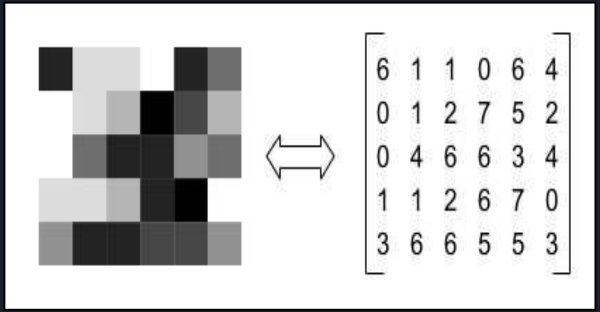
Image Thresholding

Intensity Histogram

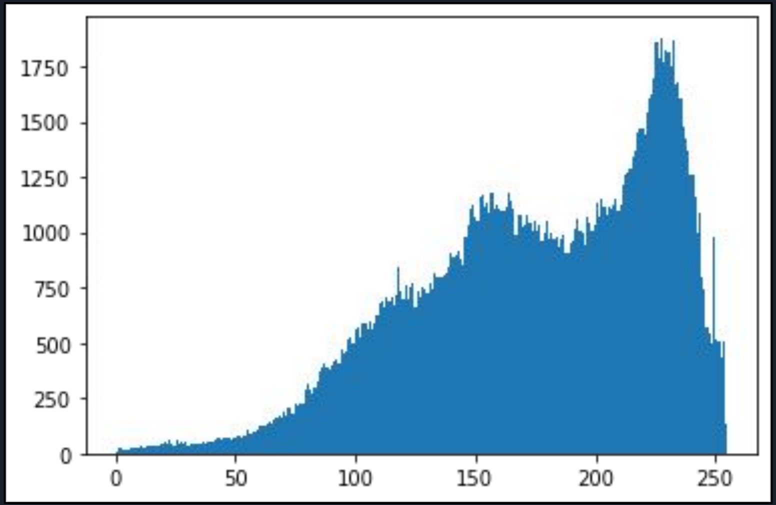
An image is a collection of pixels as mentioned previously, the below is a grayscale image with a 3-bit channel. So for a 8-bit (a total of 2^8 allowed intensity values) channel pixel values can vary from 0 (white) to 255 (black) while the intermediate pixels covering shades of grey. An intensity histogram classifies number of pixels for each value of intensity ranging from 0-255 for a 8-bit channel.



Intensity Histogram (Contd.)

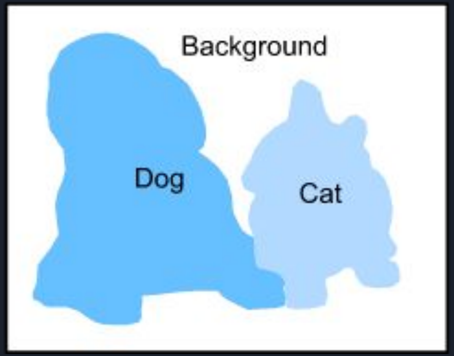
Image

Histogram



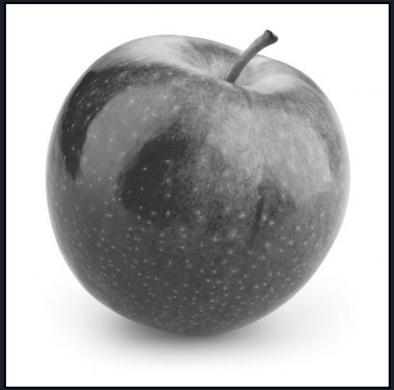
Thresholding

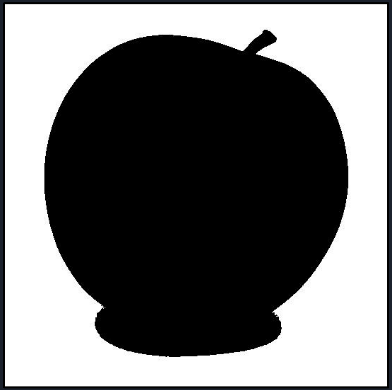
Given a particular image it is of interest to separate the different objects in the image, this is what segmentation does. Thresholding plays a central role in image segmentation.

