Image binarization

- Binarization is one of the most important preprocessing steps in most of the vision-based systems for object detection classification.
- Many image processing and computer vision applications usually require binary images as an introductory step for further processing.
- Thresholding is one of the easiest methods to automatically segment an image using a computer.
- Challenges : sensitive to lighting, shadows, noise and other atrifacts in the images.

A binary image $B(x, y)$ of a given grayscale image $I(x, y)$ is a representation of $I(x, y)$ with only two (bi) gray levels.

In a binary image, gray value 0 (black) generally represents object or foreground pixels and gray value 255 (white) represents the background pixels or vice versa.

For some threshold T (*gray level*)

$$B(x,y) = 1, \quad \text{if } I(x,y) \geq T$$
$$= 0, \quad \text{otherwise.}$$

Binarization methods can be broadly categorized into two groups; global and local methods depending on how threshold value is calculated for the image to be segmented.

If a single threshold value is used for the entire image, the corresponding method is a global binarization method.

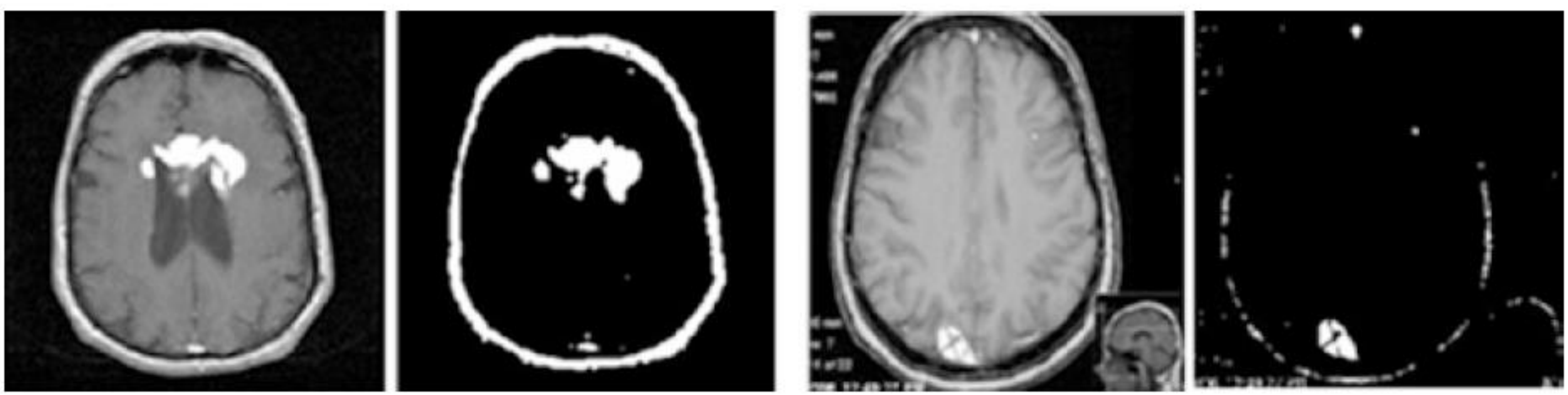
For a local method [3], a number of threshold values can be calculated for different regions of an image depending on some properties of the image.

Applications of Binarization

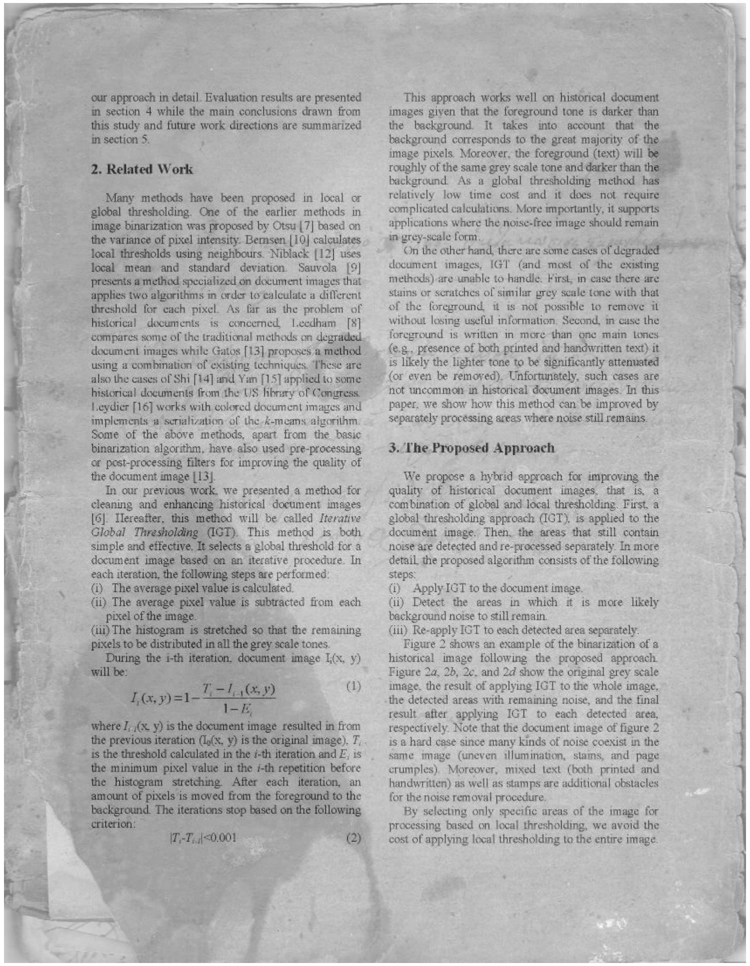
Some of the applications of binarization are listed below:

1. Document image analysis (e.g., extraction of printed logos, graphical content from document images; finding lines, legends, and characters in map processing).
2. OCR and quality inspection of materials.
3. Foreground background classification and object recognition and extraction of text embedded in images in hand-held devices.
4. Satellite image segmentation and identification of objects of interest in medical imaging.
5. Video processing, moving object detection, scene matching, etc.
6. Finger-vein pattern extraction and fingerprint preprocessing.

Document image analysis and OCR systems



Brain images (MRI)



our approach in detail. Evaluation results are presented in section 4 while the main conclusions drawn from this study and future work directions are summarized in section 5.

2. Related Work

Many methods have been proposed in local or global thresholding. One of the earlier methods in image binarization was proposed by Otsu [7] based on the variance of pixel intensity. Bernsen [10] calculates local thresholds using neighbours. Niblack [12] uses local mean and standard deviation. Sauvola [9] presents a method specialized on document images that applies two algorithms in order to calculate a different threshold for each pixel. As far as the problem of historical documents is concerned, Leedham [8] compares some of the traditional methods on degraded document images while Gatos [13] proposes a method using a combination of existing techniques. These are also the cases of Shi [14] and Yan [15] applied to some historical documents from the US library of Congress. Leydier [16] works with colored document images and implements a serialization of the *k*-means algorithm. Some of the above methods, apart from the basic binarization algorithm, have also used pre-processing or post-processing filters for improving the quality of the document image [13].

