Robotic 3D Vision

Lecture 18: Dense Stereo Reconstruction

WS 2020/21

Dr. Niclas Zeller
Artisense GmbH

What We Will Cover Today

- Stereo Rectification
- Dense Depth Reconstruction from Two and Multiple Views
  - Dense Correspondence Search
  - Regularization
- Depth Sensors
  - Structured light
  - Time-of-flight

Stereo Perception

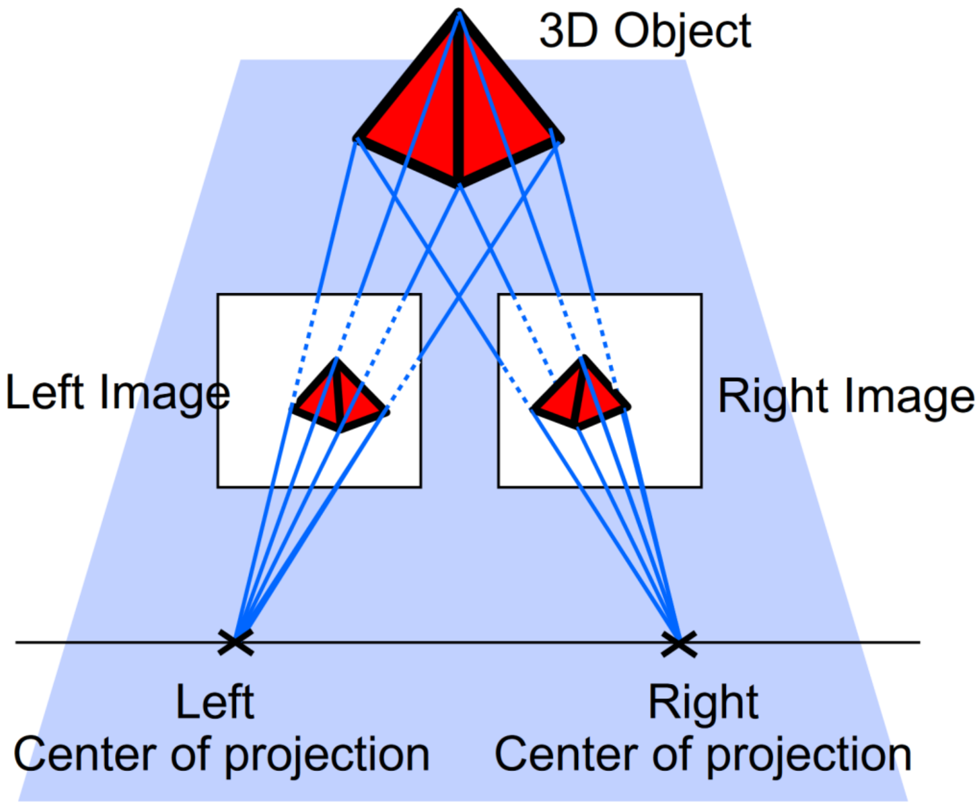
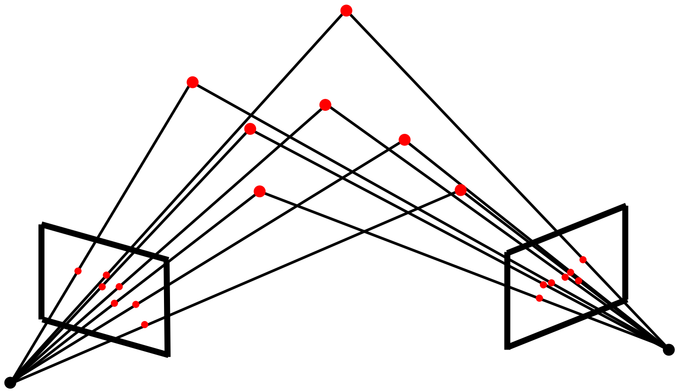


Image credit: D. Scaramuzza

Dense Depth from Two Views

- So far: triangulation of corresponding interest points between two images to find depth
- How can we obtain depth densely for all pixels in an image?
- Assume relative pose between the camera images known
- Assume intrinsic camera calibration known



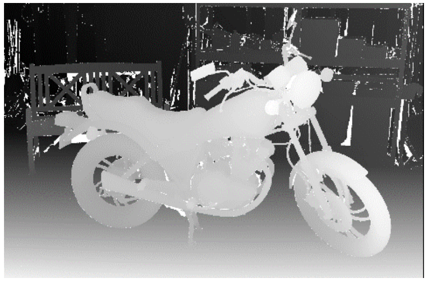
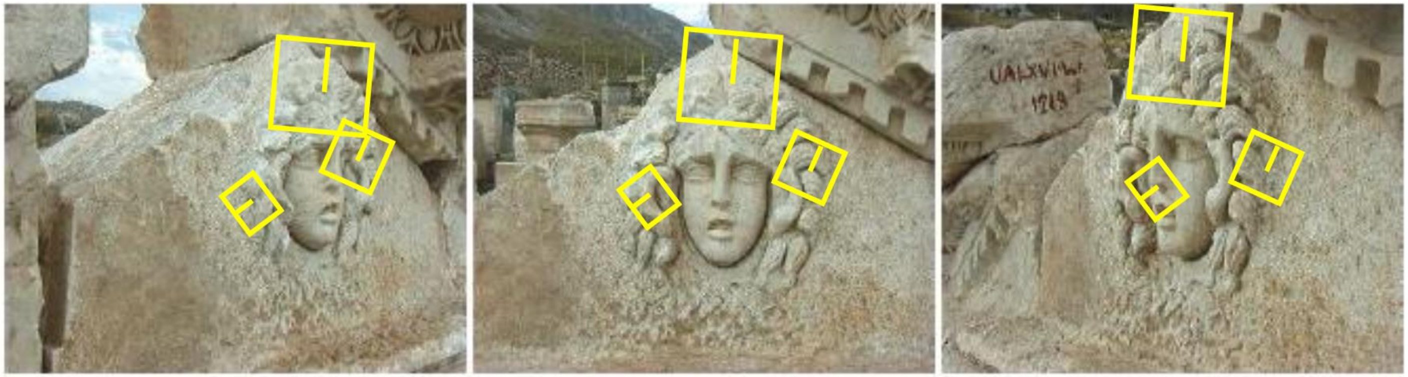


Image source: Scharstein et al., Middlebury stereo benchmark

Sparse 3D Reconstruction



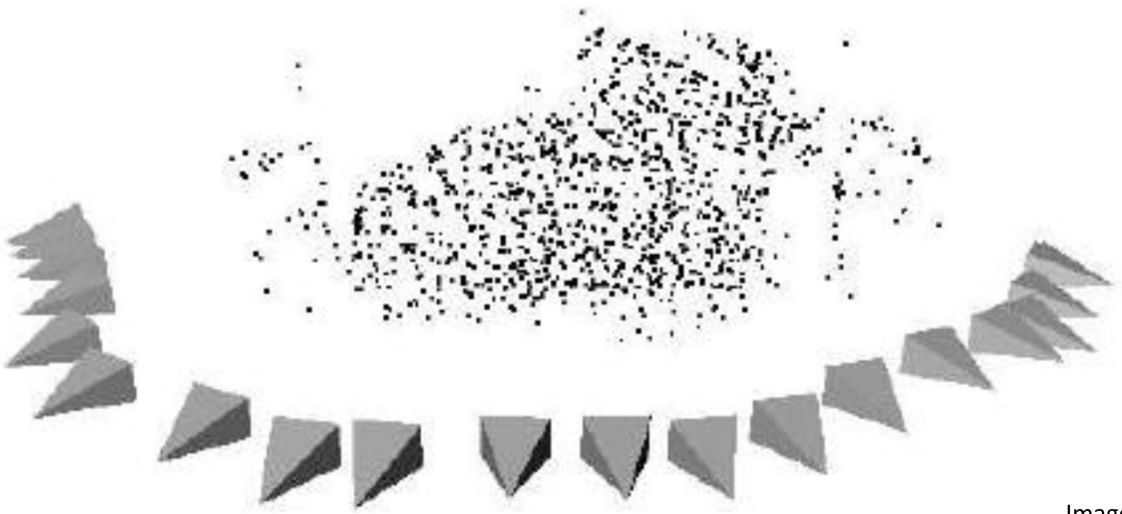
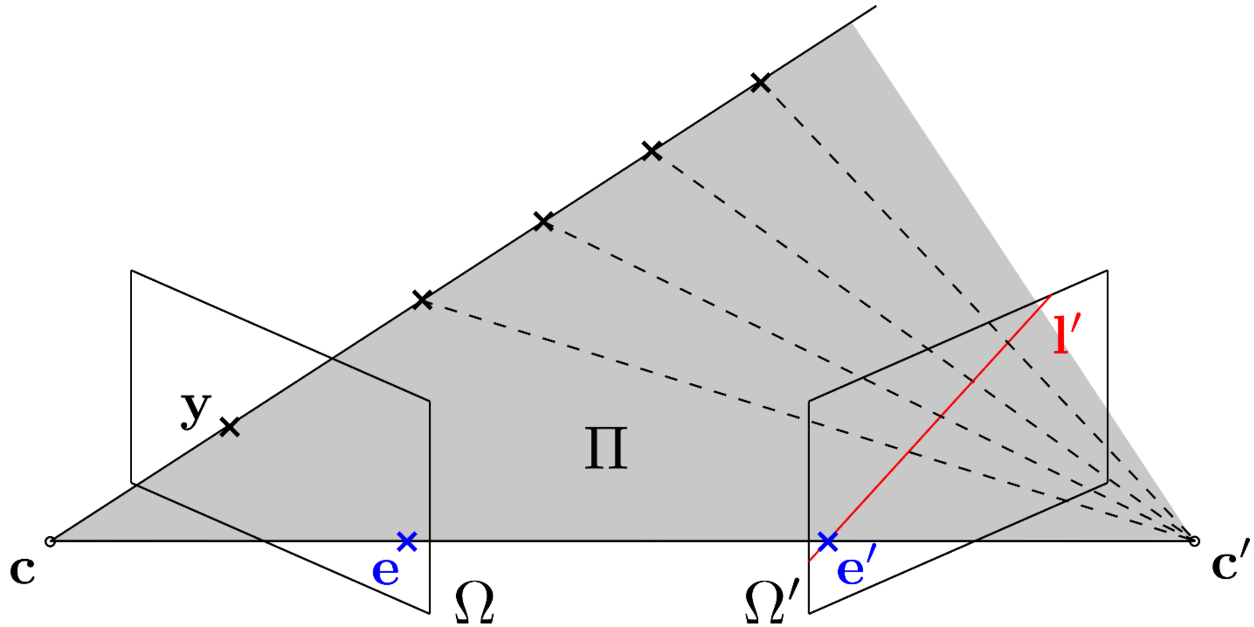


Image credit: D. Scaramuzza

Dense 3D Reconstruction

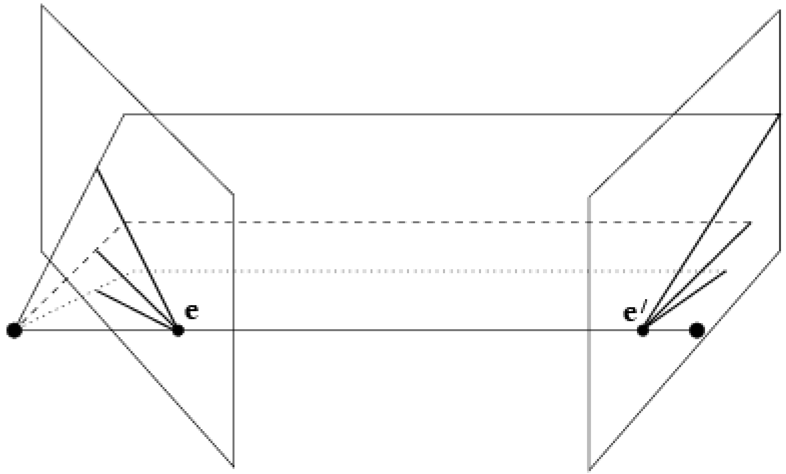
Image credit: D. Scaramuzza

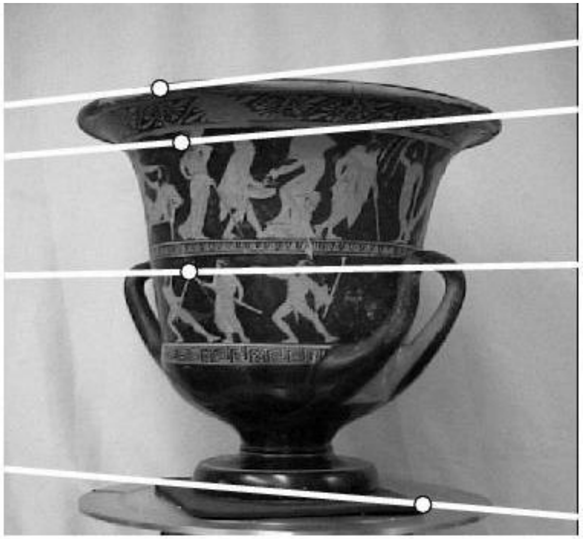
Recap: Epipolar Geometry



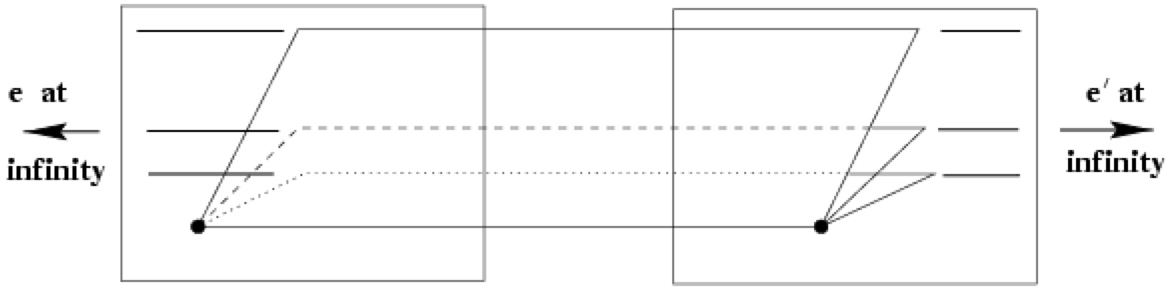
- Camera centers $\mathbf{c}$, $\mathbf{c}'$ and image point $\mathbf{y} \in \Omega$ span the epipolar plane Π
- The ray from camera center $\mathbf{c}$ through point $\mathbf{y}$ projects as the epipolar line $\mathbf{l}'$ in image plane Ω'
- The intersections of the line through the camera centers with the image planes are called epipoles $\mathbf{e}$, $\mathbf{e}'$

