interactIVe

Accident avoidance by active intervention for Intelligent Vehicles


# Multi Sensor Calibration

**Tobias Hanning**

**interactIVe Summer School**

**4-6 July, 2012**

# Agenda

- Multiple Sensors in Driver Assistance Systems
- Example: Camera calibration
  - Classic camera calibration
  - Camera calibration as multi-sensor calibration
- Applications: (Optical) Multi-Sensor Systems
  - Calibration of Two Imaging Sensors (Stereo Camera Calibration)
  - Calibration of an Imaging Sensor to a 3D-Sensor
- Conclusion

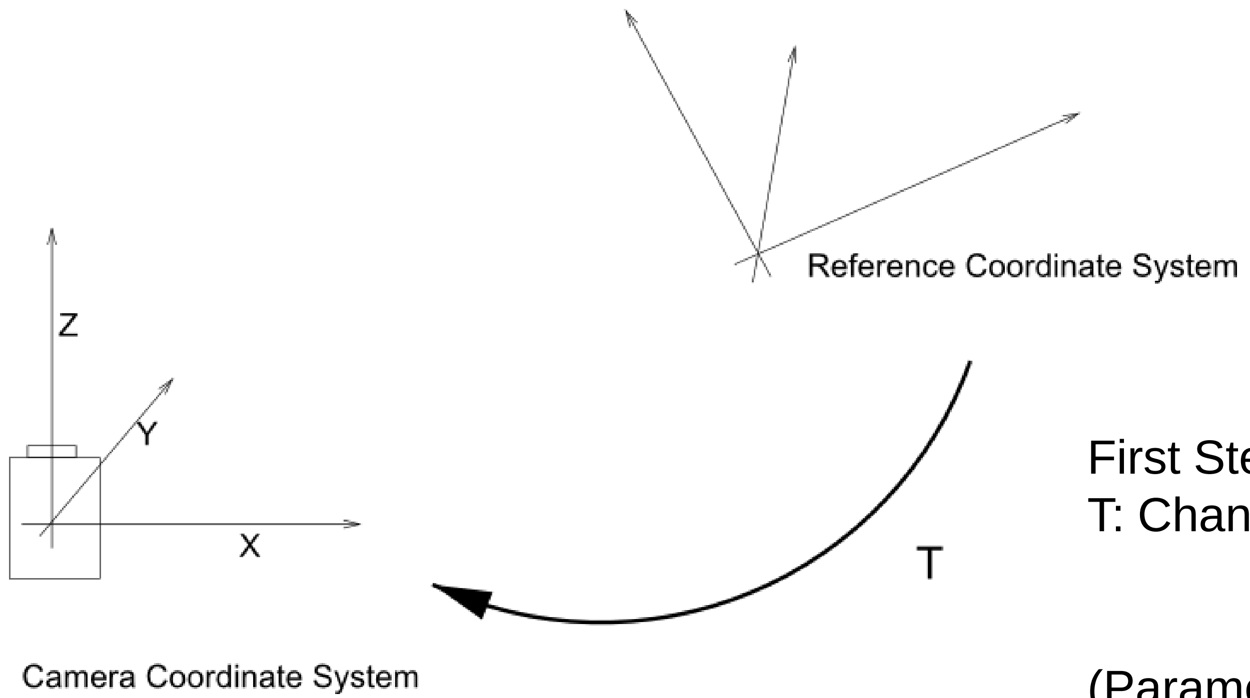
# Multi Sensor Calibration
# Overview

- Multiple Sensors In Automotive Applications:
  - Camera(s)
  - Lidar
  - Ultrasonic sensors
  - Accelerometer
  - and many more..

# Multi Sensor Calibration
# Overview

- Objective: Find parameters of each sensor
- Objective in Talk: Find pose and orientation of each sensor
  - Example: Camera Calibration as Multi-Sensor Calibration (meaning each pixel is a sensor on its own)
  - Therefore: A (very) short introduction to camera calibration
  - A new view at camera calibration as multi-sensor calibration
  - Transfer of results
    - Stereo Camera Calibration
    - Calibration of a 3D-Sensor wrt Grayvalue Camera

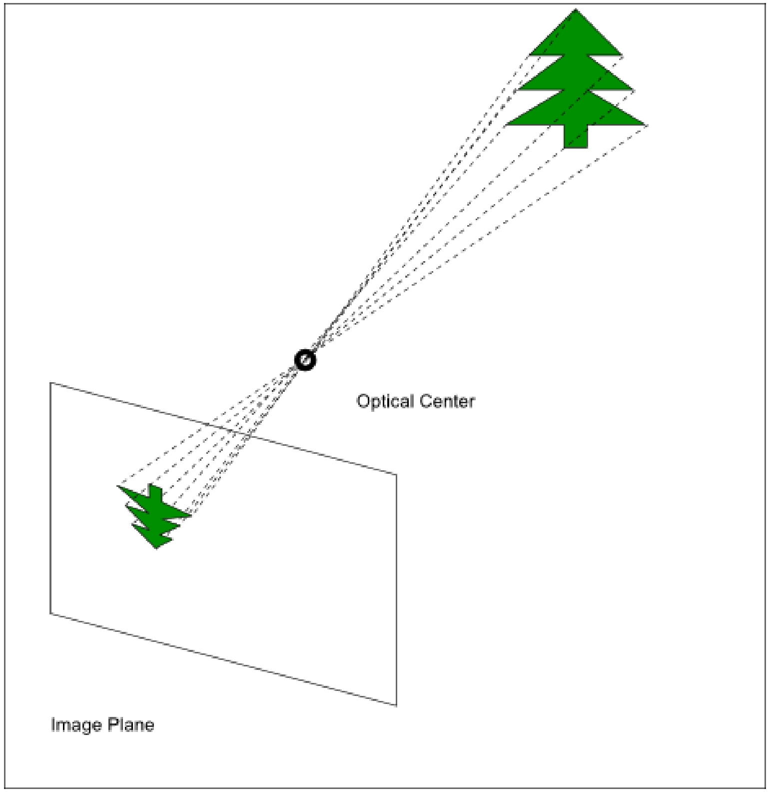
# Camera Calibration: The classic way




First Step:
T: Change Frame Of Reference

(Parameters: Rotation R and Translation t)

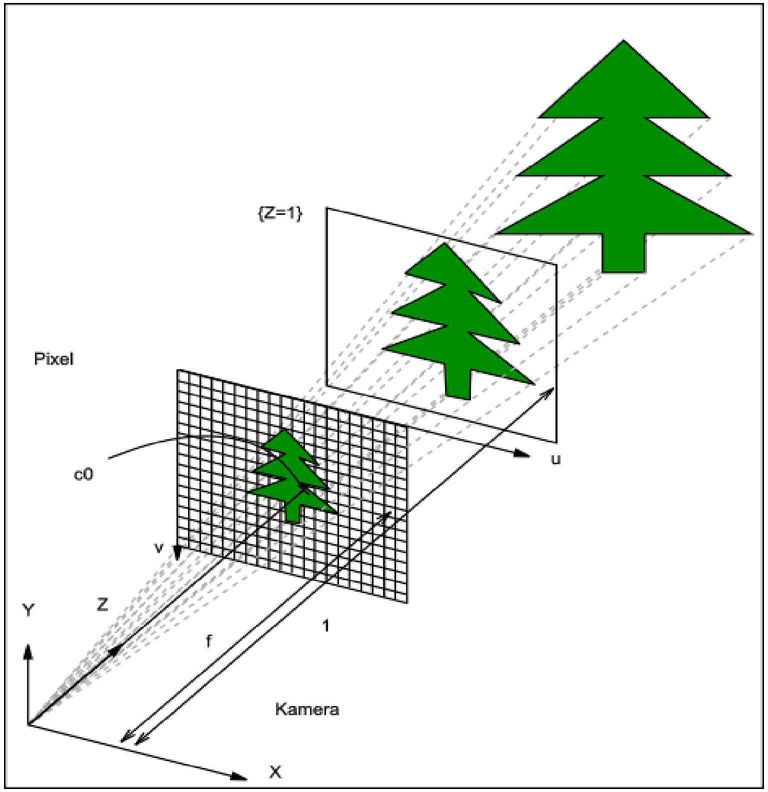
# Camera Calibration: The classic way




Second Step: Projection

- Modelled as pinhole projection
- Operation: Division by focal length
- But: Image is upside down

