COMP3204/COMP6223: Computer Vision

Types of image feature and segmentation

Jonathon Hare
jsh2@ecs.soton.ac.uk

Image Feature Morphology

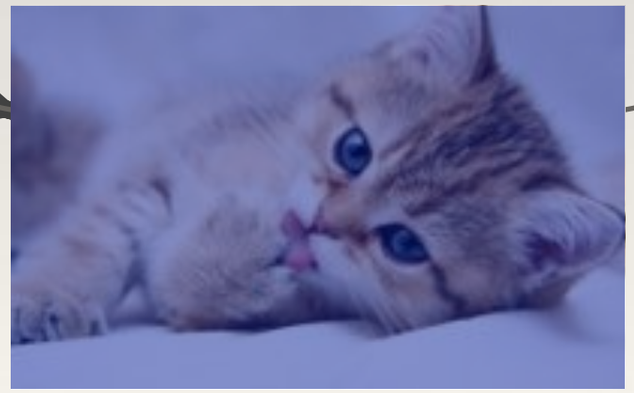
Recap: Feature Extractors

image goes in

Feature Extractor

featurevector(s) come out

Some processing and analysis occurs



Recap: Feature Extractors

image goes in

Feature Extractor

featurevector(s) come out

different morphologies ('shapes') of feature extractor output varying numbers of features

Global Features

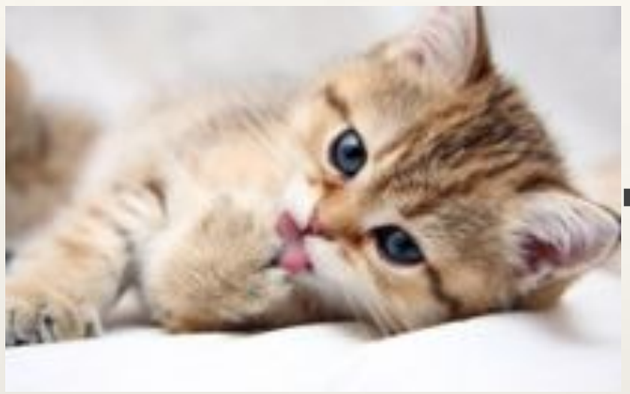
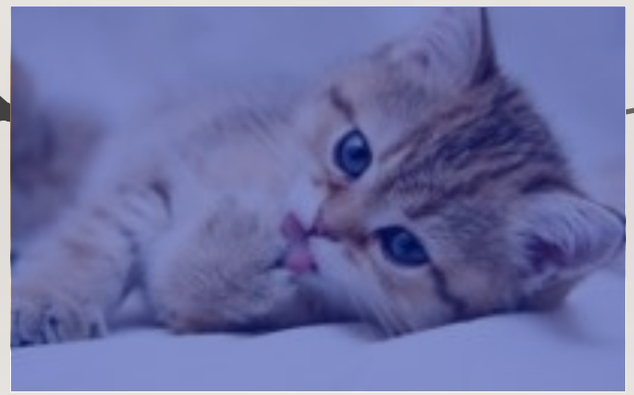


image goes in

Feature Extractor

a **single** featurevector comes out

The entire image is analysed as one



A **Global Feature** is extracted from the contents of an entire image.

