Using Image Smoothing To Improve Global Thresholding

Figure 10.40 - success

Figure 10.41 - failure

need more local approach

Variable Thresholding

Often simply can't pick effective global threshold due to

- image noise
- nonuniform illumination

Instead, vary threshold at different locations of image.

Image Partitioning

Subdivide image into nonoverlapping rectangles and perform (global) thresholding in each

Fig. 10.46. and 10.47.

Problem: what if rectangle contains only object or background?

Variable Thresholding Based on Local Image Properties

Compute threshold at every point (x,y) in image based on properties computed in neighborhood of (x,y)

Ex. Suppose σ_{xy} and m_{xy} are mean & standard deviation in neighborhood S_{xy} of pt. (x,y).

Choose $T_{xy} = a\sigma_{xy} + bm_{xy}$

where a, b are non-negative constants.

Segmented image:

$$g(x,y) = \begin{cases} 1 & \text{if } f(x,y) > T_{xy} \\ 0 & \text{if } f(x,y) \leq T_{xy} \end{cases}$$

Alternately

$T_{xy} = a\sigma_{xy} + bm_G$ where m_G is global mean.

Also, $g(x,y) = \begin{cases} 1 & \text{if } f(x,y) > a\sigma_{xy} \text{ AND } f(x,y) > bm_{xy} \\ 0 & \text{otherwise} \end{cases}$

Moving Averages

Threshold based on average of pixels in neighborhood.

Often neighborhood taken in one dimension in text segmentation.

Along each line:

- let z_{k+1} be intensity at pt. $k+1$
- Moving average

$$m(k+1) = \frac{1}{n} \sum_{i=k+2-n}^{k+1} z_i$$

$$= m(k) + \frac{1}{n}\left(z_{k+1} - z_{k-n}\right)$$

- initialize each line $m(1) = \frac{z_1}{n}$

→ efficient!

Let $T_{xy} = bm_{xy}$

Fig. 10.49 - illumination typical of photographic flash.

Here $n = 20$ ≈ 5 × average stroke width (~4 pixels here)

Fig. 10.50