Computer Vision

Multiview Geometry / Stereo Vision

Based on:

- Aanæs, H. (2014). Lecture notes on Computer Vision
- Jeremy Cohen. Pseudo-lidar - stereo vision for self-driving cars (medium.com, 2020)

THE CODING LIBRARY

Computer Vision

Multiview Geometry / Stereo Vision

Based on:

- Aanæs, H. (2014). Lecture notes on Computer Vision
- Jeremy Cohen. Pseudo-lidar - stereo vision for self-driving cars (medium.com, 2020)

Computer Vision

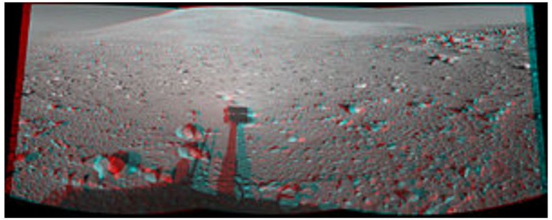
Multiview Geometry / Stereo Vision

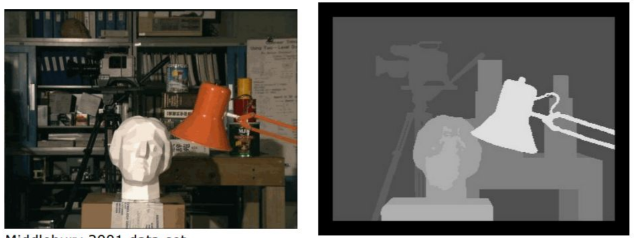
Based on:

- Aanæs, H. (2014). Lecture notes on Computer Vision
- Jeremy Cohen. Pseudo-lidar - stereo vision for self-driving cars (medium.com, 2020)

Stereo Vision

- Reconstructing 3D geometry based on camera images from two og more viewpoints
- Human Vision





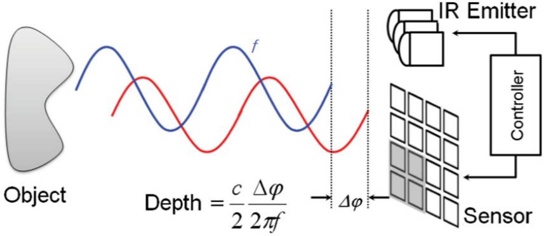
Middlebury 2001 data set.

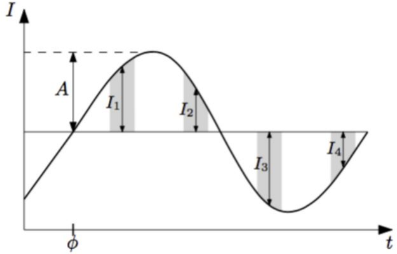
Stereo Vision

- How to estimate the distance using stereo vision?
  - Calibrate the cameras (Intrinsic and extrinsic calibration)
  - Create an epipolar scheme using epipolar geometry
  - Build a disparity and a depth map
- Depth map will be combined with an obstacle detection algorithm

Time of Flight Camera

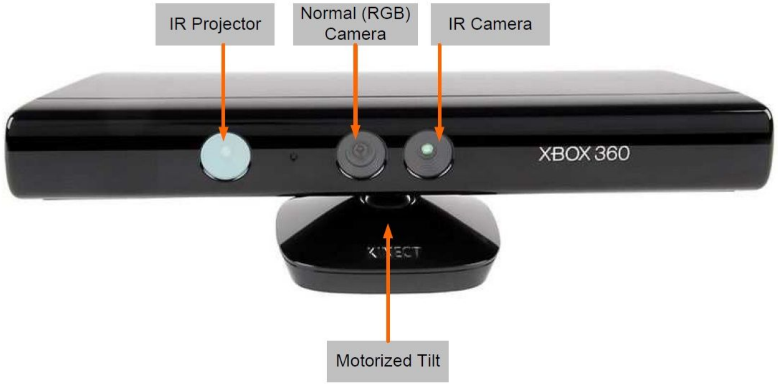
- Estimates distance by measuring the time of flight of a light signal between the camera and the subject for each point of the image



