Vision for Robotics I

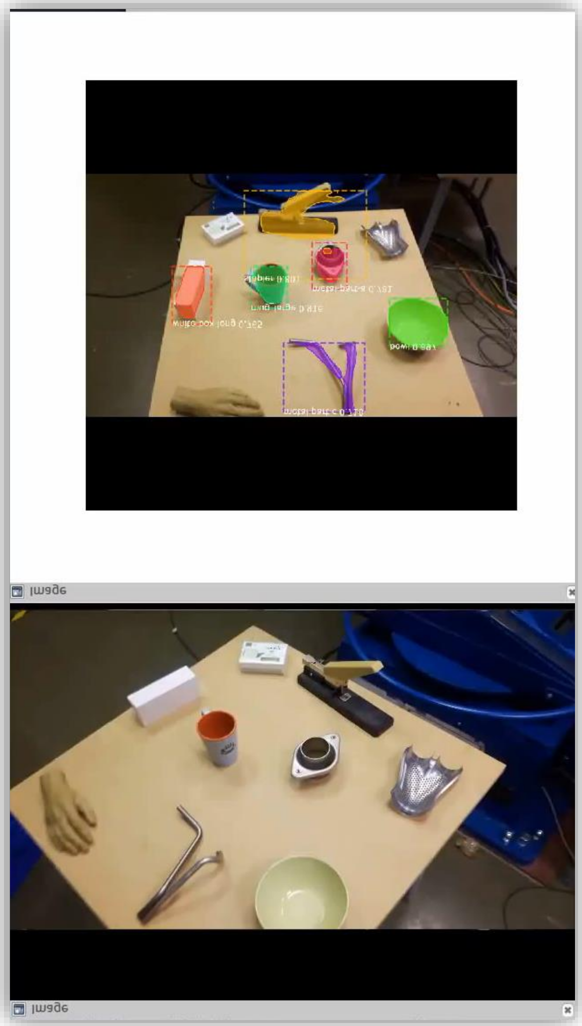
Dr Gerardo Aragon-Camarasa
gerardo.aragoncamarasa@glasgow.ac.uk