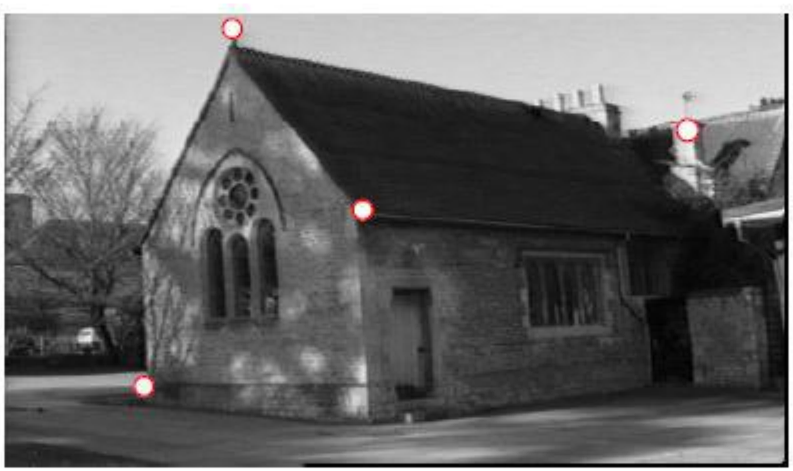
Epipolar Lines, Converging Cameras





Epipolar Lines, Parallel Cameras

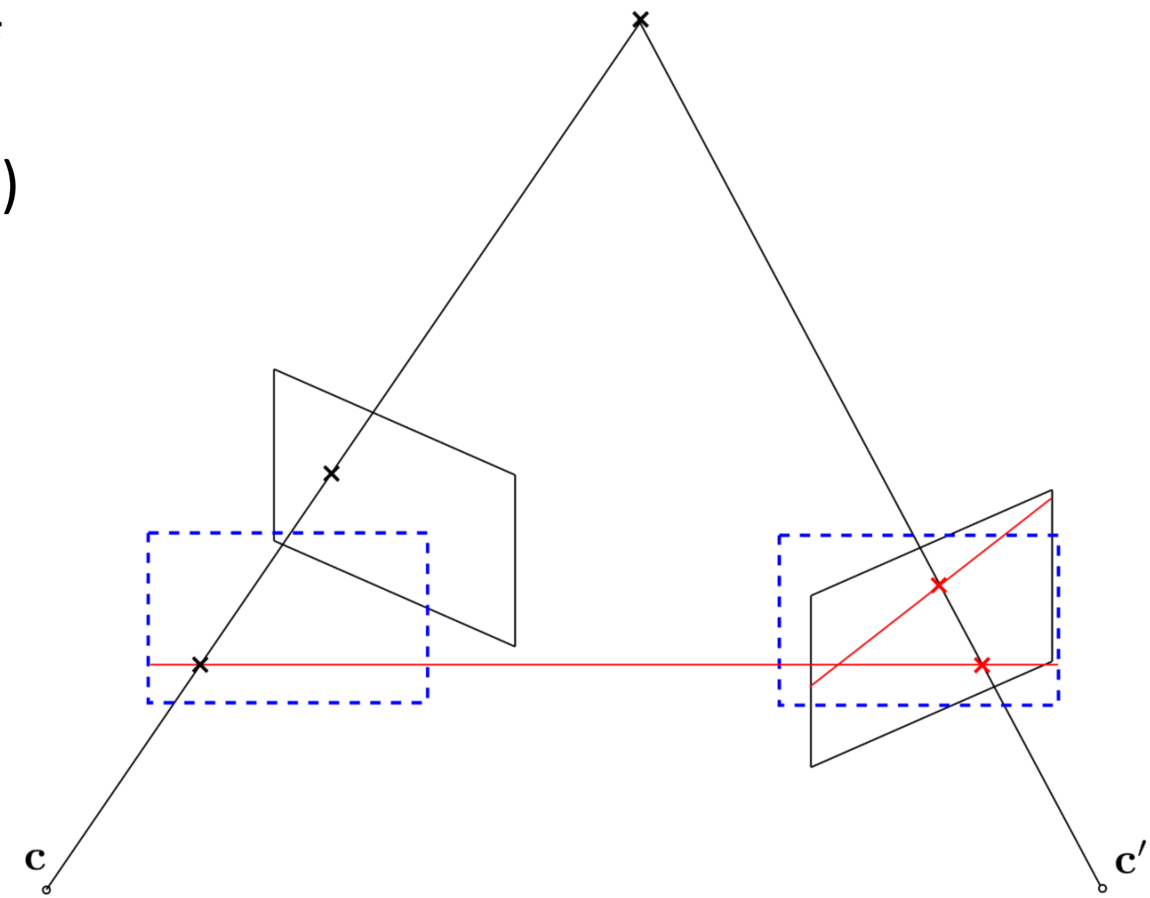




Stereo Image Rectification

- Correspondence search is simplified, if epipolar lines are horizontal (or vertical)

- Idea: Rectify images
  - warp the images onto a common image plane
  - only horizontal or vertical translation between the „new“ camera frames
  - Equal intrinsics



Stereo Rectification (1)

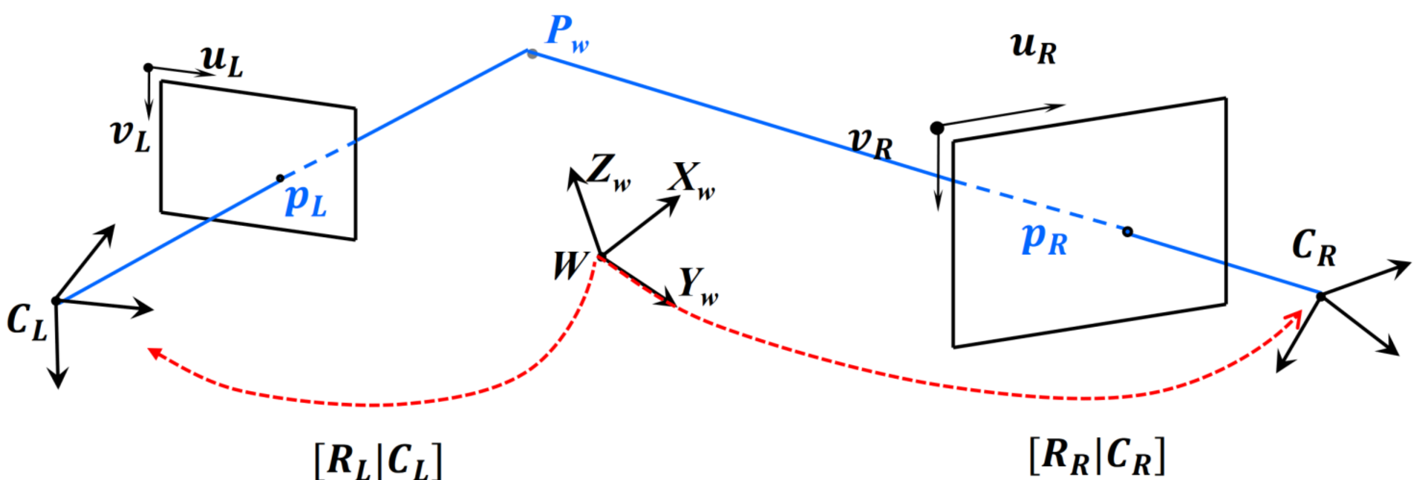
$$\lambda \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = K \begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} \qquad \begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix} = R \begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} + C$$

camera intrinsics matrix

3D point in world frame

3D rotation matrix

3D translation/ camera center in world frame



Slide adapted from D. Scaramuzza

Stereo Rectification (1)

- In the following for convenience, we will write the perspective projection of a 3D point expressed in the world frame into the camera frame as

$$\lambda \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = KR^{-1} \left[\begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix} - C \right]$$

camera intrinsics matrix

3D rotation matrix

3D point in world frame

3D translation/ camera center in world frame

Slide adapted from D. Scaramuzza

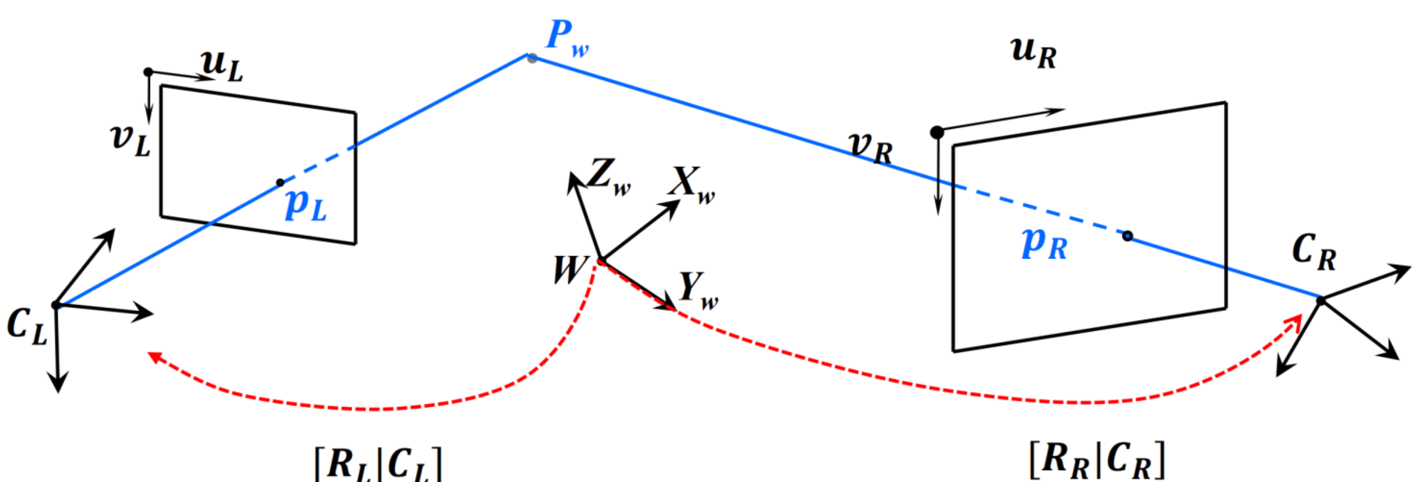
Stereo Rectification (2)

Left camera projection:

$$\lambda_L \begin{bmatrix} u_L \\ v_L \\ 1 \end{bmatrix} = K_L R_L^{-1} \left[\begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix} - C_L \right]$$

Right camera projection:

$$\lambda_R \begin{bmatrix} u_R \\ v_R \\ 1 \end{bmatrix} = K_R R_R^{-1} \left[\begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix} - C_R \right]$$