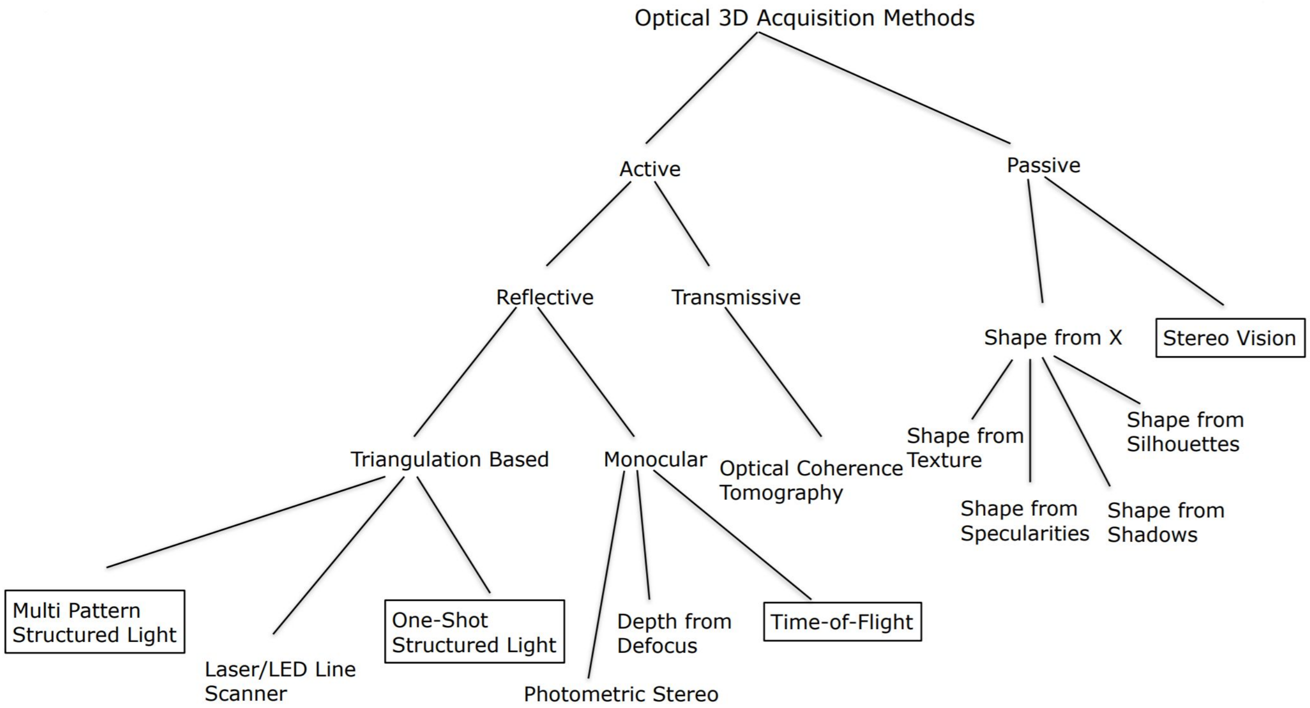


$$\varphi = \arctan \frac{I_1 - I_3}{I_2 - I_4} \qquad D = \frac{c}{4\pi f}\varphi$$

Microsoft Kinect 2



Taxonomy

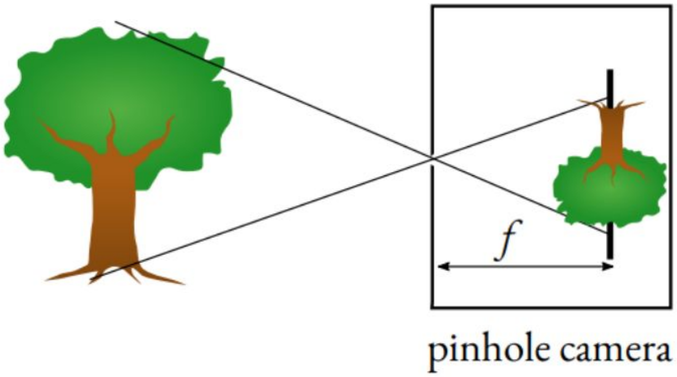


Optical 3D Acquisition Methods
Active
Passive
Reflective
Transmissive
Shape from X
Stereo Vision
Triangulation Based
Monocular
Optical Coherence Tomography
Shape from Texture
Shape from Silhouettes
Shape from Specularities
Shape from Shadows
Multi Pattern Structured Light
Laser/LED Line Scanner
One-Shot Structured Light
Photometric Stereo
Depth from Defocus
Time-of-Flight

