

1 Baseline IPA Techniques

- Image Representation
- Point-by-Point Operators
- Thresholding
- Local Operators
- Correlation, Convolution, Deconvolution
- Adaptive Threshold
- Non-Linear Local Operators

2 Baseline Vision Techniques

2.1 Image in -> Image out

- Image Acquisition
 - 2D array of integers representing pixel brightness
- Image representation
- Image processing
- Image analysis
- Image interpretation

3 Image Processing

- Image to Image operation
- Aim to produce images that will make the analysis stage simpler and more robust
- Two main classes of operations applied to images during processing
 - Point to point operations
 - Neighbourhood operations

4 Image Analysis

4.1 Image in -> Features out

- Automatic extraction of useful information from an image
- Information extracted must be:
 - Explicit
 - Useful in the subsequent decision making process

- Common image analysis techniques:
 - Template matching
 - Pattern recognition using feature extraction
 - Descriptive syntactic processes

5 Pattern Recognition Using Feature Extraction

- Images (objects described in terms of their representative features)
- Basic Steps:
 - Segmentation (divide the image into its constituent parts)
 - Feature extraction
 - Classification

6 Monochrome Image Representation

- Binary Images
- Colour images (R, G, B)
- Multispectral (Visible, IR, UV)
- Stereoscopy)
- Image Sequences

Ability to analyse image depends on how the original scene is encoded

7 Image Representation

- Example intensity assignment scheme:

$$f(i, j) = 0 \quad 0 < f(i, j) \leq 0.33W \quad 0.33W < f(i, j) \leq 0.67W \quad 0.67 < f(i, j) < W \quad f(i, j) = W.$$

- Binary image: Only two intensity levels: Black (0), White (1)

8 Elementary Image Processing Notation

- $N(i, j)$ forms a 3x3 set of pixels (neighbourhood) around pixel (i, j)
 - 8-neighbours of (i, j) (8-connected)

$$(i-1, j-1), ((i-1, j), i-1, j+1), (i, j-1), (i, j+1), (i+1, j-1), (i+1, j), (i+1, j+1).$$

- 4-neighbours of (i, j) (4-connected)

$$(i-1, j), (i, j-1), (i, j+1), (i+1, j).$$

9 Monadic, Point-by-Point Operators

- Intensity normalisation (indicated by and asterisk (*))
 - Maintain $c(i,j)$ within the same range as the input $(0,W)$
 - Permits iterative processing
- Negate $c(i,j) \Leftarrow W - a(i,j)$
- Squaring* $c(i,j) \Leftarrow [a(i,j)]^2/2$
- Intensity Shift*
- Intensity Multiply*
- Highlight
- flow(num) traps overflow and underflow situations on an integer variable num
 - if (num>WHITE) return (WHITE)
 - if (num<BLACK) return (BLACK)

10 Threshold

- Generally try to segment image regions by identifying common properties
- Simplest property is intensity \rightarrow thresholding
- Binary images from grey level images
- Problems:
 - Only consider intensity - not include relationship between pixels
 - Cannot be sure that the thresholded regions are in contact
 - Can include or miss unwanted/necessary regions
 - Very hard (or impossible) to find a satisfactory threshold value (easy to under/over threshold - noise increases this effect)
 - Illumination may be uneven across the image
- Mid-Level Threshold:
 - Threshold midway between the minimum and maximum grey level (i.e. at the grey scale value MIDGREY)
- Global: Dependent on the grey level of a given point
- Adaptive/Local: Dependent on the grey levels of the neighbouring points also
- Dynamic: Local + Dependent on the point's co-ordinates. Can be used to deal with uneven illumination.

Examine the histogram (probability distribution) to see if can get two or more distinct modes to allow separation between foreground and background.

$$\text{Probability of grey level}(g) : [p(g) = \frac{N_g}{N}].$$

$N = \text{total number pixels}, N_g = \text{number of pixels with a grey level } g.$

11 Automated Thresholds - Peaks and Valleys

Simple approach - find local minima (valley) between local maxima (peaks) in the histogram.

- Problems:
 - Noise - Multiple local maxima/minima - can be overcome by smoothing the histogram

12 Automated Thresholds - Clustering

- View as a clustering problem - problem is that the range of values may overlap
- Want to minimise the error in classification (e.g. misclassifying a foreground pixel as a background one)
- Otsu method: Simple idea, find the threshold that:
 - minimizes the weighted intra (within) class variance
 - = maximizing the inter (between) class variance
- Based solely on intensity information - No spatial context
- Set the threshold so each cluster is as tight as possible with a view to minimizing any overlap
 - Cannot change the distributions so we adjust where we separate them (threshold)
 - As we adjust the threshold one way we increase the spread of one and decrease the spread of the other - Goal: to select threshold that minimizes the combined spread
 - May fail when object and background pixels are extremely unbalanced
- For foreground object (f) and background (b)
- weights (W) and mean (μ)
- variance (σ^2 - average squared deviation of each value from its mean)
 - Within class variance
 - $\sigma_W^2 = W_b\sigma_b^2 + W_f\sigma_f^2$
 - Between class variance
 - $\sigma_B^2 = \sigma^2 - \sigma_W^2$
 - $= W_bW_f(\mu_b - \mu_f)^2$