Slide adapted from D. Scaramuzza

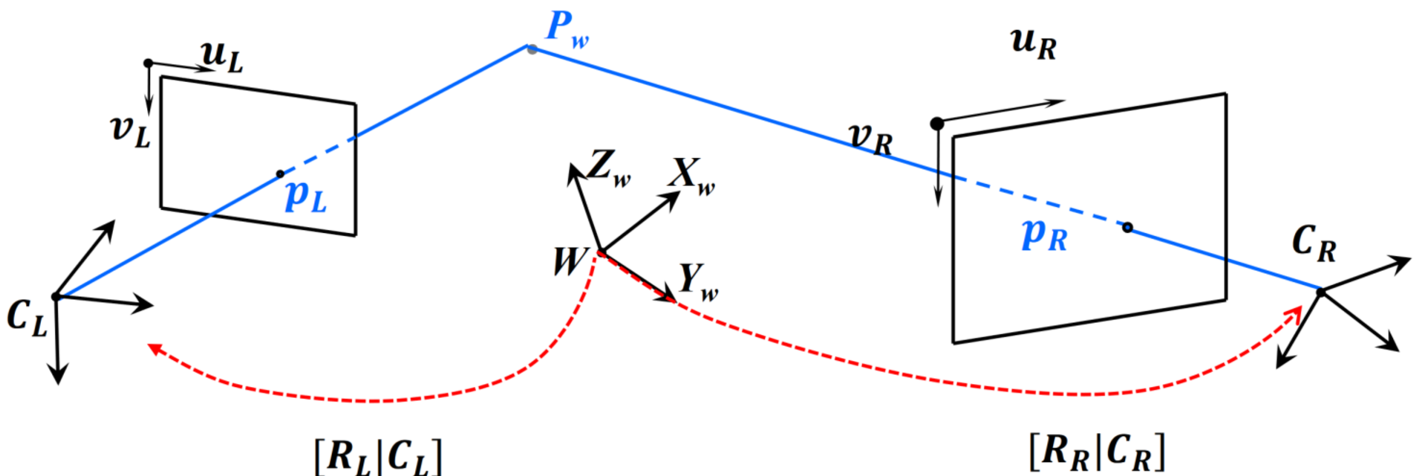
Stereo Rectification (3)

Goal: warp left and right images such that image planes coplanar and intrinsics are equal

Old Left camera	New Left camera	Old Right camera	New Right camera

$$\lambda_L \begin{bmatrix} u_L \\ v_L \\ 1 \end{bmatrix} = K_L R_L^{-1} \left[\begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix} - C_L \right] \rightarrow \bar{\lambda}_L \begin{bmatrix} \bar{u}_L \\ \bar{v}_L \\ 1 \end{bmatrix} = \bar{K}\,\bar{R}^{-1} \left[\begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix} - C_L \right]$$

$$\lambda_R \begin{bmatrix} u_R \\ v_R \\ 1 \end{bmatrix} = K_R R_R^{-1} \left[\begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix} - C_R \right] \rightarrow \bar{\lambda}_R \begin{bmatrix} \bar{u}_R \\ \bar{v}_R \\ 1 \end{bmatrix} = \bar{K}\,\bar{R}^{-1} \left[\begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix} - C_R \right]$$



Slide adapted from D. Scaramuzza

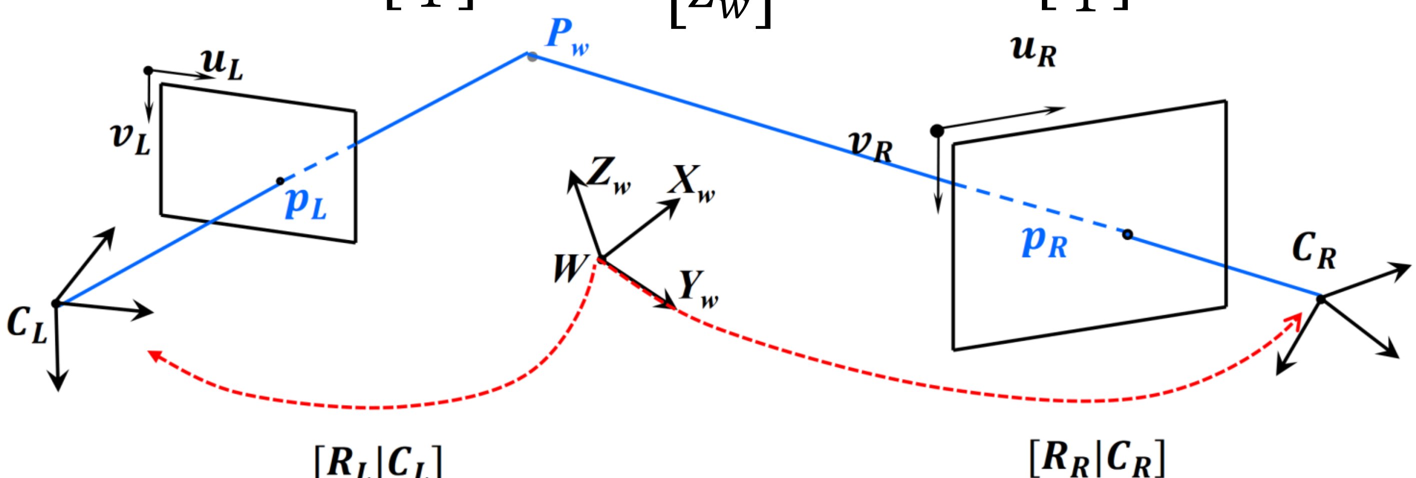
Stereo Rectification (3)

Solving for 3D point for each camera yields homographies

Left camera:

$$\lambda_L \begin{bmatrix} u_L \\ v_L \\ 1 \end{bmatrix} = K_L R_L^{-1} \left[\begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix} - C_L \right] \quad \rightarrow \quad \bar{\lambda}_L \begin{bmatrix} \bar{u}_L \\ \bar{v}_L \\ 1 \end{bmatrix} = \bar{K} \bar{R}^{-1} \left[\begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix} - C_L \right]$$

$$\lambda_L R_L K_L^{-1} \begin{bmatrix} u_L \\ v_L \\ 1 \end{bmatrix} + C_L = \begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix} = \bar{\lambda}_L \bar{R} \bar{K}^{-1} \begin{bmatrix} \bar{u}_L \\ \bar{v}_L \\ 1 \end{bmatrix} + C_L$$



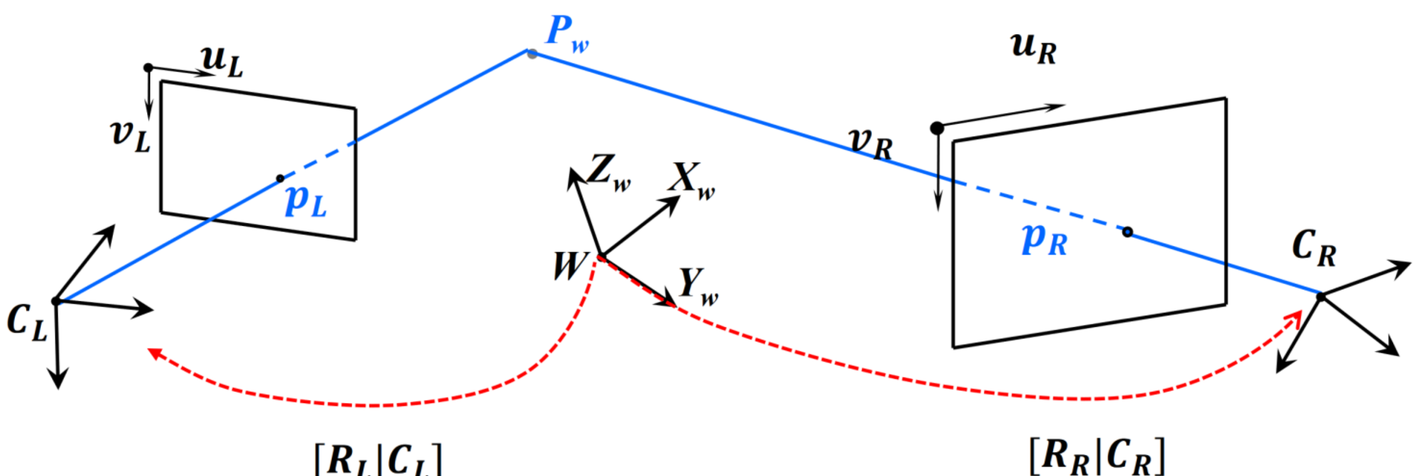
Slide adapted from D. Scaramuzza

Stereo Rectification (4)

Solving for 3D point for each camera yields homographies

$$\bar{\lambda}_L \begin{bmatrix} \bar{u}_L \\ \bar{v}_L \\ 1 \end{bmatrix} = \lambda_L \underbrace{\bar{K}\,\bar{R}^{-1}\,\mathrm{R_L}\,\mathrm{K_L}^{-1}}_{\substack{\text{Homography} \\ \text{Left Camera}}} \begin{bmatrix} u_L \\ v_L \\ 1 \end{bmatrix}$$

$$\bar{\lambda}_R \begin{bmatrix} \bar{u}_R \\ \bar{v}_R \\ 1 \end{bmatrix} = \lambda_R \underbrace{\bar{K}\,\bar{R}^{-1}\,\mathrm{R_R}\,\mathrm{K_R}^{-1}}_{\substack{\text{Homography} \\ \text{Right Camera}}} \begin{bmatrix} u_R \\ v_R \\ 1 \end{bmatrix}$$



Slide adapted from D. Scaramuzza

Stereo Rectification (5)

- How to choose the new intrinsics and rotation ?
- Fusiello et al., A Compact Algorithm for Rectification of Stereo Pairs, Mach. Vision and Appl. 1999
- Choose $\bar{K} = (K_L + K_R)/2$

$$\bar{R} = [\bar{r_1}, \bar{r_2}, \bar{r_3}]$$

where

$$\bar{r_1} = \frac{C_R - C_L}{||C_R - C_L||}$$

Vector $C_R - C_L$ is supposed to be aligned with the x-axis of the camera coordinate frame

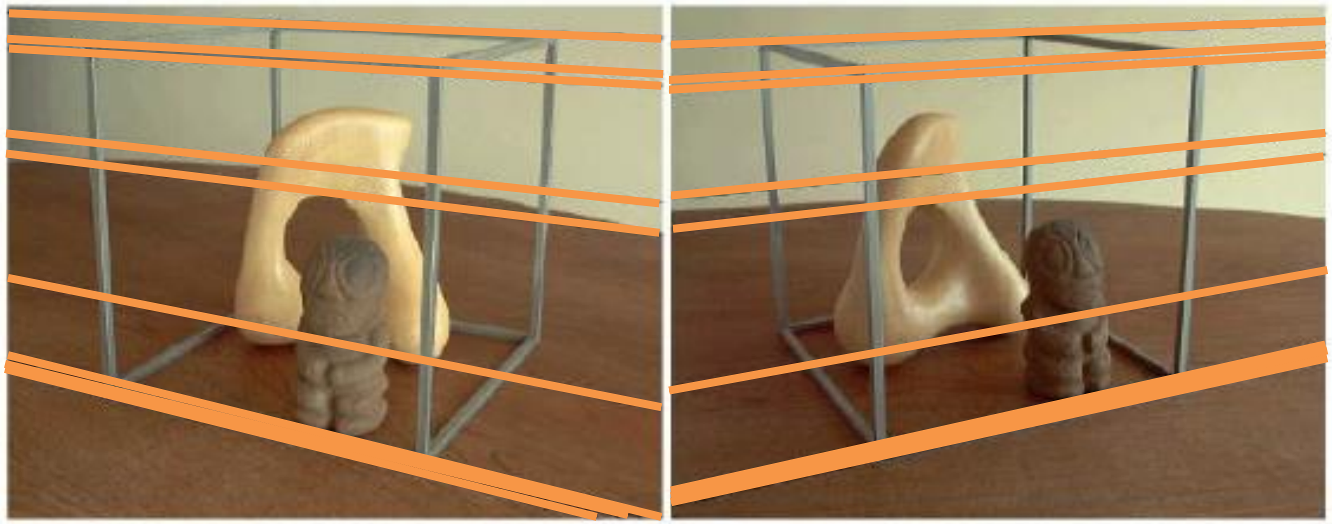
$$\bar{r_2} = r_3 \times \bar{r_1}$$

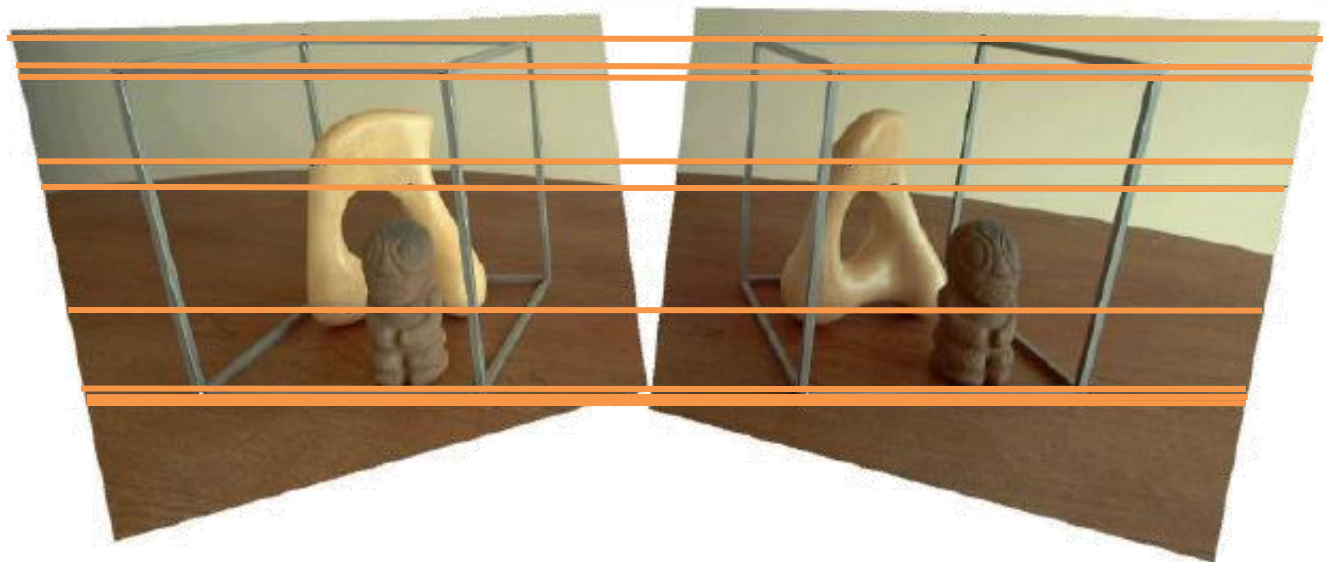
, where r_3 is the 3rd column of the rotation matrix of the left camera, i.e., R_L