In our previous work, we presented a method for cleaning and enhancing historical document images [6]. Hereafter, this method will be called *Iterative Global Thresholding* (IGT). This method is both simple and effective. It selects a global threshold for a document image based on an iterative procedure. In each iteration, the following steps are performed:

(i) The average pixel value is calculated.
(ii) The average pixel value is subtracted from each pixel of the image.
(iii) The histogram is stretched so that the remaining pixels to be distributed in all the grey scale tones.

During the i-th iteration, document image $I_i(x, y)$ will be:

$$I_i(x,y) = 1 - \frac{T_i - I_{i-1}(x,y)}{1 - E_i} \quad (1)$$

where $I_{i-1}(x, y)$ is the document image resulted in from the previous iteration ($I_0(x, y)$ is the original image), T_i is the threshold calculated in the *i*-th iteration and E_i is the minimum pixel value in the *i*-th repetition before the histogram stretching. After each iteration, an amount of pixels is moved from the foreground to the background. The iterations stop based on the following criterion:

$$|T_i - T_{i-1}| < 0.001 \quad (2)$$

This approach works well on historical document images given that the foreground tone is darker than the background. It takes into account that the background corresponds to the great majority of the image pixels. Moreover, the foreground (text) will be roughly of the same grey scale tone and darker than the background. As a global thresholding method has relatively low time cost and it does not require complicated calculations. More importantly, it supports applications where the noise-free image should remain in grey-scale form.

On the other hand, there are some cases of degraded document images, IGT (and most of the existing methods) are unable to handle. First, in case there are stains or scratches of similar grey scale tone with that of the foreground, it is not possible to remove it without losing useful information. Second, in case the foreground is written in more than one main tones (e.g., presence of both printed and handwritten text) it is likely the lighter tone to be significantly attenuated (or even be removed). Unfortunately, such cases are not uncommon in historical document images. In this paper, we show how this method can be improved by separately processing areas where noise still remains.

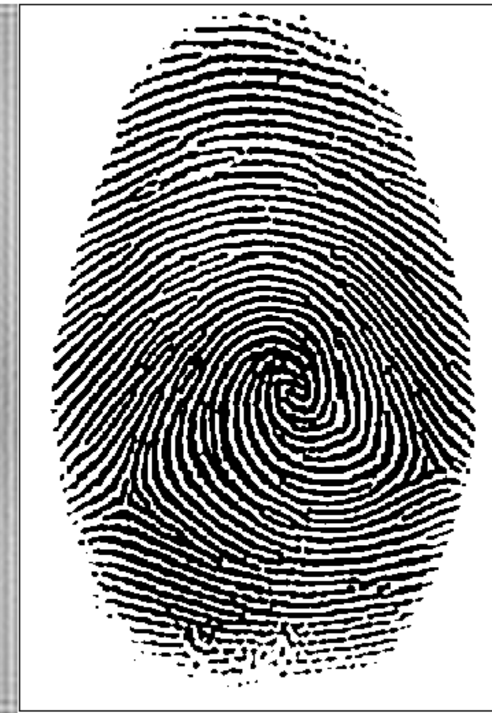
3. The Proposed Approach

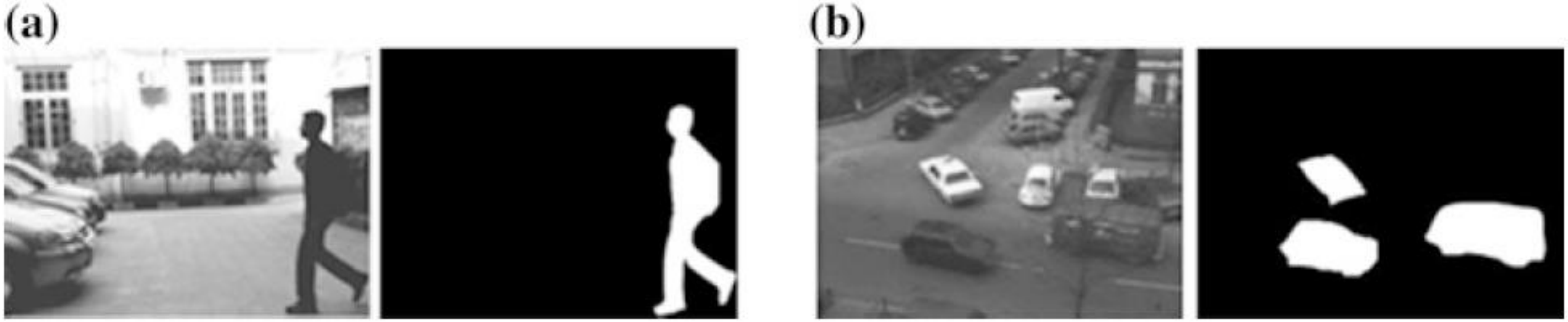
We propose a hybrid approach for improving the quality of historical document images, that is, a combination of global and local thresholding. First, a global thresholding approach (IGT), is applied to the document image. Then, the areas that still contain noise are detected and re-processed separately. In more detail, the proposed algorithm consists of the following steps:

(i) Apply IGT to the document image.
(ii) Detect the areas in which it is more likely background noise to still remain.
(iii) Re-apply IGT to each detected area separately.

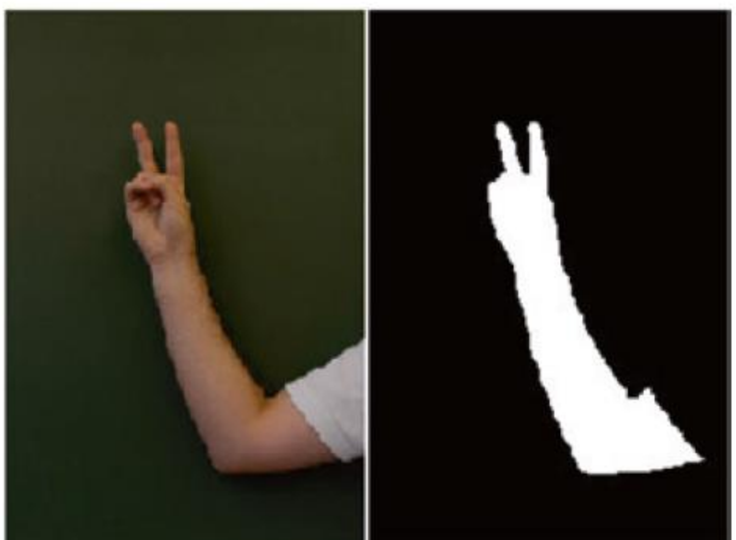
Figure 2 shows an example of the binarization of a historical image following the proposed approach. Figure 2*a*, 2*b*, 2*c*, and 2*d* show the original grey scale image, the result of applying IGT to the whole image, the detected areas with remaining noise, and the final result after applying IGT to each detected area, respectively. Note that the document image of figure 2 is a hard case since many kinds of noise coexist in the same image (uneven illumination, stains, and page crumples). Moreover, mixed text (both printed and handwritten) as well as stamps are additional obstacles for the noise removal procedure.

By selecting only specific areas of the image for processing based on local thresholding, we avoid the cost of applying local thresholding to the entire image.