Dr J. Paul Siebert
paul.siebert@glasgow.ac.uk

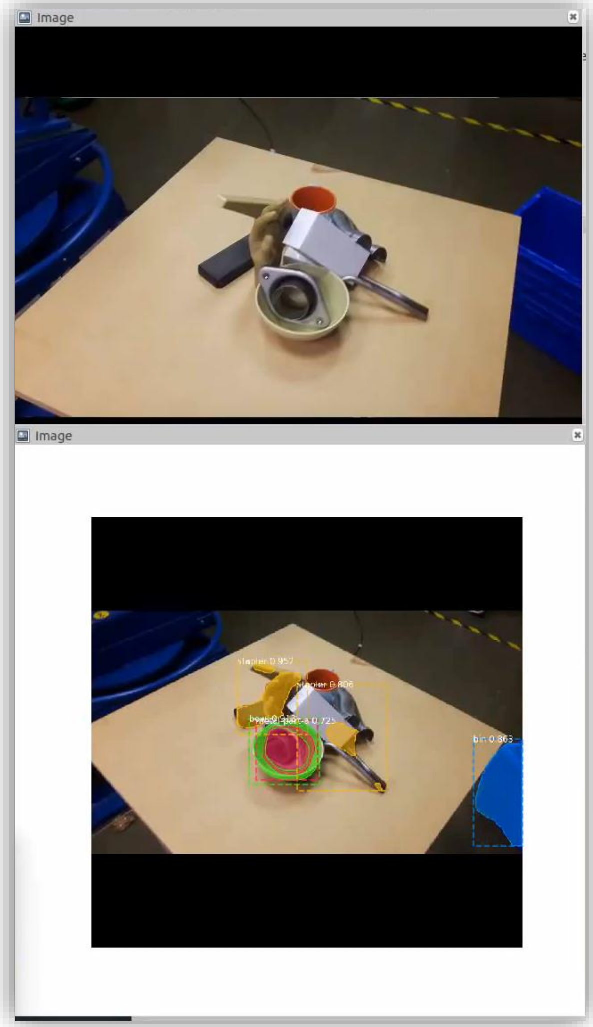
RF – University of Glasgow

Introduction

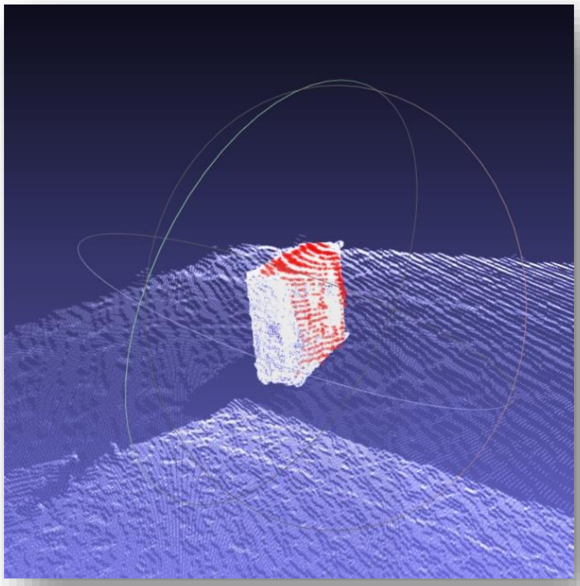
Use case 1: Objects in isolation



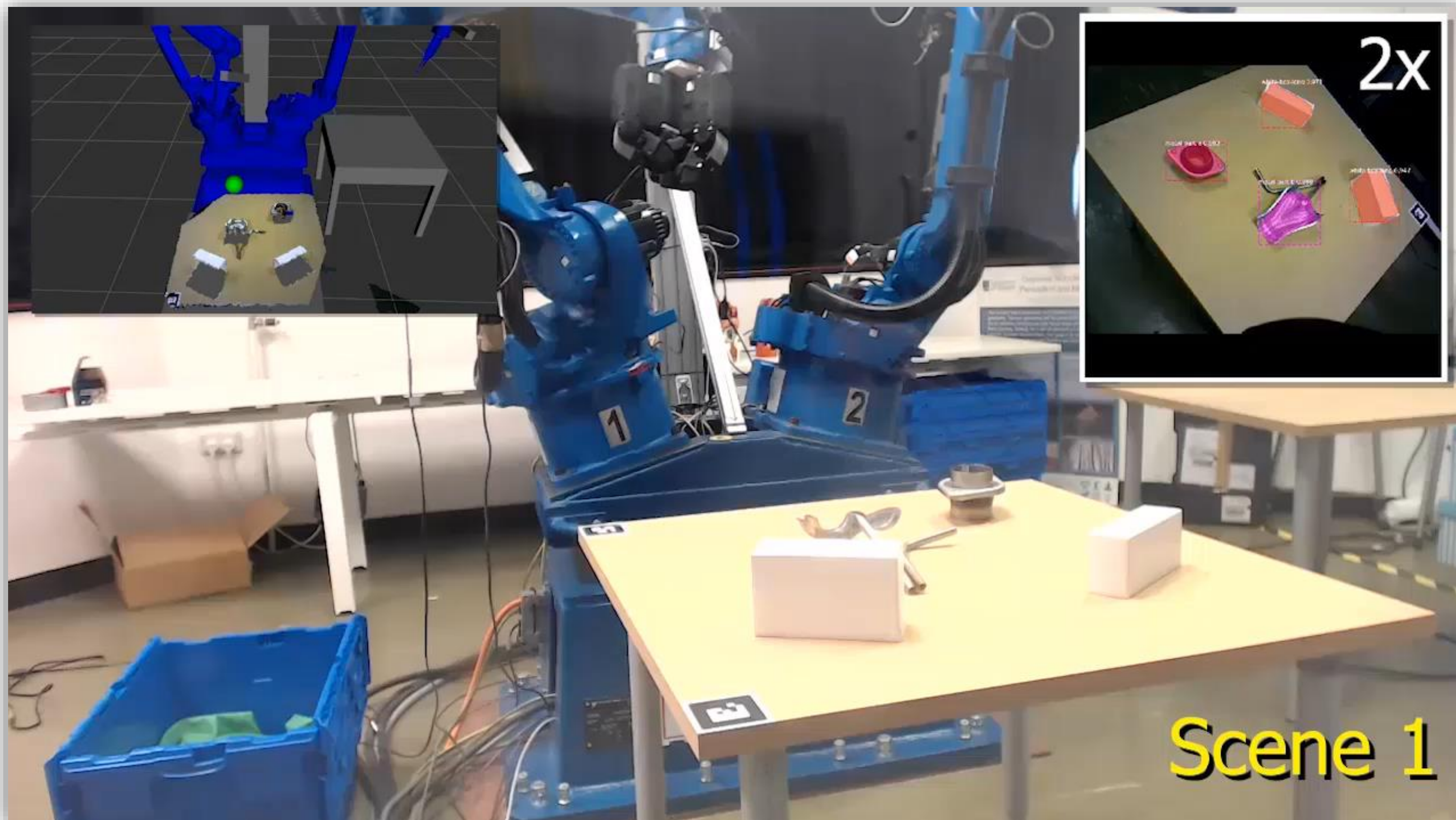
Use case 2: Pile of objects



Pose Estimation



Introduction



Introduction

- **Vision** is the **most essential** of senses – it can:
  - Provide direct understanding of the environment *without requiring direct contact with the robot*
  - Serve tasks such as object recognition, locomotion, navigation, grasping & manipulation
  - Give immediate and potentially continuous feedback to the success of an operation
  - Operate at scales of microns to light-years
- Vision is less appropriate when illumination is poor or light propagation is difficult:
  - Underwater when the conditions are turbid or dark
  - Locations where there is little light – underground, in soil – not good for guiding tunnelling robots!

Introduction

- **Visual perception** can be categorised into *what* and *where*, e.g. consider the previous video and classify vision enabled tasks:
  - **Where (visually guided behaviour)**: visual sensing of **depth** and **motion** for navigation, grasping & manipulation failure detection
  - **What (recognition & identification):** Visual perception for object recognition & scene understanding
  - https://en.wikipedia.org/wiki/Two-streams_hypothesis
- We usually further classify vision into **2D & 3D modes**:
  - 2D vision operates on conventional images, could be monochrome, colour or multi-spectral.
  - A 2.5D depth (or range) map comprises an "image" whose pixels contain distances to imaged surfaces
  - 3D vision usually operates on 3D point-cloud data which have almost always been generated from 2.5D range maps

Introduction

- In this lecture we shall survey basic machine vision algorithms for:
  - Histogramming
  - Segmentation

- In subsequent lectures we shall examine:
  - Colour perception/colour spaces
  - Shape Description & Shape hierarchy
  - SLAM, Spatial Localisation and Mapping
  - Sensing and perception for imaging, haptics, tactile, olfactory and acceleration.
  - Advanced robot vision methods, including Deep Learning