$$\bar{r_3} = \bar{r_1} \times \bar{r_2}$$

Slide adapted from D. Scaramuzza

Stereo Rectification Example



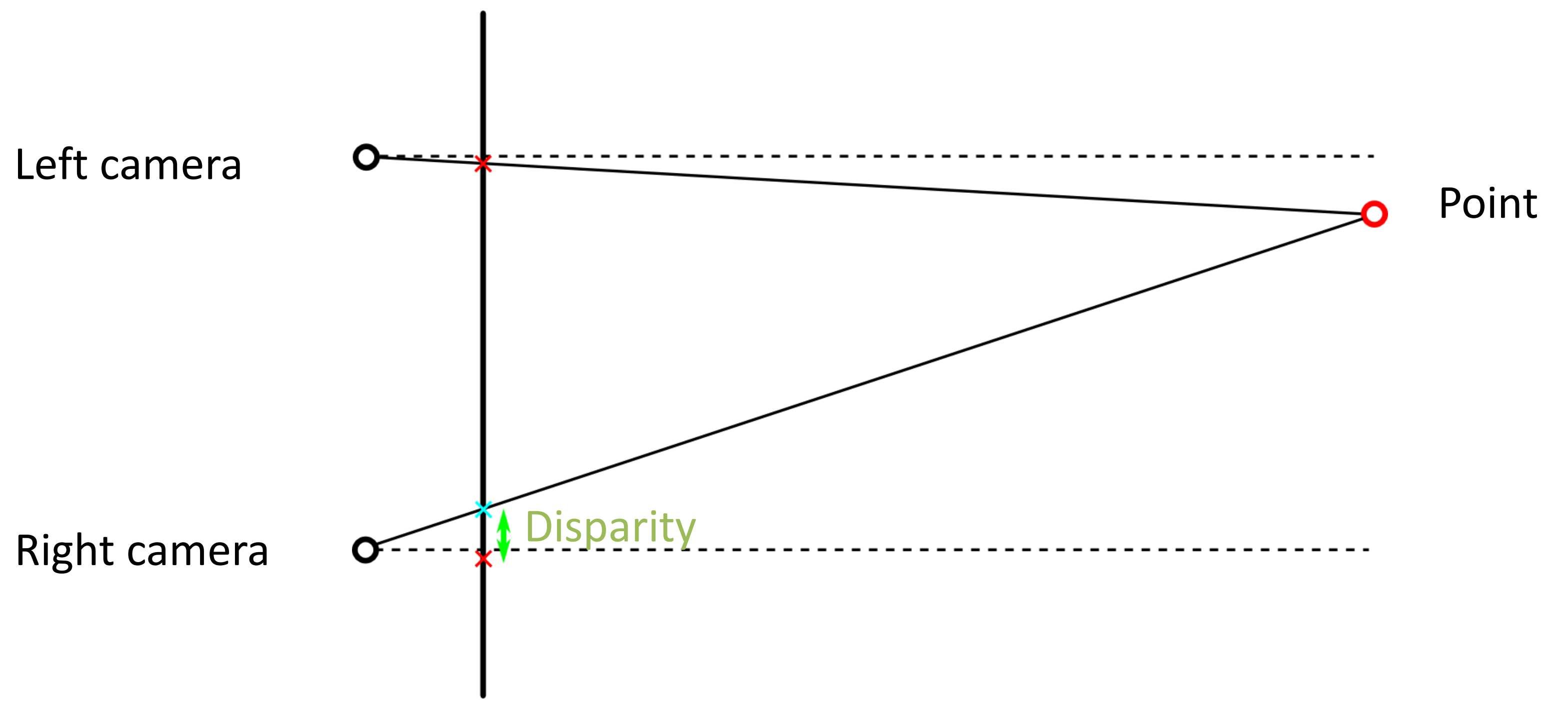


rectified

Image source: Loop and Zhang, 2001

Disparity

- Assume rectified stereo images
- Disparity: (horizontal) pixel difference of corresponding pixels between the two images

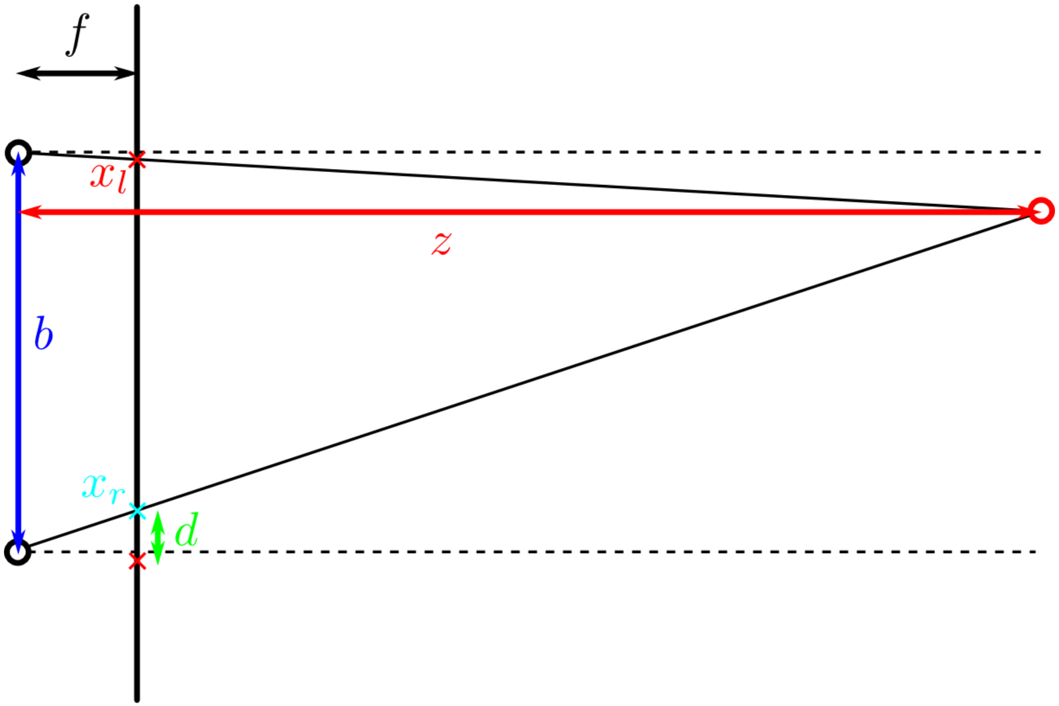


Relation of Disparity and Depth

Similar triangles:

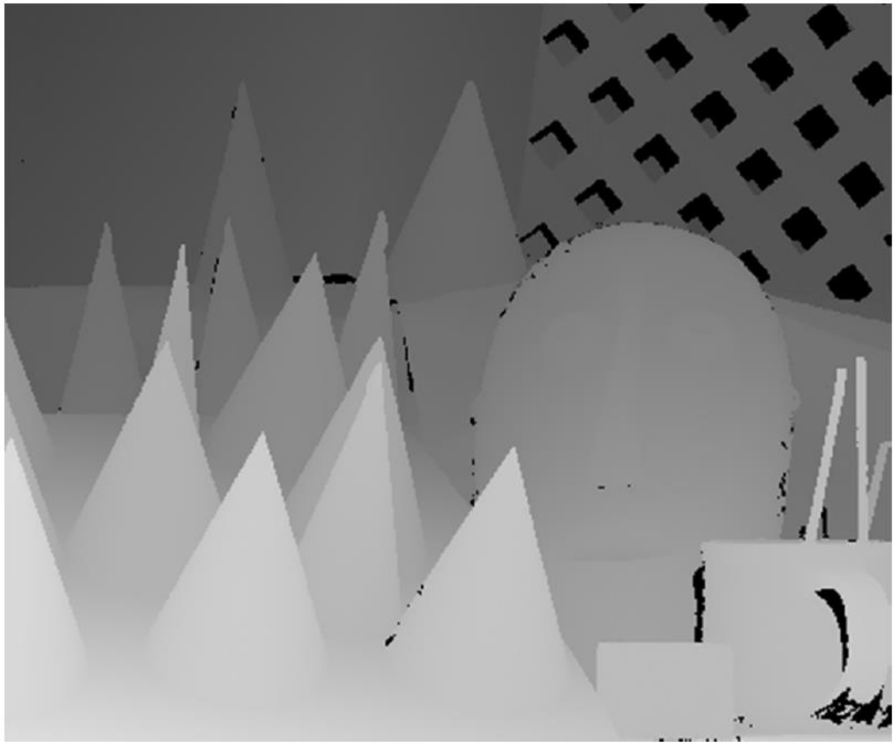
$$\frac{b}{z} = \frac{b-d}{z-f}$$

$$\rightarrow \quad d = \frac{bf}{z}$$



- Disparity is inverse proportional to depth:
  - The larger the depth, the smaller the disparity
- Disparity is proportional to the baseline:
  - The larger the baseline, the larger the disparity
  - Larger baseline means also higher depth accuracy

Relation of Disparity and Depth

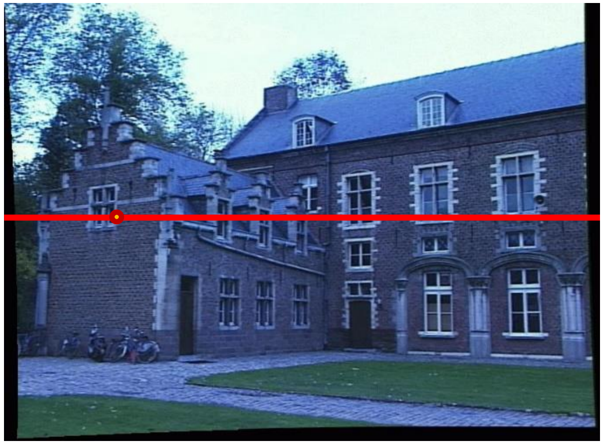


Disparity image

Depth image

Dense Stereo Depth Estimation

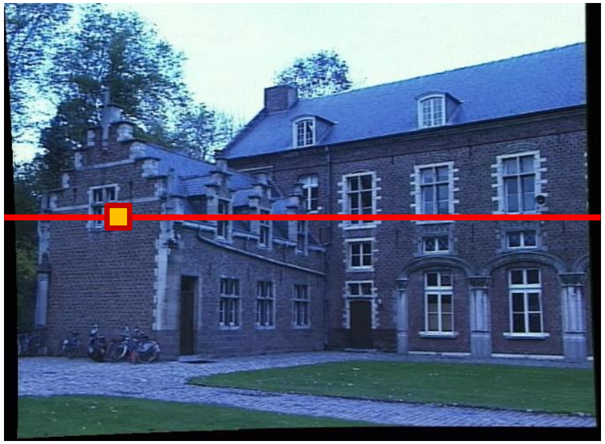
- For each pixel in left image:
  - Compare photoconsistency with every pixel on the corresponding epipolar line in the right image
  - Pick pixel with best similarity

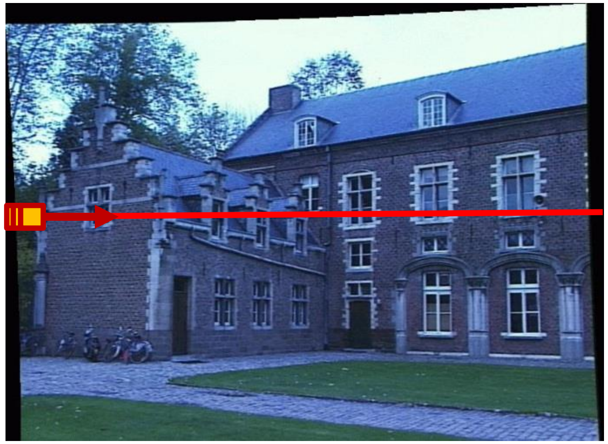


- Problems:
  - Noise
  - Intensity of a single pixel not very distinctive

Dense Stereo Depth Estimation

- Better idea: Compare patches (blocks)





- New questions:
  - What are good patch correlation measures?
  - Patch size?
  - etc.