Camera Geometry - Pinhole Model

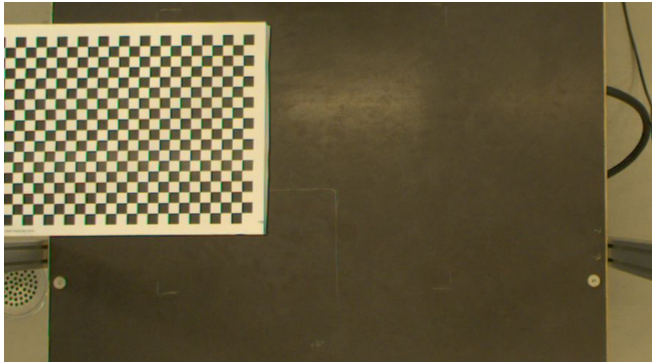
- Tag udgangspunkt i jeremy stereo vision artikel

$$w \cdot \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & 0 \\ 0 & f_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix}$$

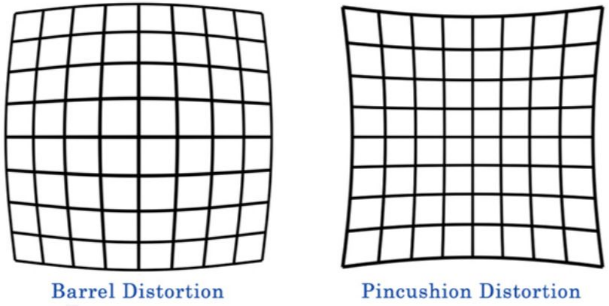


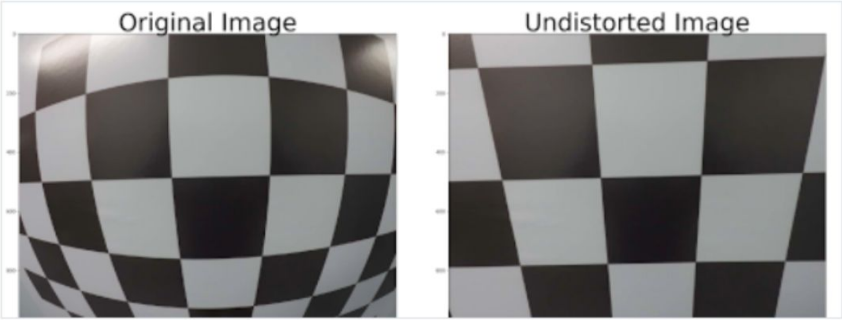
Camera Calibration

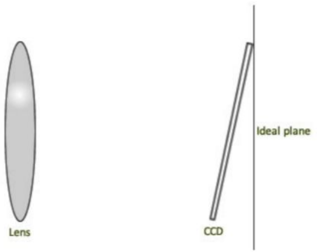
- Converting 3D point in the world to a 2D pixel
- Using multiple images
- A known object
- Different positions
  - Rotated
  - Translated
  - Tilted
- Computes camera matrix and distortion parameters
- Removes distortion



