Image Processing and Imaging
Overview and Image Acquisition

Dominik Söllinger
Fachbereich Computerwissenschaften
Universität Salzburg

Winter term 2022/23

Outline

Image Properties - Signals

A signal is a function depending on some variable with physical meaning. Signals can be:

- one-dimensional (e.g. dependent on time),
- two-dimensional (e.g. images dependent on two co-ordinates in a plane),
- three-dimensional (e.g. describing an object in space),
- or higher-dimensional

- Monochromatic image: scalar function is suitable to describe this signal
- Colour images consisting of three component colors: vector function is necessary to describe the signal

Image Representation (1)

- Instead of a signal function a two-dimensional image matrix is used.
- Each element of the image matrix represents one pixel (*picture element*) with its position uniquely identified by the row- and column-index.
- Value of the matrix element represents the brightness value of the corresponding pixel within a discrete range (e.g. 0-255)

Figure: source image

The brightness values can also be represented by bars corresponding to the brightness value:

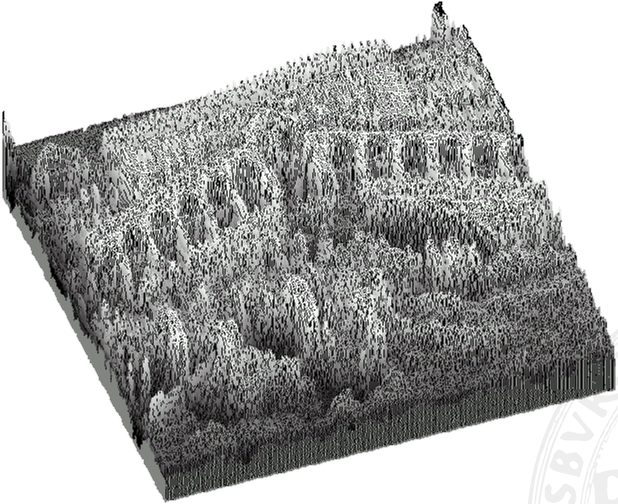


Figure: image as 2D-set of brightness values

Outline

Colour Representation - RGB and YUV (1)

- RGB colour model: luminance value is encoded in each colour channel
- YUV colour model: luminance is only encoded in the Y channel -

$$Y \approx 0.5\text{Green} + 0.3\text{Red} + 0.2\text{Blue}$$

$$Y + U = Y + (B - Y) = Y - Y + B = B$$

$$Y + V = Y + (R - Y) = Y - Y + R = R$$

$$Y - B - R = (R + G + B) - B - R = G$$

Figure: source image

red channel

green channel

blue channel

Figure: RGB decomposition of Debian image

luminance channel (Y)

red-green balance (U=B-Y)

yellow-blue balance (V=R-Y)

Figure: YUV decomposition of Debian image

Palette Images (i.e. "Malen nach Zahlen"):

- Include a lookup-table where an index identifies a certain colour in unique way
- Pixels do not carry luminance or colour information but the index
- Numerically close indices do not necessarily correspond to similar colours
- Example: GIF (graphics interchange format)

Outline

Resolution and Quantisation (1)

- Resolution is the partition of the image into fixed segments (discretisation) - each segment is represented by a picutre element (pixel) which exhibits a constant luminance or colour value (resolution is the number of pixels used to represent the image content in both image dimensions).
- Quantisation is the discrete number of gray or colour values used to represent luminance or colour information in a pixel (Bit per Pixel bpp).
- Storage requirements for an image is given as $b = N \cdot M \cdot \ln(F)$.
- The quality and perceivable detail of a digital image depends on the parameters N, M und F ab.

with N, M ... height and width of the image given in pixels; F ... number of luminance values or colour values per pixel.

raster 1/2

raster 1/4

raster 1/8

Figure: Resolution of Debian image

256 colors

64 colors

16 colors

Figure: Quantisation of Debian image

Outline

The **distance** between two pixels in a digital image is a significant quantitative measure. The distance between points with co-ordinates (i, j) and (h, k) may be defined in several different ways:

euclidean distance $D_E((i,j),(h,k)) = \sqrt{(i-h)^2 + (j-k)^2}$

city block distance $D_4((i,j),(h,k)) = |i-h| + |j-k|$

chessboard distance $D_8((i,j),(h,k)) = \max\{|i-h|, |j-k|\}$

Pixel adjacency is another important concept in digital images

- 4-neighborhood
- 8-neighborhood

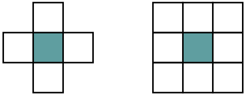
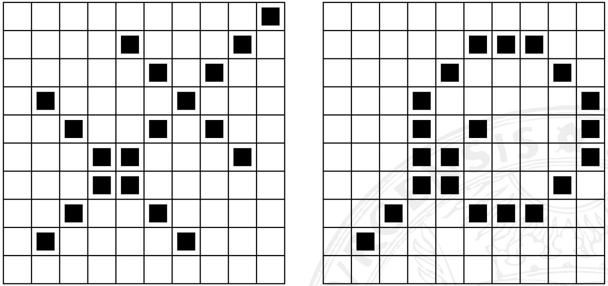


Figure: pixel neighborhoods

Region is a a contiguous set (of adjacent pixels). There exist various **contiguity paradoxes** of the square grid.



digital line | **closed curve paradox**

Figure: contiguity paradoxes

- Treat objects using 4-neighborhood and background using 8-neighborhood (or vice versa)
- Hexagonal grid solves many problems of the square grids. Any point in the hexagonal raster has the same distance to all its six neighbors.

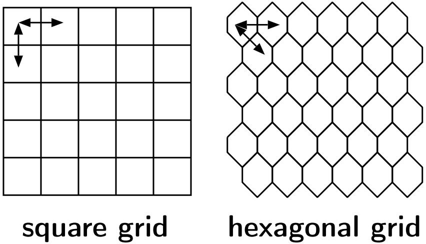


Figure: grid types

Border:

- Set of pixels within the region that have one or more neighbors outside R
- Distinguish between **inner** and **outer** borders

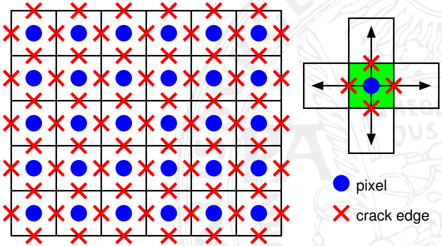
Edge:

- Local property of a pixel and its immediate neighborhood
- Vector given by a magnitude and direction
- Edge direction is perpendicular to the gradient direction, pointing in the direction of image function growth

The border is a global concept related to a region, while edge expresses local properties of an image function.

Crack edges:

- Four are attached to each pixel
- Defined by its relation to its 4-neighbors
- Direction of the crack edge is that of increasing brightness, and is a multiple of 90 degrees
- Magnitude is the absolute difference between the brightness of the relevant pair of pixels



Outline

Brightness histogram:

- Provides the frequency of the brightness value z in an image
- Lacks the positional aspect of the pixels
- Offers interesting properties, like *rotation-invariance* as well as *shift-invariance*:
  - Histogram does not change if the image is rotated or shifted (e.g. a circular wrapping of the entire image)
- Of major importance in case of object detection or tracking

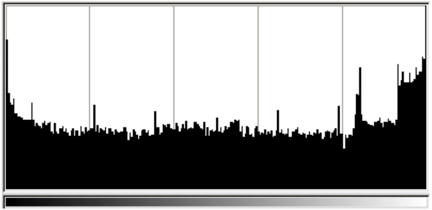


Figure: image histogram for Debian image

Figure: images with same histogram

Outline

Iconic images:

- Images containing original data; integer matrices with data about pixel brightness
- E.g. outputs of pre-processing operations (e.g., filtration or edge sharpening) used for highlighting some aspects of the image important for further treatment

Segmented images:

- Parts of the image are joined into groups that probably belong to the same objects
- Useful to know something about the application domain while doing image segmentation
- Easier to deal with noise and other problems associated with erroneous image data

Geometric representations:

- Hold knowledge about 2D and 3D shapes; quantification of a shape is very difficult but on the other hand very important

Relational models:

- Give the ability to treat data more efficiently and at a higher level of abstraction
- A priori knowledge about the case being solved is usually used in processing of this kind
- Example: counting planes standing at an airport using satellite images; a priori knowledge: position of the airport, relations to other objects, geometric models of the planes

Outline

Traditional Data Structures - Overview

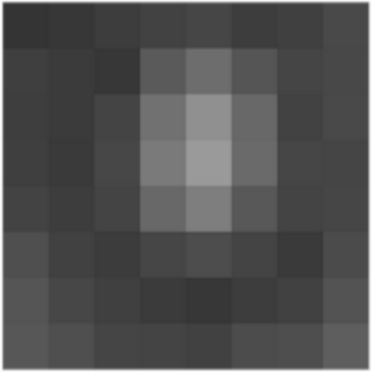
Some of the tradition data structures used for images are:

- Matrices
- Chains
- Graps
- Lists of object properties
- Relational databases
- ...