Block Matching Algorithm

- Input: Two images, intrinsics camera calibration, relative pose
- Output: Disparity image
- Algorithm:
  - Rectify images
  - For each pixel in left image:
    - Compute matching cost along epipolar line using patch comparison
    - Determine minimum in matching cost
      - with sub-pixel accuracy, e.g. using linear interpolation
  - Filter outliers

Patch Correlation Measures

If we consider rectified left/right images we don‘t have to search along the y dimension $\Delta y = 0$

- Sum-of-squared differences:

$$\mathrm{SSD}(B, (\Delta x, \Delta y)) = \sum_{(x,y)\in B} \left(I^l(x,y) - I^r(x+\Delta x, y+\Delta y)\right)^2$$

block/window

- Sum-of-absolute differences:

$$\mathrm{SAD}(B, (\Delta x, \Delta y)) = \sum_{(x,y)\in B} \left|I^l(x,y) - I^r(x+\Delta x, y+\Delta y)\right|$$

Less sensitive to outliers

- Normalized Cross-Correlation:

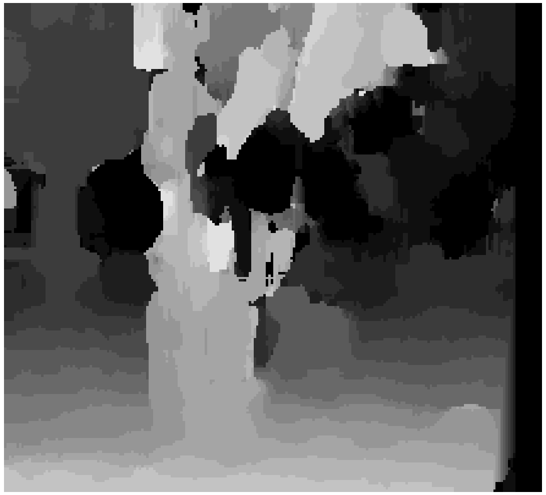
$$\mathrm{NCC}\big(B, (\Delta x, \Delta y)\big) = \frac{\sum_{(x,y)\in B} I^l(x,y) I^r(x+\Delta x, y+\Delta y)}{\sqrt{\sum_{(x,y)\in B} I^l(x,y)^2}\sqrt{\sum_{(x,y)\in B} I^r(x+\Delta x, y+\Delta y)^2}}$$

Invariant to illumination changes

Block Size

- Common choices are 5x5, 11x11, …
  - Smaller neighborhood: more details
  - Larger neighborhood: less noise
- Suppress pixels with low confidence (f.e. check ratio best match vs. second best match, examine local behavior of matching cost, etc.)

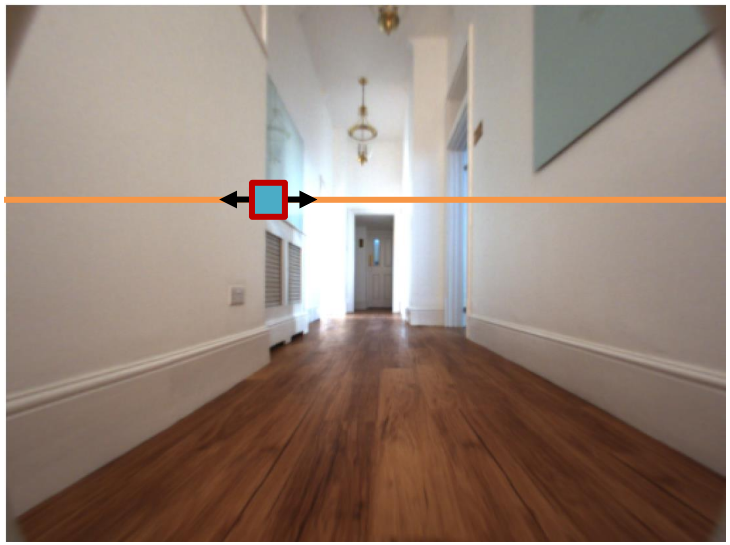
3x3 block-size

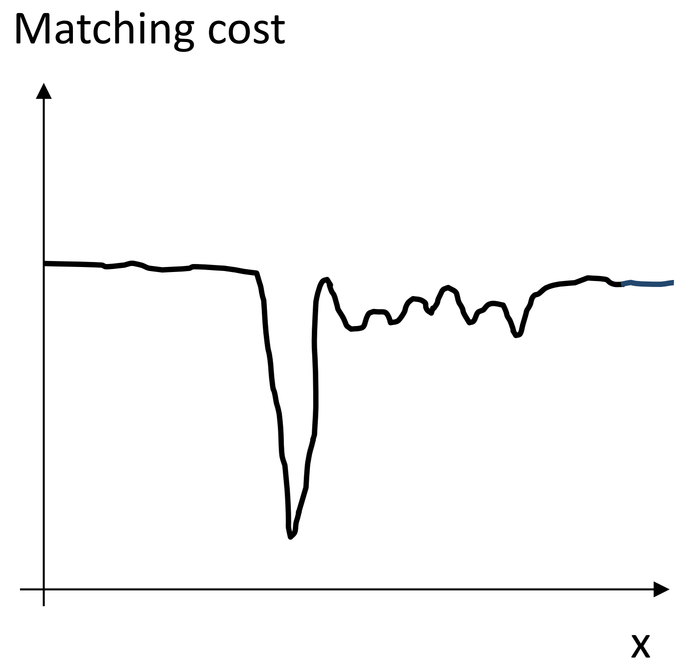


20x20 block-size

Images: R. Szeliski

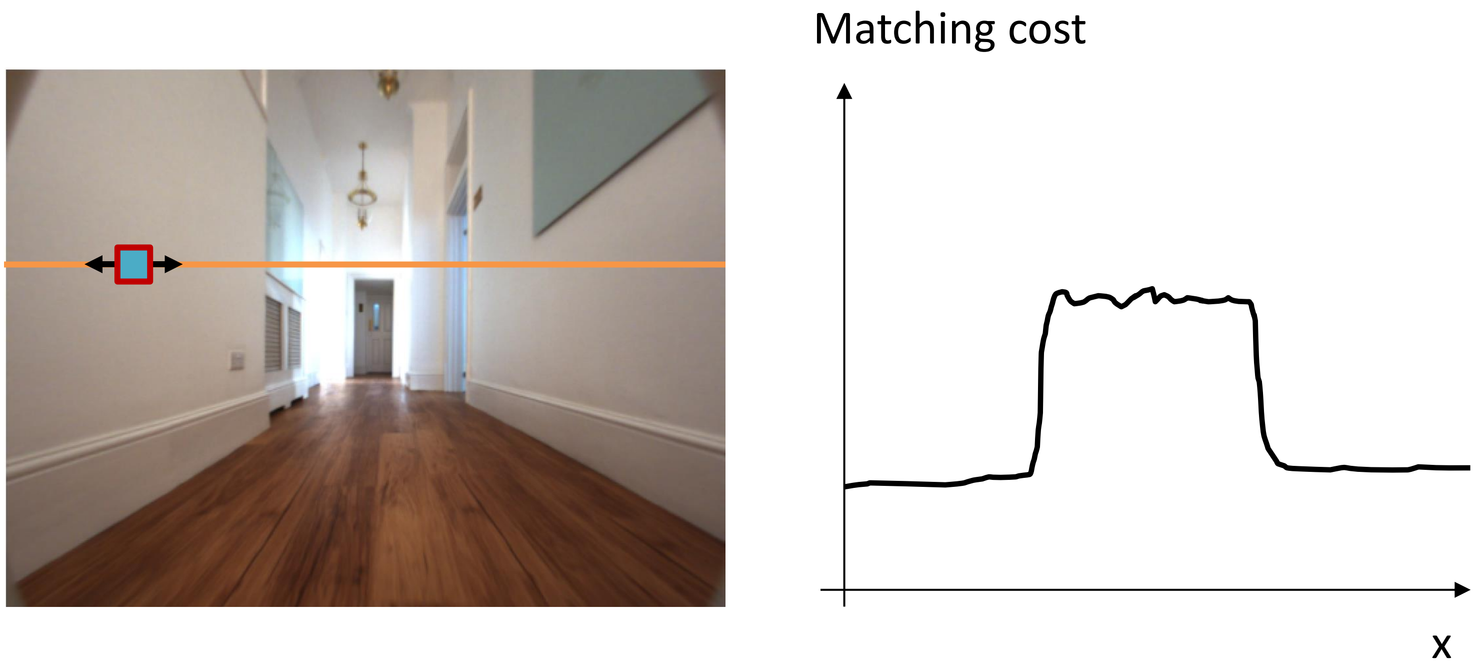
Behavior of the Correspondence Measure





Images: Pinies et al., 2015

Behavior of the Correspondence Measure



Images: Pinies et al., 2015

Challenges for Dense Correspondence Search

- Corresponding patches may differ !

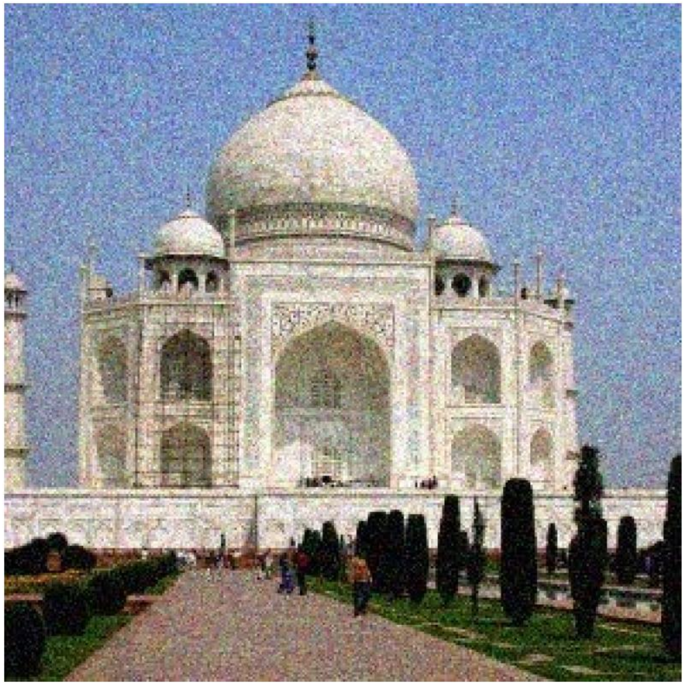


Image Noise
(Camera-related)

Images: C. Gava

Challenges for Dense Correspondence Search

- Corresponding patches may differ !

Perspective Distortion
(Viewpoint-related)

Images: R. Szeliski

Challenges for Dense Correspondence Search

- Corresponding patches may differ !

Occlusions
(Viewpoint-related)

Images: Middlebury benchmark

Challenges for Dense Correspondence Search

- Corresponding patches may differ !

Specular Reflections
(Viewpoint-related)

Images: Weinmann et al., ICCV 2013

Challenges for Dense Correspondence Search

- Correspondence can be ambiguous !

Low Texture
(Scene-related)

Images: Pinies et al., 2015

Challenges for Dense Correspondence Search

- Correspondence can be ambiguous !

Repetitive Structure/Texture
(Scene-related)

Challenges for Dense Correspondence Search

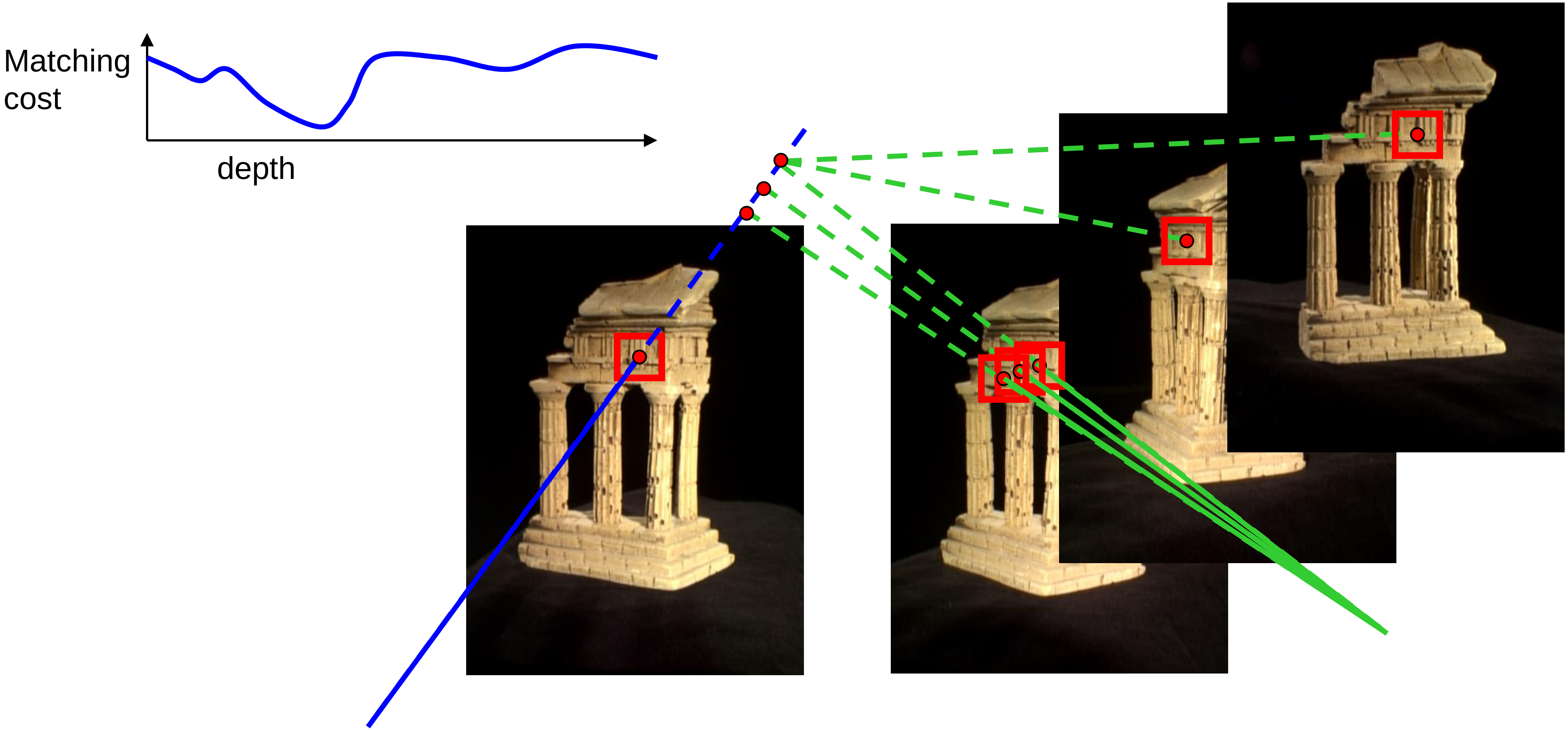
- Corresponding patches may differ !