# Camera Calibration: The classic way



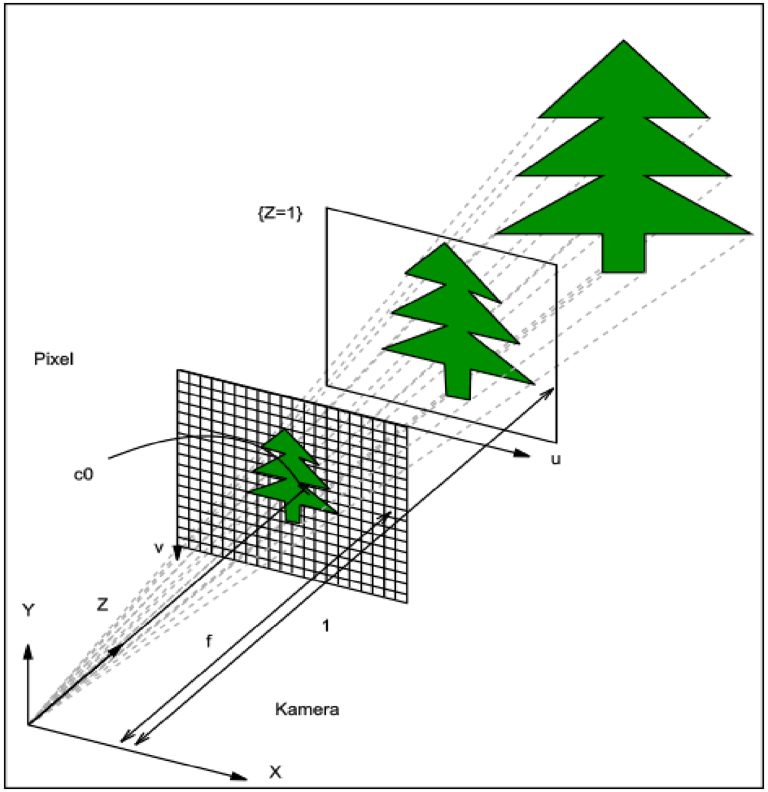

Second Step: Projection

- Project onto (virtual) plane {z=1}
- Conversion to Image Coordinates:
  - Shift
  - Scale

$$\Pi : \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \to \begin{pmatrix} X/Z \\ Y/Z \end{pmatrix}$$

$$I : \begin{pmatrix} u \\ v \end{pmatrix} \to \begin{pmatrix} \alpha & 0 \\ 0 & \beta \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix} + \begin{pmatrix} u_0 \\ v_0 \end{pmatrix}$$

# Camera Calibration: The classic way




Overall Camera Mapping

$$C = I \circ \Pi \circ T$$

Camera Calibration:

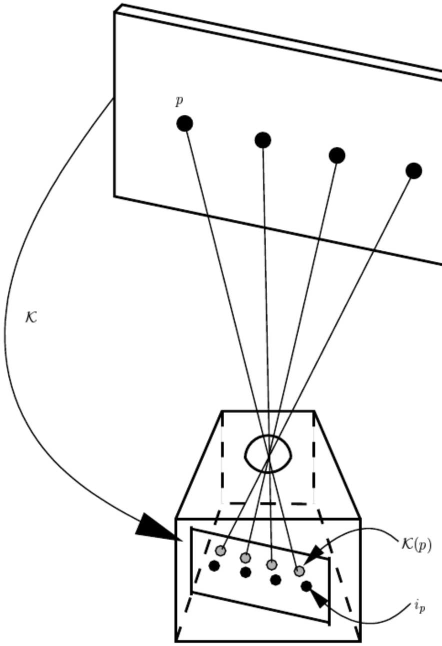
Input:

1. Set P of points in 3D
2. Observation $i_p$ for every p in P

Task:

Find $(R, t, \alpha, \beta, u_0, v_0)$ minimizing

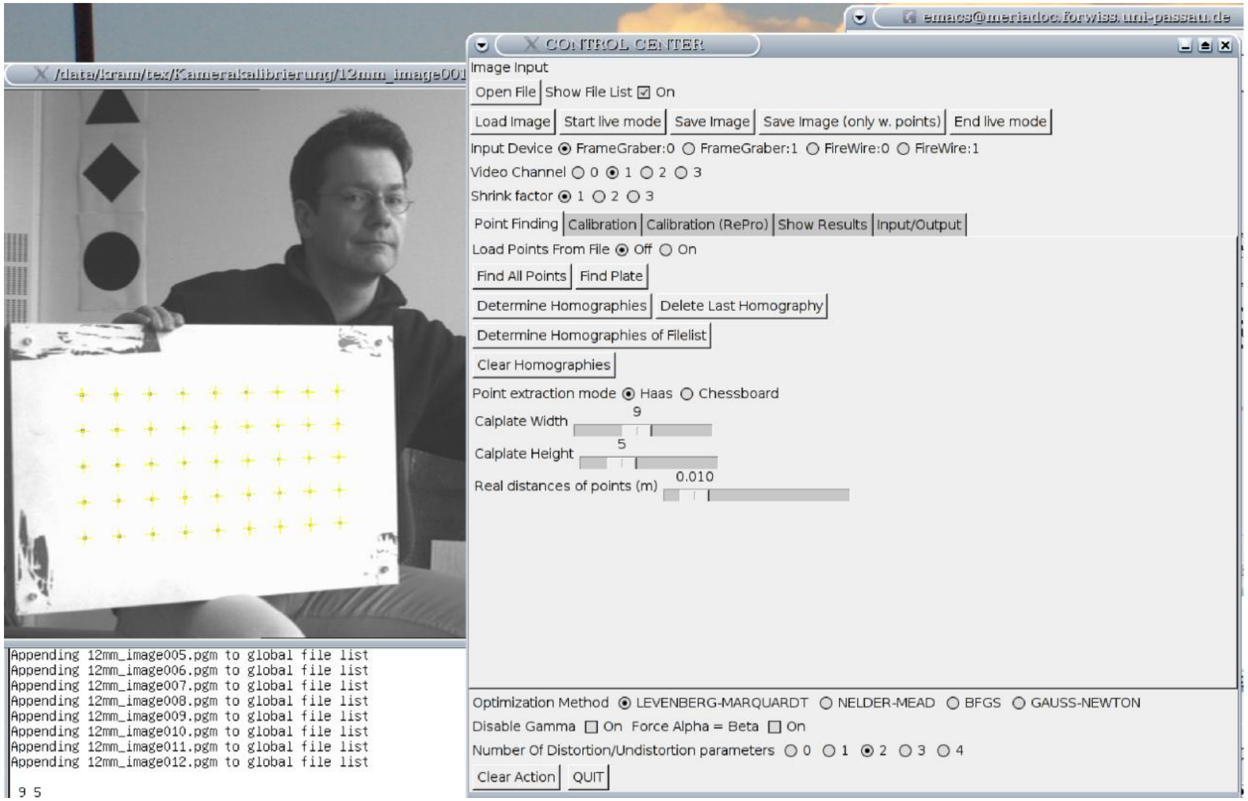
# Camera Calibration: The classic way




$$(R,t,\alpha,\beta,u_0,v_0)\mapsto \sum_{p\in P}\left\| i_p - I\circ\Pi\circ T(p)\right\|^2$$

