Thresholding Based Segmentation

Robert B. Fisher
School of Informatics
University of Edinburgh

Slide credit: Bob Fisher

Isolating flat parts

Isolate parts, then characterise later

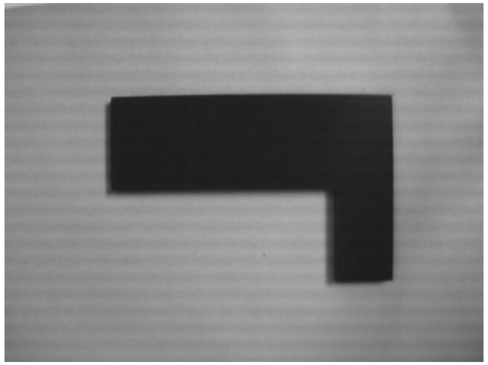
Assume

- Dark part
- Light background
- Reasonably $\boxed{?}$ illumination − > distinguishable parts

Slide credit: Bob Fisher

Motivating Example

Given this image, how might we label pixels as object and background?

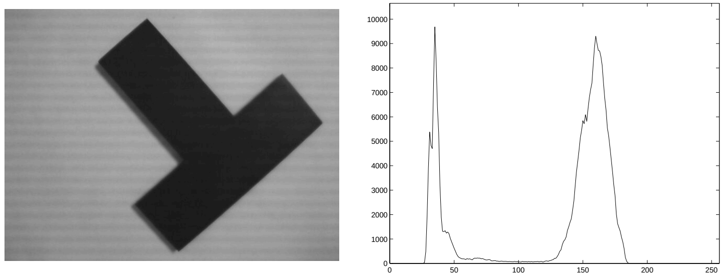


Slide credit: Bob Fisher

Thresholding Introduction

Key technique: thresholding
Assume pixel values are $\boxed{?}$

Part and typical distribution



Spread: not quite uniform illumination + part color variations + sensor noise

Slide credit: Bob Fisher

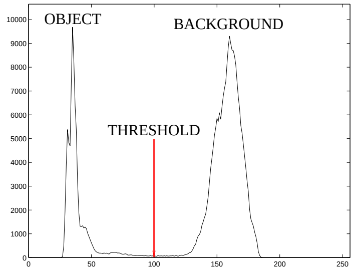
Thresholding Algorithm

Thresholding: ? technique

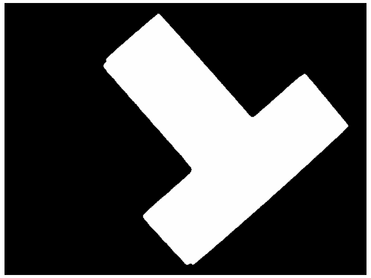
```
for row = 1 : height
 for col = 1 : width
  if value(row,col) < ThreshHigh % inside high bnd
  % & value(row,col) > ThreshLow % optional low bnd
      output(row,col) = 1;
  else
      output(row,col) = 0;
  end
```

Slide credit: Bob Fisher

Thresholding Example 1



Histogram

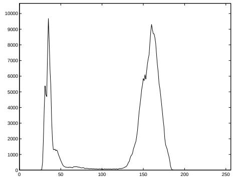


Thresholded Image

Slide credit: Bob Fisher

Threshold Selection 1

Exploit ? distribution



But:

- Distributions broad and some overlap − > misclassified pixels
- Shadows dark so might be classified with object
- Distribution has more than 2 peaks

Slide credit: Bob Fisher

So: smooth histogram to improve shape for selection

Slide credit: Bob Fisher

Convolution

General purpose image (and signal) processing function

Computed by a ? sum of image data and a fixed mask

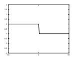
Linear operator: conv(a*B,C) = a*conv(B,C)

Used in different processes: noise removal, smoothing, feature detection, differentiation, ...