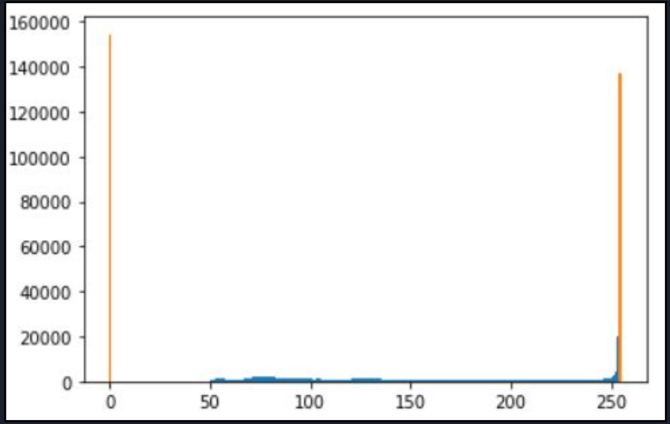


Global Thresholding

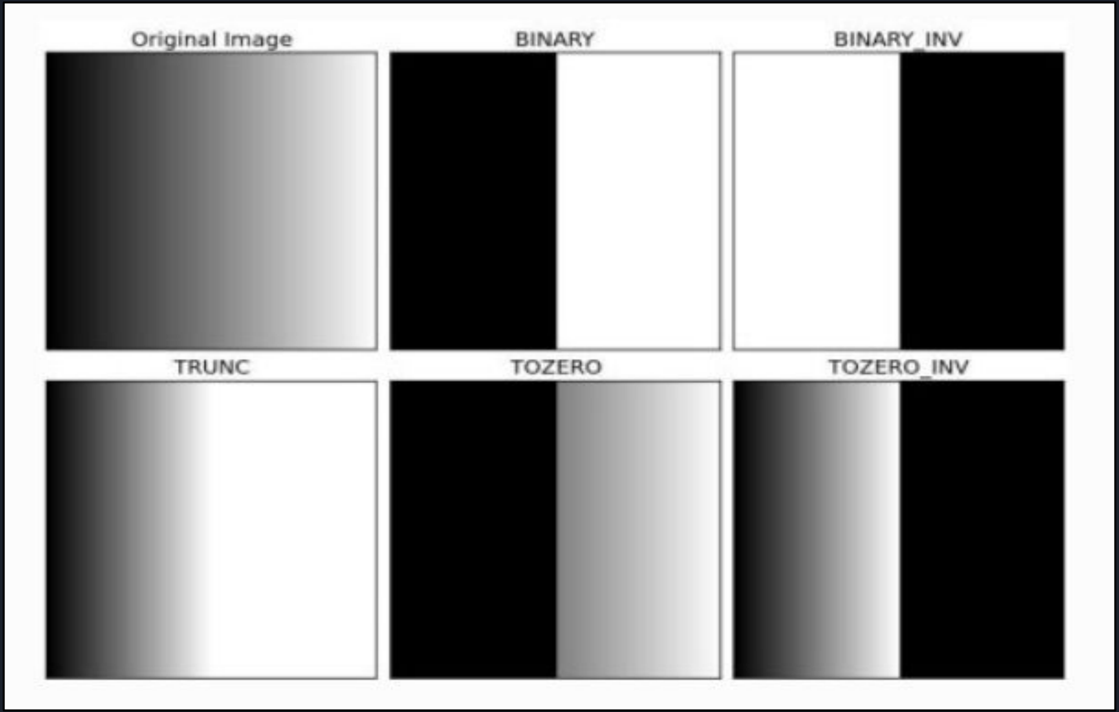
Global thresholding is a type of method that works well when the object and the background differ sufficiently in their intensity values. In terms of image histogram, a threshold value T is selected on the histogram and all the pixels with value less than the threshold are set to null and those greater than the threshold are scaled to unity. Determining the parameter T is another problem of interest. And this method works well for bi-modal distributions.