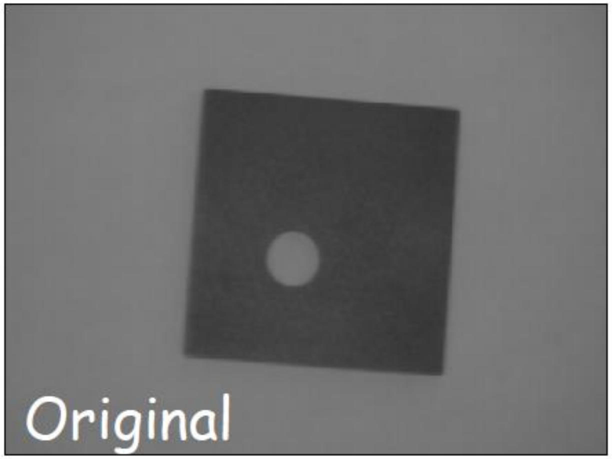


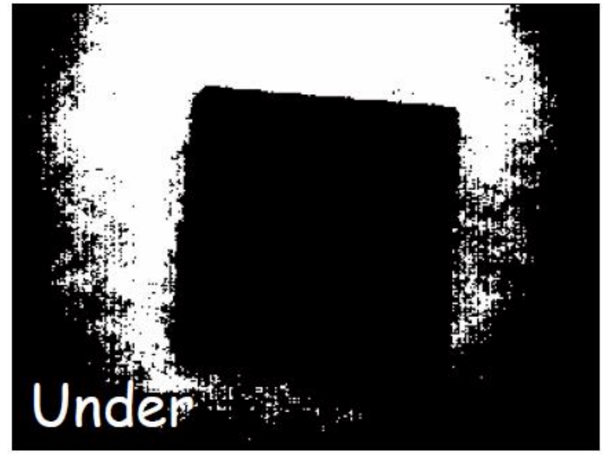


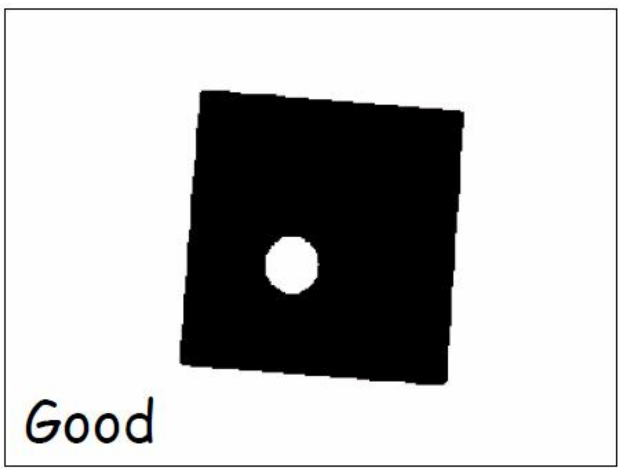
Detection of human being. **b**. Detection of vehicles in a traffic video. Moving object detection from video

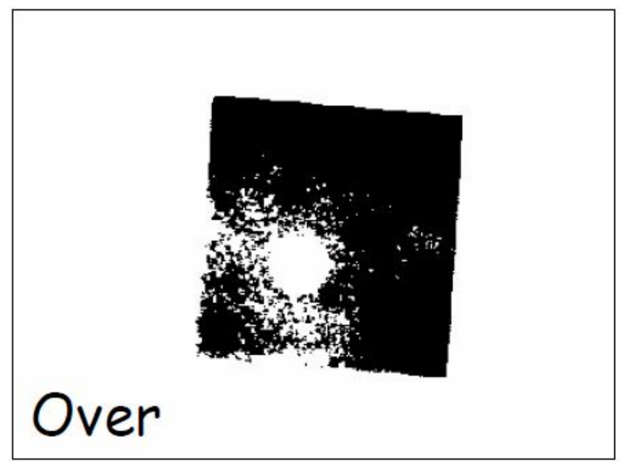


Hand Gesture Recognition. *Source* Hand Gesture Recognition (HGR) Database, Silesian University of Technology, Institute of Informatics, Poland. http://sun.aei.polsl.pl/~mkawulok/gestures/. Last Accessed: May 2014


Original


Under

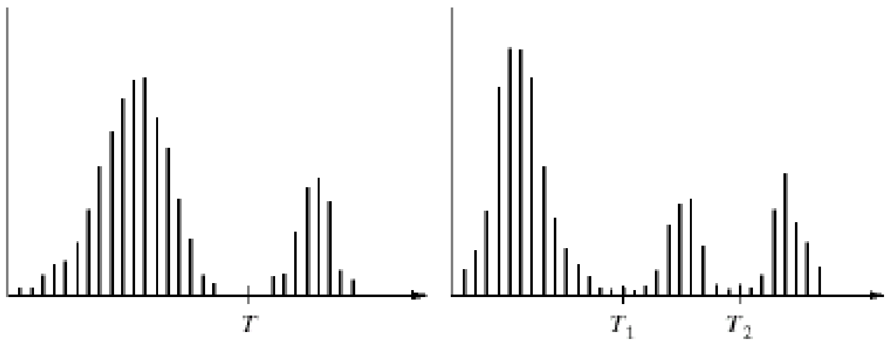

Good


Over

Optimal thresholding

- Idea: the histogram of an image is the sum of two overlapping distributions
- Optimal threshold: overlapping point of these distributions (corresponds to the minimum probability between the maxima of 2 distributions)
- Problem: distributions are unknown

- When the modes of histogram can be clearly distinguished



a b

FIGURE 10.26 (a) Gray-level histograms that can be partitioned by (a) a single threshold, and (b) multiple thresholds.

- Each mode represents either the background or an object

Comparison between conventional ar optimal thresholding

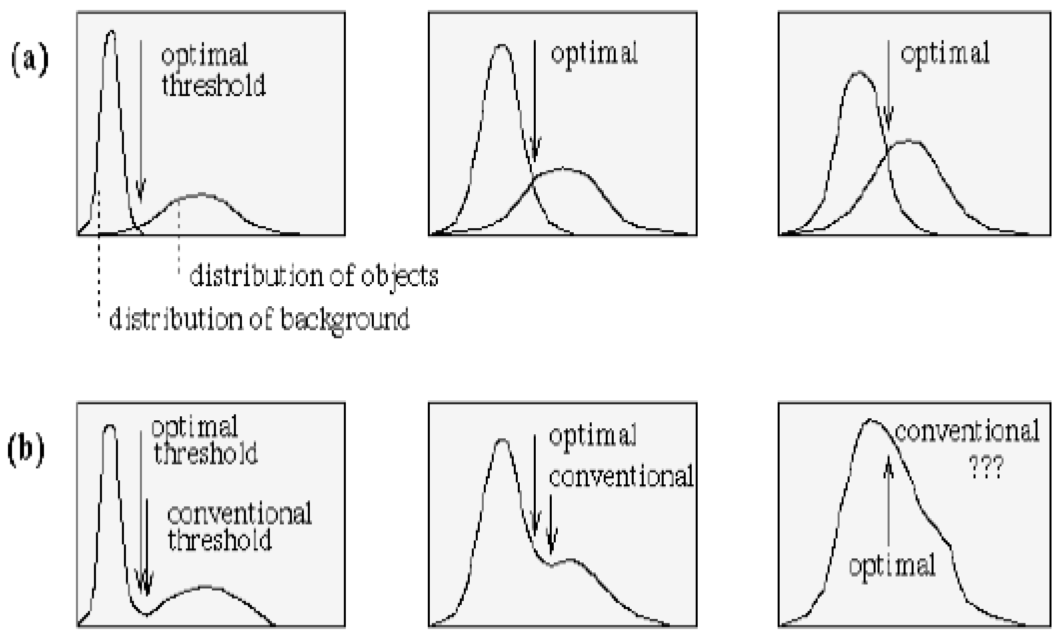


Figure 5.4 *Grey level histograms approximated by two normal distributions; the threshold is set to give minimum probability of segmentation error: (a) Probability distributions of background and objects, (b) corresponding histograms and optimal threshold.*