Motion blur
(Scene-related)

Images: C. Gava

Dense Depth from Multiple Views

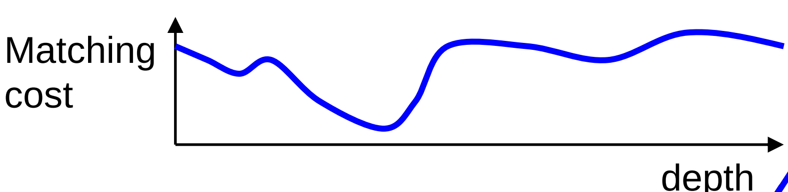
- Straightforward approach: extend two-view matching cost to sum over matching costs of an image towards multiple images

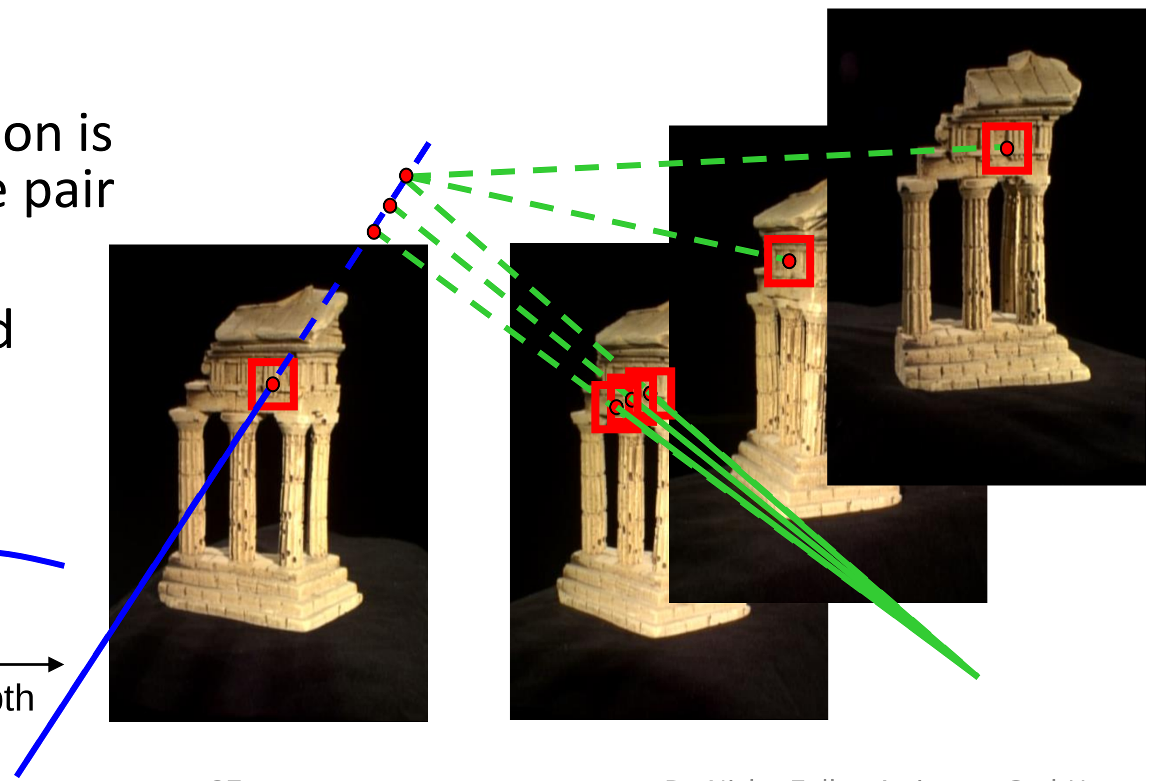


Slide adapted from R. Szeliski

Dense Depth from Multiple Views

- Straightforward approach: extend two-view matching cost to sum over matching costs of an image towards multiple images
- In general for multiple views images cannot be rectified anymore
- Disparity to depth relation is different for each image pair
- Matching cost is defined as a function of depth (or inv. depth)





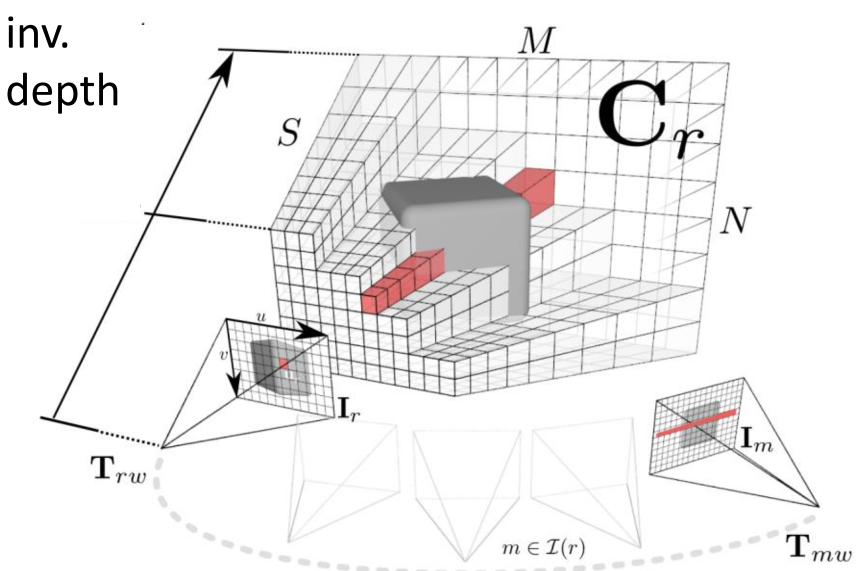
Slide adapted from R. Szeliski

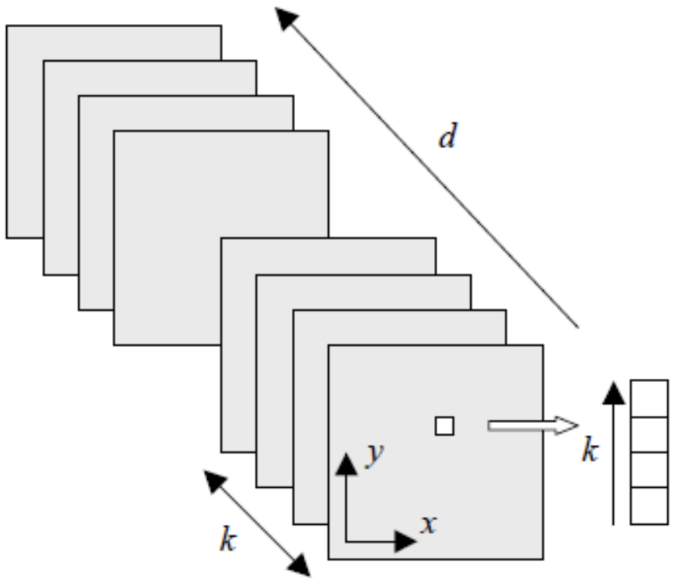
Disparity Space Image / Cost Volumes

- Sum of matching costs between reference and k images for discrete depth hypotheses in each pixel

$$C(\mathbf{y}, d) = \sum_k \rho(I_k(\omega(\mathbf{y}, d, \boldsymbol{\xi}_k)) - I_{ref}(\mathbf{y}))$$

- Multi-view: inv. depth, „cost volume“

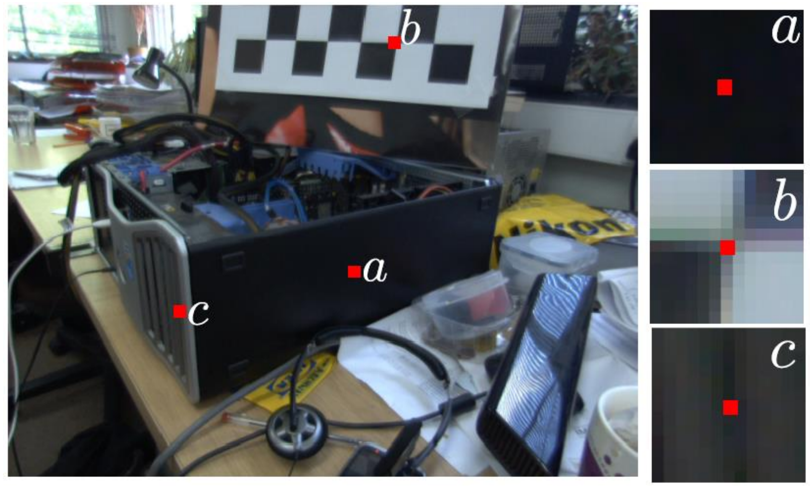


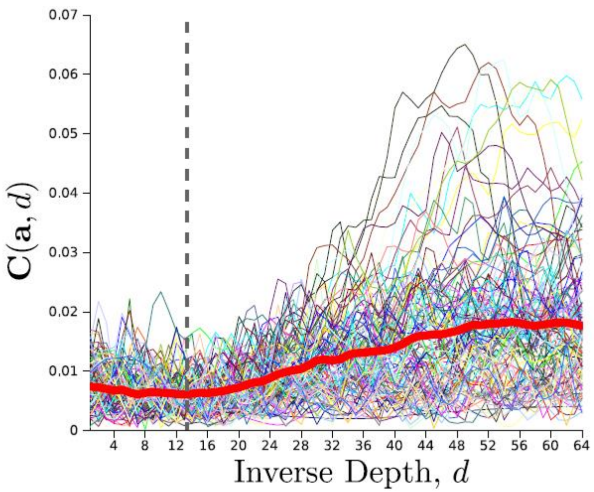


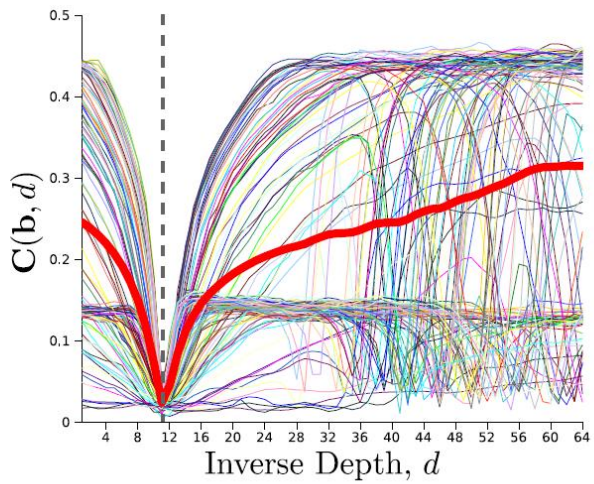
[Szeliski and Golland 1999]