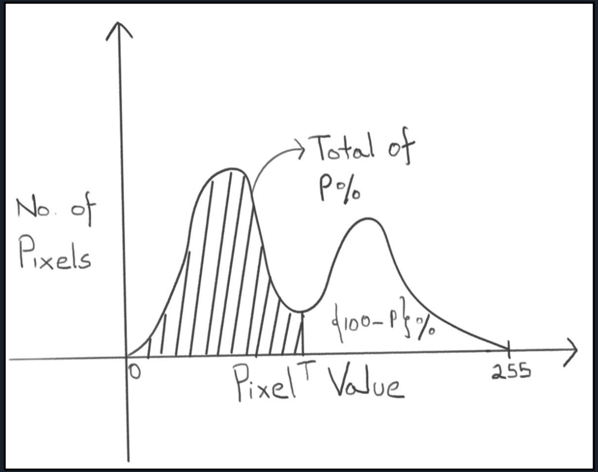


Types of simple thresholding present in Open_CV



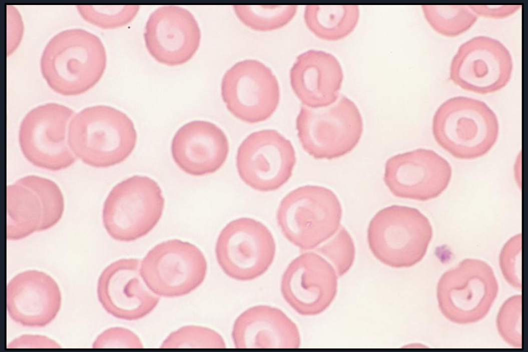
p-Tile Thresholding

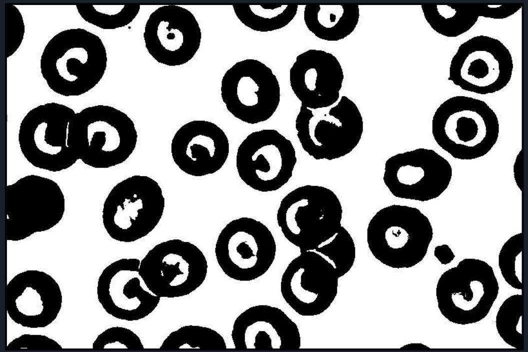
Suppose you are given an image, through inspection background and the object are of different intensities, for say light background and dark object. And if object approximately occupies p percentage of the image, then threshold T is chosen such that total number of pixels with value less than or equal to threshold is p percent.

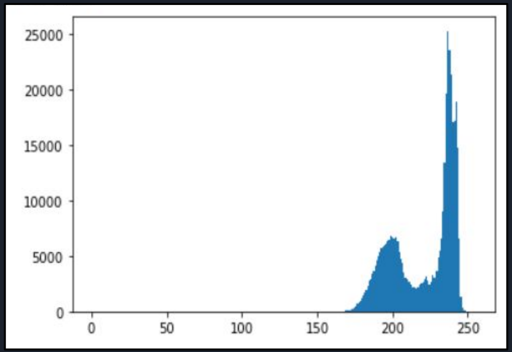


p-Tile Thresholding (Contd.)

Roughly the cells occupy 40% of the given image. From the image histogram, the threshold is determined such that there are 40% of the total number of pixels whose value is less than the threshold. The threshold value is around 220.







