Image Processing

Summary

Themes

1. Human Seeing
2. Filtering
3. Morphological Operations
4. Fourier Transform
5. Diffusion Filters
6. Continuous Energy Minimization
7. Discrete Energy Minimization
8. Interest Points
9. Image Features
10. 3D-Geometry
11. RANSAC
12. Stereo
13. Tracking

Filtering

Image generation

Original

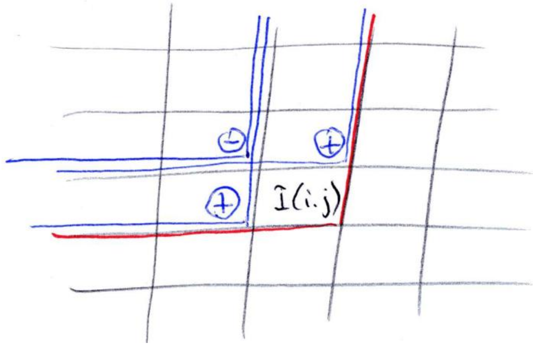
Noise

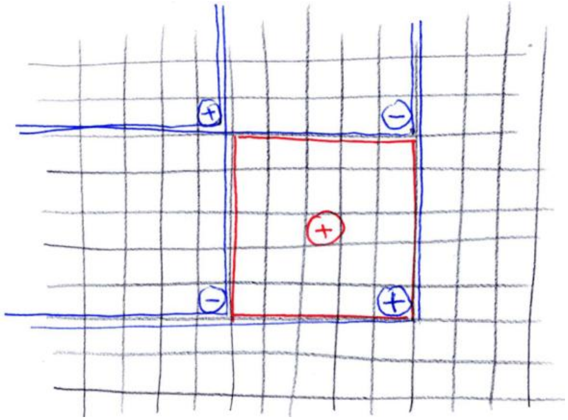
Result

Image restoration

Filtering

Integral Image:





Useful (efficient) for Mean-Filtering, Harris Detector, Haar Features and many more …

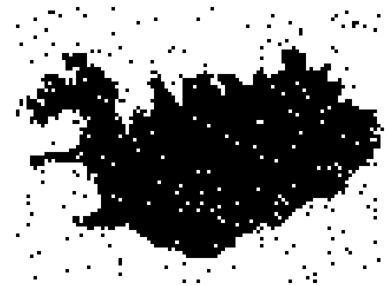
Convolution techniques, separable filters, box-filter, …

Use filtering with care and respect. Do not use filters without to know, what are they really doing, have in mind always the whole:

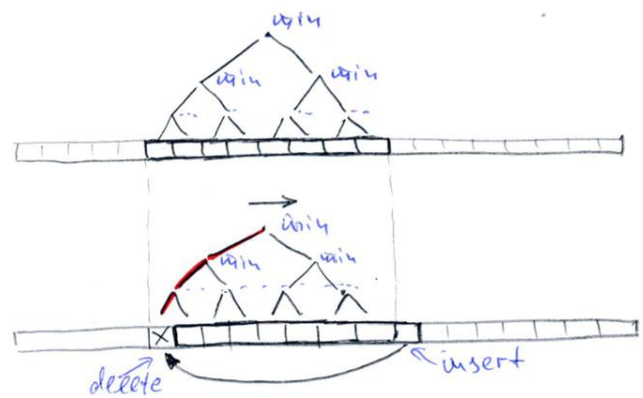
Model → Formal task → Solution → Algorithm (program)

Morphological Operations

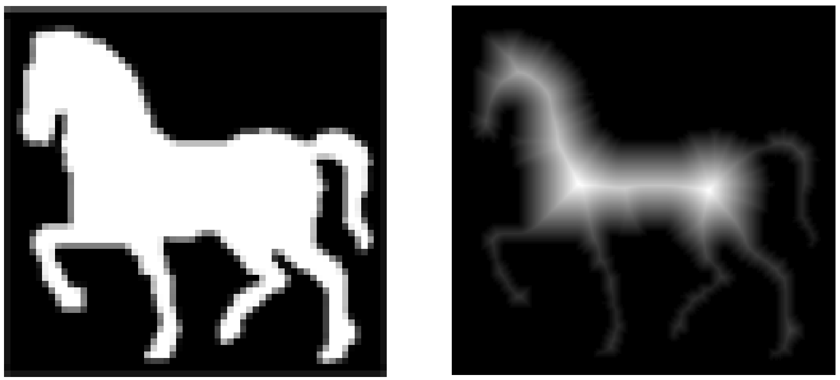
Dilation, erosion, opening, closing
structuring elements →



