Liste der noch zu erledigenden Punkte

Kapitel 1

1.Overview

- „image society“ (webpages: 1995 text-based, 2005 image based, 2015 video based . . .)

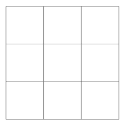
  - data transfer rates ↑, compression rates ↑
  critical shift: reading → watching

- „Photoshop“-ing (remove wrinkles, bumps, . . .)

- Images in medicine („medical image proscessing“), x-ray, CT, MRI, ultrasound, . . . („modalities“).
  different questions:

  1.) Layout!

  align bottom measurments $\overset{?}{\Rightarrow}$ image
  expl: tomography
  ⇒ difficult mathematical problems

  2.) Image enhancements
  - denoising
    simple pixels/lines: „sandpaper“ interpolation so richtig?
    global noise: smoothing
  - grayscale
    histogramm balancing (spreading)
  - distortion
    makes straight lines (in real world) straight (in the images)
  - edge detection
    contour enhancement
  - segmentation
    detect and separate parts of the image
  - registration
    sequence of images of the same object ⇒ Wort?
    , compare Skizze ↗ object following in a movie

Our Focus:

- mathematical models/methods/ideas
- (algorthms)
- ((implementation))

skipped: Very fast intro: Matlab and images

Kapitel 2

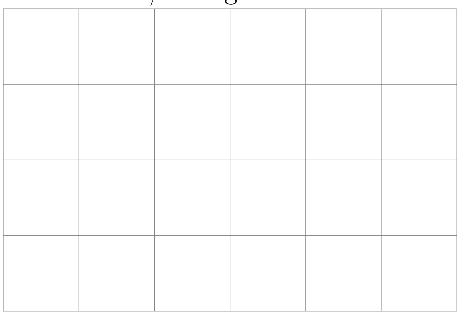
2.What is an image?

2.1 Discrete and continuous images

There are (at least) two different points of view:

- discrete/digital

- continuous/analogue



object:	matrix	function
tools:	linear algebra (SVD, . . .)	analysis (differentrage, integrate, . . .)
pros:	(finite storage) storage, complexity	freedom, tools, motions?P.4 (e.g. edge discontinuity)
cons:	limitations: zooming, rotations, . . .	storage (infinite amout of data)

arguably, one has:

- real life $\Rightarrow$ continuous „images“ (objects)
- digital camers $\Rightarrow$ discrete images

In general we will say:

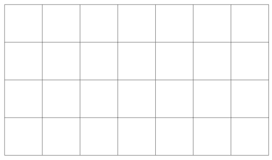
Definition 2.1 ((mathematical) image)**.** A (mathematical) *image* is a function

$$u : \Omega \to F,$$

where: $\Omega \subset \mathbb{Z}^d$ (discrete) or $\Omega \subset \mathbb{R}^d$ (continuous) . . . *domain*
$d = 2$ (typical case 2D), $d = 3$ („3D image“=body or $\underbrace{\text{2D+time}}_{\text{movie}}$)
$d = 4$ (3D + time)

F . . . *range of colours*
$F = \mathbb{R}$ or $[0, \infty]$ or $[0, 1]$ or $\{0, \dots 255\}$, . . . grayscale (light intensity)
$F \subset \mathbb{R}^3$ . . . RGB image (colored)

$F = \{0, 1\} \ldots$ black/white



3 Layers
$\Rightarrow$ colored images:w

Matlab stuff

Large parts of the course: analytical approach (i.e. continuous domain Ω)
Since we want to differentirate, ... the image u.

Still: need to assume that also F ist continuous (not as $\{0, 1\}$, $\{0, 1, \ldots, 255\}$ or $\mathbb{N}$)
since otherwise the only differentiable (actually, the only continuous) functions $u : \Omega \to F$ are *constant* functions $\Leftrightarrow$ single-colour images

Also: We usually take F one-dimensional ($F \subset \mathbb{R}$). Think of it as either

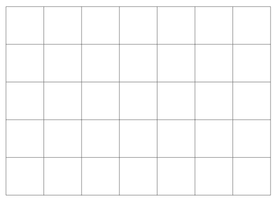
- gray scaled image, or
- treating R,G & B layer separately

2.2 Switching between discrete and continuous images

continuous $\to$ discrete:

- divide the continuous image in small squared pieces (boxes) (superimpose grid)
- now: represent each box by *one* value
  - strategy 1: take function value $u(x_i)$
    for x_i = midpoint of box B_i
  - strategy 2: use mean value

$$\frac{1}{|B_i|} \int_{B_i} u(x)dx$$



$\Rightarrow$ discrete image

strategy 1: simple (and quick) but problemativ ($u(x_i)$ might represent $u|_{B_i}$ badly; for $u \in L^p$, single point evaluation not even defined)

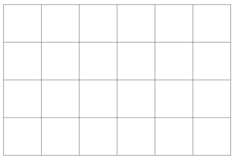
strategy 2: more komplex but also more „democratic" (actually closer to the way how CCD Sensors in digital camers work)

often the image value of the box B_i gets also digitized, i.e. fitted (by scaling & rounding) into range $\{0, 1, dots, 255\}$

discrete $\to$ continous

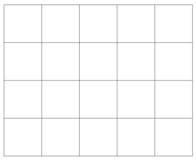
This is of course more tricky ...

- Again: each pixel of the discrete image corresponds to a „box" of the continuous image (that is still to be constructed)
- Usually: pixel value $\mapsto$ function value at the *midpoint* of the box
- Question: How to get the other function values (in the box)?



<u>idea 1:</u> just take the function value of the nearest midpoint („nearest neighbour interpolation“)

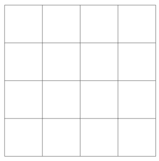
For each $x \in B_i : u(x) := u(x_j)$ where $|x - x_j| = \min_k |x - x_k|$



$\Rightarrow$ $u(x) = u(x_i)$ for all $x \in B_i$
$\Rightarrow$ each box is uni-color
$\Rightarrow$ the continuous image is essentially still discrete

<u>idea 2</u>: (bi-) lineare interpolation

hier fehlt idea 2 bis P.7

Beispiel 2.2. Roate image  by angle $\phi \neq k \cdot \frac{\pi}{2}$

hier fehlt alles bis P.8

Kapitel 3

3.Histogramm and first applicatsion

3.1 The histogramm

Definition 3.1 (histogram).