- Histogram-shaped-based
- Clustering-based
- Entropy-based
- Attribute similarity methods
- Object attribute-based
- Spatial approaches
- Local methods

Optimal thresholding by clustering

- Simplest case: segmentation into two classes (object/background).
- The intensities in each class will be our clusters.
- We want to find a threshold such that each pixel on each side of the threshold is closer in intensity to the mean of all pixels on that side of the threshold than to the mean of all pixels on the other side of the threshold.

Iterative optimal threshold selection

1. Select an initial estimate for T

2. Segment the image using T. This produces 2 groups: G1 pixels with value >T and G2 , with value <T

3. Compute μ1 and μ2, average pixel values of G1 and G2

4. New threshold: $T=1/2(\mu 1+\mu 2)$

5. Repeat steps 2 to 4 until T stabilizes.

Otsu’s Method

Nobuyuki Otsu

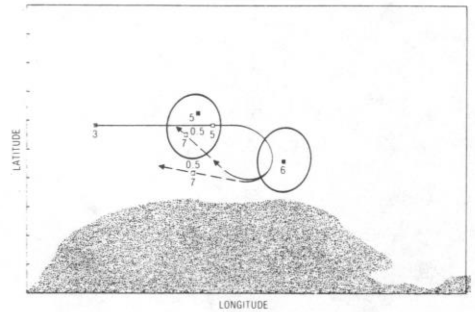


Fig. 6. ID plot of ship 10001 after the second round of operator-imposed assignment constraints.

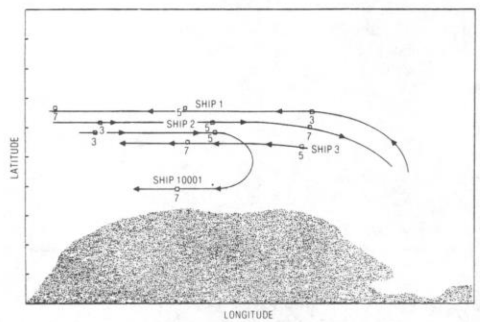


Fig. 7. Actual ship movements.

of the two last sighted locations. The true trajectories are shown in Fig. 7 where it can be seen that ship 10001 did, in fact, turn toward the coast.

IV. Concluding Remarks

The procedure of ship identification from DF sightings has been oversimplified in this discussion. Often DF sightings are not completely identified but, instead, contain only ship class information. The interactive technique still applies, but additional identification and display flexibility must be provided.

Any additional information contained in the sightings can be used to discriminate among radar and DF sightings. Factors such as measured heading and visual ID will permit further automatic reduction of the P and Q matrices.

It is also possible to automate some of the more routine manual functions. However, experience has shown that better results are obtained by having a human operator resolve ambiguous situations arising from sparse data.

References

[1] R. W. Sittler, "An optimal data association problem in surveillance theory," *IEEE Trans. Mil. Elect.*, vol. MIL-8, pp. 125–139, 1964.
[2] M. S. White, "Finding events in a sea of bubbles," *IEEE Trans. Comput.*, vol. C-20 (9), pp. 988–1006, 1971.
[3] A. G. Jaffer and Y. Bar-Shalom, "On optimal tracking in multiple-target environments," *Proc. of the 3rd Sym. on Nonlinear Estimation Theory and its Applications*, San Diego, CA, Sept. 1972.
[4] P. Smith and G. Buechler, "A branching algorithm for discrimination and tracking multiple objects," *IEEE Trans. Automat. Contr.*, vol. AC-20, pp. 101–104, 1975.
[5] D. L. Alspach, "A Gaussian sum approach to the multitarget-tracking problem," *Automatica*, vol. 11, pp. 285–296, 1975.
[6] C. L. Morefield, *Application of 0-1 Integer Programming to a Track Assembly Problem*, TR-0075(5085-10)-1, Aerospace Corp. El Segundo, CA, Apr. 1975.
[7] D. B. Reid, *A Multiple Hypothesis Filter for Tracking Multiple Targets in a Cluttered Environment*, LMSC-D560254, Lockheed Palo Alto Research Laboratories, Palo Alto, CA, Sept. 1977.
[8] P. L. Smith, "Reduction of sea surveillance data using binary matrices," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-6 (8), pp. 531–538, Aug. 1976.

A Threshold Selection Method from Gray-Level Histograms

NOBUYUKI OTSU

Abstract—**A nonparametric and unsupervised method of automatic threshold selection for picture segmentation is presented. An optimal threshold is selected by the discriminant criterion, namely, so as to maximize the separability of the resultant classes in gray levels. The procedure is very simple, utilizing only the zeroth- and the first-order cumulative moments of the gray-level histogram. It is straightforward to extend the method to multithreshold problems. Several experimental results are also presented to support the validity of the method.**

I. Introduction

It is important in picture processing to select an adequate threshold of gray level for extracting objects from their background. A variety of techniques have been proposed in this regard. In an ideal case, the histogram has a deep and sharp valley between two peaks representing objects and background, respectively, so that the threshold can be chosen at the bottom of this valley [1]. However, for most real pictures, it is often difficult to detect the valley bottom precisely, especially in such cases as when the valley is flat and broad, imbued with noise, or when the two peaks are extremely unequal in height, often producing no traceable valley. There have been some techniques proposed in order to overcome these difficulties. They are, for example, the valley sharpening technique [2], which restricts the histogram to the pixels with large absolute values of derivative (Laplacian or gradient), and the difference histogram method [3], which selects the threshold at the gray level with the maximal amount of difference. These utilize information concerning neighboring pixels (or edges) in the original picture to modify the histogram so as to make it useful for thresholding. Another class of methods deals directly with the gray-level histogram by parametric techniques. For example, the histogram is approximated in the least square sense by a sum of Gaussian distributions, and statistical decision procedures are applied [4]. However, such a method requires considerably tedious and sometimes unstable calculations. Moreover, in many cases, the Gaussian distributions turn out to be a meager approximation of the real modes.

In any event, no "goodness" of threshold has been evaluated in

Manuscript received October 13, 1977; revised April 17, 1978 and August 31, 1978.
The author is with the Mathematical Engineering Section, Information Science Division, Electrotechnical Laboratory, Chiyoda-ku, Tokyo 100, Japan.