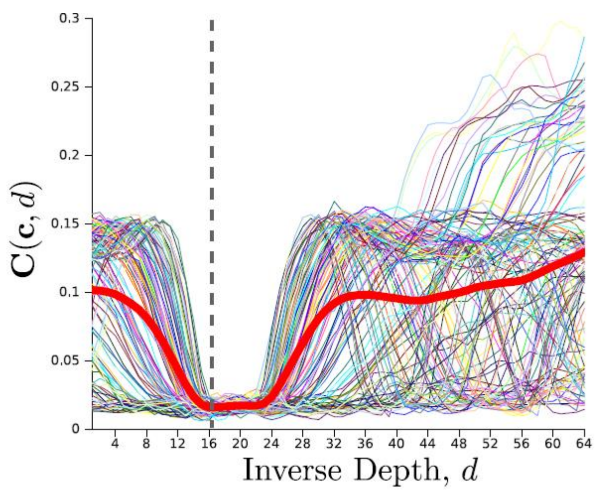
Image from Newcombe et al., 2011

Multi-View Correspondence Measure







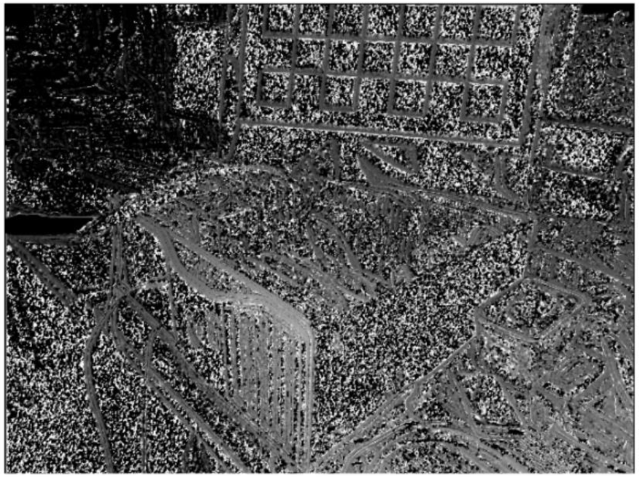


Images: R. Newcombe, 2014

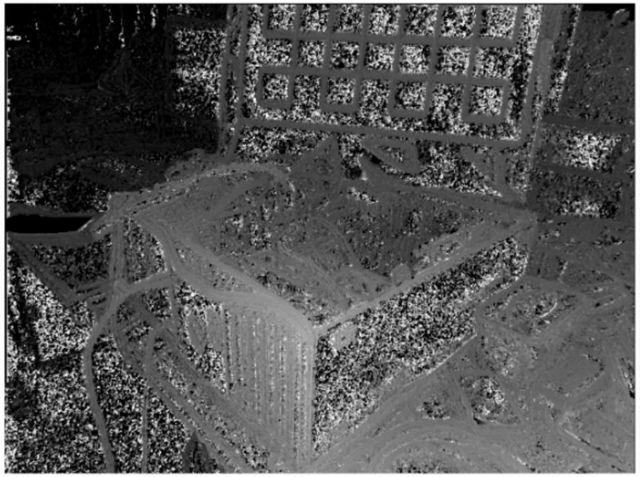
Per-Pixel Max-Likelihood Solution

- Simply choosing the depth with best matching cost at each pixel may not provide a good solution

$$\operatorname{argmin}_d C(\mathbf{y}, d)$$



2 comparison frames



10 comparison frames

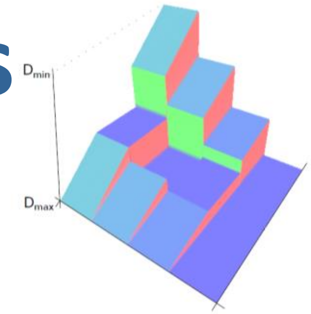
30 comparison frames

- Quite some noise in regions with little texture

Regularization

- Neighboring pixels should not be treated independently from each others
- How can we incorporate prior knowledge about the observed 3D structures such as smoothness or planarity?
- Idea: add regularizing prior term to the optimization problem

Smoothness Regularizers



- Quadratic regularizers

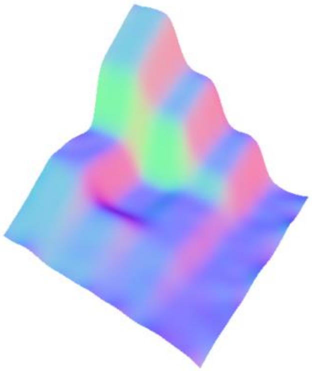
$$E(u) = \frac{1}{2}\int_{\mathbf{y}\in\Omega} \|u(\mathbf{y}) - z(\mathbf{y})\|_2^2 \, d\mathbf{y} + \lambda \frac{1}{2}\int_{\mathbf{y}\in\Omega} \|\nabla u(\mathbf{y})\|_2^2 \, d\mathbf{y}$$

Regularized depth

(multi-view)
Stereo depth

Depth gradient

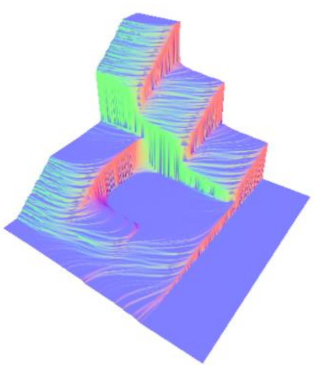
over-
smooth!



- L1-regularizer

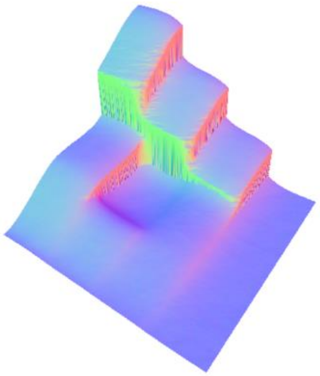
$$E(u) = \int_{\mathbf{y}\in\Omega} \|u(\mathbf{y}) - z(\mathbf{y})\|_1 \, d\mathbf{y} + \lambda \int_{\mathbf{y}\in\Omega} \|\nabla u(\mathbf{y})\|_1 \, d\mathbf{y}$$

Stair-
casing!



Images: R. Newcombe, 2014

Smoothness Regularizers



- Huber-norm regularizer as a trade-off between quadratic and L1

$$E(u) = \int_{\mathbf{y}\in\Omega} \|u(\mathbf{y}) - z(\mathbf{y})\|_{\delta_{\mathcal{F}}}\, d\mathbf{y} + \lambda \int_{\mathbf{y}\in\Omega} \|\nabla u(\mathbf{y})\|_{\delta_{\mathcal{R}}}\, d\mathbf{y}$$

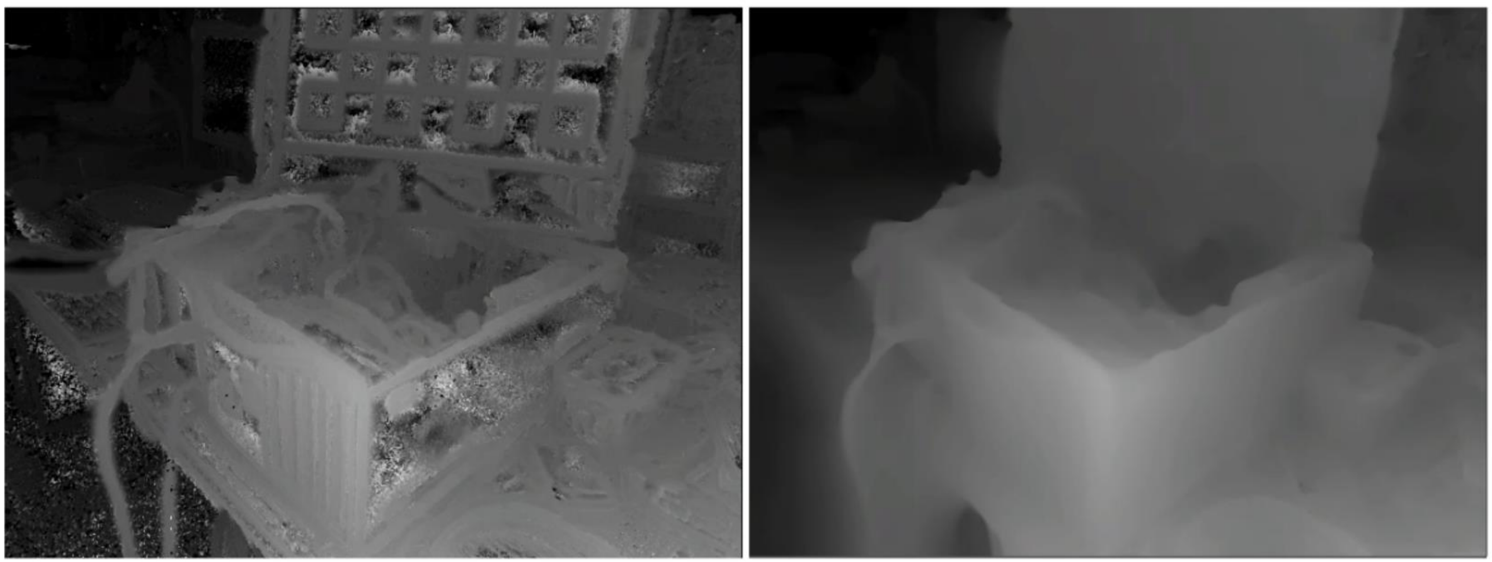
- Optimization is quite complex
  - There exist also discrete approximations
  - E.g. Semi-Global Matching

Images: R. Newcombe, 2014

Effect of Regularization

Data term: cost volume over L1-norm on photometric residuals

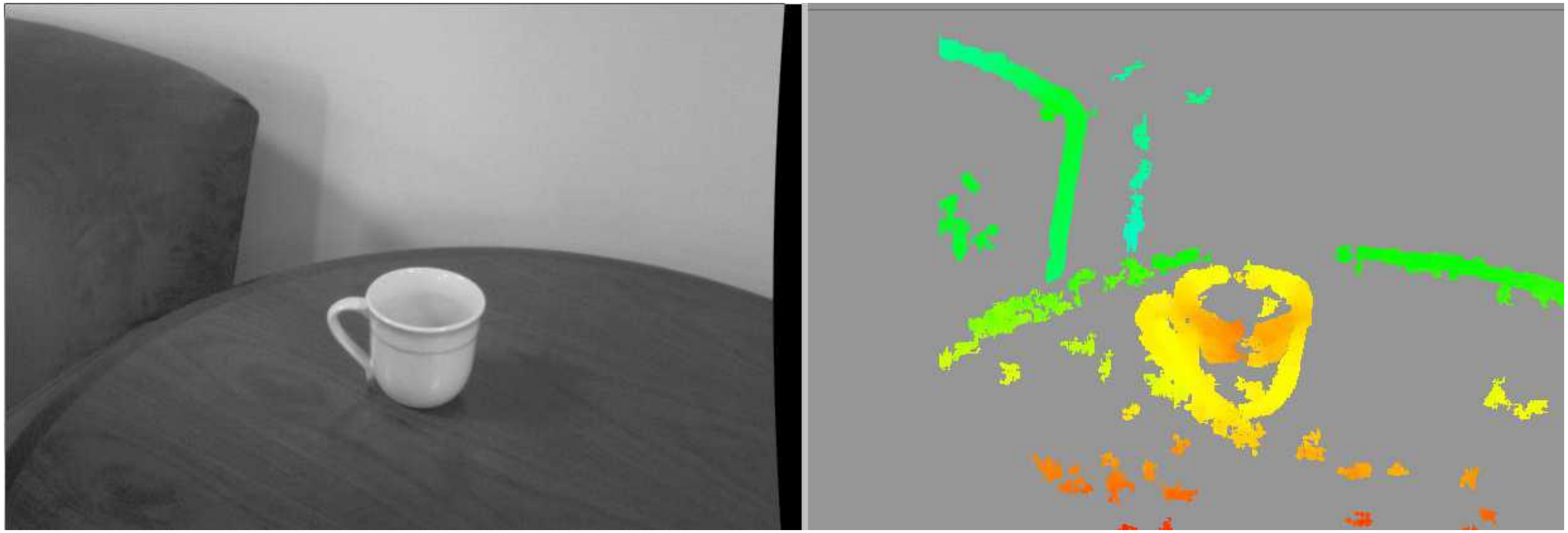
Regularizer: Huber-norm on inverse depth gradient



Images: R. Newcombe et al., 2011

Active Depth Sensing

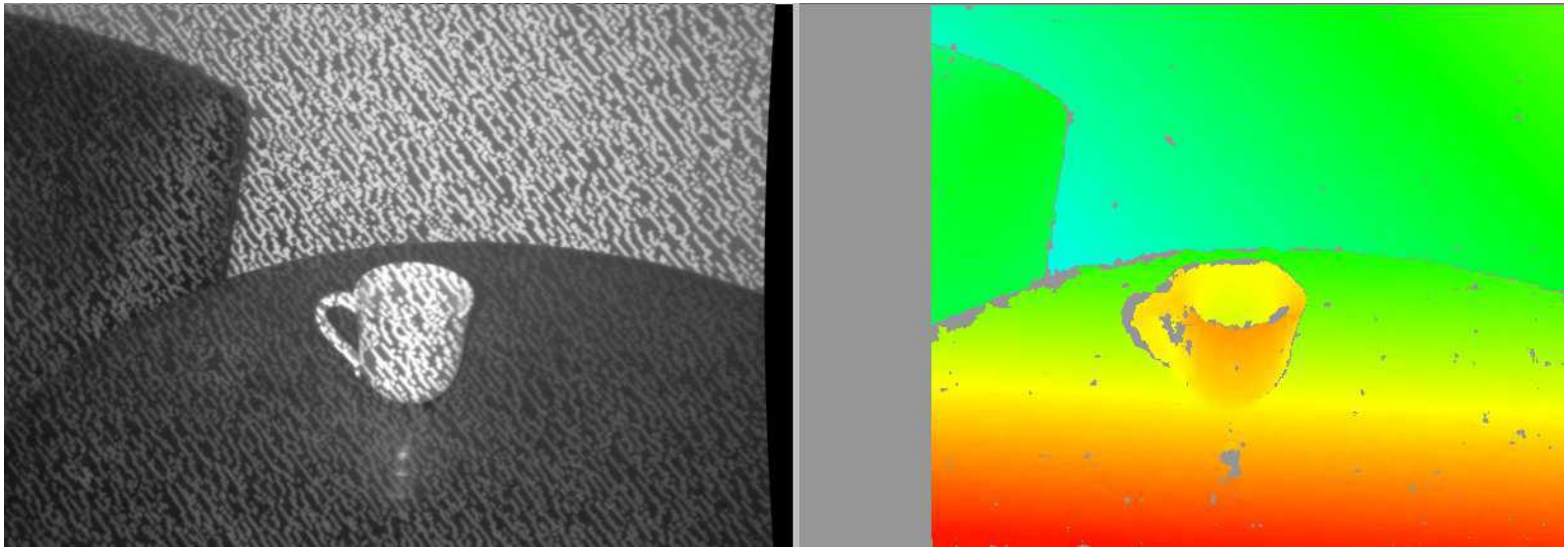
- What can we do about textureless scenes?



