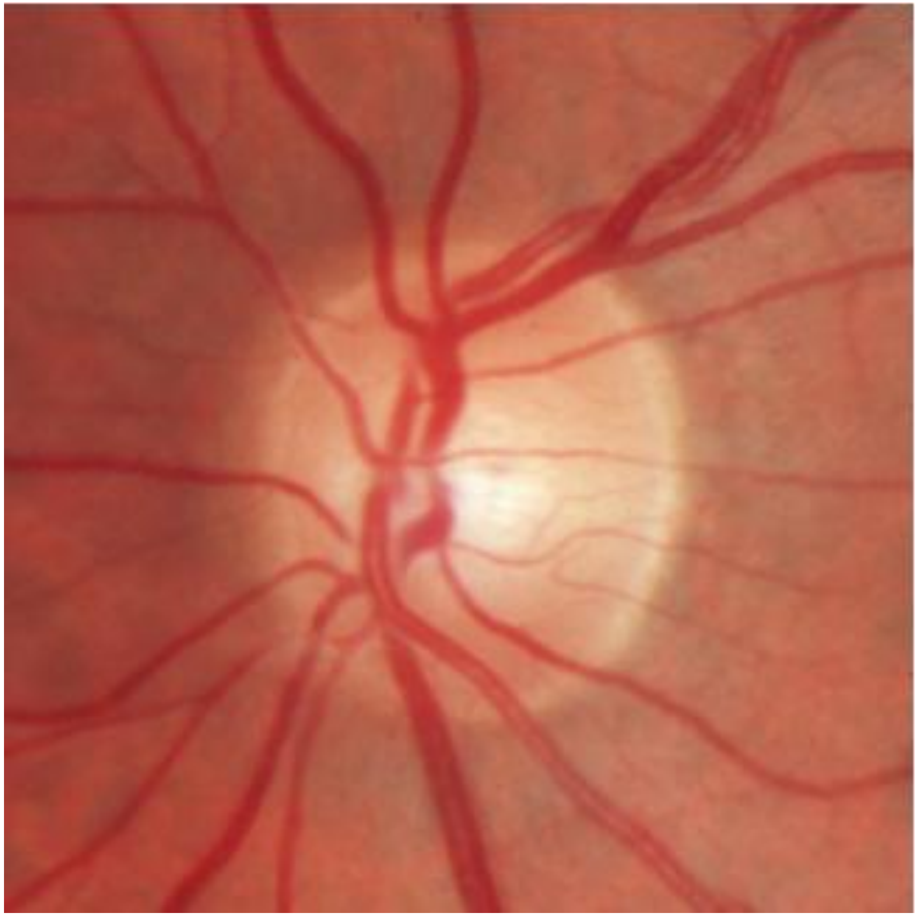
Image Processing

Stereo

Stereo vs. others

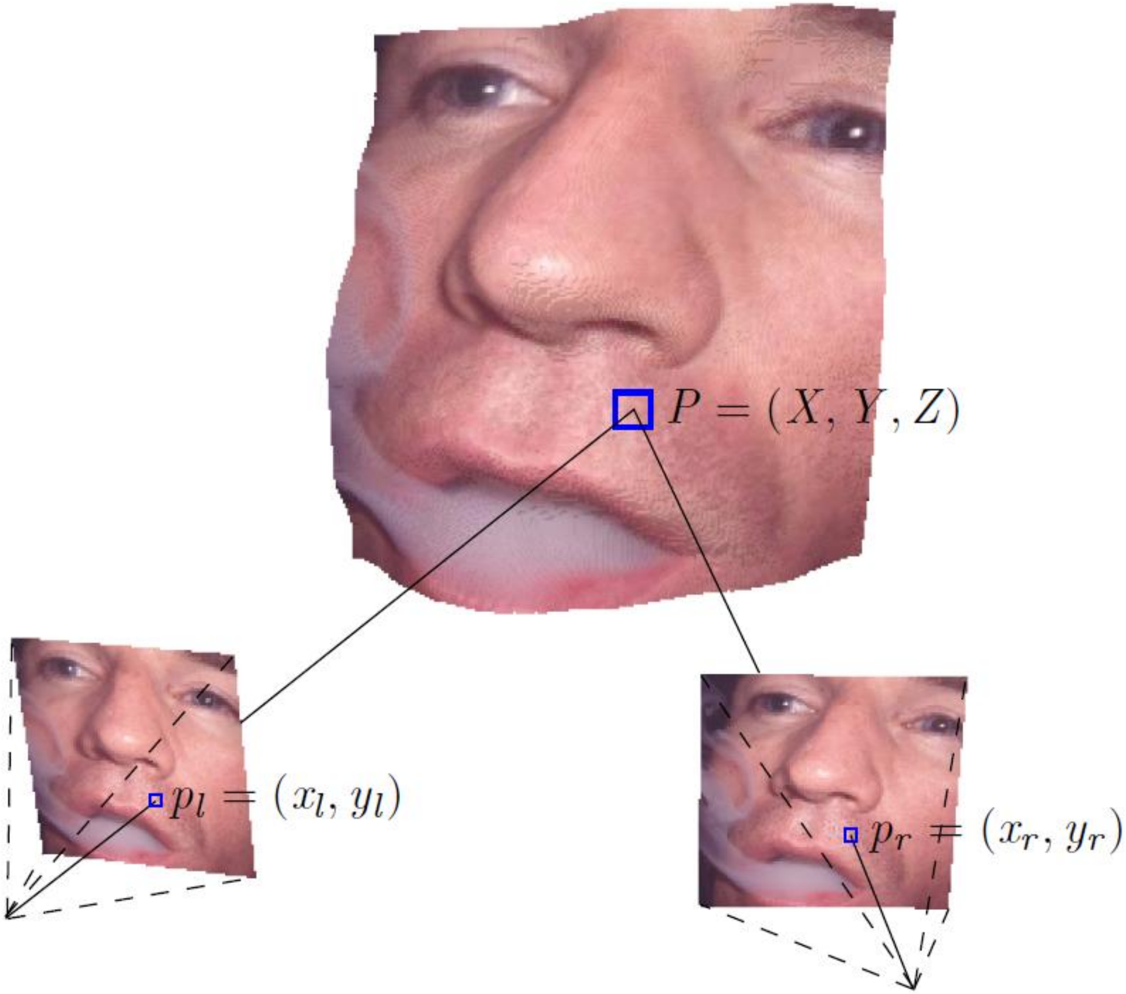
Shape from texture

Shape from shading



Stereo is especially useful if no further cues are available – i.e. the scene "can not be recognized"