0018-9472/79/0100-0062$00.75 © 1979 IEEE

- Otsu’s method is the most successful global thresholding method. It automatically performs histogram shape-based image thresholding for the reduction of a gray-level image to a binary image.

- The algorithm assumes that the image for thresholding contains two classes of pixels (e.g., foreground and background) and then calculates the optimum threshold separating those two classes so that their combined spread (intra-class variance) is minimal.

- It exhaustively searches for the threshold that minimizes the intra-class variance, defined as the weighted sum of variances of the two classes.

The *weighted within-class variance* is:

$$\sigma_w^2(t) = q_1(t)\sigma_1^2(t) + q_2(t)\sigma_2^2(t)$$

Where the class probabilities are estimated as:

$$q_1(t) = \sum_{i=1}^{t} P(i) \qquad q_2(t) = \sum_{i=t+1}^{I} P(i)$$

And the class means are given by:

$$\mu_1(t) = \sum_{i=1}^{t} \frac{iP(i)}{q_1(t)} \qquad \mu_2(t) = \sum_{i=t+1}^{I} \frac{iP(i)}{q_2(t)}$$

The *weighted within-class variance* is:

$$\sigma_w^2(t) = q_1(t)\sigma_1^2(t) + q_2(t)\sigma_2^2(t)$$

Where the class probabilities are estimated as:

$$q_1(t) = \sum_{i=1}^{t} P(i) \qquad q_2(t) = \sum_{i=t+1}^{I} P(i)$$

And the class means are given by:

$$\mu_1(t) = \sum_{i=1}^{t} \frac{iP(i)}{q_1(t)} \qquad \mu_2(t) = \sum_{i=t+1}^{I} \frac{iP(i)}{q_2(t)}$$

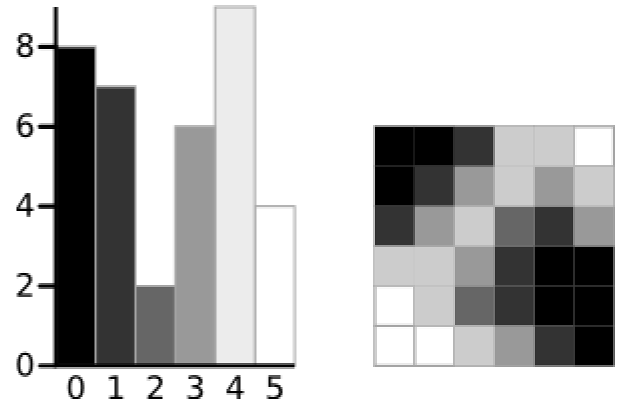
Finally, the individual class variances are:

$$\sigma_1^2(t) = \sum_{i=1}^{t} [i - \mu_1(t)]^2 \frac{P(i)}{q_1(t)}$$

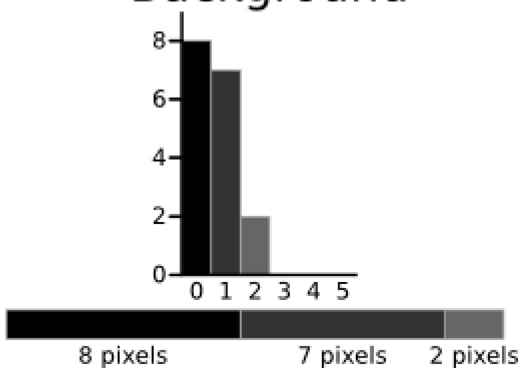
$$\sigma_2^2(t) = \sum_{i=t+1}^{I} [i - \mu_2(t)]^2 \frac{P(i)}{q_2(t)}$$

Now, we could actually stop here. All we need to do is just run through the full range of t values [1,256] and pick the value that minimizes $\sigma_w^2(t)$.

But the relationship between the within-class and between-class variances can be exploited to generate a recursion relation that permits a much faster calculation.



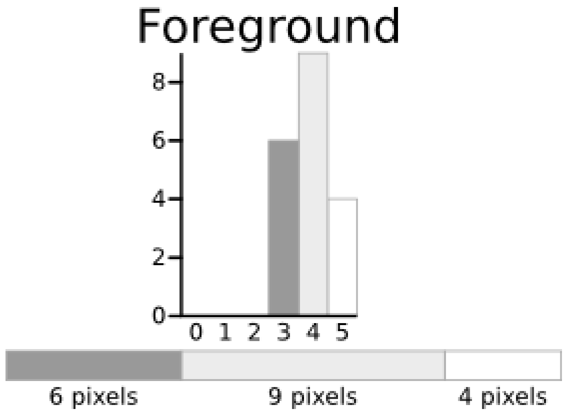
Background



$$\text{Weight} \quad W_b = \frac{8+7+2}{36} = 0.4722$$

$$\text{Mean} \quad \mu_b = \frac{(0 \times 8) + (1 \times 7) + (2 \times 2)}{17} = 0.6471$$

$$\begin{aligned} \text{Variance} \quad \sigma_b^2 &= \frac{((0 - 0.6471)^2 \times 8) + ((1 - 0.6471)^2 \times 7) + ((2 - 0.6471)^2 \times 2)}{17} \\ &= \frac{(0.4187 \times 8) + (0.1246 \times 7) + (1.8304 \times 2)}{17} \\ &= 0.4637 \end{aligned}$$

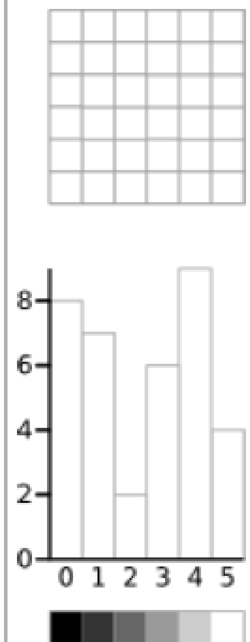
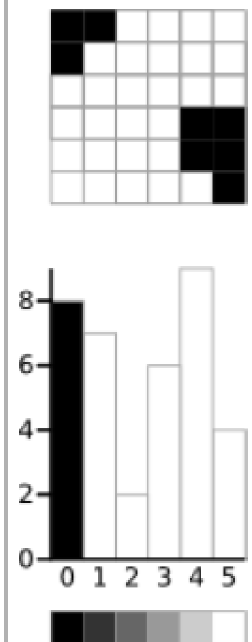
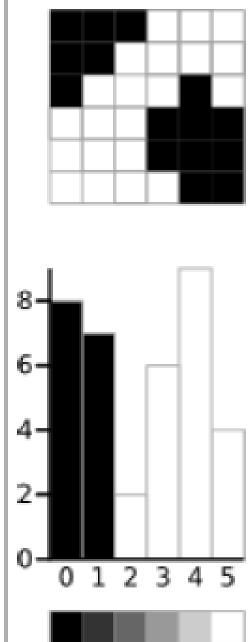
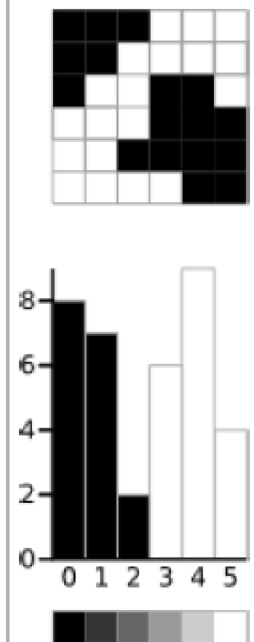
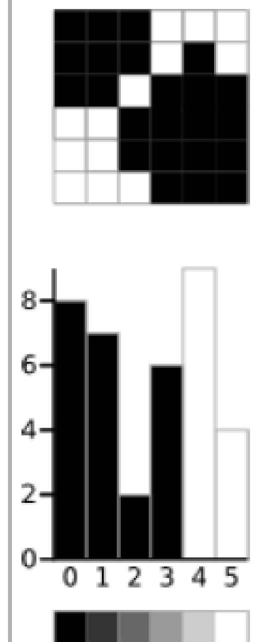
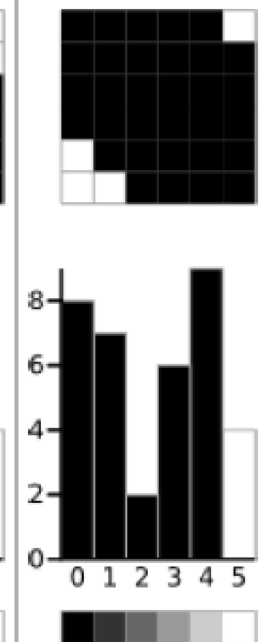