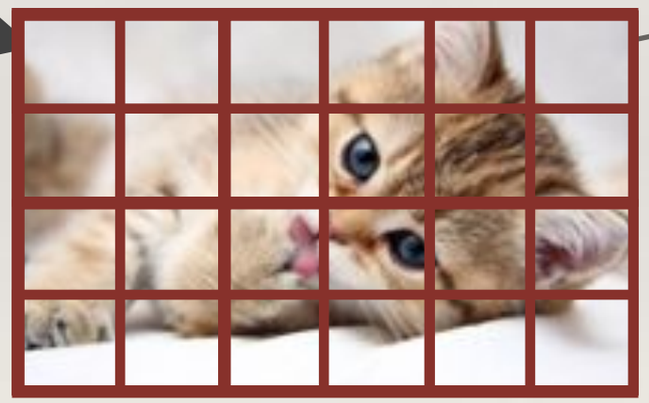
Grid or Block-based Features

image goes in

Feature Extractor

multiple featurevectors come out

The entire image split into blocks and a feature is extracted from each block



Multiple features are extracted; one per block

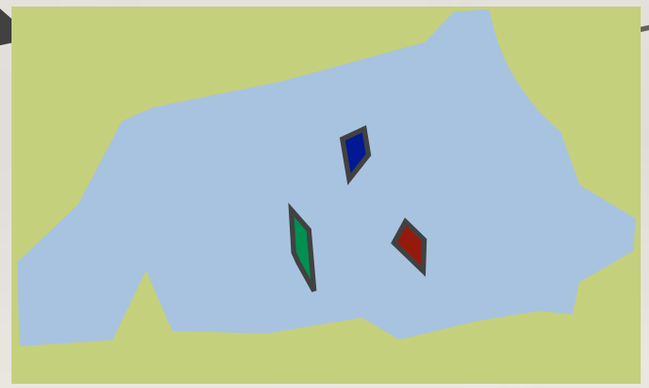
Region-based Features

image goes in

Feature Extractor

multiple featurevectors come out

The entire image split into a number of regions and a feature is extracted for each



Multiple features are extracted; one per region

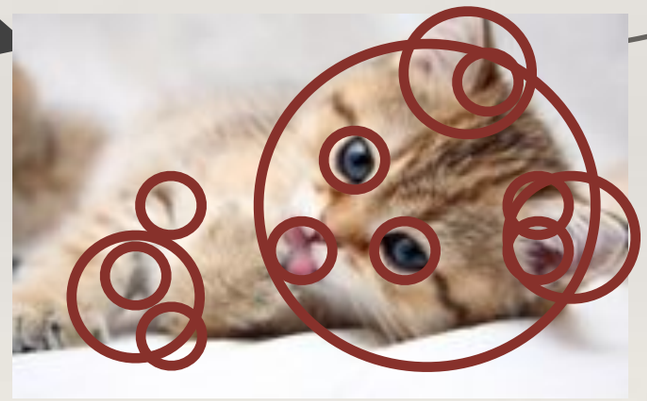
Local Features

image goes in

Feature Extractor

multiple featurevectors come out

Interest points are detected in the image and a feature is extracted from the surrounding pixels



Multiple features are extracted; one per local interest point

Global Features

Image Histograms

- In the second lecture we saw a simple global feature computed from the average of the colour bands.
  - This wasn't particularly robust, and couldn't deal well with multiple colours in the image.
- A more common approach to computing a global image description is to compute a histogram of the pixel values.

Joint-colour histograms

- A joint colour histogram measures the number of times each colour appears in an image.
  - Different to histograms in image editing programs with compute separate histograms for each channel.
- The colourspace is *quantised* into bins, and we accumulate the number of pixels in each bin.
  - Technically, it's a multidimensional histogram, but we flatten it (unwrap) to make it a feature vector

Demo: Global colour histograms

Joint-colour histograms

- Normalisation (i.e. by the number of pixels) allows the histogram to be *invariant* to image size.
- Choice of colour-space can make it invariant to uniform lighting changes (e.g. H-S histogram)
- Invariant to rotation
- But vastly different images can have the same histogram!