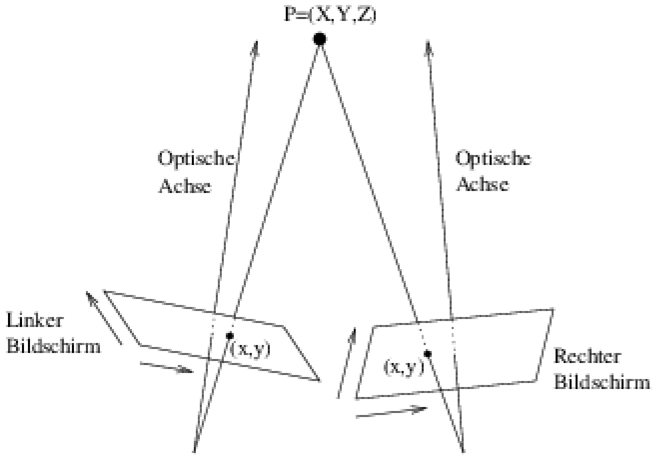
Principle

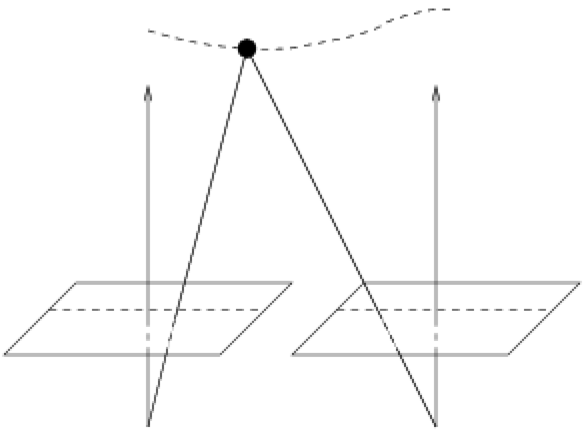


It is possible to estimate 3D-point from its two projections

Note: not all projection pairs correspond to 3D-points

Geometry





General case:

$$x_l F x_r = 0$$

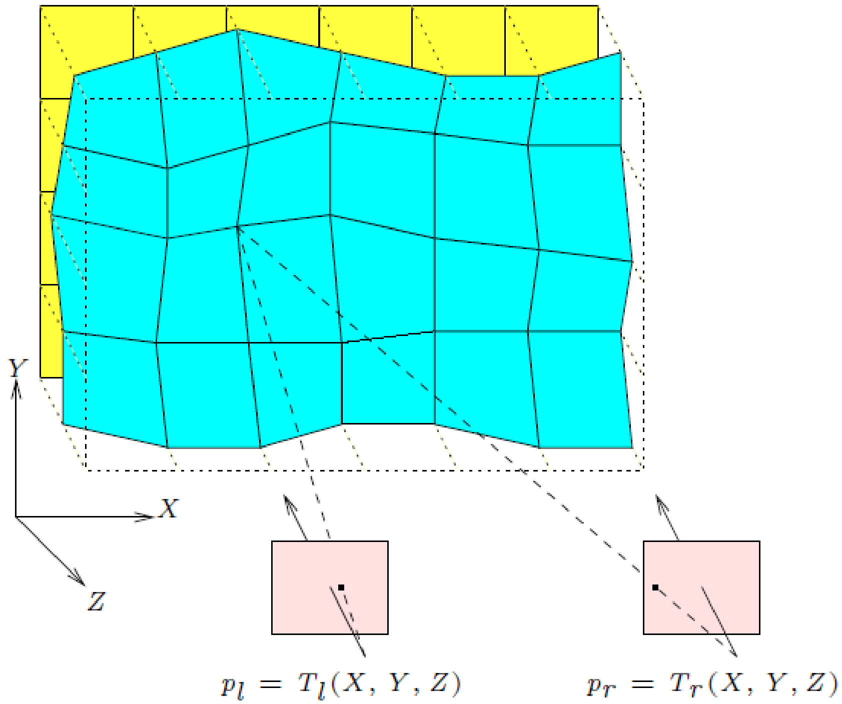
(epipolar geometry constraint)

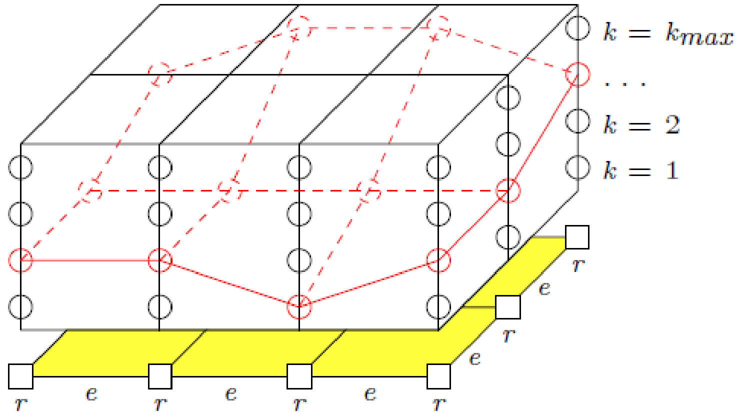
Rectified stereo:
the set of correspondences is

$$(x_l, x_r, y), \text{ d.h. } \in \mathbb{R}^3$$

Stereo = Epipolar geometry + **Correspondence problem** (today)

Depth map (discrete formulation)



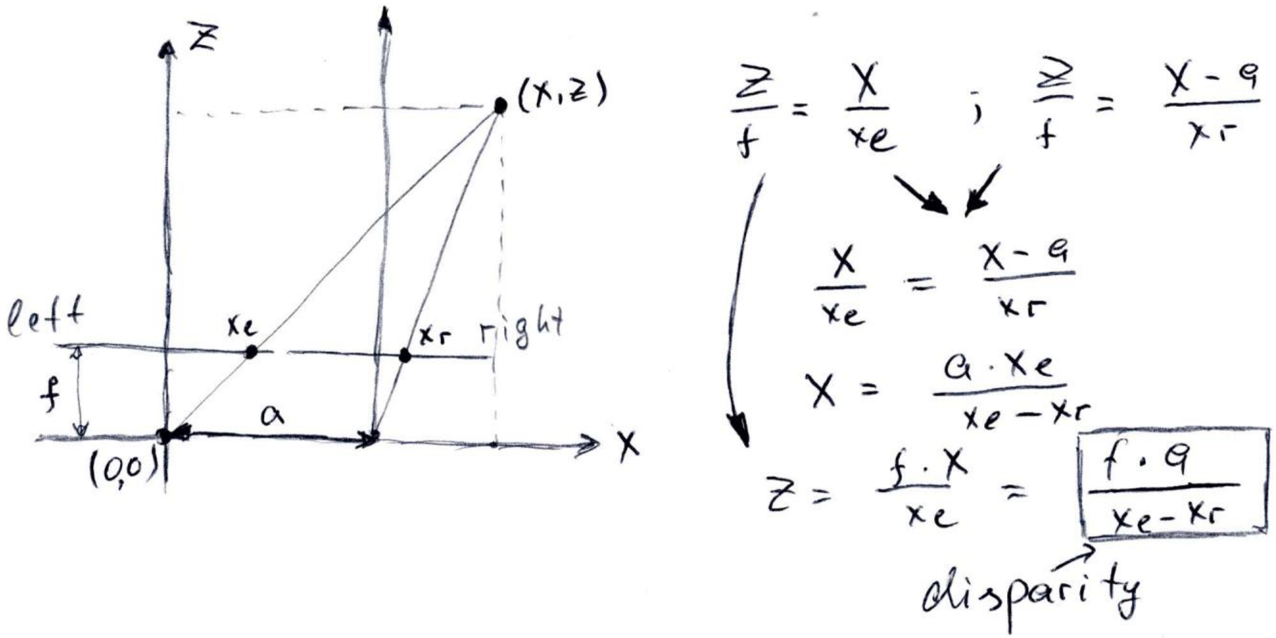


The domain of definition is a graph $V = (R, E)$, $r \in R$ is a “pixel”