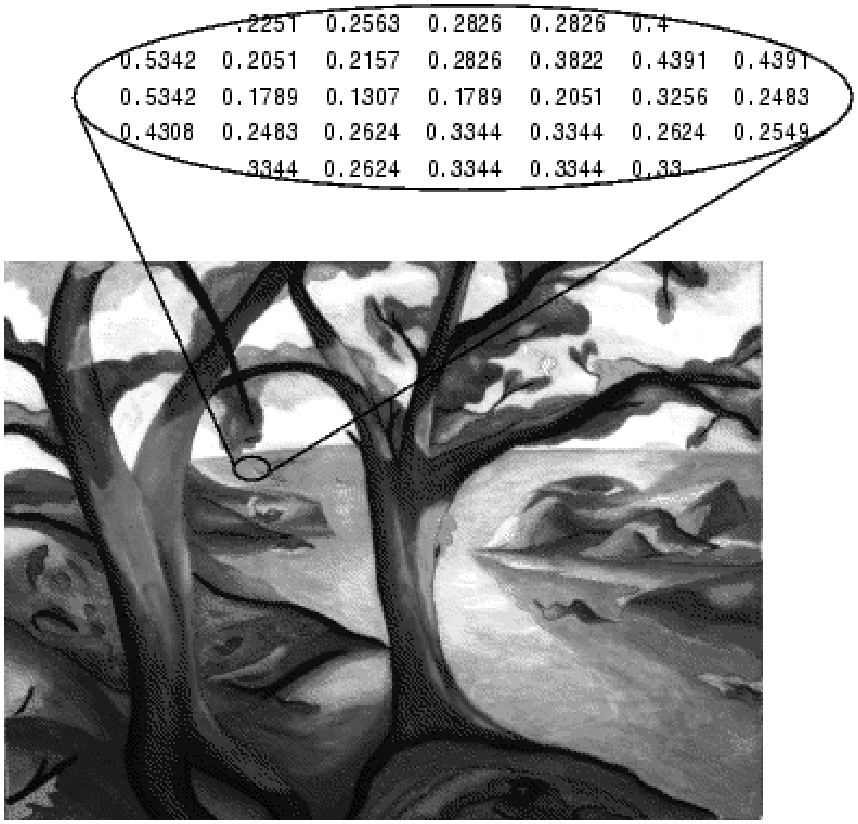
- **But first, a few basics about digital cameras and digital images…**

Digital Image Acquisition

- Digital Camera
- Lens & Projection
- Colour

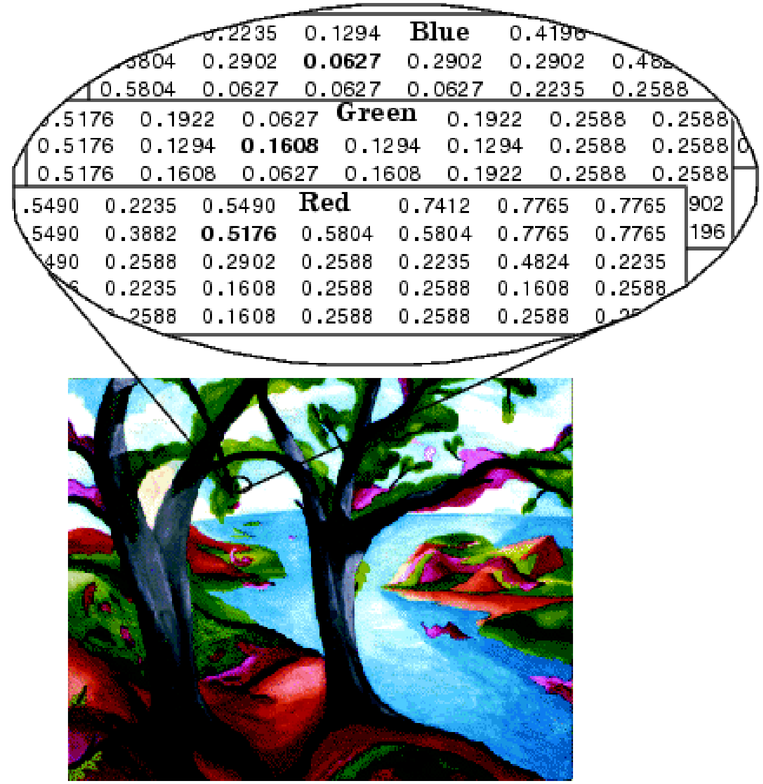
Digital Images

- A digital image is a matrix of numbers called pixels, **picture elements**, which represent the *intensity values* within small tiles composing the image.
  - **Each pixel is a number that represents intensity, I**
  - The greater I, the brighter the image pixel.
  - This number is **quantised**, often (but not always) to 8 bits [0..255], esp. for display, e.g., could be *floats* [0..1.0] as in the example image opposite.
- Digital cameras quantised 8, 10 or even 12 bits/pixel.
- Scanned material could be 14 or even 16 bits for cine.
- Must be positive range for display.
- Pixels are stored and organised in an array
  - the array organisation mirrors the image tiling
- Usually ordered 1..N, 1..M from top left hand:
  - where N,M are integers.
  - Modern GPU indexing allows *floating* point indexing!



Colour Digital Images

- The greyscale image model extends directly to colour images.
- Human day-vision (called photopic vision) is based on the cone photoreceptors in the retina:
    - there are 3 types of cones tuned to red, green and blue light, respectively.
    - each type contains red, green and blue pigment (rhodopsin) respectively.
- Colour cameras filter light into red, green & blue components.



I = ImArray(C,N,M), C [1..3] -> R,G,B

- These image components are digitised into three colour planes representing Red, Green and Blue
    - i.e. each pixel now comprises 3 numbers representing red, green & blue.
- Each plane drives the respective display colour component to generate the illusion of full colour.

Digital Depth Images

- Digital **depth images,** or *range maps***,** comprise a matrix where the value of each "pixel" is a measurement of the distance to an observed surface.
- A raw depth map encodes *dark -> near*, *light -> far*. These maps can be *rendered* to make them easier to interpret (below).
- Depth images can be computed using a variety of different methods (more in coming lecture), including by triangulation using *stereo-pair* images or devices such as the *Kinect* camera.
- Because a *single* depth image cannot contain *under cuts*, this representation is often referred to as being 2.5D