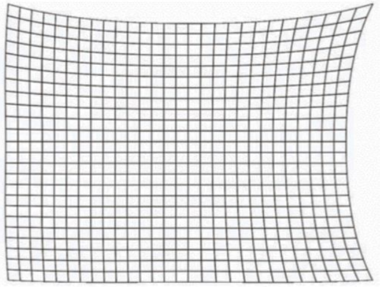
Camera Calibration - Distortion







Imperfect lens alignment



Tangential Distortion

Camera Calibration

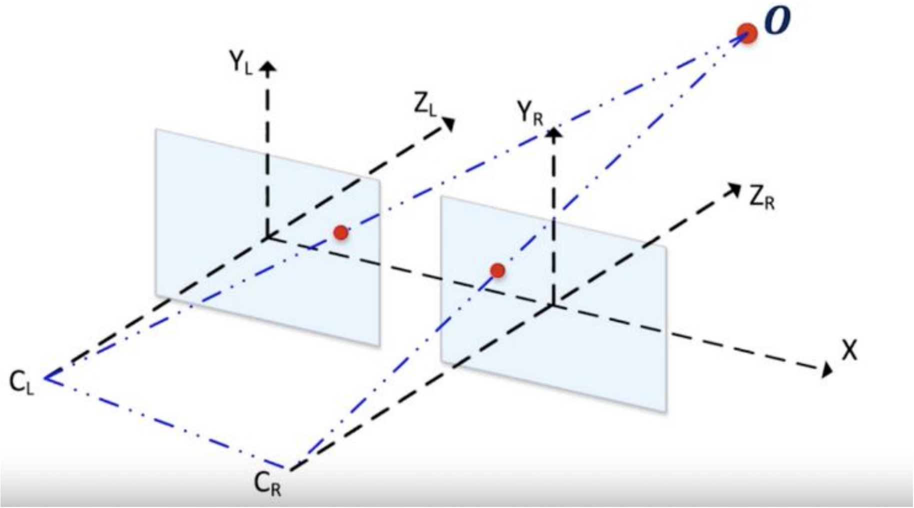
- From world coordinates to camera coordinates
  - Extrinsic parameters
    - Rotation matrix
    - Translation matrix
- From camera coordinates to pixel coordinates
  - Intrinsic parameters
    - Intrinsic matrix
    - Focal length
    - Optical centre

$$\mathbf{q_i} = \mathbf{A} \begin{bmatrix} \mathbf{R} & \mathbf{t} \end{bmatrix} Q_i = \mathbf{P} Q_i \quad , \quad \mathbf{P} = \mathbf{A} \begin{bmatrix} \mathbf{R} & \mathbf{t} \end{bmatrix}$$

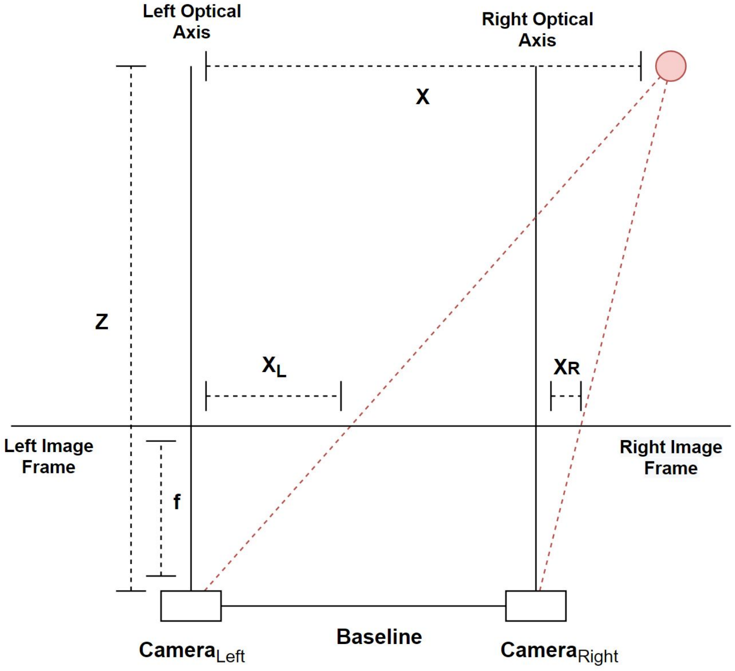
$$\mathbf{P} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix}$$

Multi View Geometry - Stereo Vision

- How can cameras estimate depth?
    - Left and right camera
    - Aligned in the same X and Y axis
    - Z is the difference of the two cameras and is used for estimating the depth



Multi View Geometry - Stereo Vision



$$\frac{Z}{f} = \frac{X}{x_L} \qquad \frac{Z}{f} = \frac{X-b}{x_R}$$

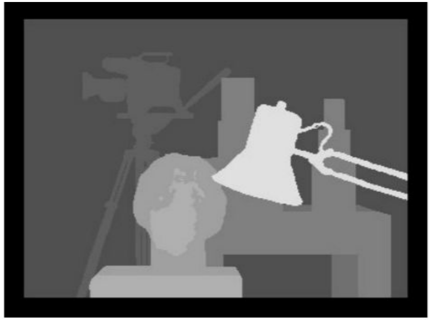
$$x_L = \frac{X}{Z} \cdot f \qquad x_R = \frac{X-b}{Z} \cdot f$$

$$Disparity = x_L - x_R$$

Disparity and Depth Maps

- Disparity is the difference in image location of the same 3D point from 2 different camera angles.
- In a pair of images, you can measure the apparent motion in pixels for every point and make an intensity image from the measurements

Middlebury 2001 data set.