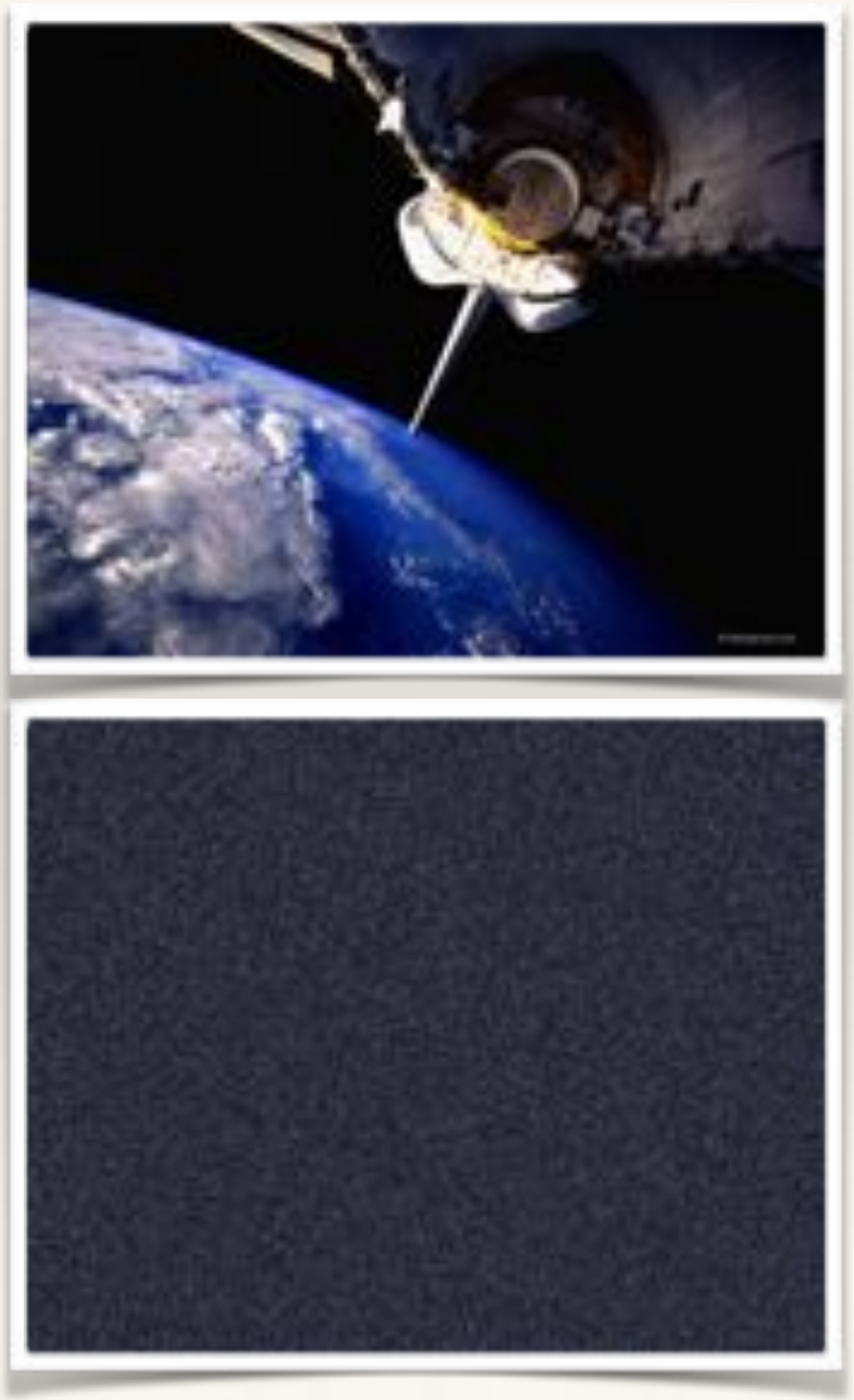
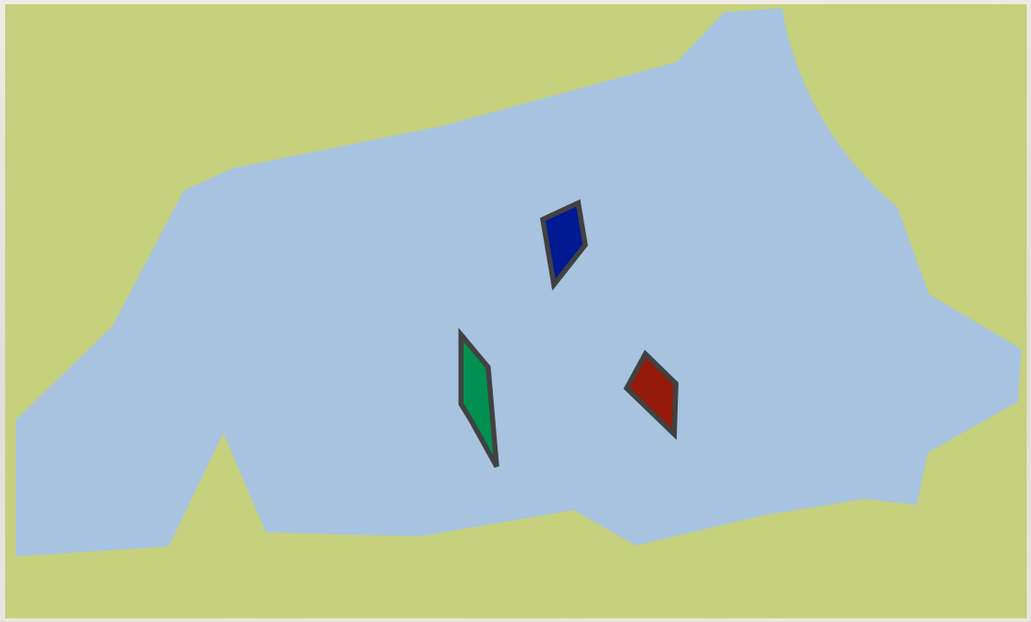