A raw depth map representing a human face

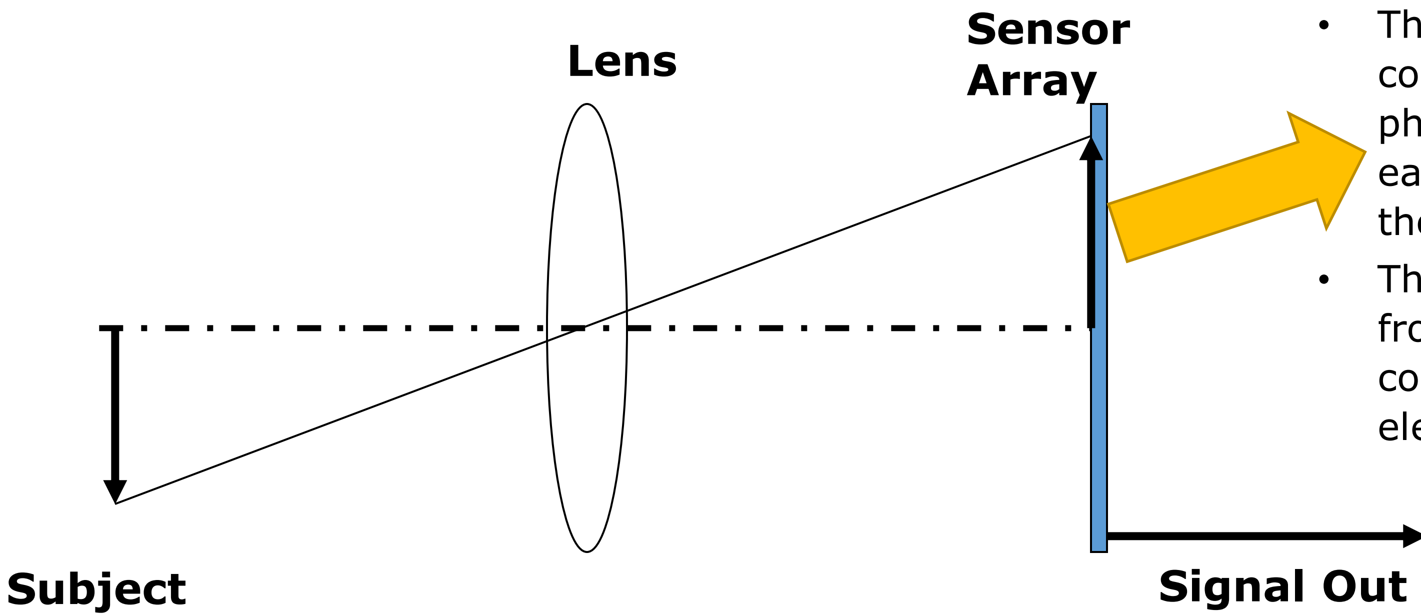
Stereo-pair cameras

View from left camera

View from right camera

Rendered depth image

Digital Cameras



- The sensor matrix comprises a matrix of photosensitive cells, each of which samples the image.
- The image sample from each cell is converted into an electrical signal.

- An image of the subject is *projected* onto the sensor array in the camera.
- The sensor array converts the image into a matrix of charges, proportional to the incoming photon flux intensity. Hence the image is sampled as *pixels,* “picture elements”.
- This charge matrix is read out sequentially in the form of an electrical signal.
- The above signal is digitised and interfaced usually via USB to the computer host

Digital Cameras

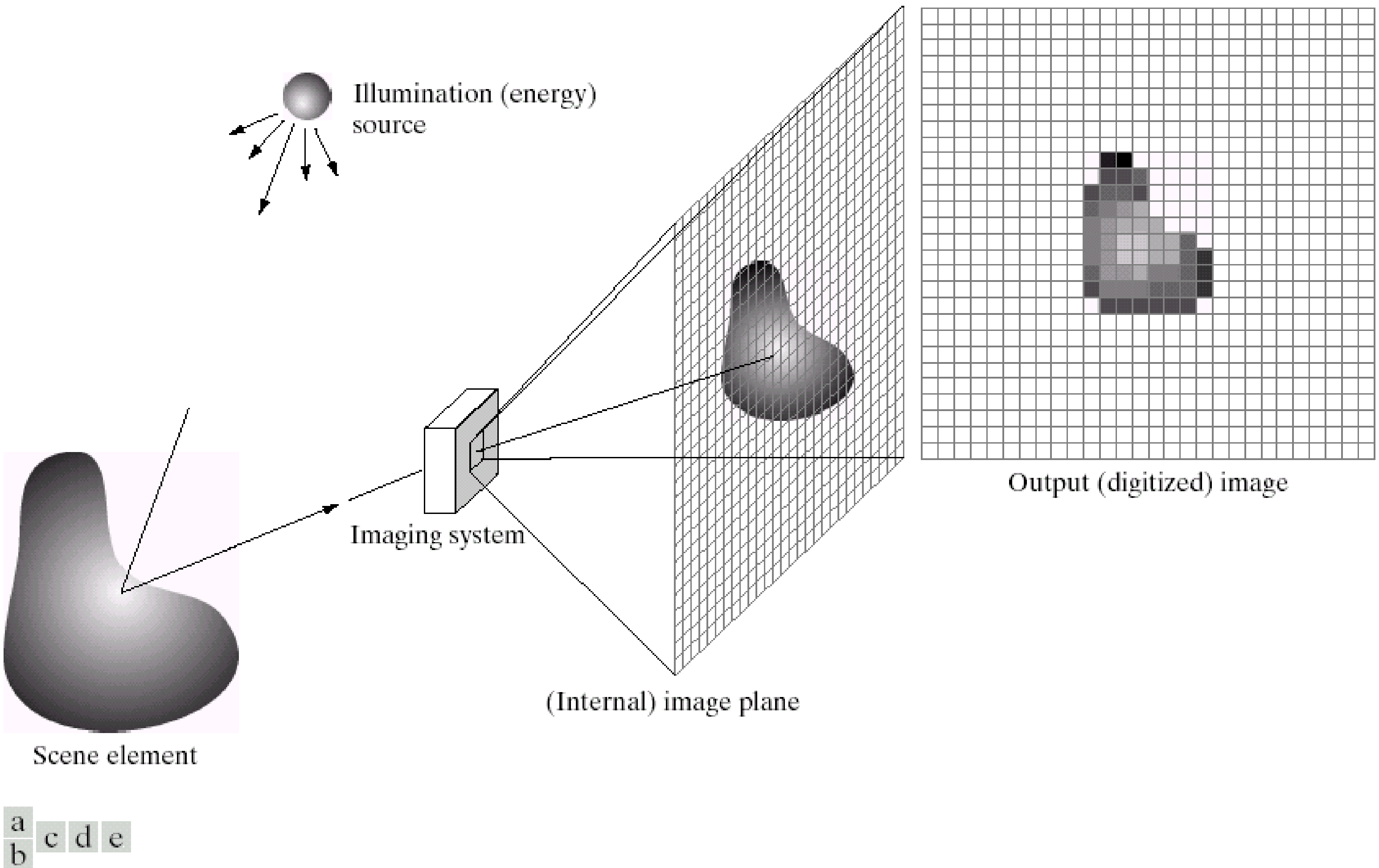
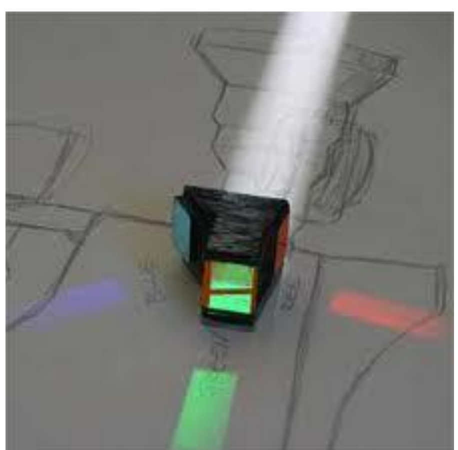