"Minimize the distance between model projection and observation"

# Camera Calibration: The classic way




Sucessful Calibration: Prototype projected in input image

# Summary: Classic Camera Calibration

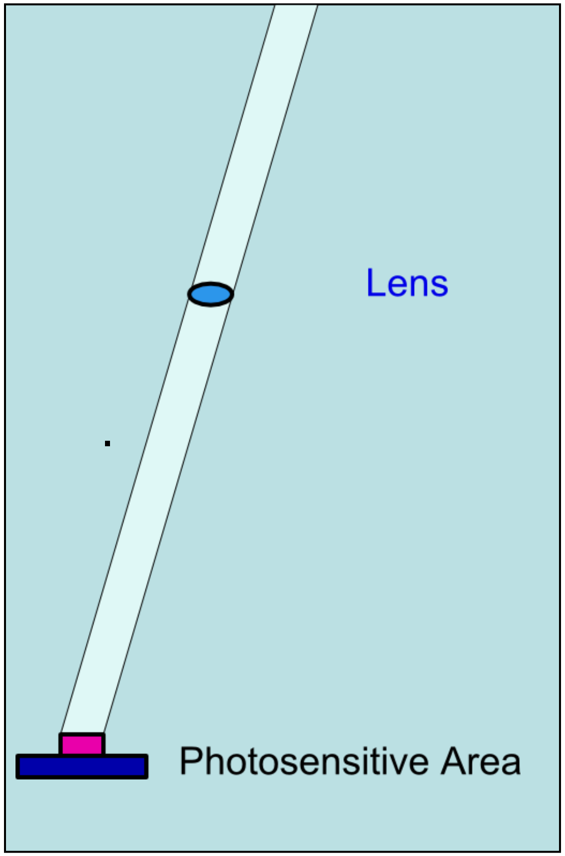
Camera is modelled as pinhole camera
Calibration input on point to point base
Sensor (= Imager) not mentioned in model

Let's take a different point of view:

Each pixel is a sensor on its own!

# Camera Calibration As Multi-Sensor Calibration

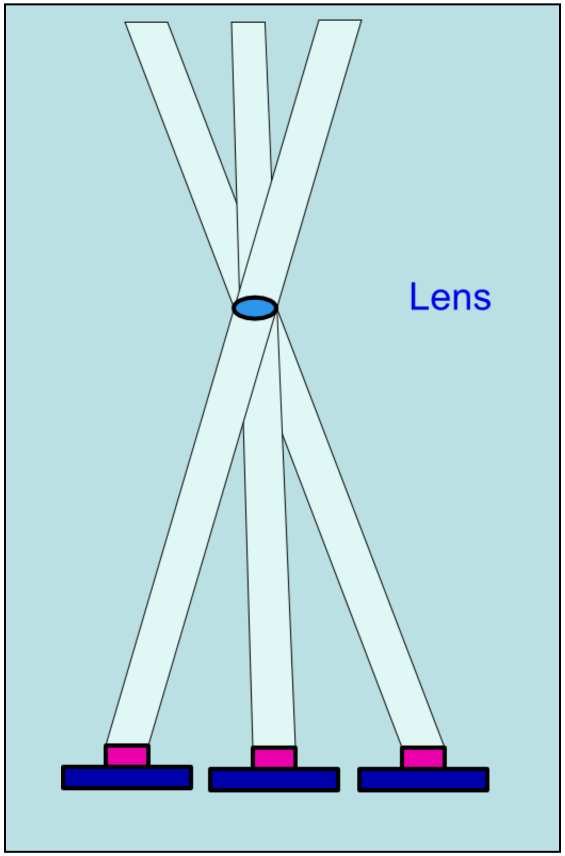
## Camera Pixel As Single Sensor




- Sensor measures intensity of light
- Together with lens:
  - Measuring intensity of a ray of light

# Camera Calibration As Multi-Sensor Calibration

## Camera Modelled As Multi-Sensor

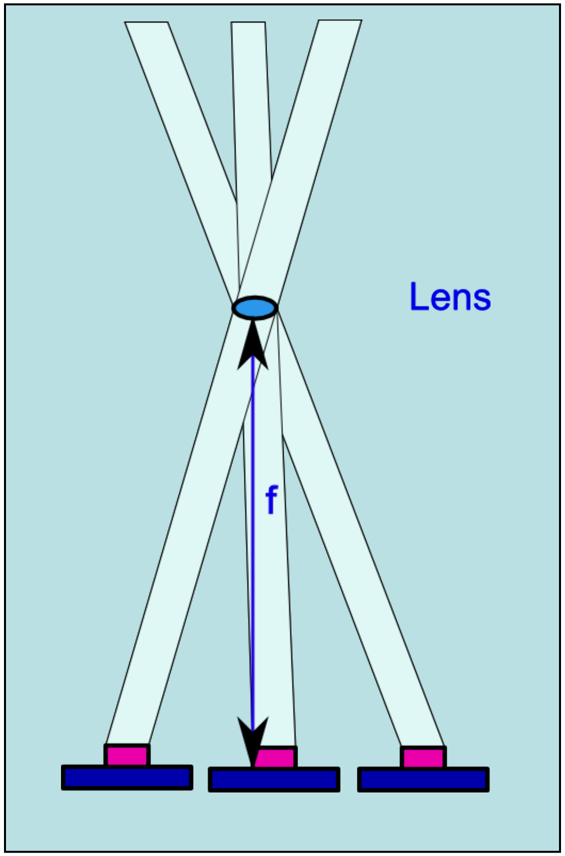



- Camera = Grid of Sensors
- Parameters:
  - Position and Orientation wrt lens
- Constraints:
  - All rays share one point (optical sensor)
  - Sensors are arranged in fixed grid (perpendicular to the optical axis)

→ Camera parameters

# Camera Calibration As Multi-Sensor Calibration

## Camera Modelled As Multi-Sensor

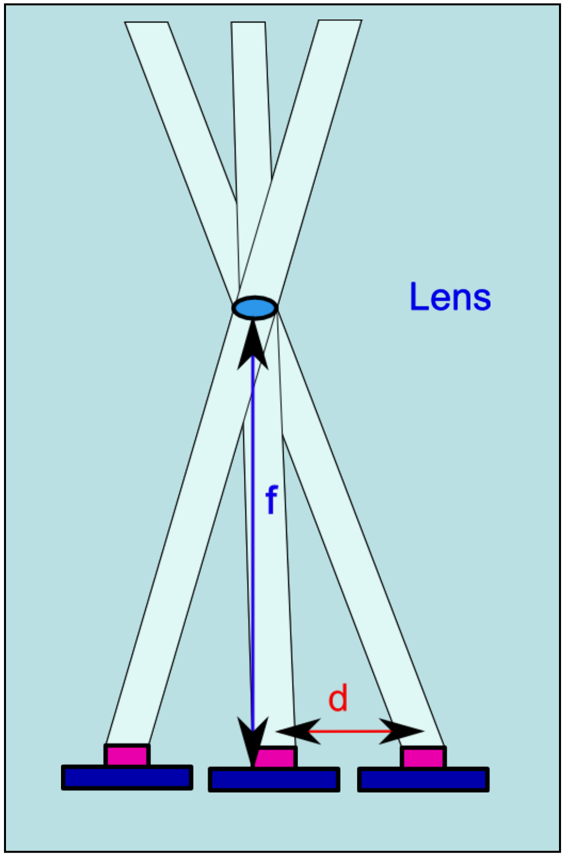



Parameters:

- Distance to Lens f

# Camera Calibration As Multi-Sensor Calibration

## Camera Modelled As Multi-Sensor

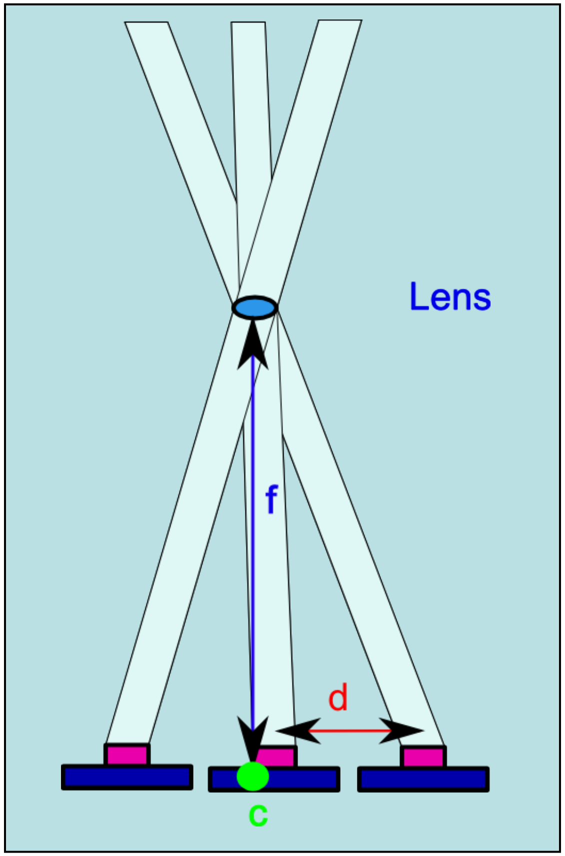



Parameters:

- Distance to Lens f
- Distance between Sensors d

# Camera Calibration As Multi-Sensor Calibration

## Camera Modelled As Multi-Sensor

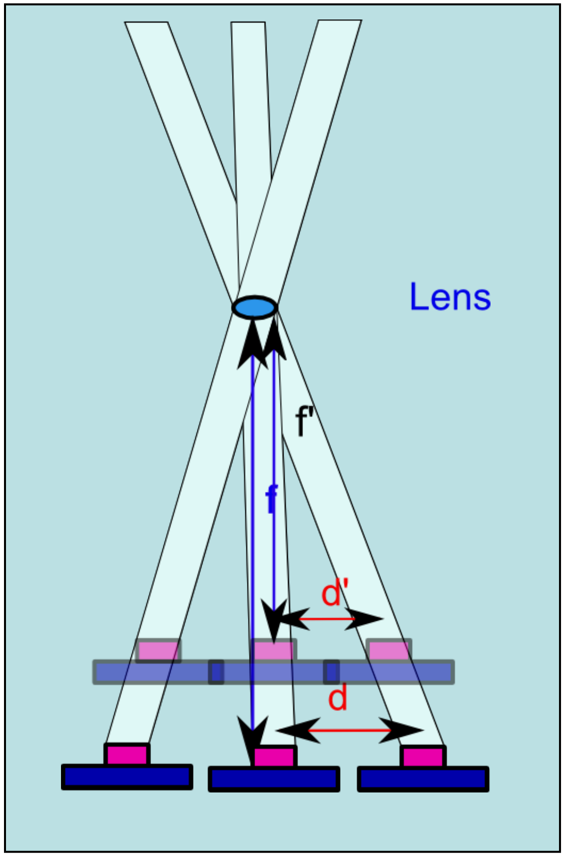



Parameters:

- Distance to Lens f
- Distance between Sensors d
- Point of orthogonal Projection of Lens c

# Camera Calibration As Multi-Sensor Calibration

Camera Modelled As Multi-Sensor




Parameters:

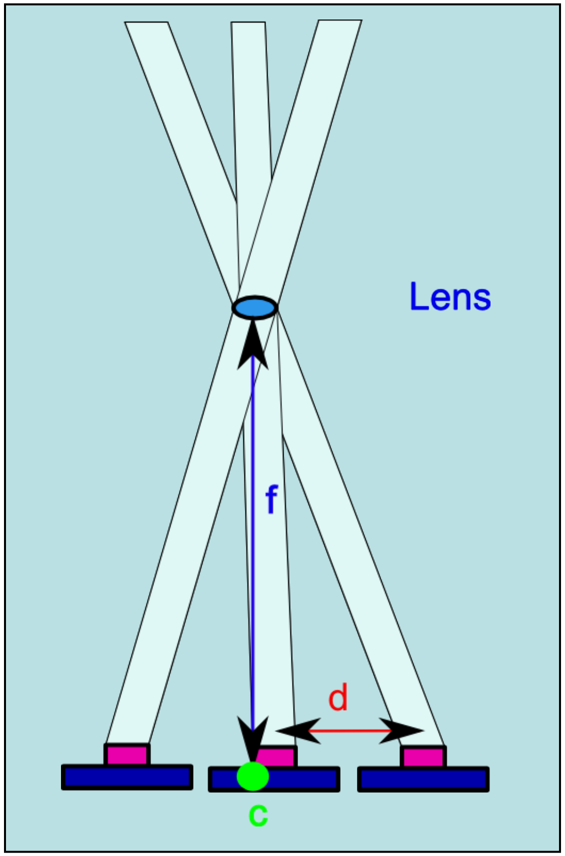
- Distance to Lens f
- Distance between Sensors d
- Point of orthogonal Projection of Lens c

<u>But:</u> Any f'/d' with f'/d' = f/d will deliver the same sensor!

<u>Thus:</u> Not f,d are not free parameters, but f/d is!

# Camera Calibration As Multi-Sensor Calibration

## Camera Modelled As Multi-Sensor




Link to classic camera parametrization:

$$\frac{f}{d_u} \cong \alpha$$

$$\frac{f}{d_v} \cong \beta$$

$$c \cong \begin{pmatrix} u_0 \\ v_0 \end{pmatrix}$$

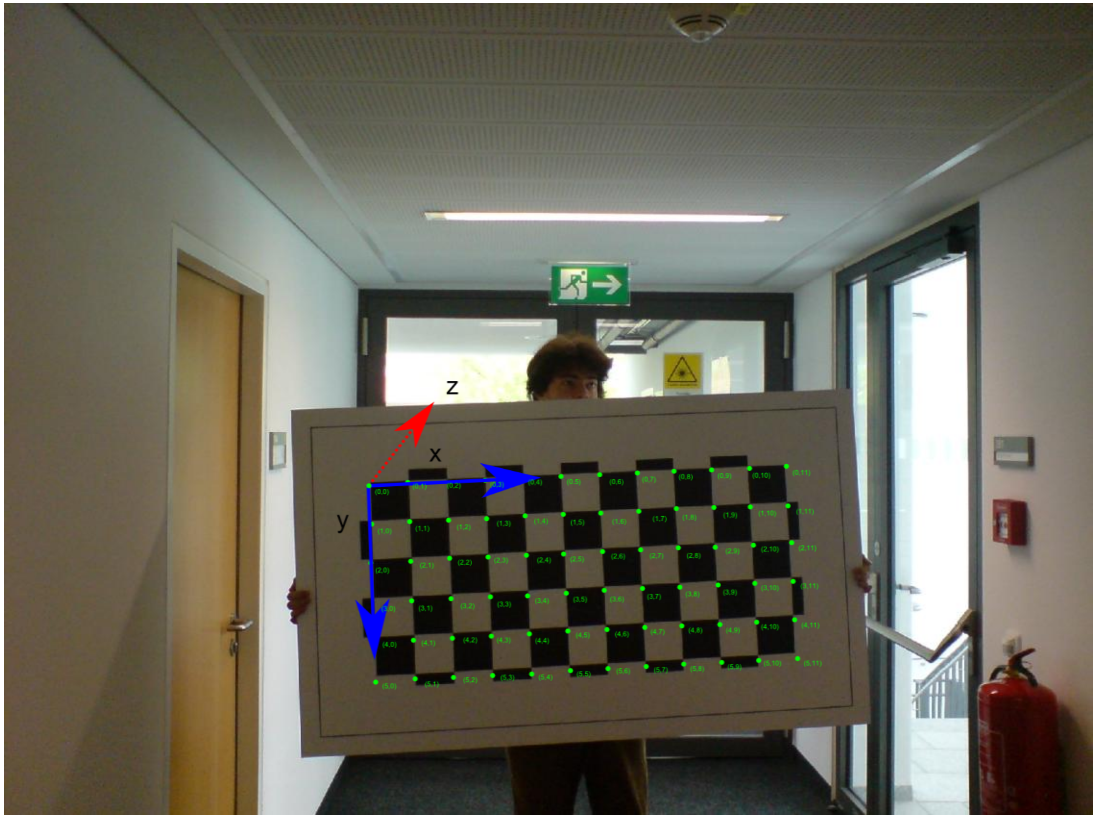
(R,t)