Image Segmentation

What is segmentation?

- The first part in the process of creating region-based descriptions…
- The process of partitioning the image into ***sets*** of pixels often called ***segments***.
- Pixels within a segment typically share certain visual characteristics.

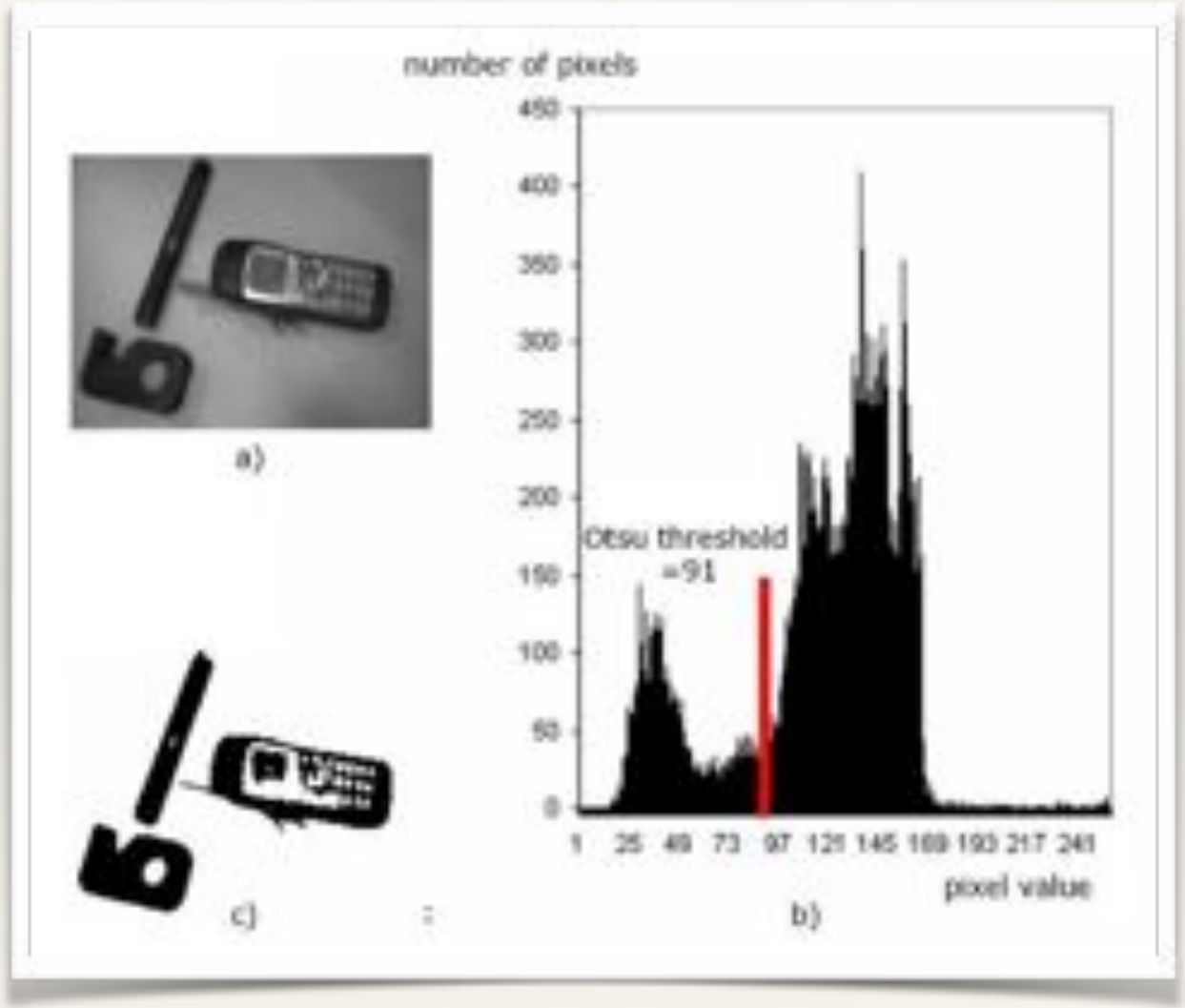


Global Binary Thresholding

- Thresholding is the simplest from of segmentation
  - Turns grey level images into binary images (i.e. two segments) by assigning all pixels with a value less than pre-determined threshold to one segment, and all other pixels to the other.
- Really fast
- Requires a manually set static threshold
  - Not robust to lighting changes
  - Can work well in applications with lots of physical constraints (lighting control and/or high-contrast objects)

Otsu's thresholding method

- Otsu's method (named after Nobuyuki Otsu) provides a way to automatically find the threshold.
- Assume there are two classes (i.e. foreground & background)
  - The histogram must have two peaks
- Exhaustively search for the threshold that maximises interclass variance



Demo: Global Thresholding and Otsu

Demo: When Global Thresholding Breaks

Adaptive/local thresholding

- Local (or Adaptive) thresholding operators compute a different threshold value for every pixel in an image based on the surrounding pixels.
- Usually a square or rectangular window around the current pixel is used to define the neighbours

Mean adaptive thresholding