FIGURE 2.15 An example of the digital image acquisition process. (a) Energy ("illumination") source. (b) An element of a scene. (c) Imaging system. (d) Projection of the scene onto the image plane. (e) Digitized image.

Colour Cameras

- Colour cameras either comprise 3 sensor array devices or single array mosaics.
- 3 sensor devices use a complex optical arrangement to split the incoming light into 3 *channels,* i.e. 3 separate but identical images, each sampled by a separate imaging array.
- Each channel split into red, green and blue components using red, green and blue filters.
- Mosaic devices place micro filters over each pixel in a regular array of red, green blue and sometimes with additional colours, e.g. IR or teal (blue/green).
- Th colour image is represented as *three* image matrices comprising Red, Green & Blue image planes, constructed by *interpolation,* so that final colour image has the same the number of *RGB pixel triples* as the image sensor plane has pixels.

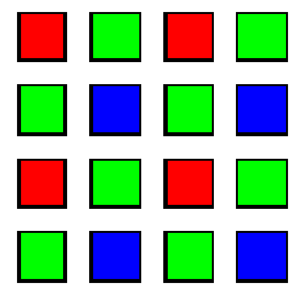
Prism separates out 3 optical channels, each covered by a different colour of filter to produce R,G,B images sampled by 3 separate sensor arrays.



RGRGRGRG

GBGBGBGB

RGRGRGRG



Typical RGB Mosaic

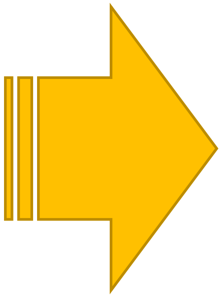
Image Histogram Analysis

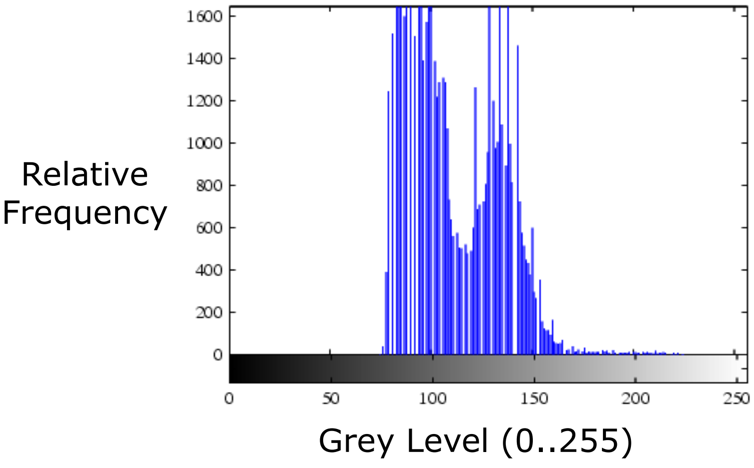
- Histogramming
- Segmentation
- Colour perception/colour spaces
- Edge Detection & Convolution

The Grey Level Histogram (Image Statistics)

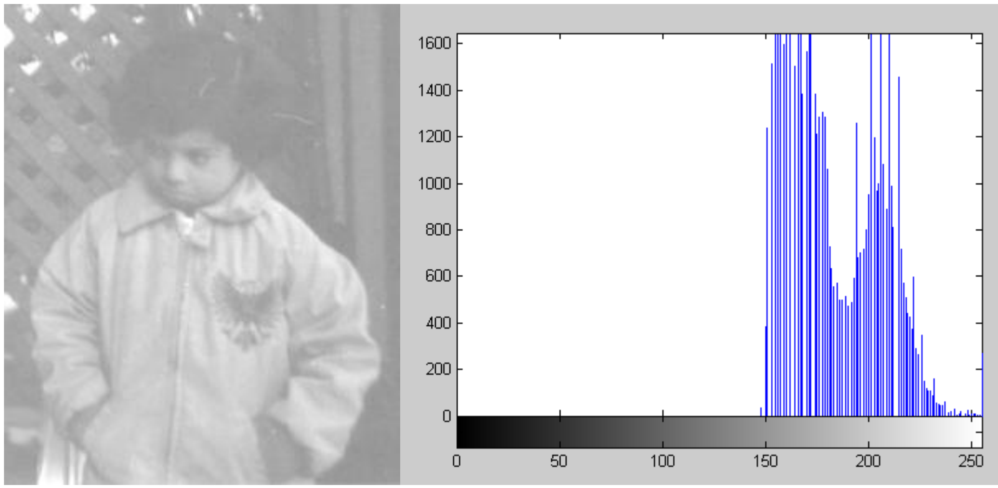
- Consider an image as a distribution of pixel values, i.e. treat the image as a "**bag of numbers**"
  - This implies disregarding the spatial structure of the image for the moment
- Count the number of pixels of each specific intensity present in the image
  - i.e., how many pixels = 0,1,2….,max intensity
- Plot the *(relative) frequency* of each intensity value present as a *histogram*
- This representation provides a useful *summary* of an image
- Consider an image as orthogonal distributions of grey levels in space and in intensity.