# Camera Calibration As Multi-Sensor Calibration

Scene Interpretation

# Camera Calibration As Multi-Sensor Calibration

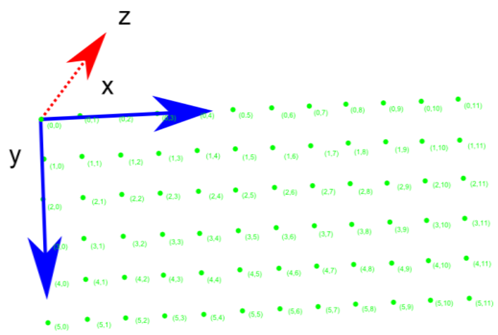
## Scene Interpretation



# Camera Calibration As Multi-Sensor Calibration

Scene Interpretation:

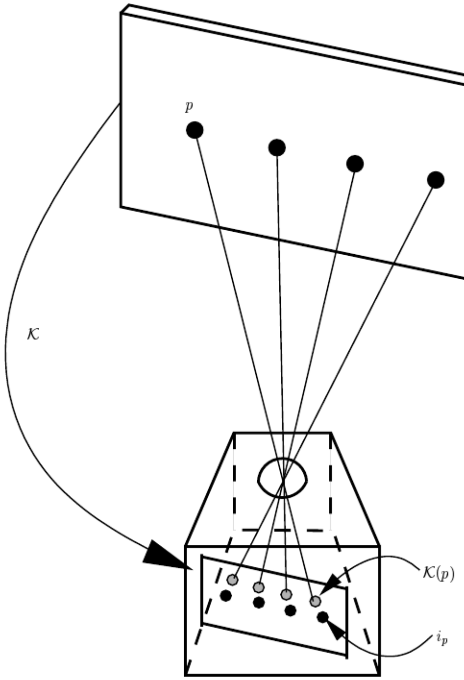
- Each sensor shares an subset of the same interpretation of the scene
- Sensor may not observe the whole scene
- Scene introduces new parameters (e.g. position of the calibration target)



# Camera Calibration As Multi-Sensor Calibration

Multi-Sensor Calibration:

- Determine parameters
  - of the scene
  - of the Multi-Sensor-Array
- such that distance of the
  - interpretation of the observation
  - and the interpretation of the scene
- becomes minimal



# Camera Calibration As Multi-Sensor Calibration

Further Topics of Classic Camera Calibration:

(Not covered in this talk:)

- (Radial) Distortion (“fish-eye“)
- Sensor Correlations (e.g. blurring introduced by lens)
- Error Functions
- Initial Solution (Projective Geometry)
- Optimization Techniques

# Applications: Multi-sensor systems

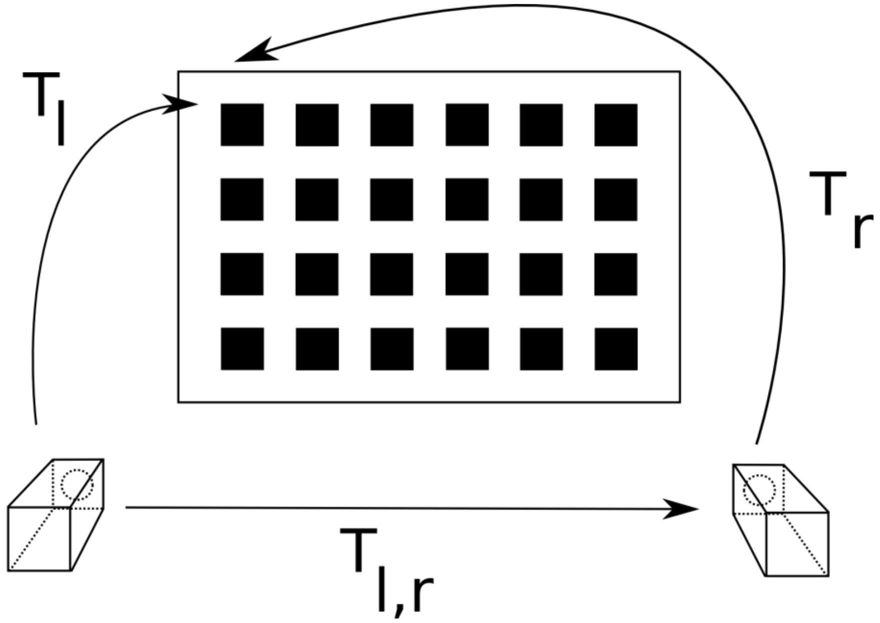
Examples:

- Homogeneous Multi-Sensor-System: Stereo Camera
- Heterogeneous Multi-Sensor-System: 3D Camera to Mono-Camera

In both examples the sensors are related by rotation and translation

# Multi-sensor calibration: Share the same interpretation

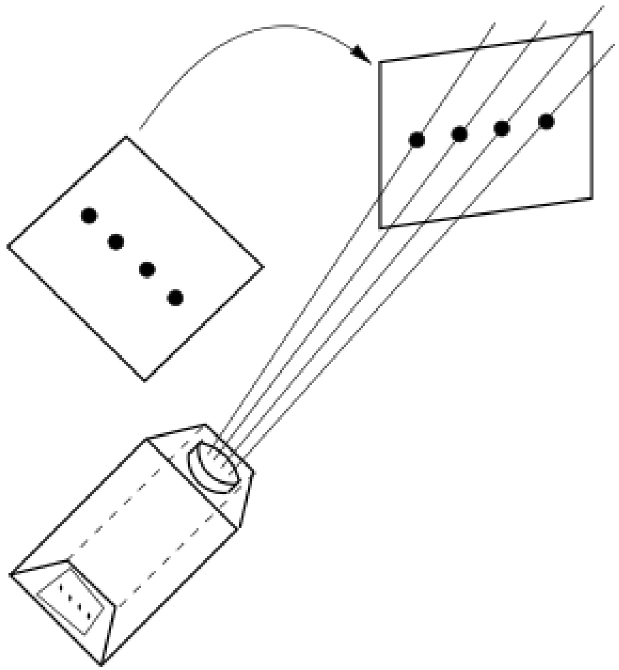
Stereo-Camera-Calibration




Left Camera and right camera observe the same scene

# Multi-sensor calibration: Share the same interpretation

Stereo-Camera-Calibration

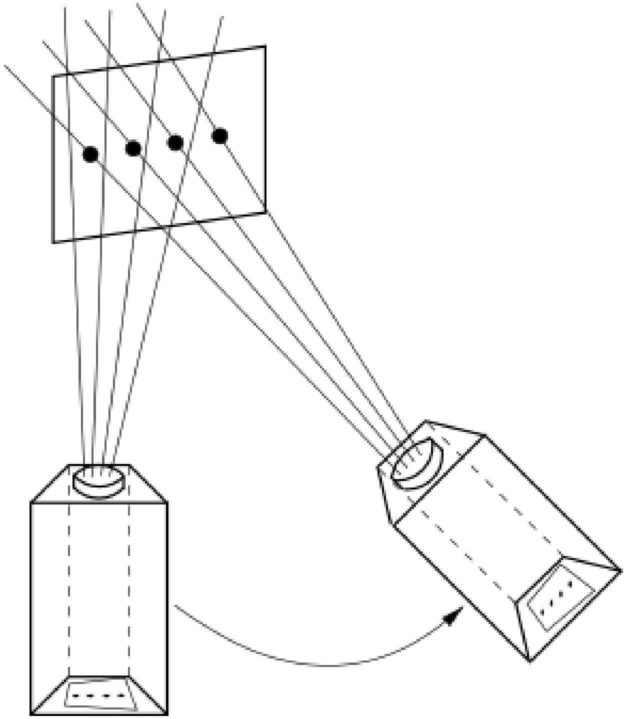


Left Camera:

Determine position of observed calibration pattern wrt left camera

# Multi-sensor calibration: Share the same interpretation

Stereo-Camera-Calibration

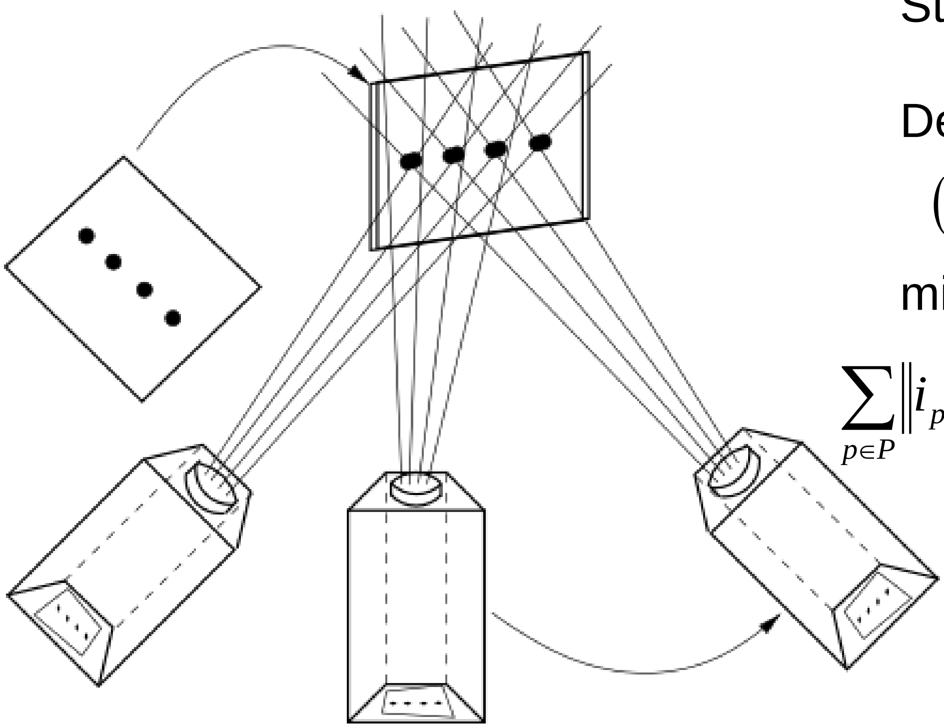


Right Camera:

Determine position of right camera wrt observed calibration pattern

# Multi-sensor calibration: Share the same interpretation

Stereo-Camera-Calibration



Stereo Camera Calibration:

Determine parameters

$$\left(R_l, t_l, \alpha_l, \beta_l, u_{0,l}, v_{0,l}, R_r, t_r, \alpha_r, \beta_r, u_{0,r}, v_{0,r}\right)$$

minimizing

$$\sum_{p \in P} \left\| i_{p,l} - I_l \circ \Pi \circ T_l(p) \right\|^2 + \left\| i_{p,r} - I_r \circ \Pi \circ T_r(p) \right\|^2$$

$i_{p,l}$ observation of p in left image
$i_{p,r}$ observation of p in right image

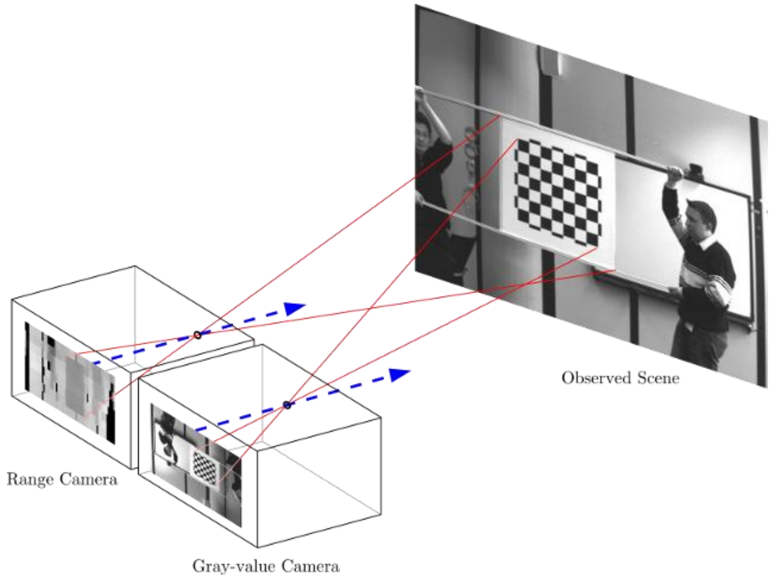
# Multi-sensor calibration: Share the same interpretation

Stereo-Camera-Calibration

- Scene interpretation by two sensors of same kind
- Calibration: Determine parameters of both sensors such that the observation matches the interpretation of the model

# Multi-sensor calibration: Share the same interpretation

3D-Camera and 2D-Camera



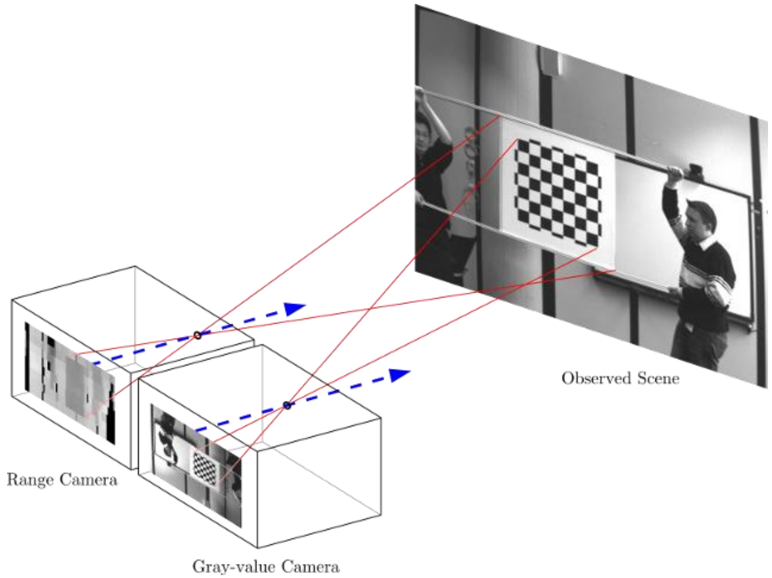

- 2D Camera and
- 3D Camera observe
- the same scene

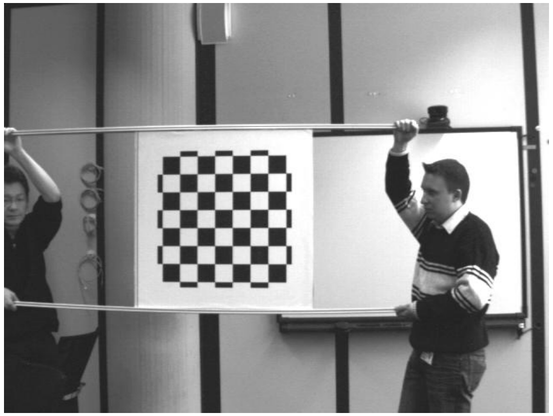
Sensors are related by Rotation and Translation


