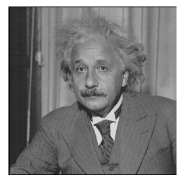
Binary Image Analysis

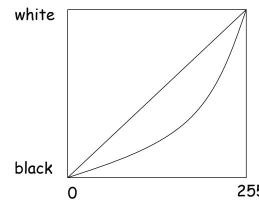
Jana Kosecka
http://cs.gmu.edu/~kosecka/cs482.html

Pointwise Image Operations

- Lookup table – match image intensity to the displayed brightness values

Manipulation of the lookup table – different
Visual effects – mapping is often non-linear
image is not changed





Contrast

Contrast gamma

$g(u) = u^{\alpha}$

Brightness

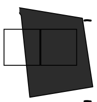
Binary Images

- Go from gray–level images to black–white.
- Industrial applications
- Simple tasks – counting objects, regions, connected components, thinning, thickening
- Thresholding

1. Select pixels as foreground pixels and background pixels

Image segmentation

- Ideally, object pixels would be black (0 intensity) and background pixels white (maximum intensity)
- But this rarely happens
  - pixels overlap regions from both the object and the background, yielding intensities between pure black and white - edge blur
  - cameras introduce "noise" during imaging - measurement "noise"
  - potatoes have non-uniform "thickness", giving variations in brightness in X-ray - model "noise"



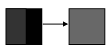


Image segmentation by thresholding

- But if the objects and background occupy different ranges of gray levels, we can "mark" the object pixels by a process called **thresholding:**
  - Let F(i,j) be the original, gray level image
  - B(i,j) is a **binary image** (pixels are either 0 or 1) created by **thresholding** F(i,j)
    - B(i,j) = 1 if F(i,j) < t
    - B(i,j) = 0 if F(i,j) >= t
    - We will assume that the 1's are the object pixels and the 0's are the background pixels

Quantization

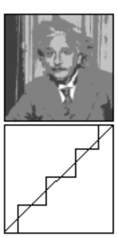
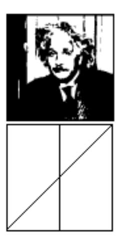


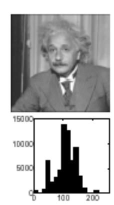
Image quantized to 5 different gray-levels

Thresholding



2 levels

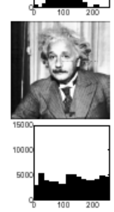
Histogram



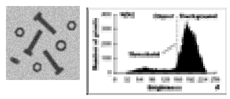
Histogram – frequency gray-level -> empirical distribution of the intensity values
h[i] – number of pixels of intensity i

Histogram equalization – making histogram flat

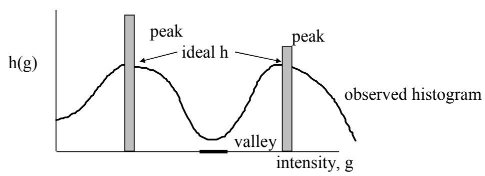
Passing the image through look-up table of the form of cumulative distribution function



Thresholding



- How do we choose the threshold t for segmentation?
- Histogram (h) - gray level frequency distribution of the gray level image F.
  - $h_F(g)$ = number of pixels in F whose gray level is g
  - $H_F(g)$ = number of pixels in F whose gray level is <=g

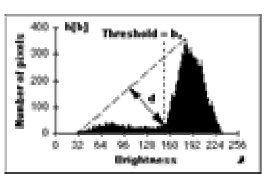


Thresholding

- P-tile method
  - in some applications we know approximately what percentage, p, of the pixels in the image come from objects
    - might have one potato in the image, or one character.
  - H_F can be used to find the gray level, g, such that ~p% of the pixels have intensity <= g
  - Then, we can examine h_F in the neighborhood of g to find a good threshold (low valley point)

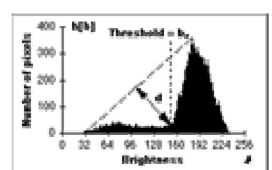
Thresholding

- Peak and valley method
  - Find the two most prominent peaks of h
    - g is a peak if $h_F(g) > h_F(g \pm \Delta g)$, $\Delta g = 1, \ldots, k$
  - Let g_1 and g_2 be the two highest peaks, with $g_1 < g_2$
  - Find the deepest valley, g, between g_1 and g_2
    - g is the valley if $h_F(g) <= h_F(g')$, g,g' in $[g_1, g_2]$
  - Use g as the threshold



Triangle algorithm

- A line is constructed between the maximum of the histogram at brightness b_{max} and the lowest value b_{min} = (p=0)% in the image.
- The distance d between the line and the histogram h[b] is computed for all values of b from $b = b_{min}$ to $b = b_{max}$.
- The brightness value b_o where the distance between $h[b_o]$ and the line is maximal is the threshold value.
- This technique is particularly effective when the object pixels produce a weak peak in the histogram.