Fast algorithms →



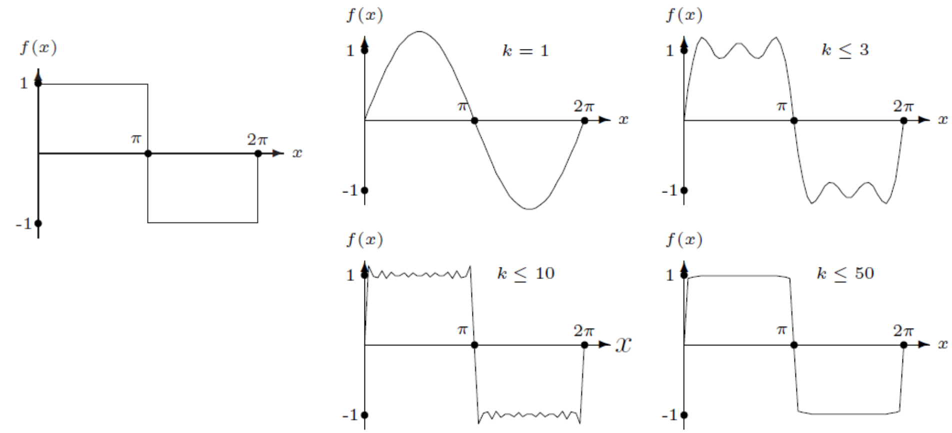
Distance transform
structuring functions →