Iterative Thresholding

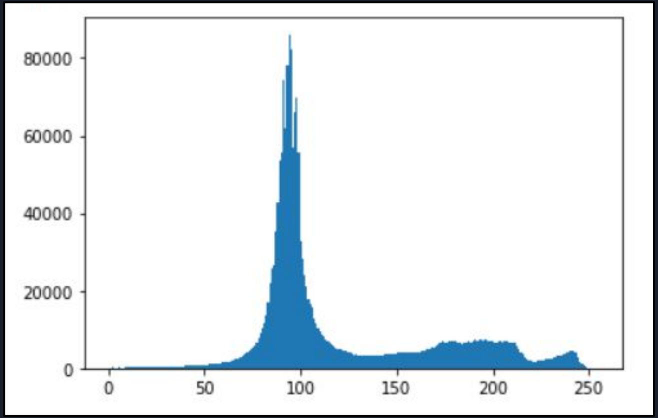
The method of iterative thresholding is given as follows :

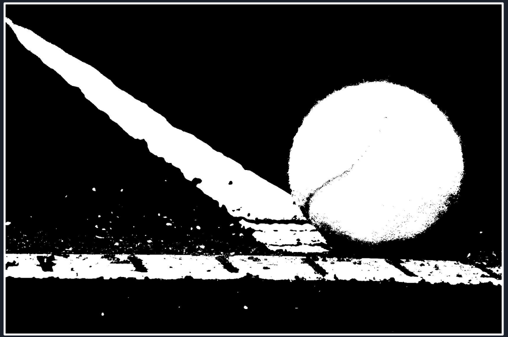
- Let T_0 be the initial estimate of the global threshold, thus dividing the image histogram into two classes C_1 and C_2
- M_1 is the mean intensity of pixels belonging to C_1, while M_2 for that of C_2
- The new threshold is given by the arithmetic mean of M_1 and M_2
- The process is repeated until the difference between the successive thresholds values falls below certain tolerance value.

This method works well when there is a clear separation or valley between the object and the background classes and often it can be started-off by taking T_0 as mean intensity of pixel values.

Iterative thresholding (Contd.)

In this case the T value converges to 143, either we start from 100 or 200. For $T_0 = 100$, values of T after each iteration are listed.



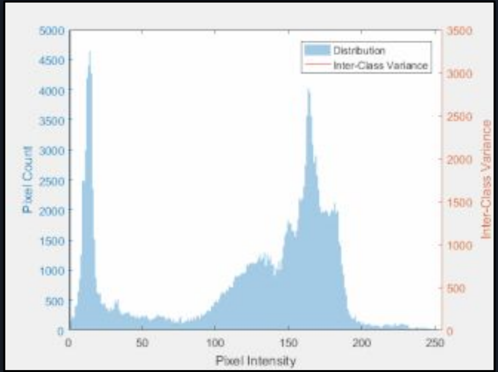


Otsu’s Binarization

Iterative thresholding works well when there is a clear valley between the classes, and eventually the threshold value converges to the neighbourhood of the depression of the valley. But when there isn't such depression a more superior method - Otsu's Binarization which tries to maximise ‘between-class variance’ (a measure of separability of classes) is employed. Even this works on the principle of different intensities of background and object classes, although the difference is small.

The algorithm that lies behind Otsu’s approach is as follows :

- Consider the normalised histogram of an image, where pixel value at each location is divided by total number of pixels. Let p_i be the normalized value at pixel value i.
- For a threshold value k, class C_1 contains all the pixels whose value is less than k while class C_2 contains the rest of the pixels.