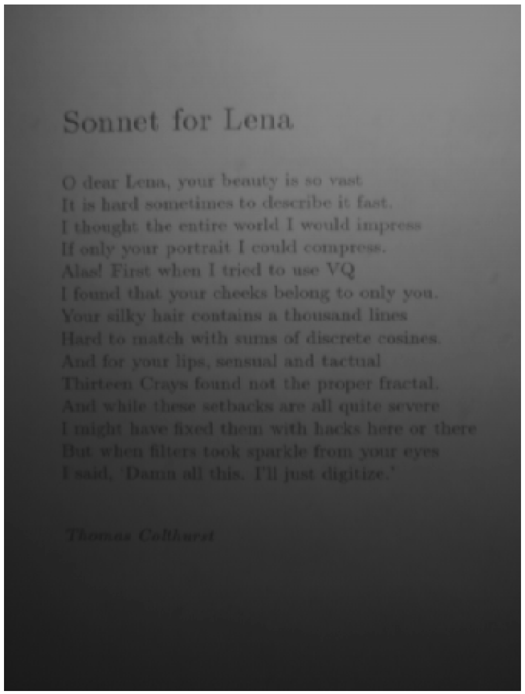
$$\text{Weight} \quad W_f = \frac{6+9+4}{36} = 0.5278$$

$$\text{Mean} \quad \mu_f = \frac{(3 \times 6) + (4 \times 9) + (5 \times 4)}{19} = 3.8947$$

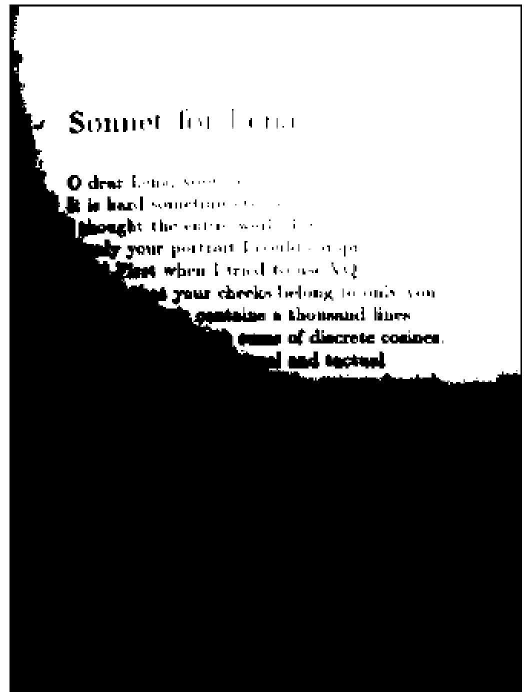
$$\begin{aligned}
\text{Variance} \quad \sigma_f^2 &= \frac{((3 - 3.8947)^2 \times 6) + ((4 - 3.8947)^2 \times 9) + ((5 - 3.8947)^2 \times 4)}{19} \\
&= \frac{(4.8033 \times 6) + (0.0997 \times 9) + (4.8864 \times 4)}{19} \\
&= 0.5152
\end{aligned}$$

$$\begin{aligned}
\text{Within Class Variance} \quad \sigma_W^2 = W_b\, \sigma_b^2 + W_f\, \sigma_f^2 &= 0.4722 * 0.4637 + 0.5278 * 0.5152 \\
&= 0.4909
\end{aligned}$$

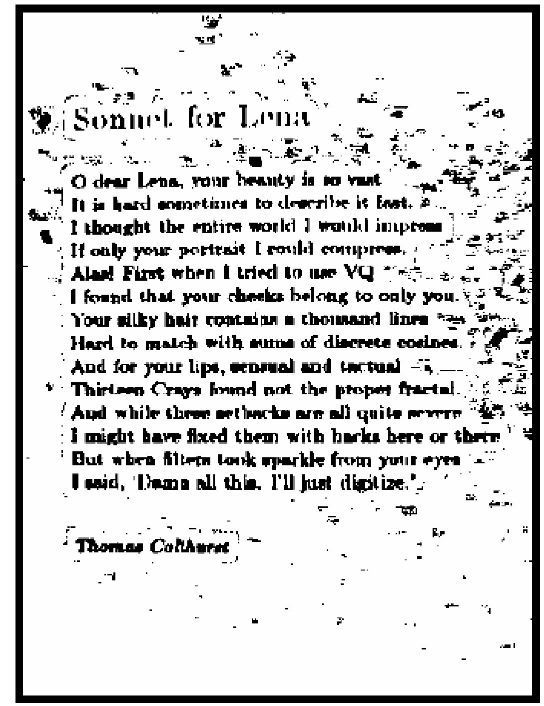
Threshold	T=0	T=1	T=2	T=3	T=4	T=5
						
Weight, Background	$W_b = 0$	$W_b = 0.222$	$W_b = 0.4167$	$W_b = 0.4722$	$W_b = 0.6389$	$W_b = 0.8889$
Mean, Background	$M_b = 0$	$M_b = 0$	$M_b = 0.4667$	$M_b = 0.6471$	$M_b = 1.2609$	$M_b = 2.0313$
Variance, Background	$\sigma^2{}_b = 0$	$\sigma^2{}_b = 0$	$\sigma^2{}_b = 0.2489$	$\sigma^2{}_b = 0.4637$	$\sigma^2{}_b = 1.4102$	$\sigma^2{}_b = 2.5303$
Weight, Foreground	$W_f = 1$	$W_f = 0.7778$	$W_f = 0.5833$	$W_f = 0.5278$	$W_f = 0.3611$	$W_f = 0.1111$
Mean, Foreground	$M_f = 2.3611$	$M_f = 3.0357$	$M_f = 3.7143$	$M_f = 3.8947$	$M_f = 4.3077$	$M_f = 5.000$
Variance, Foreground	$\sigma^2{}_f = 3.1196$	$\sigma^2{}_f = 1.9639$	$\sigma^2{}_f = 0.7755$	$\sigma^2{}_f = 0.5152$	$\sigma^2{}_f = 0.2130$	$\sigma^2{}_f = 0$
Within Class Variance	$\sigma^2{}_W = 3.1196$	$\sigma^2{}_W = 1.5268$	$\sigma^2{}_W = 0.5561$	$\boldsymbol{\sigma^2{}_W = 0.4909}$	$\sigma^2{}_W = 0.9779$	$\sigma^2{}_W = 2.2491$