Fourier Transform

Bases in function spaces – Math

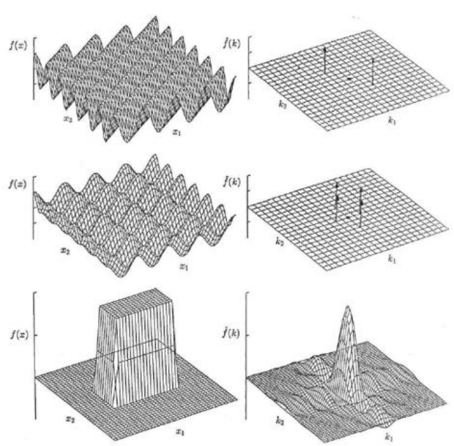
Fourier transform

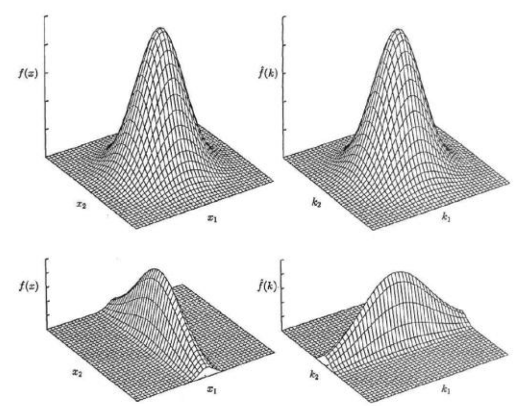


Convolution theorem + consequences

$$\mathcal{F}[f * g] = \mathcal{F}[f] \cdot \mathcal{F}[g]$$

Spectrums





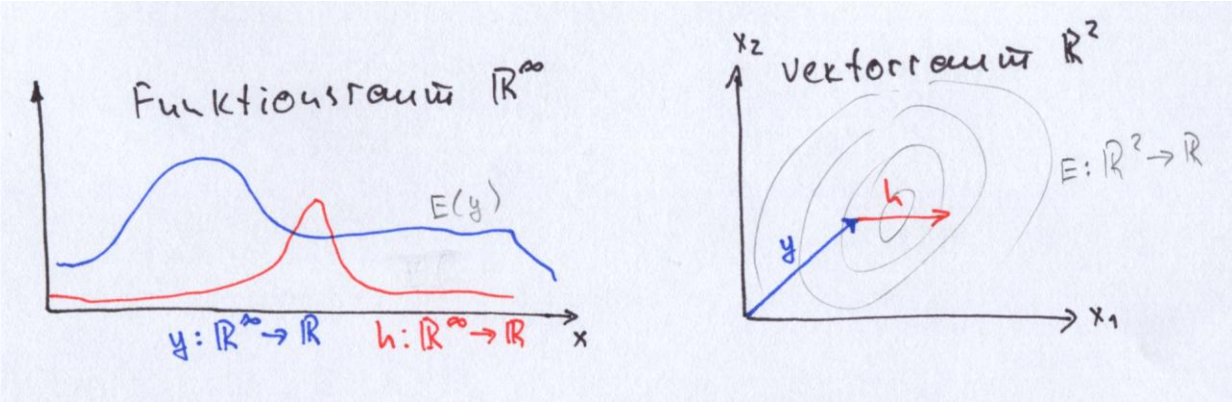
Diffusion Filters

Physical background – Fick`s law, flux, tensors, gradients, divergences etc.

Linear, non-linear, isotropic, anisotropic diffusions, edge information for tensors, diffusivity (robust functions), TV-flow

Numerical schemes: explicit, implicit, reduction to linear systems

Continuous Energy Minimization



Relations: continuous ↔ discrete (domain of definition, range)

Example: denoising, energy terms

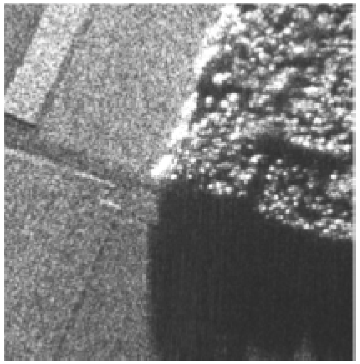
Discrete domain of definition: reduction to large linear systems

Continuous domain of definition: calculus of variations, Gâteaux-derivative, Euler-Lagrange equations

Relation to diffusion, regularizers

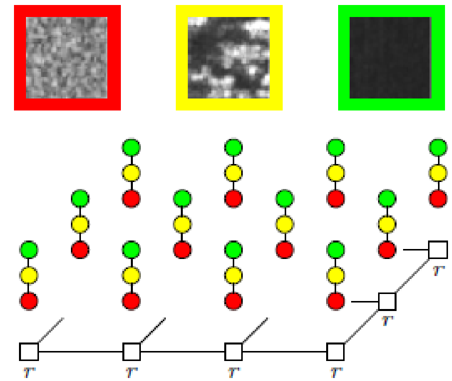
Discrete Energy Minimization

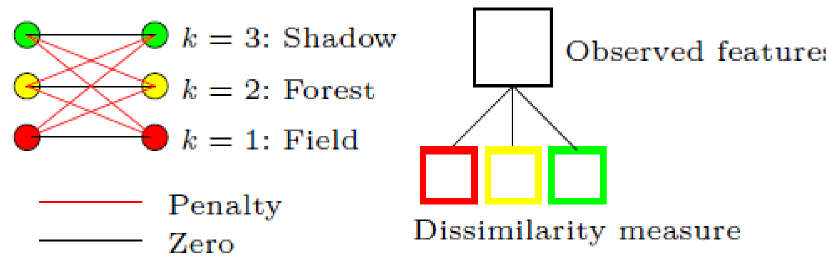
Example: segmentation



Original

A possible segmentation





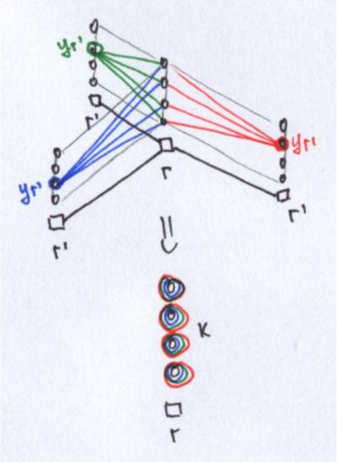
$$y^* = \arg\min_y \left[\sum_r q_r(y_r) + \sum_{rr'} g_{rr'}(y_r, y_{r'}) \right]$$

Discrete Energy Minimization

Iterated Conditional Modes:

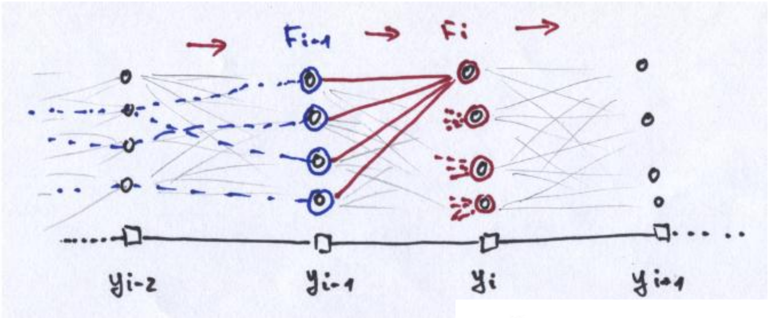
$$y_r = \arg\min_k \left[q_r(k) + \sum_{r':rr' \in E} g_{rr'}(k, y_{r'}) \right]$$

→



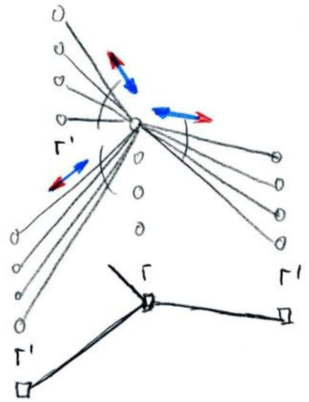
Dynamic Programming:

→

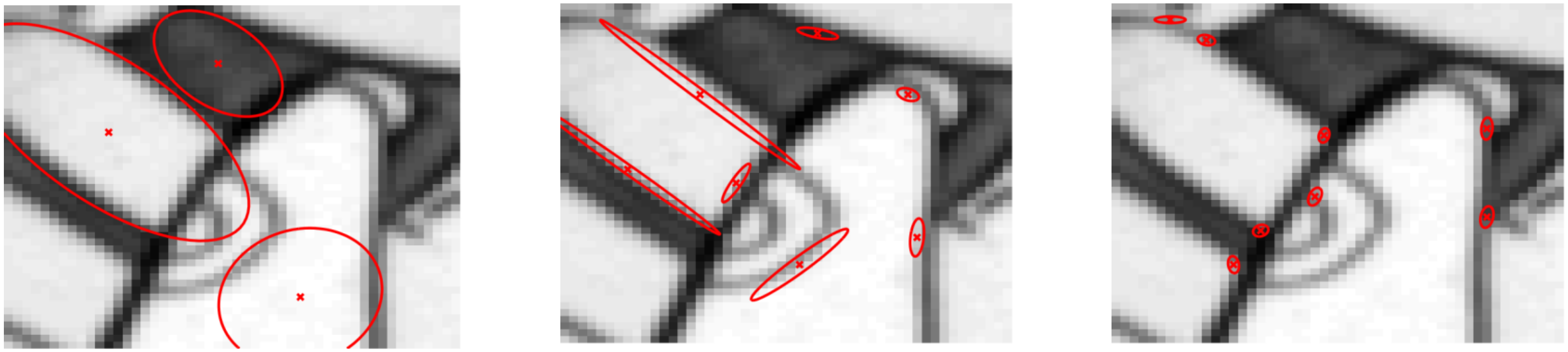


Equivalent transformations, Seeming Quality,
Diffusion Algorithm:

→



Harris detector: auto-correlation function, fast algorithm



Maximally stable extremal regions (MSER): definition, properties

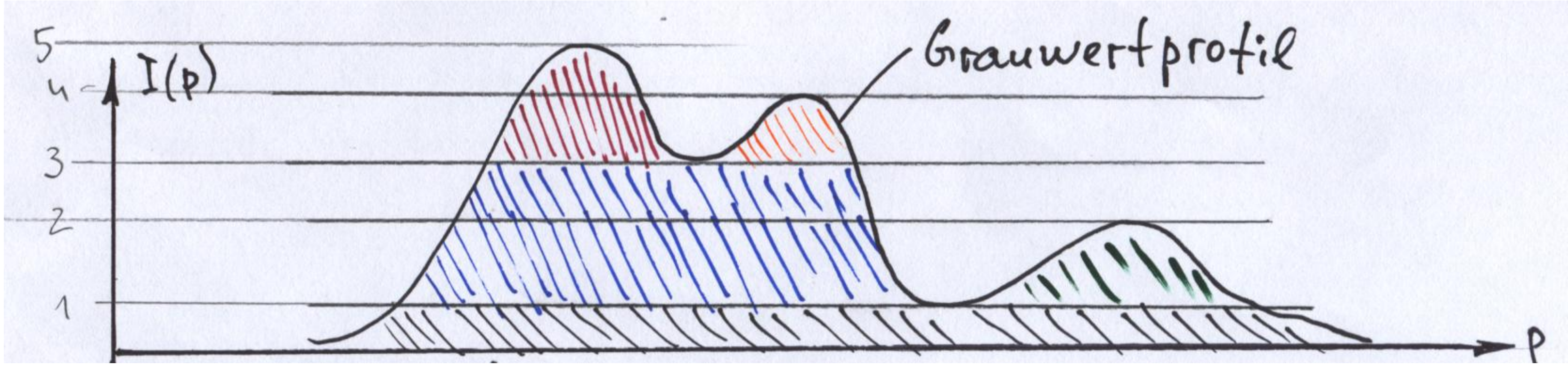
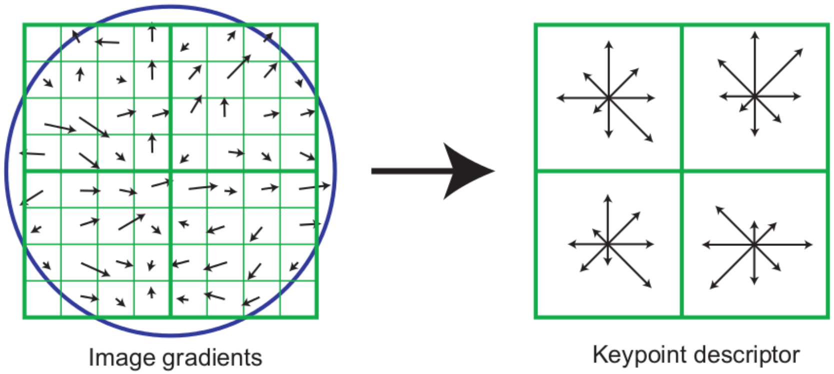
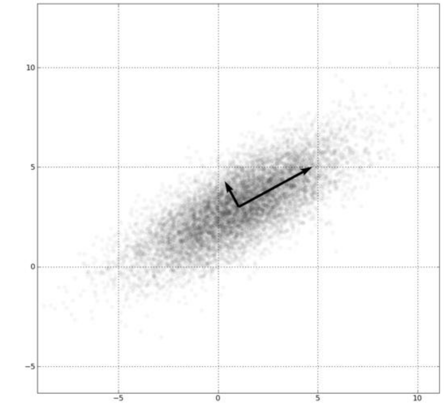


Image Features

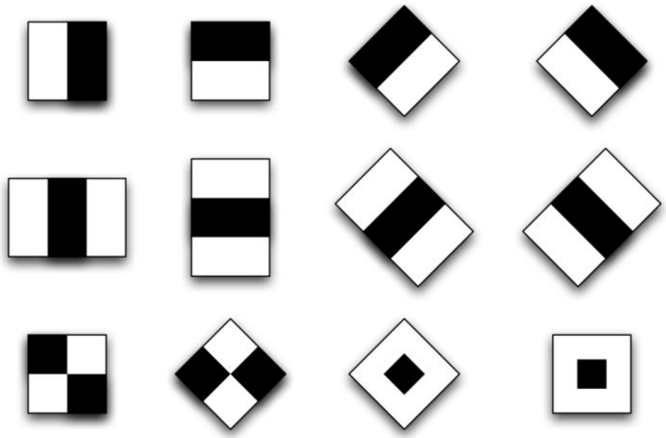
SIFT:



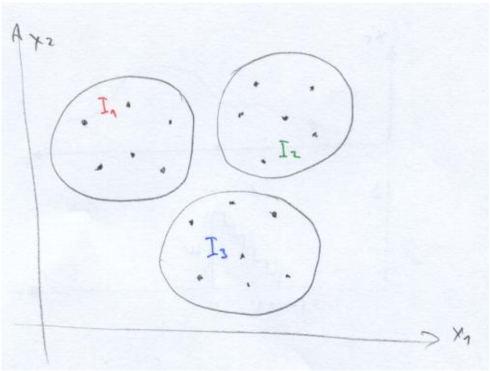
PCA:

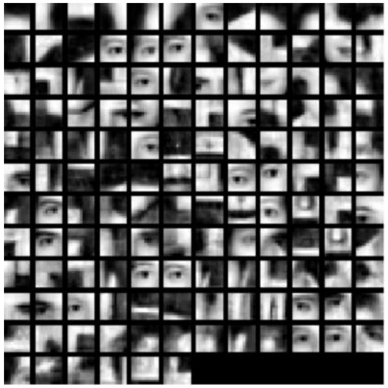


Haar Features:



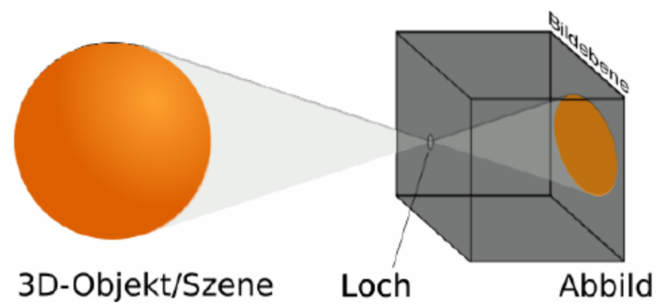
Clustering, Visual Words:



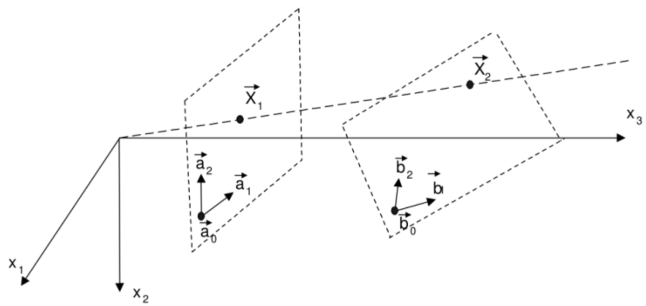


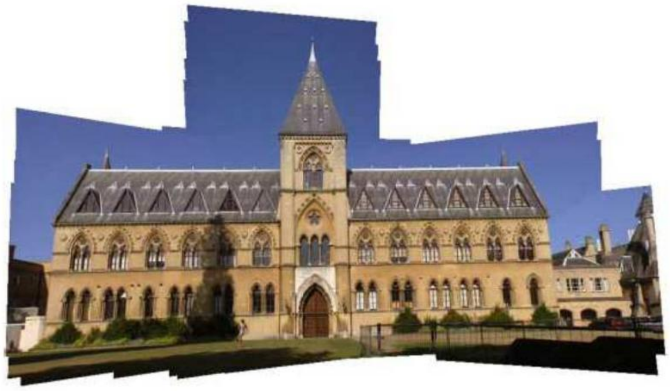
3D-Geometry

Pinhole camera model:



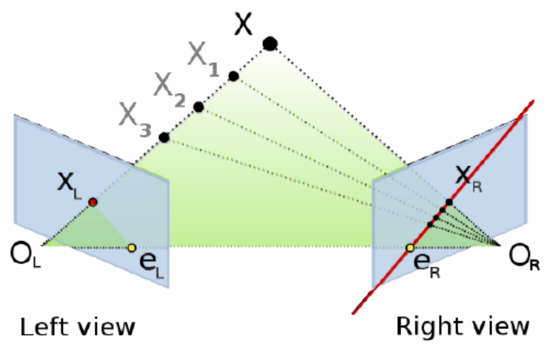
Homography:



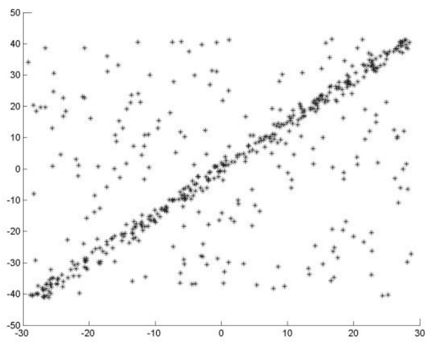


Epipolar Geometry:

$$x_l^T F x_r = \begin{bmatrix} x_{l1} & x_{l2} & 1 \end{bmatrix} \begin{bmatrix} f_{11} & f_{12} & f_{13} \\ f_{21} & f_{22} & f_{23} \\ f_{31} & f_{32} & f_{33} \end{bmatrix} \begin{bmatrix} x_{r1} \\ x_{r2} \\ 1 \end{bmatrix} = 0$$



Model fitting, Outliers:



Algorithm:

Wiederhole oft

Würfele $L' \subset L, \ |L'| = d$

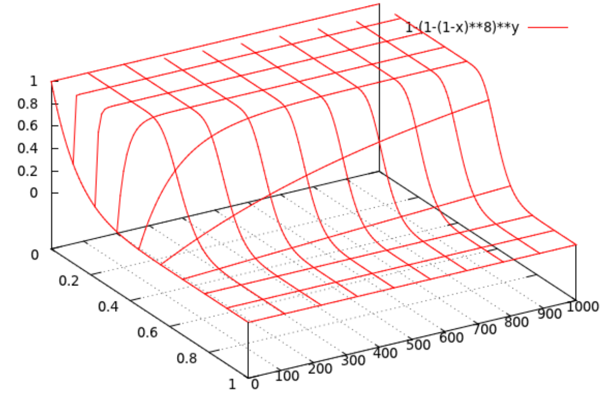
Schätze $y = g(L')$

Bewerte $f(y) = \sum_i f(x^i, y)$

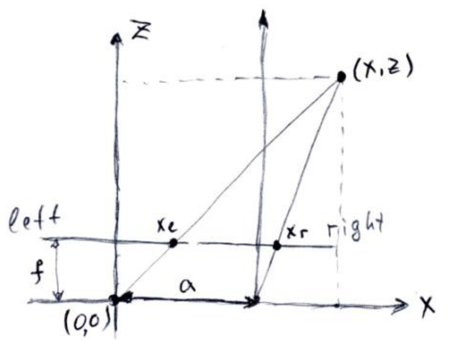
wenn $f(y) > f(y^*)$

`setze` $y^* = y$ `und merke` $f(y^*)$

Convergence:



Disparities:



$$\frac{Z}{f} = \frac{X}{x_l} \quad ; \quad \frac{Z}{f} = \frac{X-a}{x_r}$$

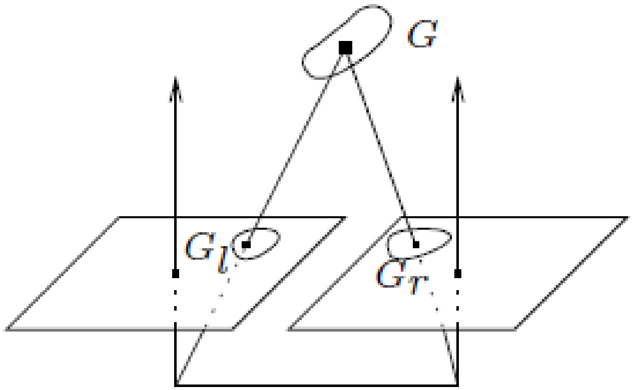
$$\frac{X}{x_l} = \frac{X-a}{x_r}$$

$$X = \frac{a \cdot x_l}{x_l - x_r}$$

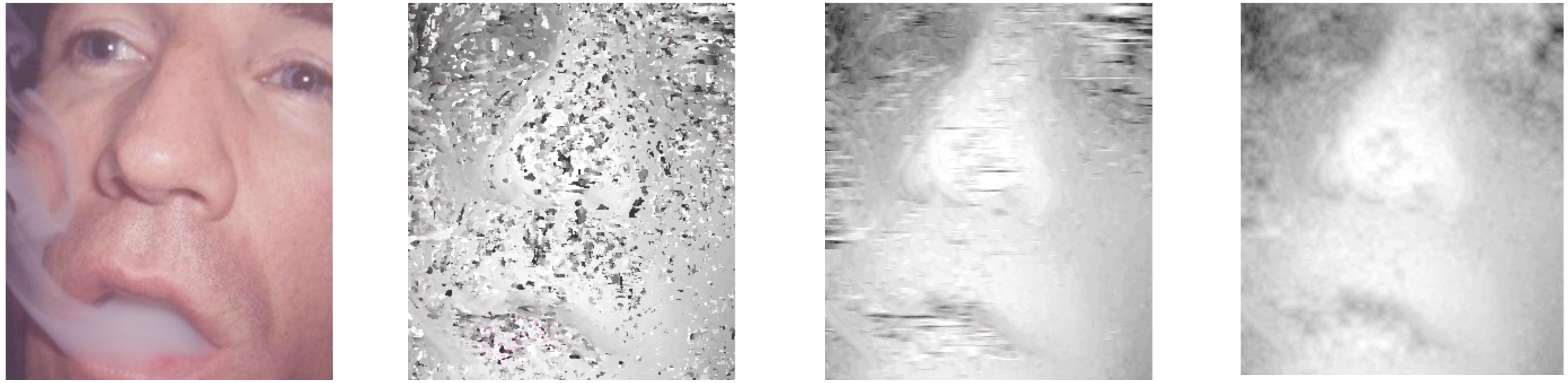
$$Z = \frac{f \cdot X}{x_l} = \boxed{\frac{f \cdot a}{x_l - x_r}}$$