The range is the discretized set of depth values K

The depth map is a mapping $y : R \to K$
(assigns a depth value $k \in K$ to each position $r \in R$)

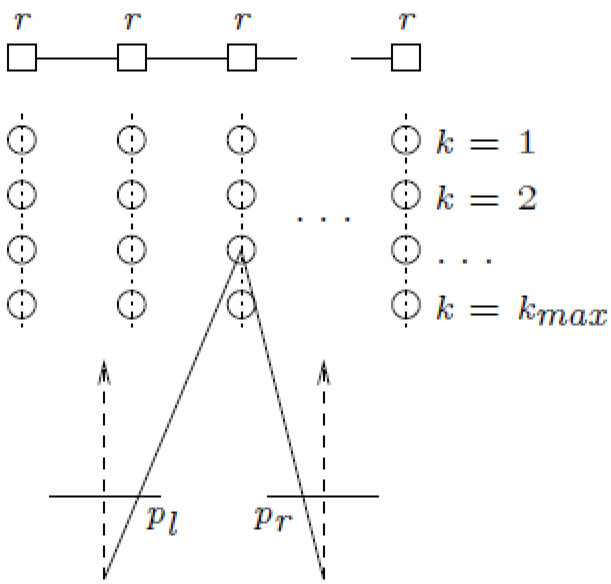
Disparities



A slightly other formulation: the set of variables (the domain of definition) is the set of pixels of the **left** image, the range consists of all **disparity** values.

For **each** pixel of the left image **exactly one** correspondent pixel is to be given (a non-symmetric formulation).

Dissimilarity measures



Each pair (r, k) (position, depth value) corresponds to a 3D-point and consequently to two projections

$$p_l \in \mathbb{R}^2, p_r \in \mathbb{R}^2$$

How to rate it?

A simple choice – quadratic difference between the color values:

$$A(r,k) = A(p_l, p_r) = \Big(I_l(p_l) - I_r(p_r)\Big)^2$$

A bit more complex – average over a small neighborhood:

$$A(p_l, p_r) = \sum_{\triangle p \in F} \left[I_l(p_l + \triangle p) - I_r(p_r + \triangle p)\right]^2$$

Dissimilarity measures

Further step – allow certain color transformation, e.g. allow image patches to be more light or dark → search for an additive constant C_v so that the **residual squared difference** is minimal:

$$A(p_l, p_r) = \min_{C_v} \sum_{\triangle p \in F} \left[I_l(p_l + \triangle p) + C_v - I_r(p_r + \triangle p) \right]^2$$

$$\frac{\partial A}{\partial C_v} = \sum_{\triangle p \in F} 2 \left[I_l(p_l + \triangle p) + C_v - I_r(p_r + \triangle p) \right] = 0$$

$$\Rightarrow \quad C_v = \frac{1}{|F|} \sum_{\triangle p \in F} \left[I_r(p_r + \triangle p) - I_l(p_l + \triangle p) \right]$$

Set the optimal value of C_v into the subject ↑ and obtain

$$A(p_l, p_r) = \sum_{\triangle p \in F} \left[I_l(p_l + \triangle p) - I_r(p_r + \triangle p) \right]^2 - C_v^2 \cdot |F|$$

Dissimilarity measures

Further allowed color transformation – e.g. contrast change
(a multiplicative constant in addition to the additive one)

$$A(p_l, p_r) = \min_{C_v, C_s} \sum_{\triangle p \in F} \left[I_l(p_l + \triangle p) \cdot C_s + C_v - I_r(p_r + \triangle p) \right]^2$$

→ “almost” the correlation coefficient

$$A(p_l, p_r) = \frac{\sum_{\Delta p} \left(I_l(p_l + \triangle p) - \bar{I}_l \right) \left(I_r(p_r + \triangle p) - \bar{I}_r \right)}{\sqrt{\sum_{\Delta p} \left(I_l(p_l + \triangle p) - \bar{I}_l \right)^2} \sqrt{\sum_{\Delta p} \left(I_r(p_r + \triangle p) - \bar{I}_r \right)^2}}$$

with

$$\bar{I}_{lr} = \frac{1}{|F|} \sum_{\triangle p} I_{lr}(p_{lr} + \triangle p)$$

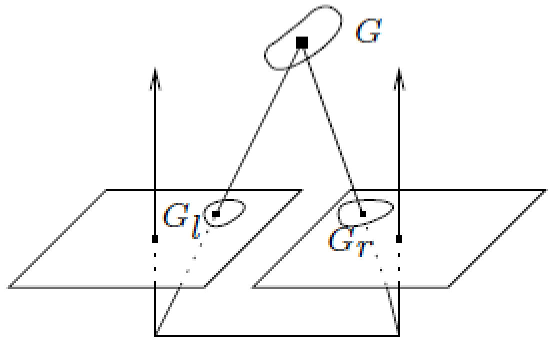
Further possibilities – general monotone color transformation etc.