Images: J. Sturm

Active Depth Sensing

- Idea: Project light/texture



Images: J. Sturm

Depth Cameras

...

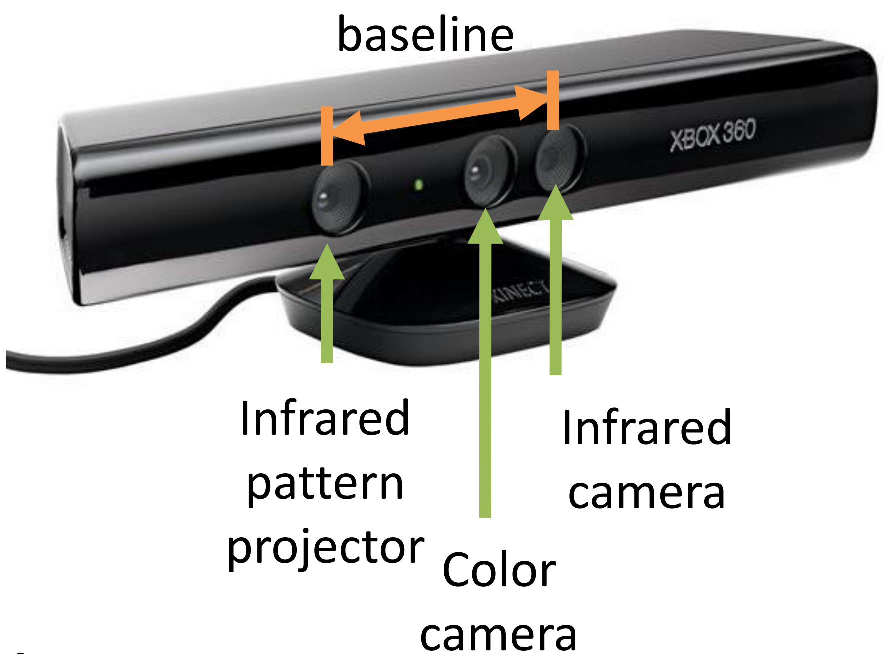
Time-of-Flight

...

Structured Light

Structured Light Measurement Principle

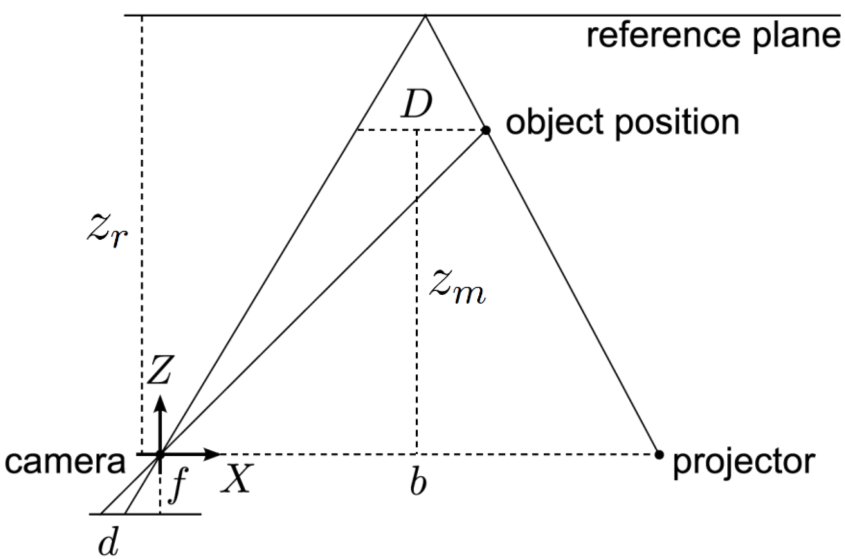
- Project speckle pattern using infrared laser and diffraction element
- Measure infrared speckles using infrared camera
- Measure corresponding RGB image using color camera



Slide adapted from J. Sturm

Structured Light Measurement Principle

- Use known baseline and reference pattern for depth measurement

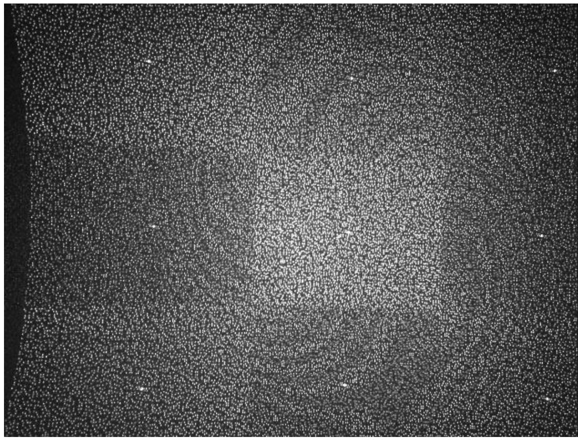


$$\frac{d}{f} = \frac{D}{z_m}$$

$$\frac{D}{z_r - z_m} = \frac{b}{z_r}$$

$$\Rightarrow \quad z_m = \frac{z_r}{\frac{d z_r}{b f} + 1}$$

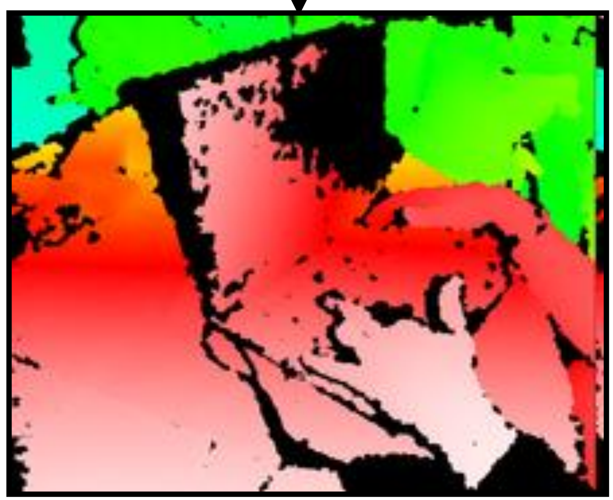
Structured Light Measurement Principle



IR reference pattern

Block matching (9x9)

IR pattern in actual scene

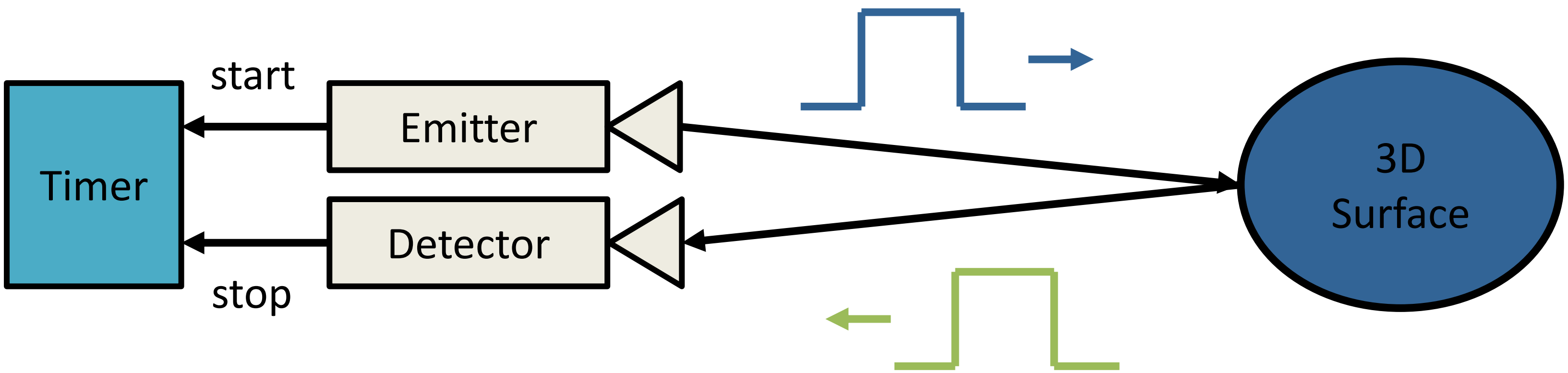


Depth image

Slide adapted from J. Sturm

Time-of-Flight Measurement Principle

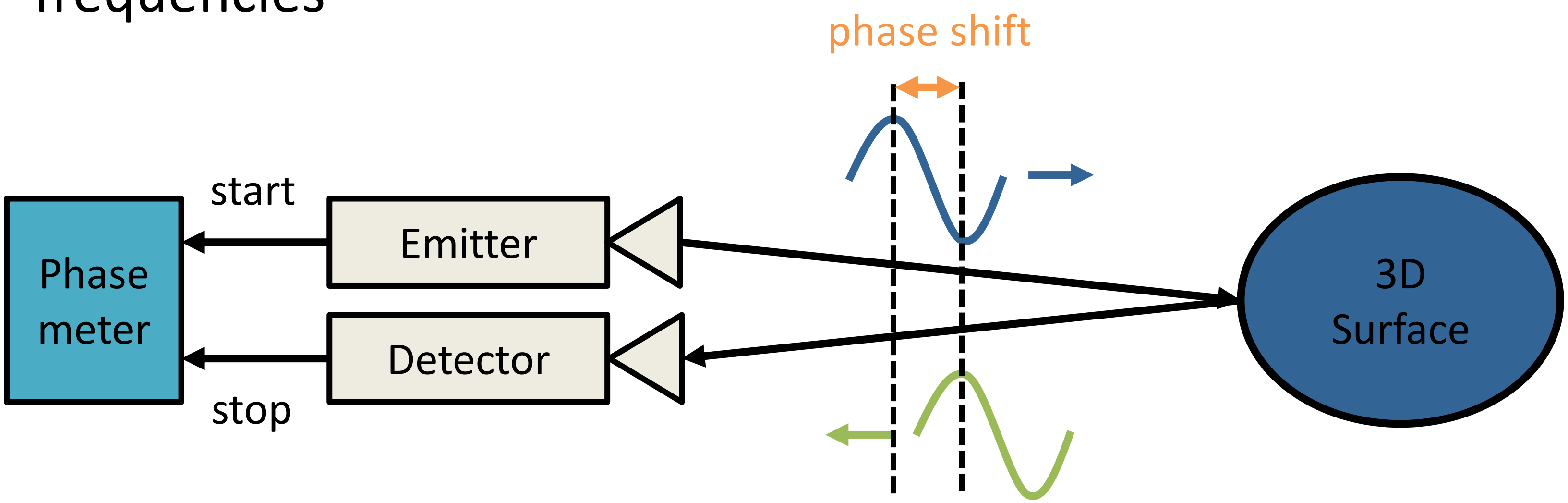
- Idea: emit timed IR pulse and measure its time of return
- Difficult to create pulses and measure time precisely



Slide adapted from N. Navab

Time-of-Flight Measurement Principle

- Idea: emit continous modulated IR wave signal and measure phase shift
- Signal periodicity creates phase ambiguities: use multiple frequencies



Slide adapted from N. Navab

Active vs. Passive Sensors

- Active Sensors
  - Surfaces do not need to be textured
  - Bring their own light, also work in low-light scenarios
  - But: Diffuse IR sunlight typically overrides emitted light
  - Difficulties for IR-absorbing or reflective materials

- Passive Sensors (e.g RGB-only)
  - Do not rely on measuring emitted light
  - Are not limited by the resolution of the projection pattern or ToF measurement principle
    - Distance
    - Multi-path noise (ToF)

Lessons Learned Today

- Stereo depth reconstruction from two and multiple views
  - Stereo rectification simplifies correspondence search for two views
  - Dense correspondence search using block matching
  - Correspondences can be ambiguous
  - Regularization with priors to help with noisy and ambiguous data terms

- Depth cameras
  - Structured light principle
  - Time-of-flight principle

Thanks for your attention!

Slides Information

- These slides have been initially created by Jörg Stückler as part of the lecture “Robotic 3D Vision” in winter term 2017/18 at Technical University of Munich.
- The slides have been revised by myself (Niclas Zeller) for the same lecture held in winter term 2020/21
- Acknowledgement of all people that contributed images or video material has been tried (please kindly inform me if such an acknowledgement is missing so it can be added).