disparity

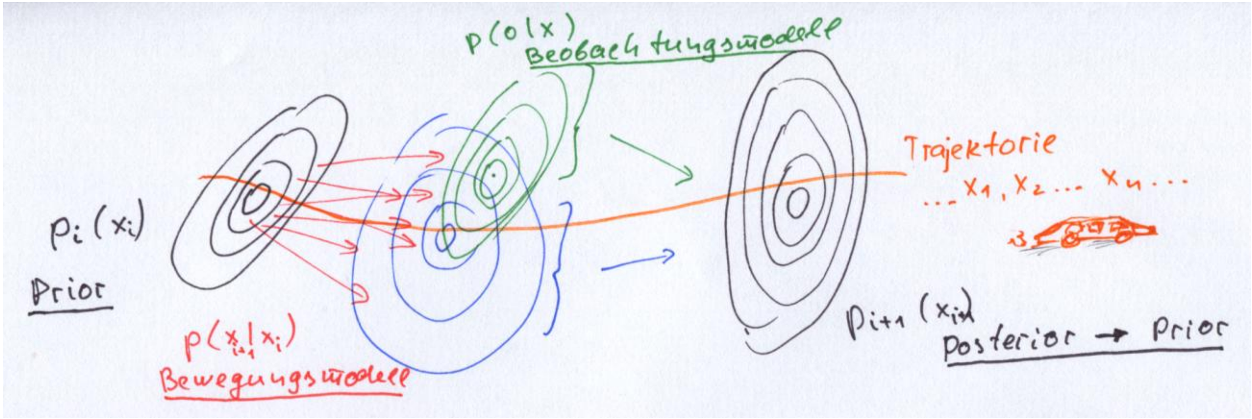
Dissimilarity measures:



Block Matching → Row-wise stereo → Energy Minimization



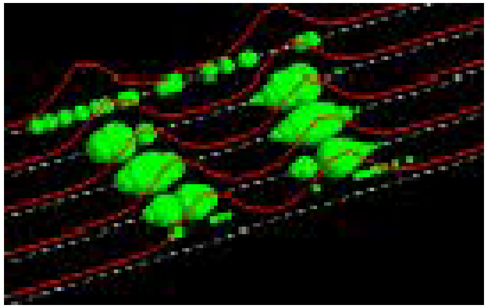
Bayesian Filtering:



Kalman Filter:

$$\begin{bmatrix} x_{i+1} \\ y_{i+1} \\ v_{x,i+1} \\ v_{y,i+1} \end{bmatrix} = \begin{bmatrix} 1 & 0 & \Delta t & 0 \\ 0 & 1 & 0 & \Delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_i \\ y_i \\ v_{x,i} \\ v_{y,i} \end{bmatrix} + \epsilon \qquad \begin{bmatrix} o_{x,i+1} \\ o_{y,i+1} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_i \\ y_i \\ v_{x,i} \\ v_{y,i} \end{bmatrix} + \delta$$

Particle Filtering:



WS 2013/2014

New Prof. – Carsten Rother

New courses:

Computer Vision (Carsten Rother):

similar to Image Processing, but with “more CV”: more geometry (especially 3D), image formation and cameras, motion estimation (Optical Flow), recognition, Pictorial Structures …

Practice: different compared to IP – more structured:
There are three block, each one consisting of a “lecture” (fast implementations, algorithmic tricks and techniques etc.) and “free work” (assignment).

Combination BV(SS2013)+CV(WS2013/2014) is allowed for exams.

WS 2013/2014

Machine Learning (Dmitrij Schlesinger):

Almost the same as Pattern Recognition, but:

- a bit less Neural Networks,
- more Machine Learning: regressions, other learning principles (e.g. reinforcement learning), other classifiers (e.g. decision trees) etc.

Combination ME(SS2013)+ML(WS2013/2014) is not allowed.

Other courses (Holger Heidrich with others):

- Einführungspraktikum Computer Vision
- Komplexpraktikum Computer Vision
- Projektpraktikum Computer Vision
- Hauptseminar Bildanalyse