Dissimilarity measures

How to choose the window size?

- The larger the more robust against noises
- But: even for really correspondent points it may lead to geometric distortions

see →



The way out – allow geometric transformations (affine, projective etc.)

$$A(p_l, p_r) = \min_{C_v, C_s, Tr} \sum_{\triangle p \in F} \Big[I_l(p_l + \triangle p) \cdot C_s + C_v - I_r\big(Tr(p_r + \triangle p)\big)\Big]^2$$

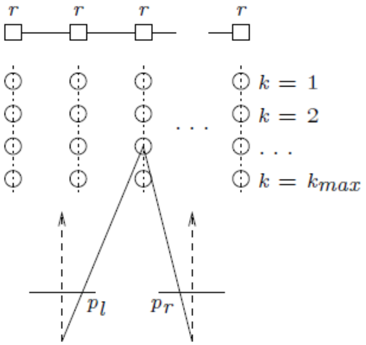
Generally: the more allowed transformation, the less "discriminative" the dissimilarity measure

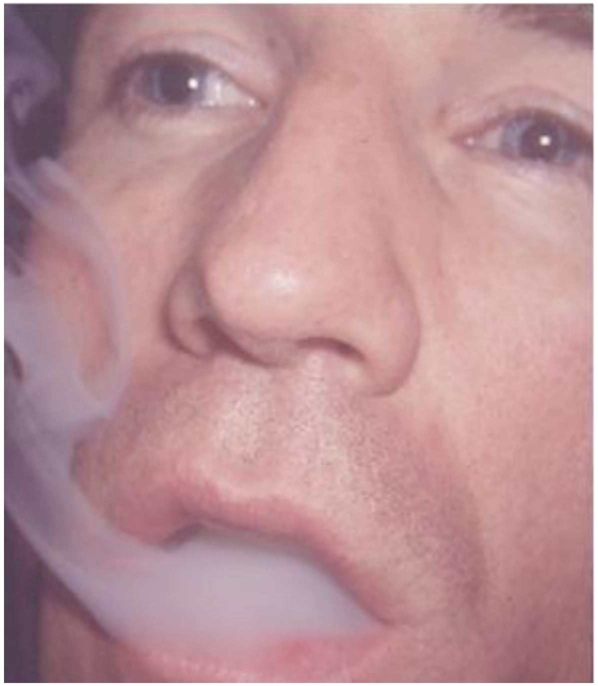
→ a compromise "noise ↔ signal" is crucial.

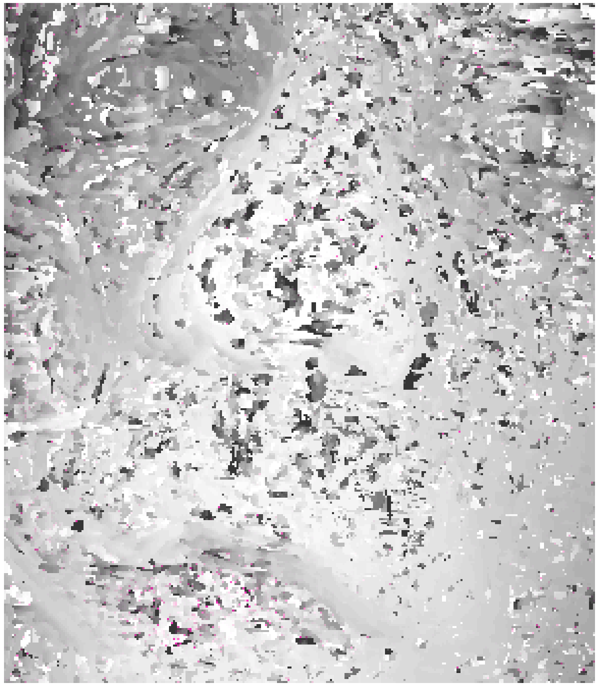
Block matching

No further assumptions about the depth map – decisions are done **independently** for all r