Original



Otsu



Local Otsu
on 7x7 window

Given a set of observations ($\mathbf{x}_1$, $\mathbf{x}_2$, …, $\mathbf{x}_n$), where each observation is a *d*-dimensional real vector, *k*-means clustering aims to partition the *n* observations into *k* (≤*n*) sets $\mathbf{S} = \{S_1, S_2, \ldots, S_k\}$ so as to minimize the within-cluster sum of squares (WCSS) (sum of distance functions of each point in the cluster to the K center). In other words, its objective is to find:

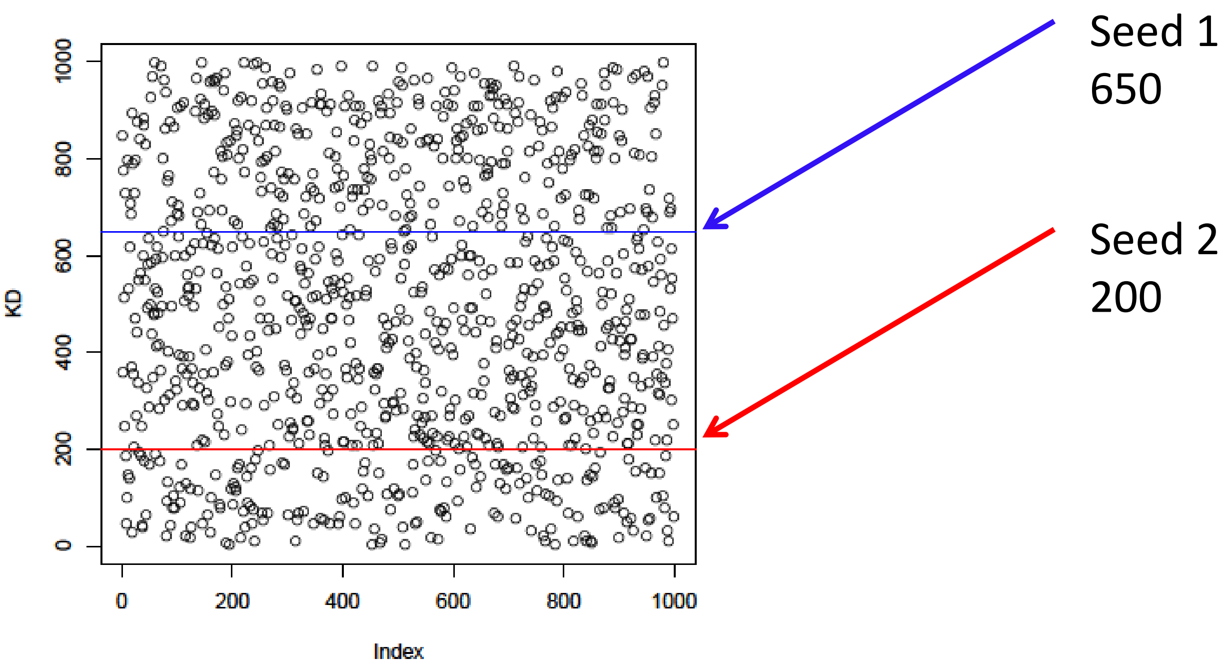
$$\underset{\mathbf{S}}{\arg\min} \sum_{i=1}^{k} \sum_{\mathbf{x} \in S_i} \left\| \mathbf{x} - \boldsymbol{\mu}_i \right\|^2$$

where $\boldsymbol{\mu}_i$ is the mean of points in S_i.

K-means Clustering

Basic Algorithm:

- Step 0: select K
- Step 1: randomly select initial cluster seeds



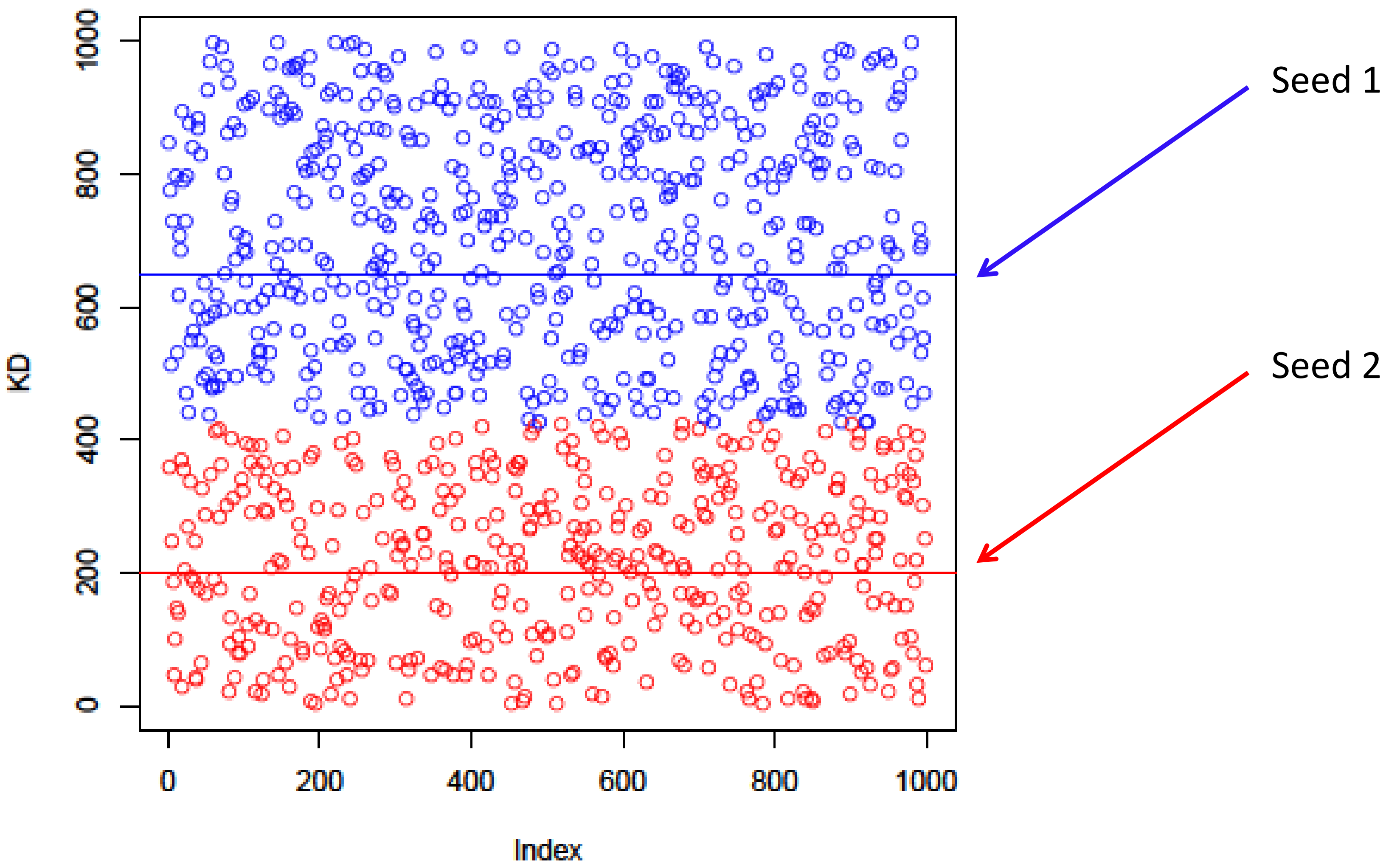
K-means Clustering

- An initial cluster seed represents the “mean value” of its cluster.
- In the preceding figure:
  - Cluster seed 1 = 650
  - Cluster seed 2 = 200

K-means Clustering

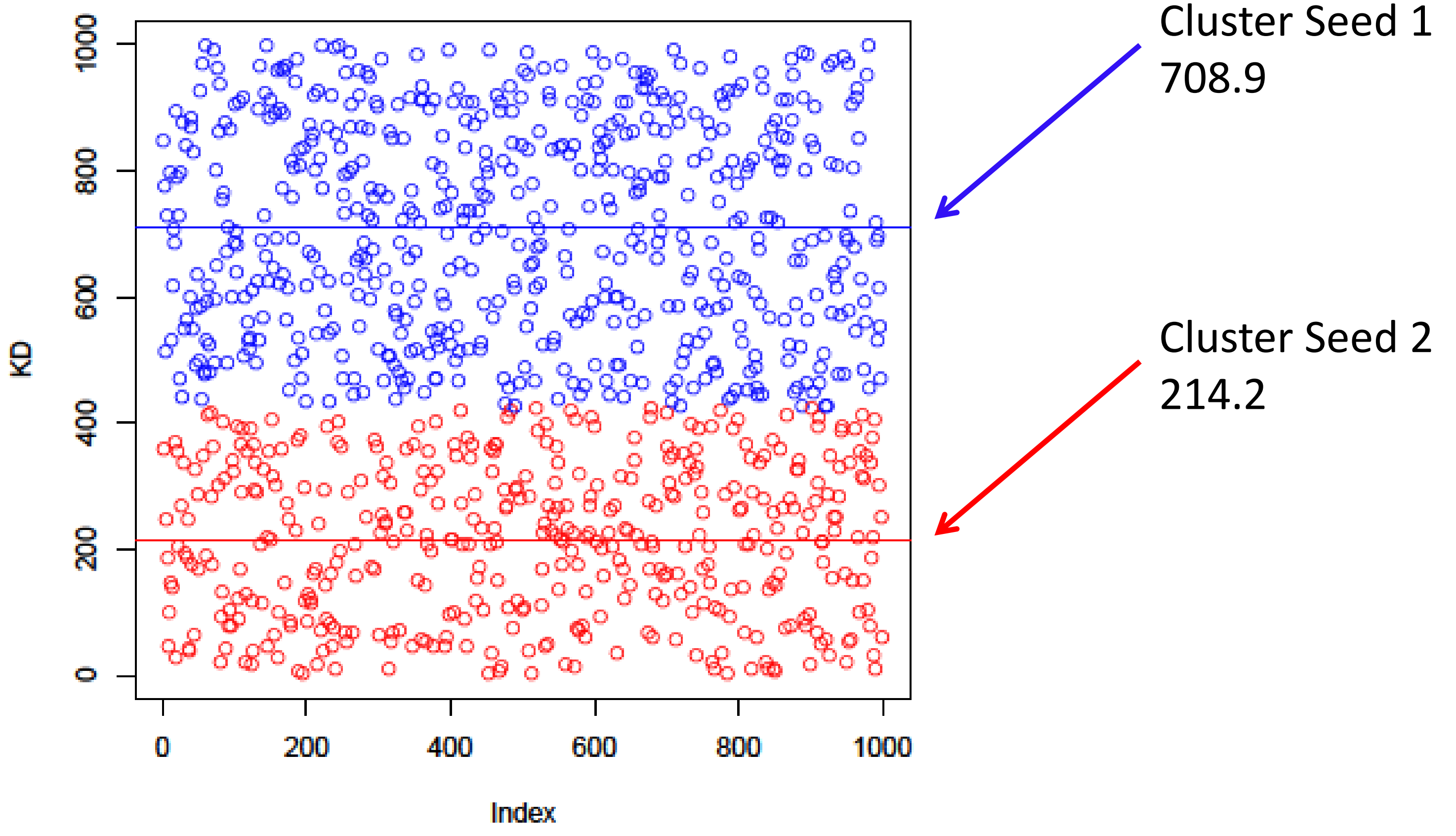
- Step 2: calculate distance from each object to each cluster seed.
- What type of distance should we use?
  - Squared Euclidean distance
- Step 3: Assign each object to the closest cluster

K-means Clustering



K-means Clustering

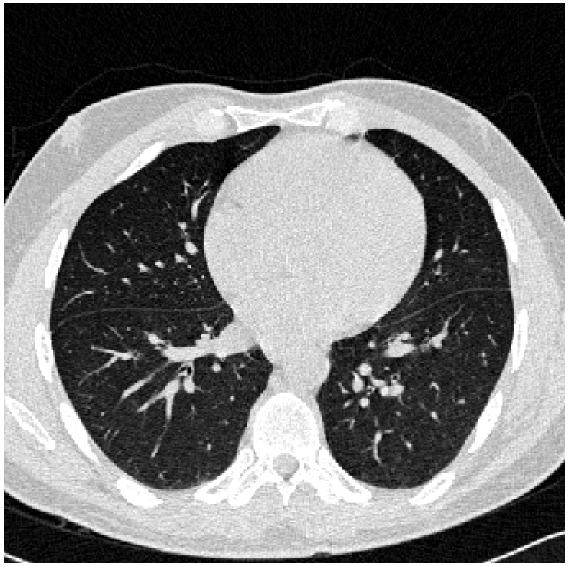
- Step 4: Compute the new centroid for each cluster



K-means Clustering

- Iterate:
  - Calculate distance from objects to cluster centroids.
  - Assign objects to closest cluster
  - Recalculate new centroids
- Stop based on convergence criteria
  - No change in clusters
  - Max iterations

K-means Image Segmentation



An image (*I*)

Three-cluster image (*J*) on gray values of *I*

Matlab code:

```
I = double(imread('…'));

J = reshape(kmeans(I(:),3),size(I));
```

Note that *K*-means result is "noisy"