Input Image

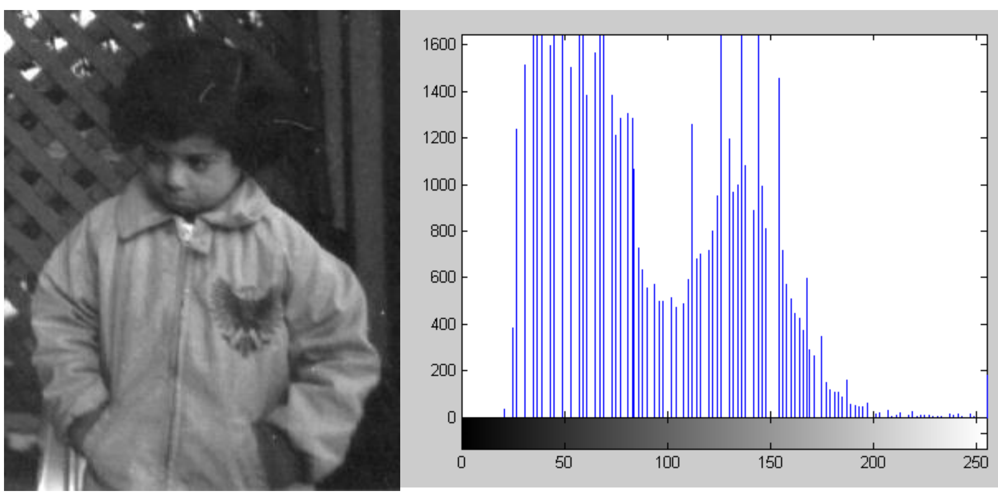




Gain and Black Level changes



black level increase



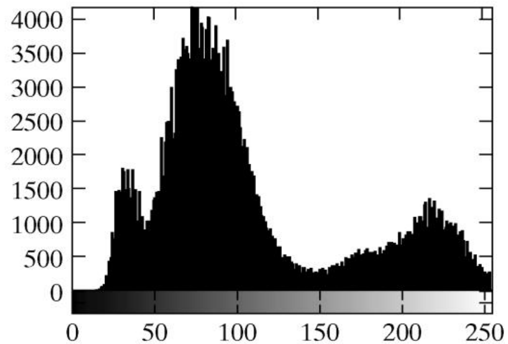
gain increase

- Image Addition/Subtraction by a constant:
  - shifts histogram right/left
- Image Multiplication/Division by a constant
  - expands/contracts histogram
  - This operation will shift as well if left un-normalised

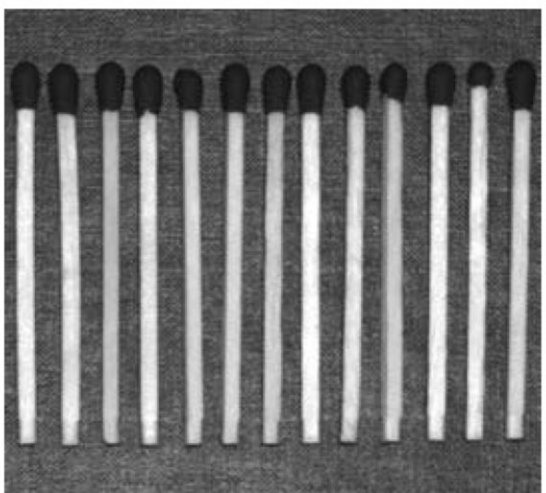
Grey-Level Histogram Analysis

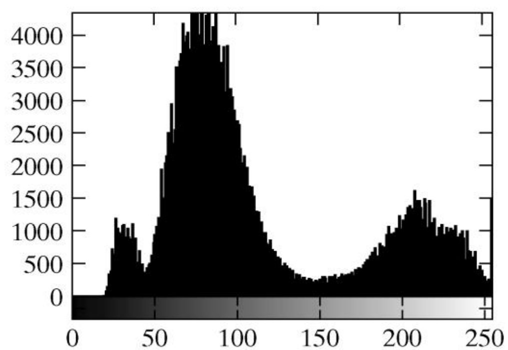
- Image dynamic range check
  - some cameras provide an inbuilt histogram display for checking exposure
- Evidence of homogenous image regions as peaks
  - Basic tools used for deconstructing an image into regions, segmentation
- Extends to colour images with 3 image planes (RGB) which can be expressed by 3 histograms accordingly
  - Can now summarise colour images
- Histogram shape can be used to roughly characterise or describe an image
  - Can compare image histograms directly by treating them as vectors
  - The first Content Based Image Retrieval systems were based on comparing the colour histogram of a *query image* to the histograms stored in an image database
  - The above systems work at a very general level, but are not very specific ***as no image spatial structure remains in the histogram***

Grey-Level Histogram Properties



<u>Note:</u> the histogram only preserves the relative frequency of grey levels





All spatial information is lost regarding the relative location of these grey levels.