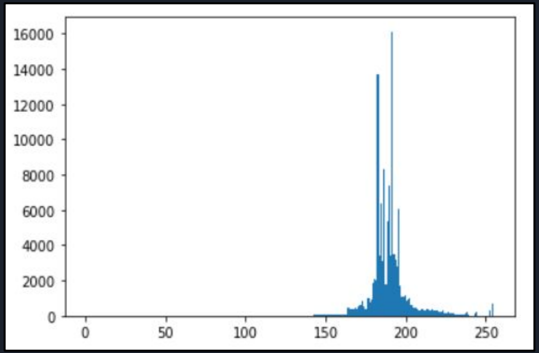


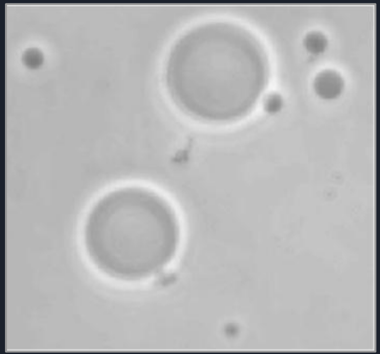
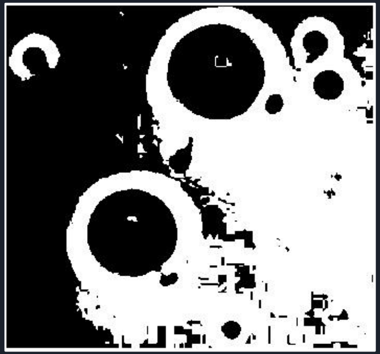
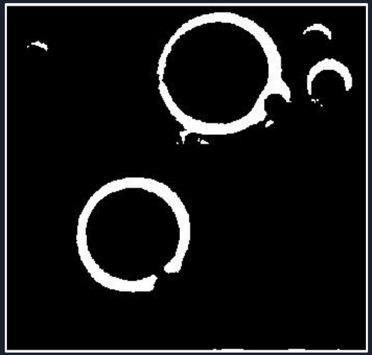
Otsu's Binarization (Contd.)

- Consider the following measures :
  - Probability that pixel selected at random belongs to C_1 : $\mathbf{P_1 = \sum p_i}$; $i \in [0, k]$
  - Probability that pixel selected at random belongs to C_2 : $\mathbf{P_2 = \sum p_i}$; $i \in [k+1, 255]$
  - The average intensity of the image is given by : $\mathbf{m = \sum i.p_i}$; $i \in [0, 255]$
  - The global variance of the image is given by : $\mathbf{\sigma_G^2 = \sum (i - m)^2.p_i}$; $i \in [0, 255]$
  - The mean intensity of pixels belonging to C_1 : $\mathbf{m_1 = \sum i.q_i}$; $i \in [0,k]$, $q_i = p_i/P_1$
  - The mean intensity of pixels belonging to C_2 : $\mathbf{m_2 = \sum i.q_i}$; $i \in [k+1,255]$, $q_i = p_i/P_2$
  - The between class variance : $\mathbf{\sigma_B^2 = P_1(m_1 - m)^2 + P_2(m_2 - m)^2}$
- The separability measure η is the ratio of between-class variance and global variance. The value of k is chosen such that the value of between class-variance or equivalently η is the maximum for $k \in [0, 255]$.
- For a histogram with a valley between the classes to be seperated the separability measure is higher than those which doesn't have a clear visible valley.

Otsu’s Binarization (Contd.)

T determined using iterative method is 189, using otsu’s method is 205.

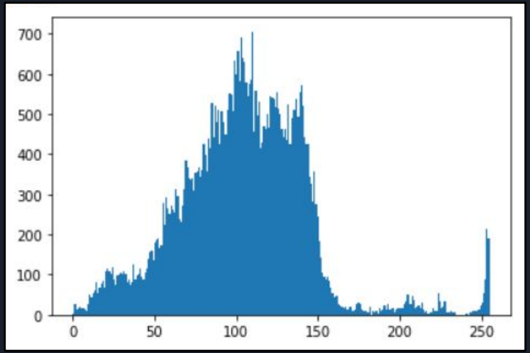


Histogram - Image - Iterative TH - Otsu’s TH

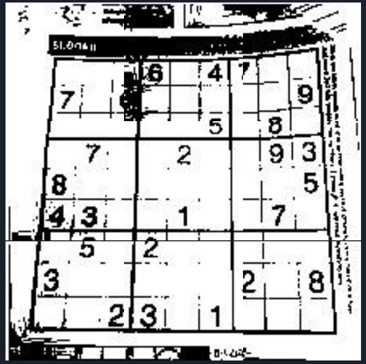
Demerits of Global Thresholding

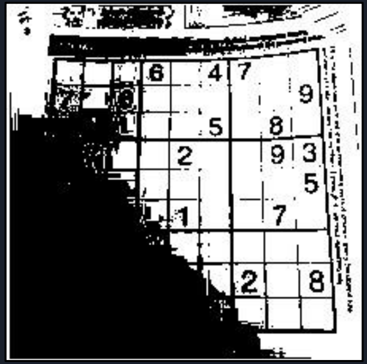
Global Thresholding suffers from issues like noise in the input image, uneven illumination and reflectance of objects because of which the output thresholded image is not often useful. Such effects are taken care by approaches such as variable thresholding. Consider the following image with uneven illumination and thresholded image obtained using Otsu's method, clearly illumination has affected the output.