Set the current pixel to 0 if its value is less than the ***mean*** *of its neighbours plus a* ***constant*** *value; otherwise set to 1.*

- *Parameters:*
  - *size of window*
  - *constant offset value*

Demo: Adaptive Thresholding

Adaptive/local thresholding

- Good invariance to uneven lighting/contrast
- But…
  - Computationally expensive (at least compared to global methods)
  - Can be difficult to choose the window size
    - If the object being imaged can appear at different distances to the camera then it could break…

Demo: Adaptive Thresholding Again

Segmentation with K-Means

- K-Means clustering also provides a simple method for performing segmentation:
  - Cluster the colour vectors (i.e. [r, g, b]) of all the pixels, and then assign each pixel to a segment based on the closest cluster centroid.
- Works best if the colour-space and distance function are compatible
  - e.g. Lab colour-space is designed so that Euclidean distances are proportional to perceptual colour differences

Demo: k-means colour clustering

- Naïve approach to segmentation using k-means doesn't attempt to preserve continuity of segments
  - Might end up with single pixels assigned to a segment, far away from other pixels in that segment.
- Can also encode spatial position in the vectors being clustered: [r, g, b, x, y]
  - Normalise x and y by the width and height of the image to take away effect of different images sizes
  - Scale x and y so they have more or less effect than the colour components.