Thresholding

- Hand selection
  - select a threshold by hand at the beginning of the day
  - use that threshold all day long!
- Many threshold selection methods in the literature
  - Probabilistic methods
    - make parametric assumptions about object and background intensity distributions and then derive "optimal" thresholds
  - Structural methods
    - Evaluate a range of thresholds wrt properties of resulting binary images
      - one with straightest edges, most easily recognized objects, etc.
  - Local thresholding
    - apply thresholding methods to image windows

An advanced probabilistic threshold selection method - minimizing Kullback information distance

- Suppose the observed histogram, f, is a mixture of the gray levels of the pixels from the object(s) and the pixels from the background
  - in an ideal world the histogram would contain just two spikes (this depends of the class of images/objects)
  - but
    - measurement noise
    - model noise (e.g., variations in ink density within a character)
    - edge blur (misalignment of object boundaries with pixel boundaries and optical imperfections of camera)

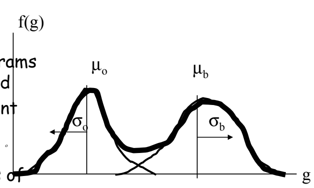
  spread these spikes out into hills

Kullback information distance

- Make a parametric model of the shapes of the component histograms of the objects(s) and background
- Parametric model - the component histograms are assumed to be Gaussian
  - p_o and p_b are the proportions of the image that comprise the objects and background

    μ_o and μ_b are the mean gray levels of the objects and background

    σ_o and σ_b- are their standard deviations



$$f_o(g) = \frac{p_o}{\sqrt{2\pi}\sigma_o} e^{-1/2(\frac{g-\mu_o}{\sigma_o})^2}$$

$$f_b(g) = \frac{p_b}{\sqrt{2\pi}\sigma_b} e^{-1/2(\frac{g-u_b}{\sigma_b})^2}$$

Kullback information distance

- Now, if we hypothesize a threshold, t, then all of these unknown parameters can be approximated from the image histogram.
- Let f(g) be the observed and normalized histogram
  - f(g) = percentage of pixels from image having gray level g

$$p_o(t) = \sum_{g=0}^{t} f(g) \qquad p_b(t) = 1 - p_0(t)$$

$$\mu_o(t) = \sum_{g=0}^{t} f(g)g \qquad \mu_b(t) = \sum_{g=t+1}^{\max} f(g)g$$

Kullback information distance

- So, for any hypothesized t, we can “predict” what the total normalized image histogram **should** be if our model (mixture of two Gaussians) is correct.
  - $P_t(g) = p_o f_o(g) + p_b f_b(g)$
- The total normalized image histogram is **observed to be** f(g)
- So, the question reduces to:
  - determine a suitable way to measure the <u>similarity</u> of P and f
  - then search for the t that gives the highest similarity

Kullback information distance

- A suitable similarity measure is the Kullback directed divergence, defined as

$$K(t) = \sum_{g=0}^{\max} f(g) \log[\frac{f(g)}{P_t(g)}]$$

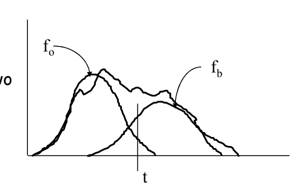
  If P_t matches f exactly, then each term of the sum is 0 and K(t) takes on its minimal value of 0
- Gray levels where P_t and f disagree are penalized by the log term, weighted by the importance of that gray level (f(g))

An alternative - minimize probability of error

- Using the same mixture model, we can search for the t that minimizes the predicted probability of error during thresholding
- Two types of errors
  - background points that are marked as object points. These are points from the background that are darker than the threshold
  - object points that are marked as background points. These are points from the object that are brighter than the threshold

An alternative - mimimize probability of error

- For each “reasonable” threshold
  - compute the parameters of the two Gaussians and the proportions
  - compute the two probability of errors
- Find the threshold that gives
  - minimal overall error
  - most equal errors



$$e_b(t) = p_b \sum_{g=0}^{t} f_b(g)$$

$$e_o(t) = p_o \sum_{g=t+1}^{\max} f_o(g)$$