. D., and A. K. Jain. 1995. Goal-directed evaluation of binarization methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 17 (12):1191–201. doi:[10.1109/34.476511](https://doi.org/10.1109/34.476511).
- Wang, Z., A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. 2004. Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing* 13 (4):600–12. doi:[10.1109/TIP.2003.819861](https://doi.org/10.1109/TIP.2003.819861).
- Weszka, J. S., R. N. Nagel, and A. Rosenfeld. 1974. A threshold selection technique. *IEEE Transactions on Computers* C-23 (December):1322–26. doi:[10.1109/T-C.1974.223858](https://doi.org/10.1109/T-C.1974.223858).
- Ye, Q.-Z., and P.-E. Danielsson. 1988. On minimum error thresholding and its implementations. *Pattern Recognition Letters* 7 (4):201–06. doi:[10.1016/0167-8655\(88\)90103-1](https://doi.org/10.1016/0167-8655(88)90103-1).
- Ye, Z., H. Zhengbing, X. Lai, and H. Chen. 2012. Image segmentation using thresholding and swarm intelligence. *Journal of Software* 7 (5):1074–82. doi:[10.4304/jsw.7.5.1074-1082](https://doi.org/10.4304/jsw.7.5.1074-1082).
- Zou, Y., H. Liu, E. Song, and Z. Huang. 2012. Image bilevel thresholding based on multiscale gradient multiplication. *Computers and Electrical Engineering* 38 (4, July):853–61. doi:[10.1016/j.compeleceng.2012.03.009](https://doi.org/10.1016/j.compeleceng.2012.03.009).
- Zou, Y., H. Liu, and Q. Zhang. 2013. Image bilevel thresholding based on stable transition region set. *Digital Signal Processing* 23 (1):126–41. doi:[10.1016/j.dsp.2012.08.004](https://doi.org/10.1016/j.dsp.2012.08.004).