$$y(r) = \arg\min_k A(r,k) \quad \forall r$$



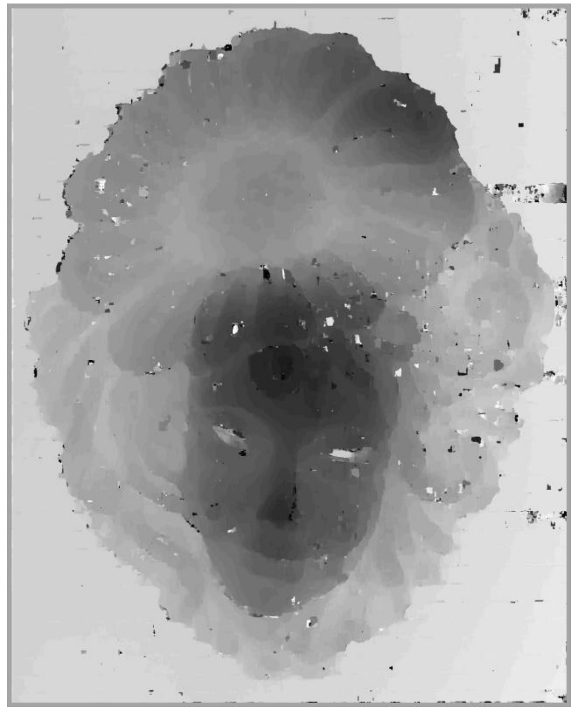




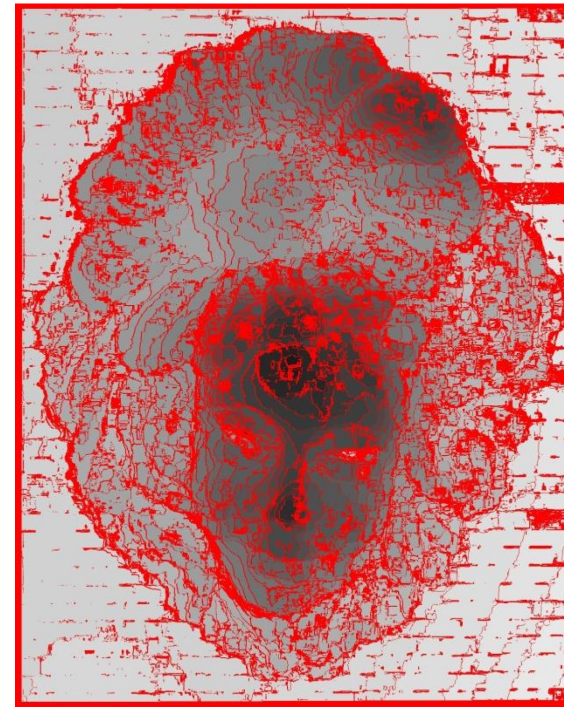
Block matching

Some wrong matches can be filtered out by the cross-check

Original (left)



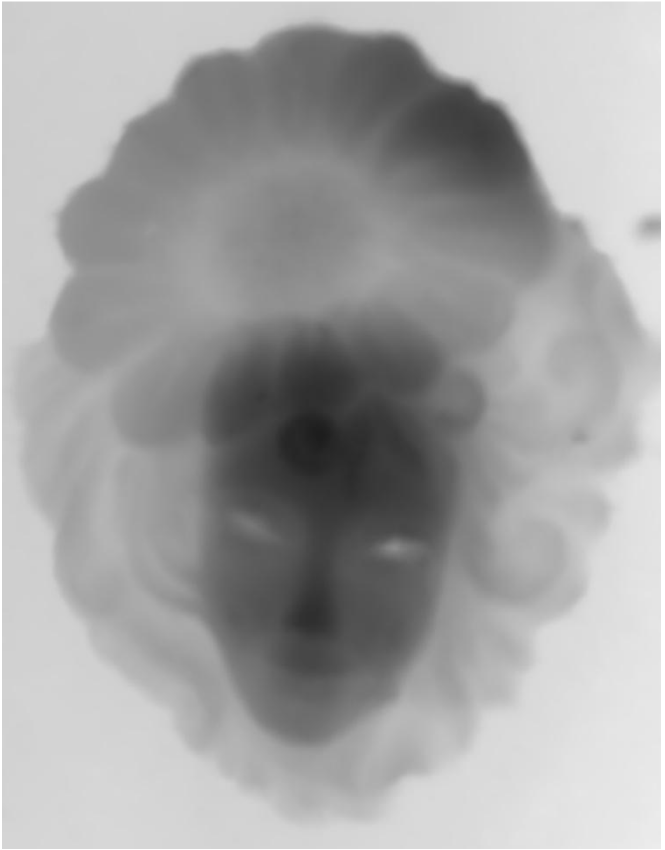
Block Matching



Filtered

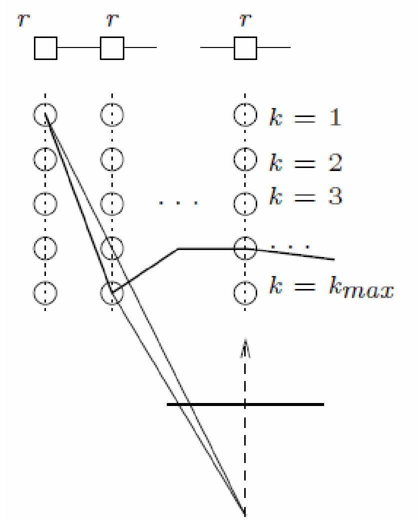
The **disparity maps** are estimated for both the left and the right image. Those pixels are left out whose “partners have another partner”.