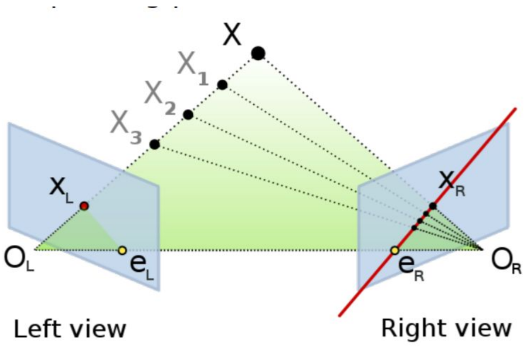
$$Z = \frac{fb}{d}$$

$$X = \frac{Zx_L}{f}$$

$$Y = \frac{Zy_L}{f}$$

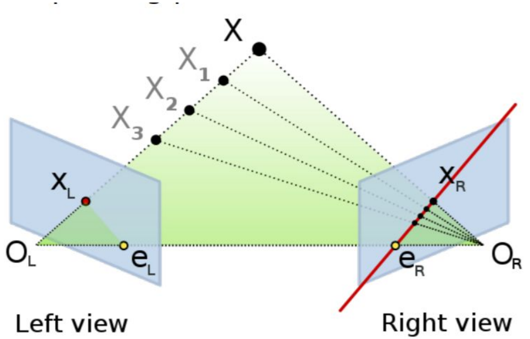
Epipolar Geometry

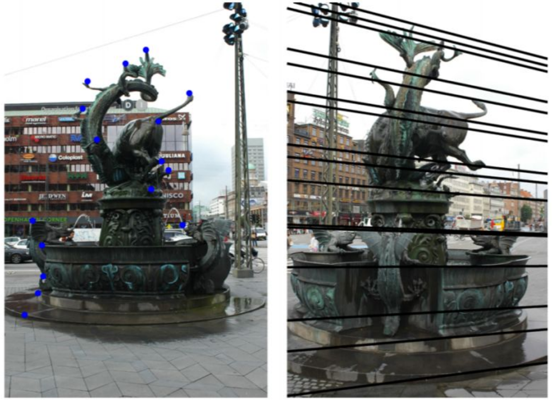
- Given a point in one camera - its corresponding point in the other camera lies on the epipolar line
- 7 Degree of freedom fundamental matrix



Stereo Correspondence Problem

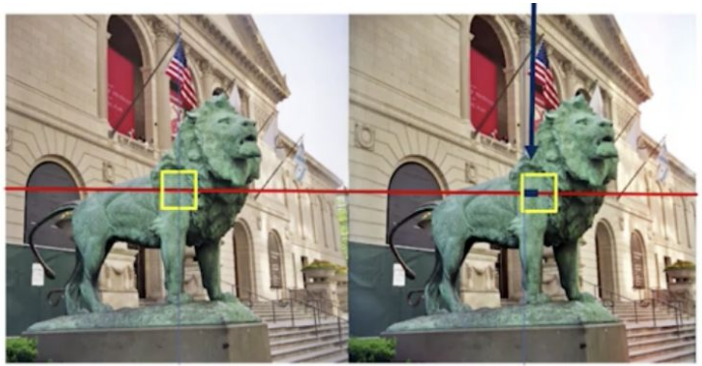
- How to solve it and compute the disparity?
  - Take a pixel in the left image
  - Search on the epipolar line for the pixel in the right image
  - The point is located on the epipolar line and the search is 1D
  - Cameras need to be aligned along the same axis





