WIKIPEDIA

Thresholding (image processing)

In digital image processing, **thresholding** is the simplest method of segmenting images. From a grayscale image, thresholding can be used to create binary images. [1]

Contents

Original image

Example of a threshold effect used on an image

Definition

The simplest thresholding methods replace each pixel in an image with a black pixel if the image intensity $I_{i,j}$ is less than some fixed constant T (that is, $I_{i,j} < T$), or a white pixel if the image intensity is greater than that constant. In the example image on the right, this results in the dark tree becoming completely black, and the white snow becoming completely white.

Categorizing thresholding methods

To make thresholding completely automated, it is necessary for the computer to automatically select the threshold T. Sezgin and Sankur (2004) categorize thresholding methods into the following six groups based on the information the algorithm manipulates (Sezgin et al., 2004):

- **Histogram shape**-based methods, where, for example, the peaks, valleys and curvatures of the smoothed histogram are analyzed
- **Clustering**-based methods, where the gray-level samples are clustered in two parts as background and foreground (object), or alternately are modeled as a mixture of two Gaussians
- **Entropy**-based methods result in algorithms that use the entropy of the foreground and background regions, the cross-entropy between the original and binarized image, etc.[2]
- **Object Attribute**-based methods search a measure of similarity between the gray-level and the binarized images, such as fuzzy shape similarity, edge coincidence, etc.

- **Spatial** methods [that] use higher-order probability distribution and/or correlation between pixels
- **Local** methods adapt the threshold value on each pixel to the local image characteristics. In these methods, a different T is selected for each pixel in the image.

Multiband thresholding

Colour images can also be thresholded. One approach is to designate a separate threshold for each of the RGB components of the image and then combine them with an AND operation. This reflects the way the camera works and how the data is stored in the computer, but it does not correspond to the way that people recognize colour. Therefore, the HSL and HSV colour models are more often used; note that since hue is a circular quantity it requires circular thresholding. It is also possible to use the CMYK colour model (Pham et al., 2007).

Probability distributions

Histogram shape-based methods in particular, but also many other thresholding algorithms, make certain assumptions about the image intensity probability distribution. The most common thresholding methods work on bimodal distributions, but algorithms have also been developed for unimodal distributions, multimodal distributions, and circular distributions.

Automatic thresholding

Automatic thresholding is a great way to extract useful information encoded into pixels while minimizing background noise. This is accomplished by utilizing a feedback loop to optimize the threshold value before converting the original grayscale image to binary. The idea is to separate the image into two parts; the background and foreground.[3]

1. Select initial threshold value, typically the mean 8-bit value of the original image.
2. Divide the original image into two portions;

   1. Pixel values that are less than or equal to the threshold; background
   2. Pixel values greater than the threshold; foreground
3. Find the average mean values of the two new images
4. Calculate the new threshold by averaging the two means.
5. If the difference between the previous threshold value and the new threshold value are below a specified limit, you are finished. Otherwise apply the new threshold to the original image keep trying.

Note about limits and threshold selection

The limit mentioned above is user definable. A larger limit will allow a greater difference between successive threshold values. Advantages of this can be quicker execution but with a less clear boundary between background and foreground. Picking starting thresholds is often done by taking the mean value of

the grayscale image. However, it is also possible to pick out the starting threshold values based on the two well separated peaks of the image histogram and finding the average pixel value of those points. This can allow the algorithm to converge faster; allowing a much smaller limit to be chosen.

Method limitations

Automatic thresholding will work best when a good background to foreground contrast ratio exists. Meaning the picture must be taken in good lighting conditions with minimal glare.

See also

- Otsu's method
- Balanced histogram thresholding

References

1. (Shapiro, et al. 2001:83)
2. Zhang, Y. (2011). "Optimal multi-level Thresholding based on Maximum Tsallis Entropy via an Artificial Bee Colony Approach". *Entropy*. **13** (4): 841–859. Bibcode:2011Entrp..13..841Z (https://ui.adsabs.harvard.edu/abs/2011Entrp..13..841Z). doi:10.3390/e13040841 (https://doi.org/10.3390%2Fe13040841).
3. E., Umbaugh, Scott (2017-11-30). *Digital Image Processing and Analysis with MATLAB and CVIPtools, Third Edition* (3rd ed.). ISBN 9781498766074. OCLC 1016899766 (https://www.worldcat.org/oclc/1016899766).

Sources

- Pham N, Morrison A, Schwock J et al. (2007). Quantitative image analysis of immunohistochemical stains using a CMYK color model. *Diagn Pathol.* **2:**8 (http://www.pubmedcentral.nih.gov/articlerender.fcgi?tool=pubmed&pubmedid=17326824).
- Shapiro, Linda G. & Stockman, George C. (2002). "Computer Vision". Prentice Hall. ISBN 0-13-030796-3
- Mehmet Sezgin and Bulent Sankur, Survey over image thresholding techniques and quantitative performance evaluation, Journal of Electronic Imaging 13(1), 146–165 (January 2004). doi:10.1117/1.1631315 (https://doi.org/10.1117%2F1.1631315)

Further reading

- Gonzalez, Rafael C. & Woods, Richard E. (2002). Thresholding. In Digital Image Processing, pp. 595–611. Pearson Education. ISBN 81-7808-629-8
- M. Luessi, M. Eichmann, G. M. Schuster, and A. K. Katsaggelos, Framework for efficient optimal multilevel image thresholding, Journal of Electronic Imaging, vol. 18, pp. 013004+, 2009. doi:10.1117/1.3073891 (https://doi.org/10.1117%2F1.3073891)
- Y.K. Lai, P.L. Rosin, Efficient Circular Thresholding, IEEE Trans. on Image Processing 23(3), pp. 992–1001 (2014). doi:10.1109/TIP.2013.2297014 (https://doi.org/10.1109%2FTIP.2013.2297014)
- Scott E. Umbaugh (2018). Digital Image Processing and Analysis, pp 93–96. CRC Press. ISBN 978-1-4987-6602-9