Demo: k-means spatial colour clustering

Advanced segmentation techniques

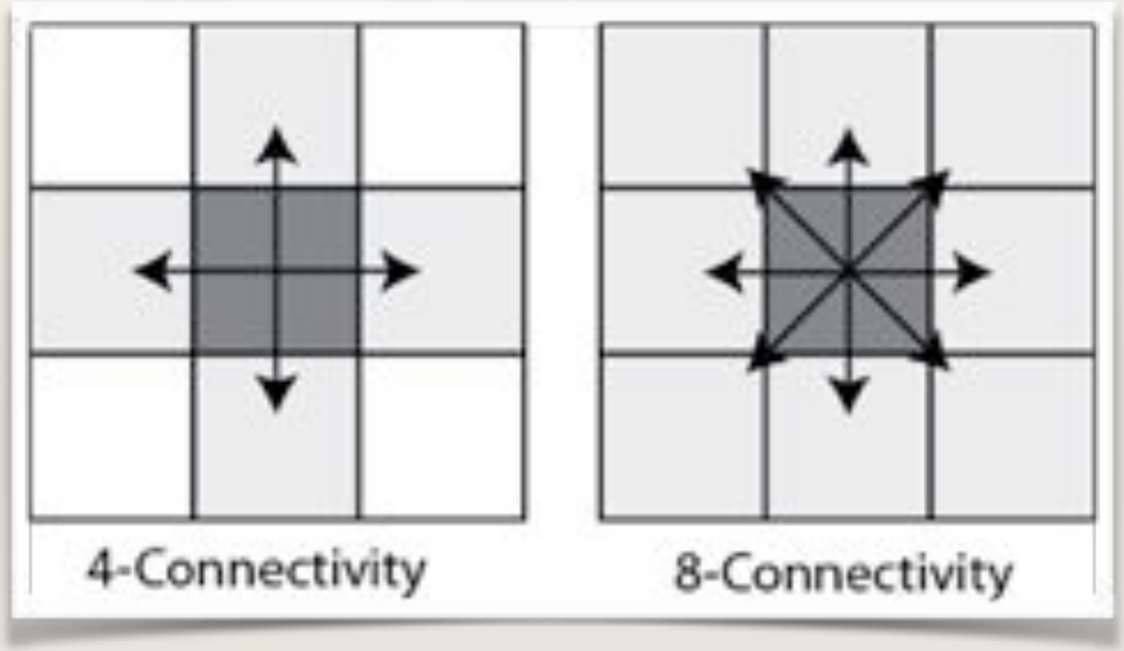
- Lots of ongoing research into better segmentation techniques:
  - Techniques that can automatically determine the number of segments
  - “semantic segmentation” techniques that try to create segments that fit the objects in the scene based on training examples