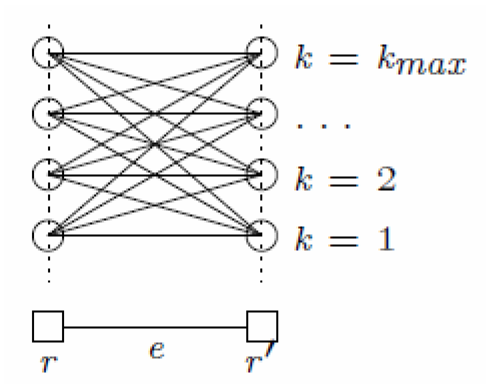
Block matching



- is very simple
- is very efficient (thanks to the **integral image** approach!)
- can be easily parallelized
- can be used to estimate non-dense disparity maps
- can serve as a reasonable initialization for more elaborated techniques
  ← see

Row-wise approaches





Certain combinations of the depth values in the neighboring (along the horizontal direction) pixels are impossible.

→ Introduce a function that forbids (penalizes) "wrong" pairs

Solve for each row:

$$y^* = \arg\min_y \left[\sum_{i=1}^{n} q_i(y_i) + \sum_{i=2}^{n} g(y_{i-1}, y_i) \right]$$

(Energy Minimization on a chain)

Row-wise approaches

Solve for each row:

$$y^* = \arg\min_y \left[\sum_{i=1}^{n} q_i(y_i) + \sum_{i=2}^{n} g(y_{i-1}, y_i) \right]$$

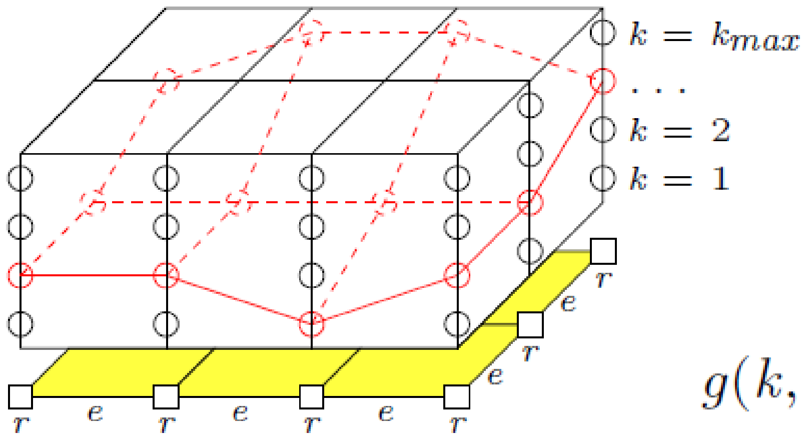
Dynamic programming

G. Gimel'farb (long time ago)

Energy Minimization

There are functions which penalize pairs of depth values for both vertical and horizontal directions. The task reads:

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



Some popular choices:

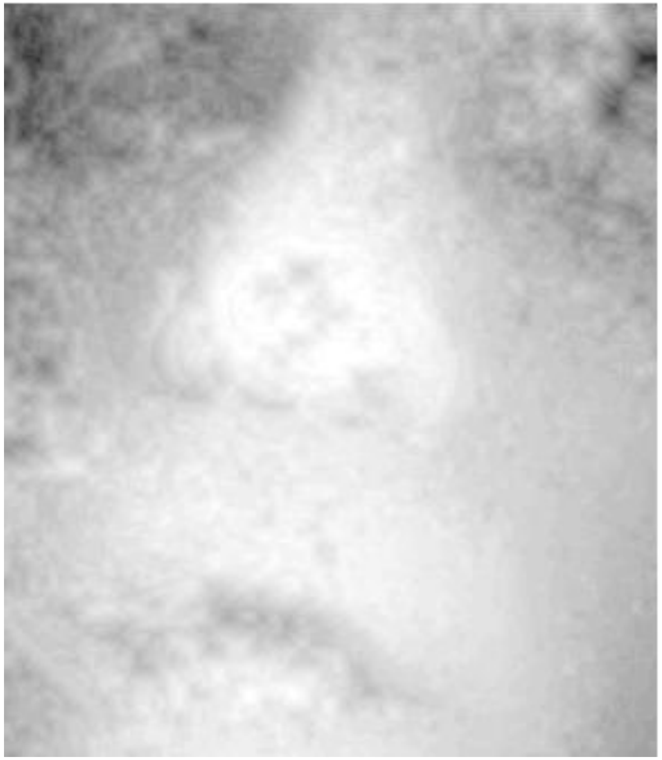
$$g(k, k') = \begin{cases} 0 & wenn \quad |k - k'| \leq \delta \\ \infty & sonst \end{cases}$$