Segmentation

- In order to analyse an image in terms of content:
  - Must be able to break the image into isolated components, i.e. segment the image
  - These segments can be processed into *symbols* or *tokens*
  - Subsequent reasoning can then be applied to the symbolic form
- Assume that image components comprise:
  - Regions of uniform intrinsic properties, grey level or colour being the simplest,
  - Could be uniform texture – more difficult to analyse though
  - Where uniform regions meet *edges* are formed
  - So in some sense edges and regions are dual (redundant) representations

Original Image

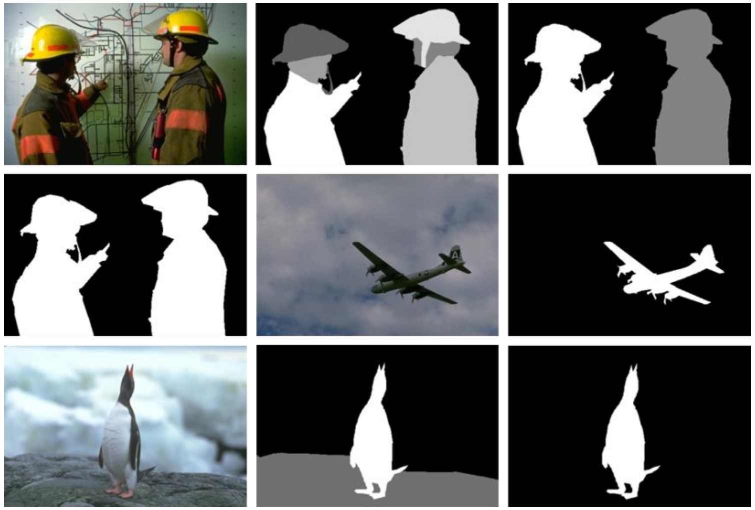
Basic Segmentation

Segmentation

- Image segmentation is not grounded in a general theory
- Appropriate segmentations *can* be achieved by *ad hoc* means (heuristics) for *specific* tasks
- Simple underlying assumptions that clearly do NOT hold generally, but can serve as the basis for useful segmentation algorithms:
  - Uniform image regions correspond to uniform surfaces in the scene
  - Edges correspond to the interface at real object boundaries
  - Edges and contours are *dual* representations of each other
- So uniform regions and their edges may have both *physical* and also *perceptual* significance

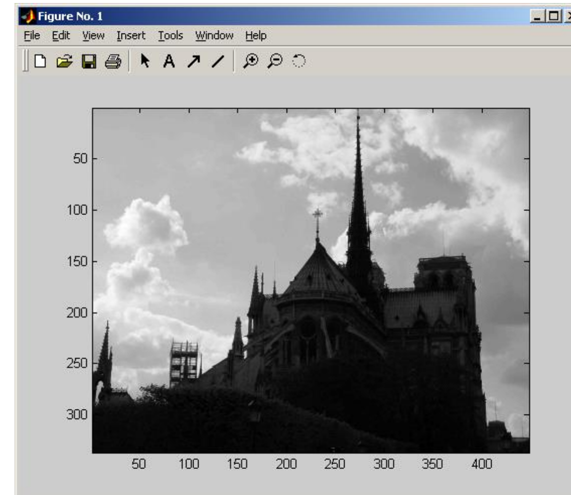
In the figure we can observe different criteria for segmentation:

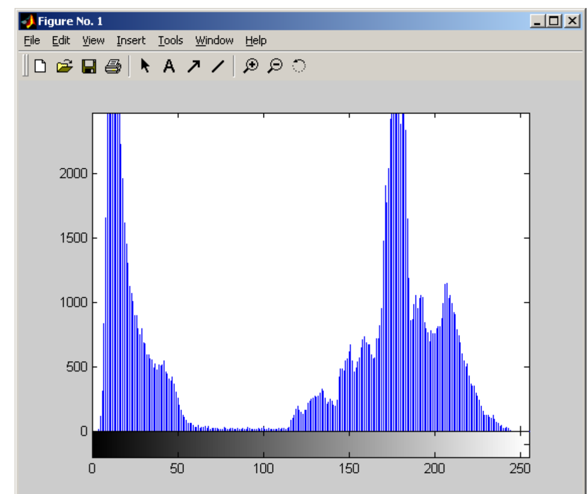
Can you deduce what is being segmented and why?



Histogram-Based Segmentation

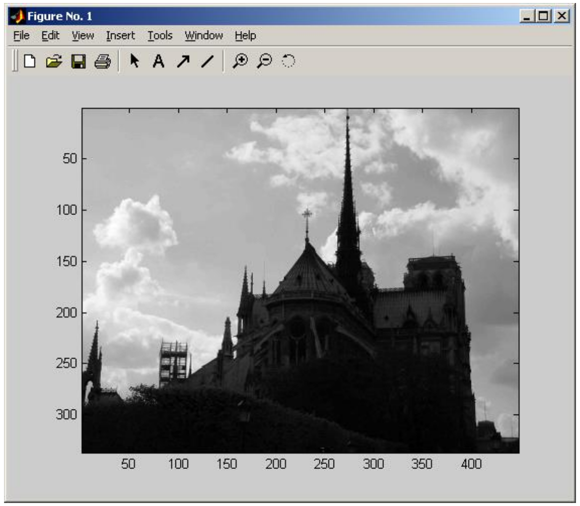
- Basic approach: consider an image region to have uniform grey-level properties
  - Near uniform grey-levels (real-world situation) is reflected in a peak in the image histogram centred at the (local) mode grey-level value
  - There will be a cluster of grey levels varying about this local mode
- Each uniform region will be represented as a separate peak in the histogram if:
  - It is sufficiently large (in area) w.r.t other peaks
  - Of a sufficiently different mode value to avoid merging with other peaks