Outline

Matrices

- Most common data structure for low level image representation
- Elements of the matrix are integer numbers
- Image data of this kind are usually the direct output of an image capturing device



$$\begin{bmatrix} 52 & 55 & 61 & 66 & 70 & 61 & 64 & 73 \\ 63 & 59 & 55 & 90 & 109 & 85 & 69 & 72 \\ 62 & 59 & 68 & 113 & 144 & 104 & 66 & 73 \\ 63 & 58 & 71 & 122 & 154 & 106 & 70 & 69 \\ 67 & 61 & 68 & 104 & 126 & 88 & 68 & 70 \\ 79 & 65 & 60 & 70 & 77 & 68 & 58 & 75 \\ 85 & 71 & 64 & 59 & 55 & 61 & 65 & 83 \\ 87 & 79 & 69 & 68 & 65 & 76 & 78 & 94 \end{bmatrix}$$

Figure: Matrix representing an image

Outline

- Used for description of object borders
- Symbols in a chain usually correspond to the neighborhood of primitives in the image
- The chain code for the example in the figure starting at the red marked pixel is:

000077665555556600000644444444222111111223444456522 11

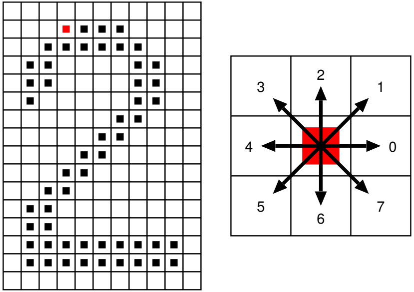


Figure: chain data structure

Chains (2)

- If local information is needed from the chain code, it is necessary to search through the whole chain systematically
  - Does the border turn somewhere to the left by 90 degrees?
  - A sample pair of symbols in the chain must be found - simple
- If global information is needed, the situation is much more difficult
- E.g. questions about the shape of the border represented by chain codes are not trivial
- Chains can be represented using static data structures (e.g., one-dimensional arrays)
  - Their size is the longest length of the chain expected
- Dynamic data structures are more advantageous to save memory

Outline

Topological Data Structures

- Describe images as a set of **elements and their relations**
- Can be expressed in graphs (evaluated graphs, region adjacency graphs)

- Region adjacency graph as an example of a topological data structure:

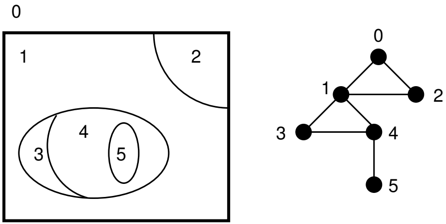
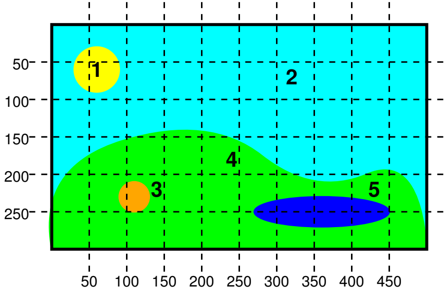


Figure: region adjacency graph

Outline

- Information is concentrated in relations between semantically important parts of the image: objects, that are the result of segmentation
- Especially appropriate for higher level image understanding



no.	object name	colour	min. row	min. col.	inside
1	sun	yellow	30	30	2
2	sky	blue	0	0	–
3	ball	orange	210	80	4
4	hill	green	140	0	–
5	lake	blue	225	275	4

Outline

Hierarchical Data Structures (1)

- Computer vision is by its nature very computationally expensive
  - If for no other reason than the amount of data to be processed
- One of the solutions is using parallel computers (*brute force*).
- Many computer vision problems are difficult to divide among processors, or decompose in any way

- Hierarchical data structures make it possible to use algorithms which decide a strategy for processing on the basis of relatively small quantities of data
- Work at the finest resolution only with those parts of the image for which it is necessary
- Using knowledge instead of brute force to ease and speed up the processing

- Two typical structures are **pyramids** and **quadtrees**

Problems associated with hierarchical image representation are:

- Dependence on the position, orientation and relative size of objects
- Two similar images with just very small differences can have very different pyramid or quadtree representations
- Even two images depicting the same, slightly shifted scene, can have entirely different representations

Outline

Matrix pyramid (M-pyramid):

- Sequence $\{M_L, M_{L-1}, \ldots\}$ of images
- M_L has the same dimensions and elements as the original image
- M_{i-1} is derived from M_i by reducing the resolution by one half
- Square matrices with dimensions equal to powers of two are required
- M_0 corresponds to one pixel only

- Only the difference between the prediction for M_i (computed from M_{i-1}) and the real M_i is saved for each level i of the pyramid

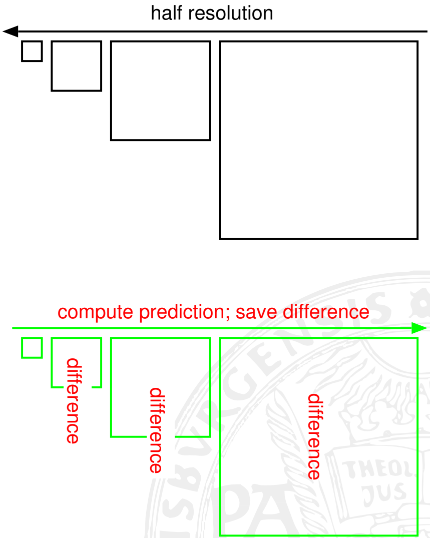


Figure: image pyramids

Pyramids (2)

Methods to reduce the image can be:

- Subsampling (only use a fraction of the input pixels)
- Averaging (average neighbouring pixels)
- Weighted averaging (Gaussian averaging filter - Gauss'sche Bildpyramide)
- Laplace pyramid (difference of neighbouring Gauss pyramid levels - DoG)

- Pyramids are used when it is necessary to work with an image at *different resolutions* simultaneously
- An image having one degree smaller resolution in a pyramid contains four times less data
- Processed approximately four times as quickly

A T-pyramid is a tree, where every node of the T-pyramid has 4 child nodes

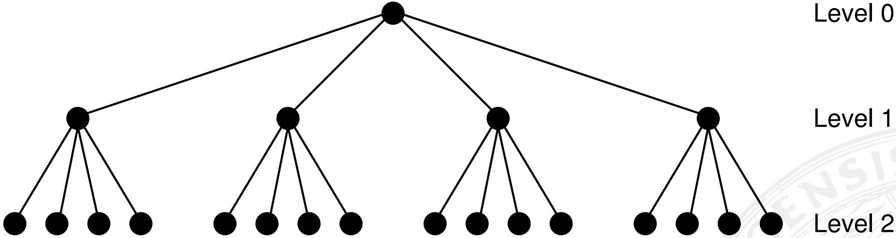


Figure: T-pyramid data structure

Quadtrees

Quadtrees:

- Modifications of T-pyramids
- Every node of the tree except the leaves has four children (NW: north-western, NE: north-eastern, SW: south-western, SE: south-eastern)
- Similarly to T-pyramids, image is divided into four quadrants at each hierarchical level
- However it is **not necessary** to keep nodes at all levels
- If a parent node has four children of the same value (e.g., brightness), it is not necessary to record them

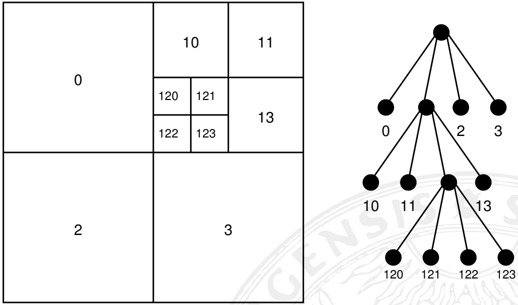


Figure: quadtree

Outline

We already observed that visual perception according to the Human Visual System (HVS) does not exactly correspond to the numerical values of single pixels but is highly context dependent.

Contrast

is the local change of luminance / intensity, defined as the fraction between the average object luminance and the average background luminance.

- Contrast is a logarithmic property:
  - No linear relation between the numerical contrast values and the perceptual ones
  - In case of higher luminance we need more contrast to achieve the same impression as in low luminance conditions
- Acuity ("Sehschärfe") is best when luminance does not change too fast or too slow. In either cases HVS performance is much weaker and we face the problem of high inter-observer variability.
- In addition, we have the "multiresolution property" caused by the non-uniform distribution of receptors in the retina.