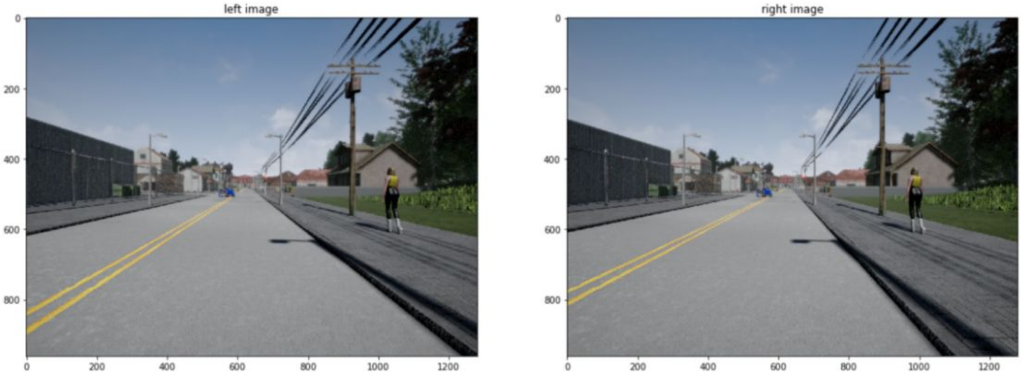
Epipolar Search in Images

- Take each pixel on the line in the left image
- Compare the left image pixels to the right image pixels on the epipolar line
- Take the pixel with the minimum cost
- Compute the disparity



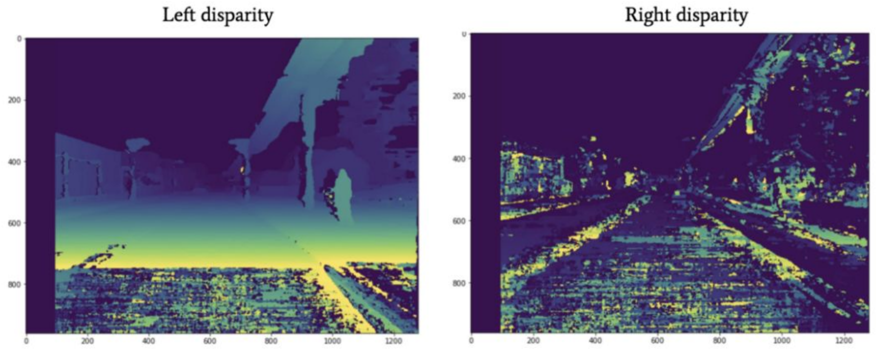
Practical Example of Using Stereo Vision

- First we calibrate the camera/images
- Then we have the image parameters for both images



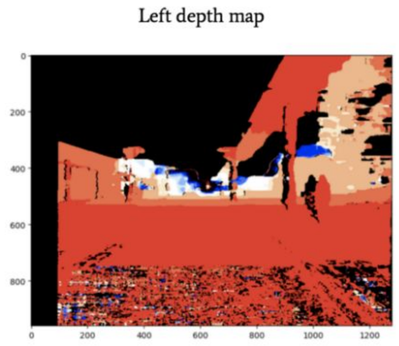
Compute Disparity Maps

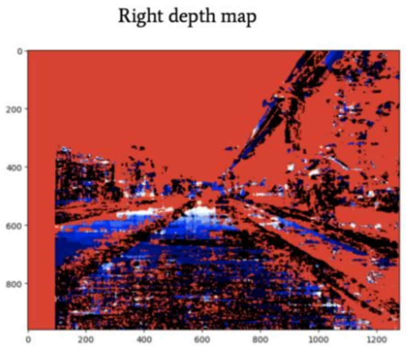
- Determine disparity between the two images
- Decompose the projection into the camera matrix
  - Both intrinsic and extrinsic parameters
- Estimate the depth by using the information from the last steps



Create the Depth Maps

- Use the projection matrix for each camera
  - Use the focal length
  - Compute baseline using corresponding values from the translation vectors
- Compute depth map of images from the disparity maps





$$Z = \frac{fb}{d}$$

$$X = \frac{Zx_L}{f}$$

$$Y = \frac{Zy_L}{f}$$

Depth Estimation - Results

- Run an obstacle detection algorithm and use it together with a depth map
- Take the closest point of the detected obstacle
    - We know the distance of every single point in the image

Left detection

closest_point_depth 28.927

Right detection

closest_point_depth 28.927

OpenCV Example

- Kig på opgave til lektionen

Homographies

Stereo Vision Triangulation

Direct Linear Transform (DLT)

Triangulation Error

Non-Linear optimization