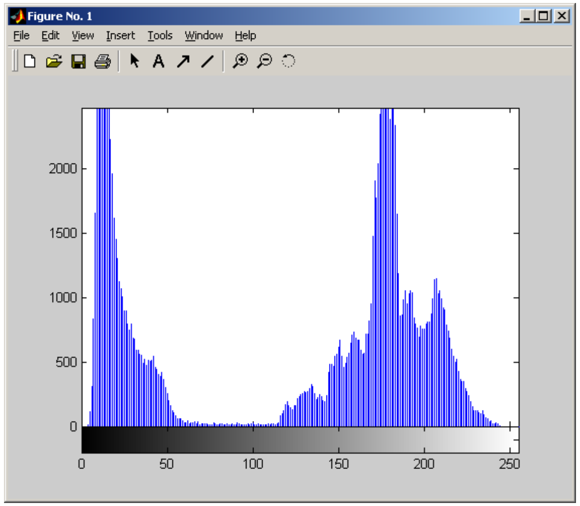




Histogram-Based Segmentation

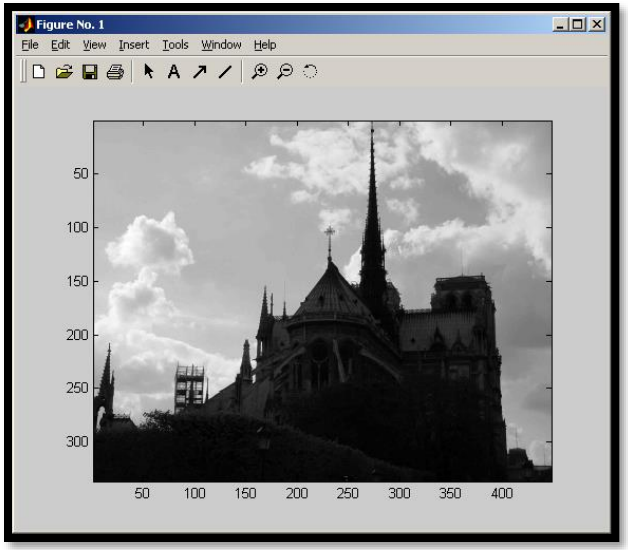
- The histogram of the (top) example is shown (below):
  - Select output pixels O less than intensity threshold,t
  - O(x,y) = I(x,y,) < t; to select dark region,
  - O(x,y) = I(x,y,) > t; to select light region
- The relationship of the image and its histogram:
  - Darker regions appear as peaks on the left of the histogram
  - Lighter regions appear as peaks on the right of the histogram
  - Histogram valleys often correspond to the edge pixels between regions
- *All* explicit spatial information is *lost* in the histogram
- Try to relate regions in the image opposite to peaks in the histogram below

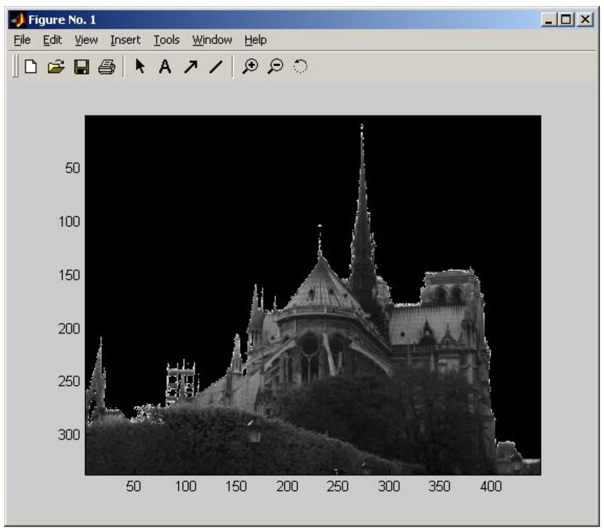


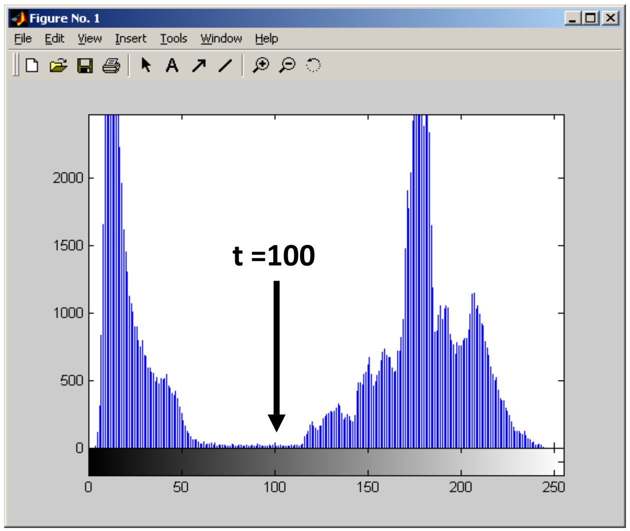


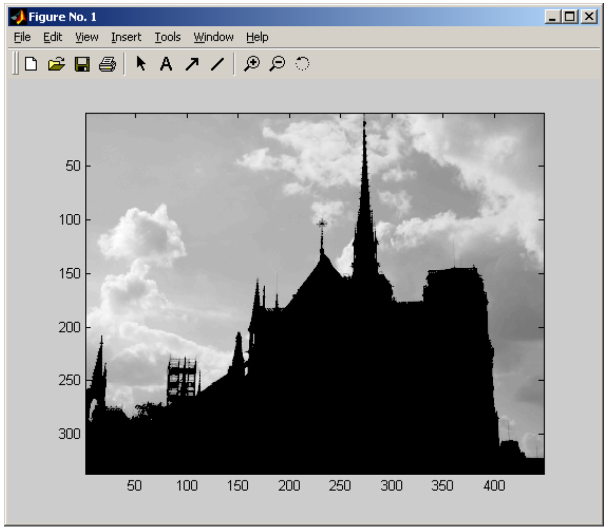
Intensity Thresholding

- Top: Original input image I
- Below left: O(x,y) = I(x,y) < t=100; to select dark region
- Below right: O(x,y) = I(x,y) > t=100; to select light region









Estimating a Segmentation Threshold

An alternative approach to segmentation can be based on the following heuristic:

1. Select an initial estimate for t
2. Segment the image using t. This will produce two groups of pixels:
   1. G_1 consisting of all pixels > t
   2. G_2 consisting of all pixels <= t
3. Compute the average grey level values, μ_1 and μ_2, for the pixels in the regions G_1 and G_2
4. Compute a new threshold value:

$$t = (\mu_1 \text{ and } \mu_2)/2$$

5. Repeat steps 2 through 4 until the difference in t in successive iterations is less than a predefined parameter t_o