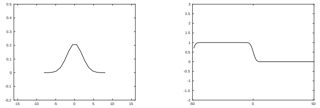
Slide credit: Bob Fisher

Convolution in 1D

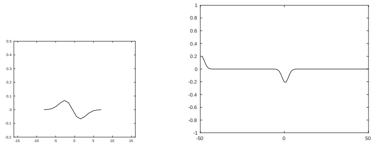
$$Output(x) = \sum_{i=-N}^{N} weight(i) * input(x-i)$$

Input: 

Gaussian Mask and Output:



? of Gaussian Mask and Output:



Slide credit: Bob Fisher

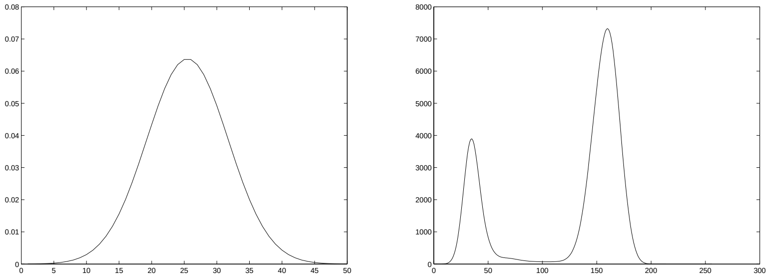
Histogram ? for Threshold Selection

Histogram Smoothing (in `findthresh.m`)
Convolve with a Gaussian smoothing window

```
filterlen = 50;                        % filter length
thefilter = gausswin(filterlen,sizeparam); % size=4
thefilter = thefilter/sum(thefilter); % unit norm
tmp2=conv(thefilter,thehist); % makes longer output
% select corresponding portion
offset = floor((filterlen+1)/2);
tmp1=tmp2(offset:len+offset-1);
```

Slide credit: Bob Fisher

Convolved Histogram Example



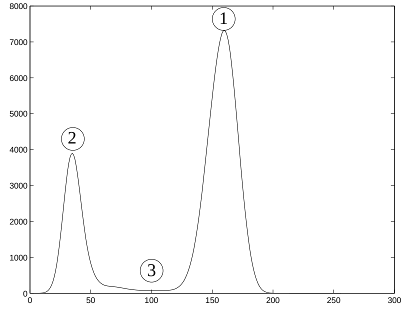
FILTER SHAPE SMOOTHED HISTOGRAM

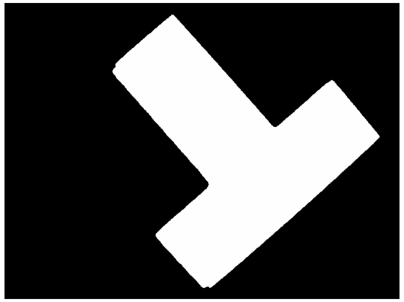
Slide credit: Bob Fisher

Threshold Selection 2

Assume 2 big peaks, brighter ? is higher:

1. Find biggest peak (background)
2. Find next biggest peak in darker direction
3. Find lowest point in trough between peaks





Slide credit: Bob Fisher

Peak Pick Code

Omit special cases for ends of array and closing ‘end’s.

```
peak = find(tmp1 == max(tmp1));   % find largest peak

% find highest peak to left
xmaxl = -1;
for i = 2 : peak-1
  if tmp1(i-1) < tmp1(i) & tmp1(i) >= tmp1(i+1) ...
  & tmp1(i)>xmaxl
      xmaxl = tmp1(i);
      pkl = i;

% find deepest valley between peaks
```

Slide credit: Bob Fisher

```
xminl = max(tmp1)+1;
for i = pkl+1 : peak-1
  if tmp1(i-1) > tmp1(i) & tmp1(i) <= tmp1(i+1) ...
  & tmp1(i)<xminl
      xminl = tmp1(i);
      thresh = i;
```

Slide credit: Bob Fisher

Lecture Overview

1. Thresholding to differentiate object from a constant and simple background (not just white backgrounds: see also *bluescreening* or *chroma keying*)
2. 1D ?
3. Histogram smoothing & threshold selection

Slide credit: Bob Fisher