Adaptive Thresholding

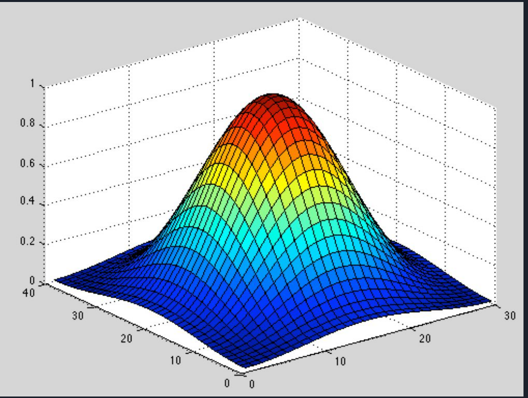
Dividing the image into sub- regions and applying global thresholding to each of the sub-region reduces the effect of illumination. This is the principle behind adaptive thresholding. Basic intuition behind variable thresholding is that for every pixel, the threshold is calculated based on the properties of the pixels that lie in the neighbourhood of the selected pixel. Because of this threshold value is a function of the pixel which was not the case in global thresholding.





Adaptive Thresholding (Contd.)

- Adaptive Mean Thresholding - In this method threshold value is taken as the mean intensity of the pixels in a certain window around the pixel and is shifted by a constant -C.
- Adaptive Gaussian Thresholding - In this method threshold value is taken as the mean intensity of the pixels weighted by a Gaussian window (images on the right) around the pixel and is shifted by a constant -C.
- Block Size, N - It decides the size of neighbourhood area.
- Constant, C - It is subtracted from the mean or weighted mean calculated.
- This approach is more robust to illumination gradient because of the reason mentioned previously.



$$\frac{1}{16}\begin{array}{|c|c|c|}\hline 1 & 2 & 1 \\ \hline 2 & 4 & 2 \\ \hline 1 & 2 & 1 \\ \hline \end{array}$$

Example

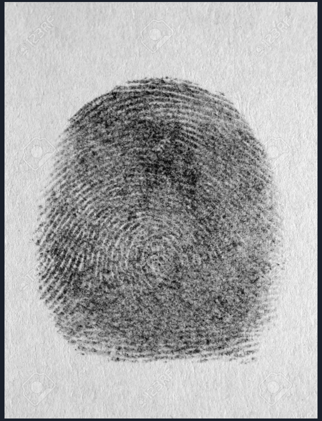
Clockwise from left to right :

- Image
- Otsu’s Binarization
- Adaptive Gaussian
- Adaptive mean

Applications

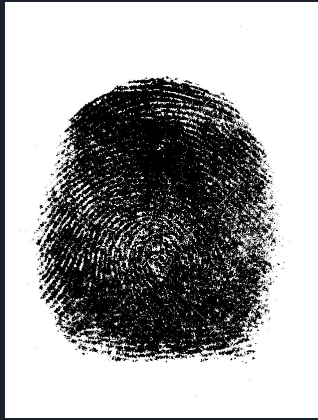
- Number plate segmentation
- Cell segmentation
- Apps such as Adobe Scan, Office Lens
- Fingerprint scanning

etc.



Input

Image thresholding using Otsu's method



References and Sources

- Digital Image Processing - 4th edition - Gonzalez, C. Rafael • Woods, E. Richard
- https://docs.opencv.org/master/d7/d4d/tutorial_py_thresholding.html
- https://en.wikipedia.org/wiki/Thresholding_(image_processing)
- Google Images