$$g(k, k') = c \cdot (k - k')^2$$

$$g(k, k') = \begin{cases} 0 & wenn \quad k = k' \\ a > 0 & sonst \end{cases}$$

Energy Minimization

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



NP-complete in general

Polynomial solvable for some energies

Efficient approximations for some others

Boykov, Kolmogorov, Veksler, Zabih, around 2001

α-expansion, α-β-swap

Combinations

There is a lot of ways to combine the previous stuff.

For example:

1. Start with Block Matching (or row-wise dynamic programming)

2. Filter out the obviously wrong matches

3. Interpolate depth values in the “unknown” areas

4. Use it as initialization

5. Proceed with an approximate/local approach for the Energy Minimization (e.g. Iterated Conditional Mode, row-wise ICM, α-expansion, α-β-swap etc.)

Statistic models – MRF

The a-posteriori probability distribution of depth maps:

$$p(y) \sim \exp\left[\sum_r q_r(y_r) + \sum_{rr'} g(y_r, y_{r'})\right]$$

Maximum a-posteriori decision is an Energy Minimization task:

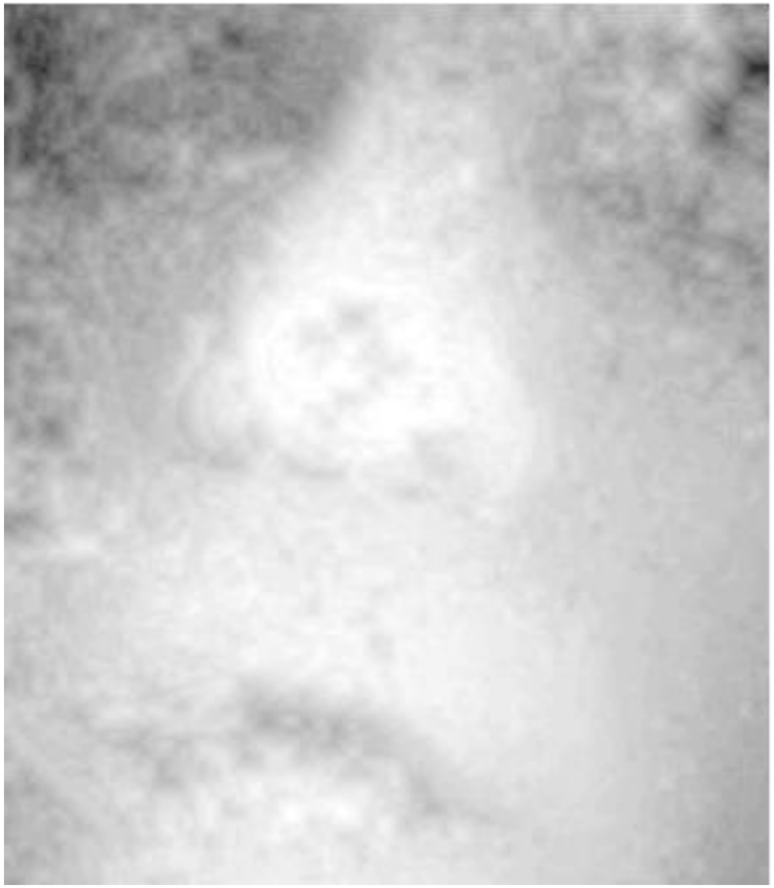
$$y^* = \arg\min_y \left[\sum_r q_r(y_r) + \sum_{rr'} g(y_r, y_{r'})\right]$$

Better cost function (see the Bayesian Decision Theory)
→ another decision strategy – Minimum Marginal Squared Error

$$y_r^* = \sum_k k \cdot p(y_r = k) \quad \forall r$$

Statistic models – MRF

MAP



MMSE

Schlesinger, 2003

Non-rectified stereo

A simple approach:

1. Rectify images (SIFT, RANSAC, apply Homographies etc.)
2. Solve rectified stereo

A better approach: both the depth map and the epipolar geometry are unknowns in a unified model.

Iterate:

1. Depth map → correspondences → epipolar geometry (e.g. 8-point algorithm)
2. Epipolar geometry → rectified stereo → depth map (allow deviations from the actual epipolar geometry)

- Continuous: Valgaerts, Bruhn, Mainberger, Weickert, 2010
- Discrete: Schlesinger, Flach, Shekhovtsov, 2004