Demo: Felzenswalb-Huttenlocher segmentation

Connected Components

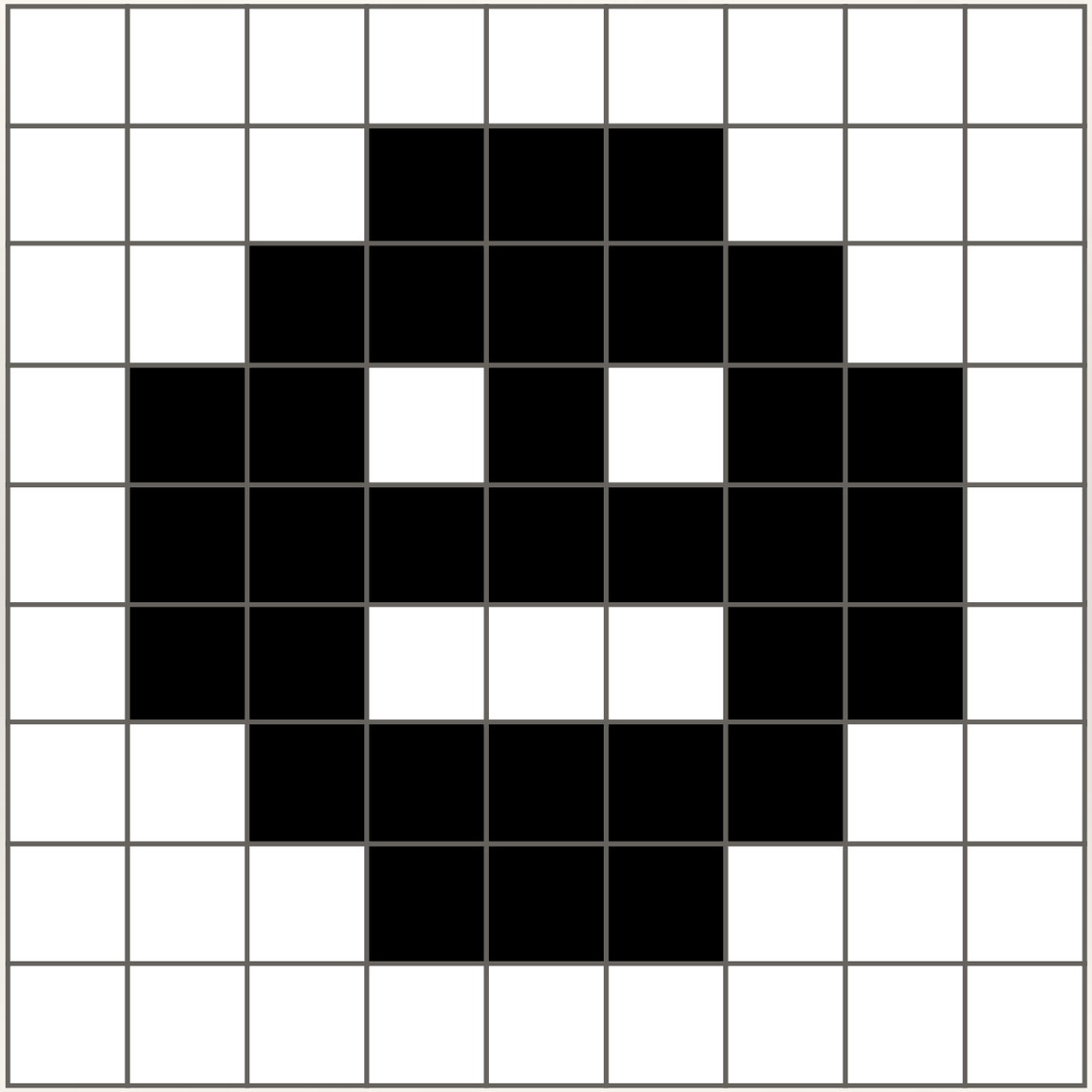
Pixel Connectivity

- A pixel is said to be connected with another if they are spatially adjacent to each other.
- Two standard ways of defining this adjacency:
  - 4-connectivity
  - 8-connectivity



Connected Component

A connected component is a set of pixels in which every pixel is connected either directly or through any connected path of pixels from the set.



Connected Component Labelling

- Connected Component Labelling is the process of finding all the connected components within a binary (segmented) image.
  - Each connected segment is identified as a connected component.
- Lots of different algorithms to perform connected component labelling
  - Different performance tradeoffs (memory versus time)

The two-pass algorithm

1. On the first pass:
   a. Iterate through each element of the data by column, then by row (Raster Scanning)
      i. If the element is not the background
         1. Get the neighbouring elements of the current element
         2. If there are no neighbours, uniquely label the current element and continue
         3. Otherwise, find the neighbour with the smallest label and assign it to the current element
         4. Store the equivalence between neighbouring labels
2. On the second pass:
   a. Iterate through each element of the data by column, then by row
      i. If the element is not the background
         1. Relabel the element with the lowest equivalent label

Demo: Connected Component Labelling

Summary

- Image features can be categorised in one of four main categories: **global**, **grid-based**, **region-based** and **local**.
- A common type of global feature is a global colour histogram.
- Region-based methods need regions to be detected - this process is called segmentation
  - Many different approaches…
- Connected components are segments in which the pixels are all reachable by a connected path.