- **Output:** Image-like distance information (64 × 8-pixel matrix, 42° horizontal, 16° vertical angular resolution)
- Distance measurements in **outdoor** environments up to 25m
- Automotive and safety-applications capable

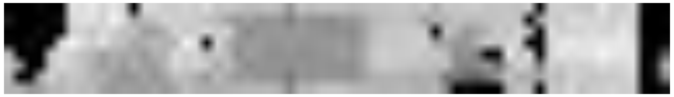
# Multi-sensor calibration: Share the same interpretation

3D-Camera and 2D-Camera



Scene observed by grayvalue camera



Scene observed by 3D-Camera

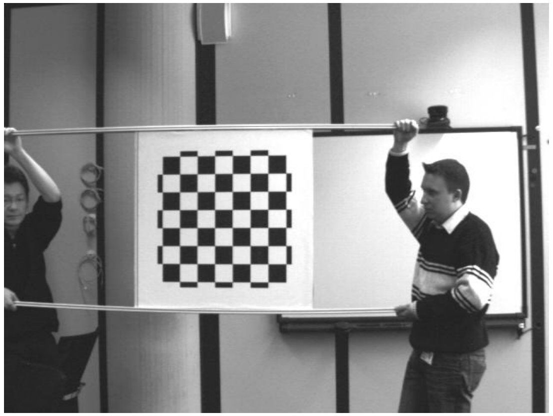
# Multi-sensor calibration: Share the same interpretation

3D-Camera and 2D-Camera

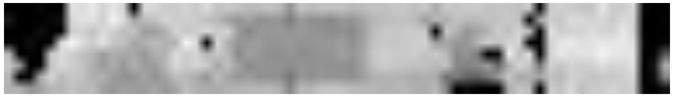
Canonical scene interpretation of both sensors differ:

- Light intensity per pixel
- Depth information per pixel

Task:
Find an interpretation of the scene that applies for both sensors !



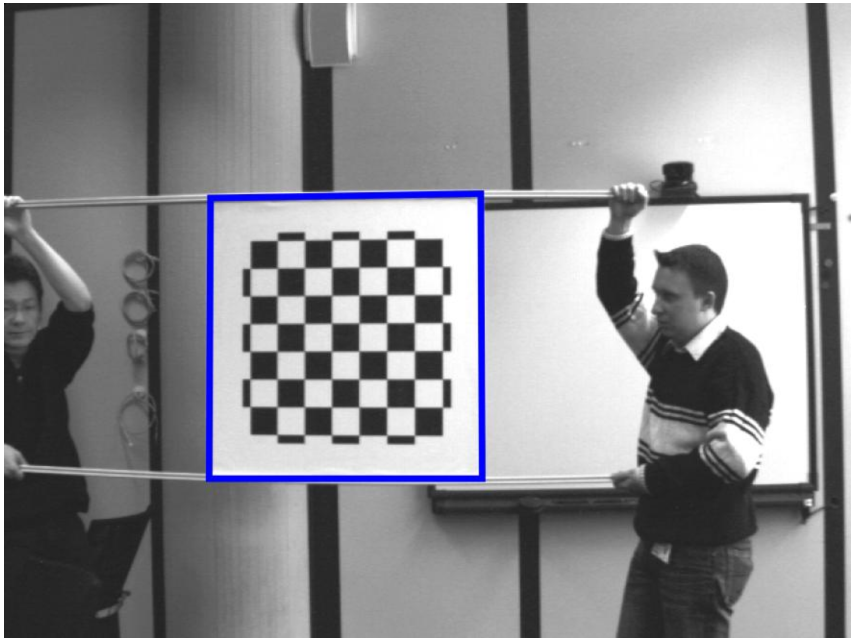
Scene observed by grayvalue camera



Scene observed by 3D-Camera

# Multi-sensor calibration: Share the same interpretation

3D-Camera and 2D-Camera

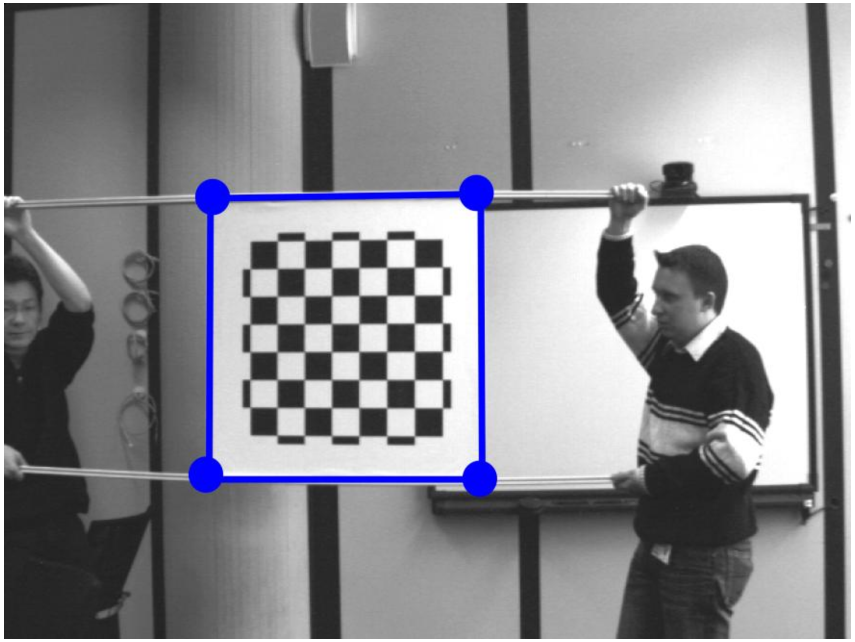


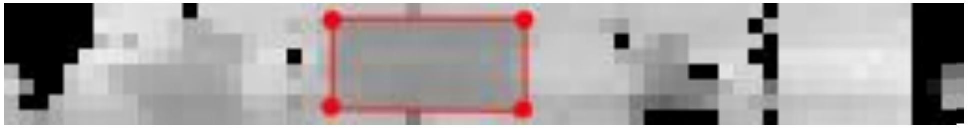
*Shared Scene Interpretation*:

Plane (with defined borders) at fixed position

# Multi-sensor calibration: Share the same interpretation

3D-Camera and 2D-Camera





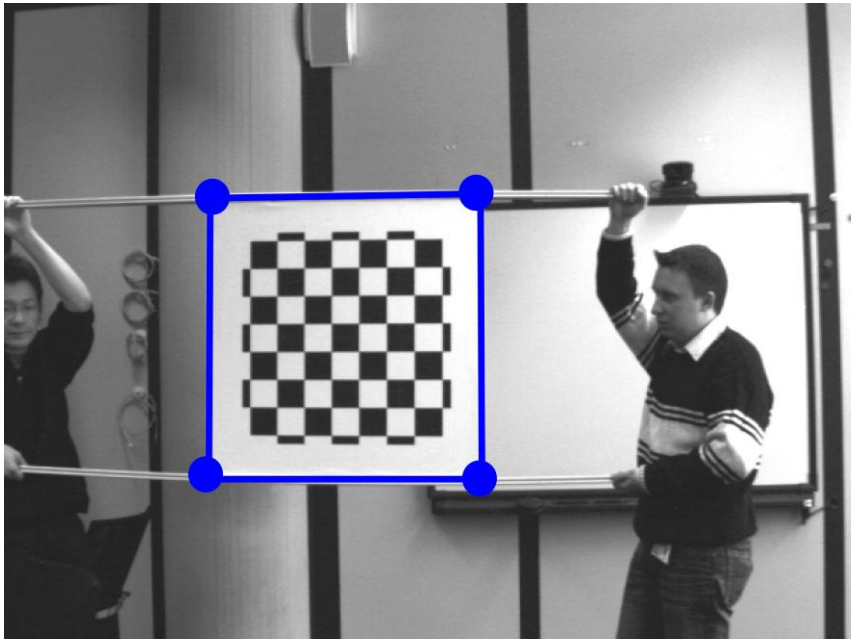
*Shared Scene Interpretation*:

Points determining borders of plane

Calibration: Determine Rotation R and translation t that maps 3D points on 3D points

# Multi-sensor calibration: Share the same interpretation

3D-Camera and 2D-Camera



*Monocular Prototype Reconstruction*:

Input:

1. Set P of four points in 3D (planar)
2. Observation $i_p$ for every p in P
3. Calibrated Camera C

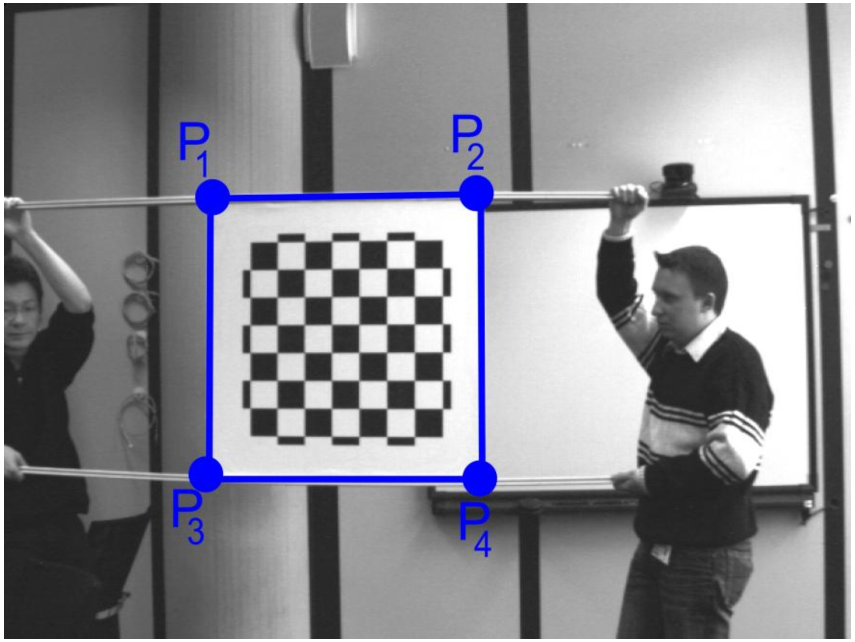
Task:

Determine $(R_0,t_0)$ minimizing

$$\sum_{p \in P} \left\| C(Rp+t) - i_p \right\|^2$$

# Multi-sensor calibration: Share the same interpretation

3D-Camera and 2D-Camera




$$(R_0, t_0) = \arg\min_{(R,t)} \sum_{p \in P} \left\| C(Rp + t) - i_p \right\|^2$$

*Monocular Prototype Reconstruction*:

Result:

$$P_1 = R_0 p'_1 + t_0$$

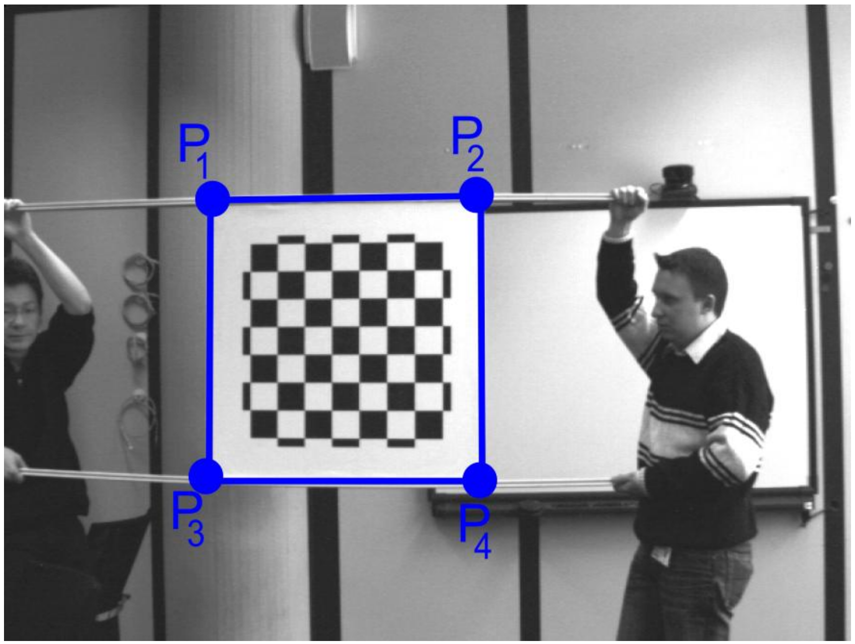
$$P_2 = R_0 p'_2 + t_0$$

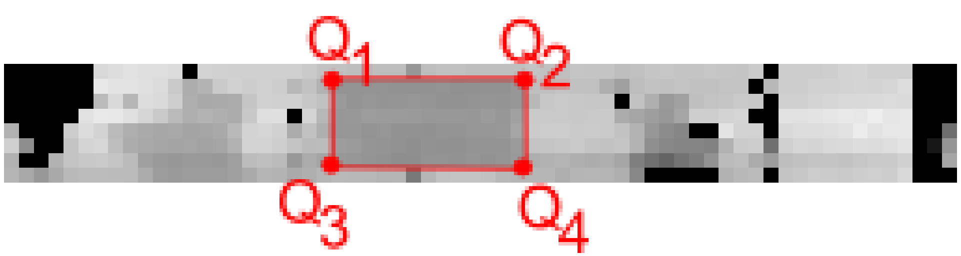
$$P_3 = R_0 p'_3 + t_0$$

$$P_4 = R_0 p'_4 + t_0$$

# Multi-sensor calibration: Share the same interpretation

3D-Camera and 2D-Camera




*Result of Object Segmentation in 3D Camera*:

$$Q_1$$

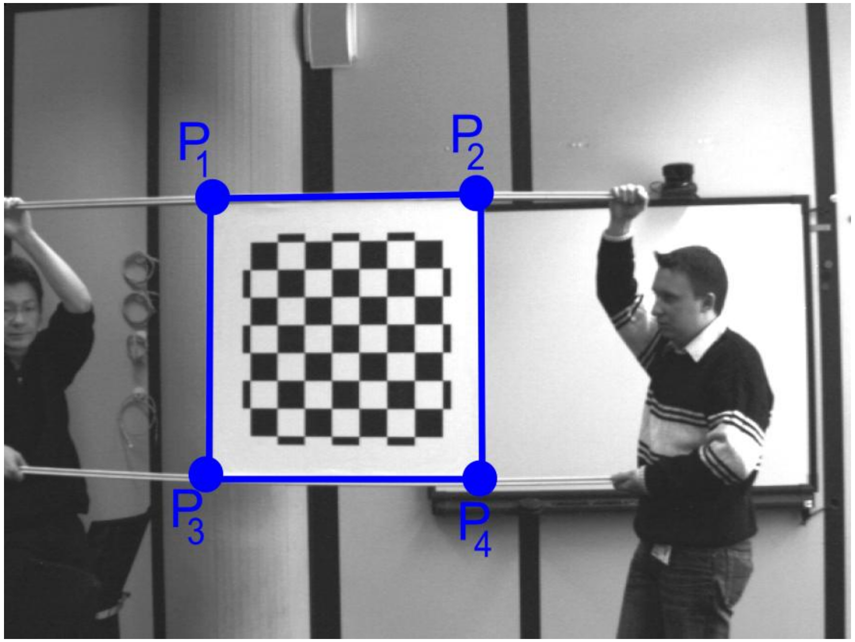
$$Q_2$$

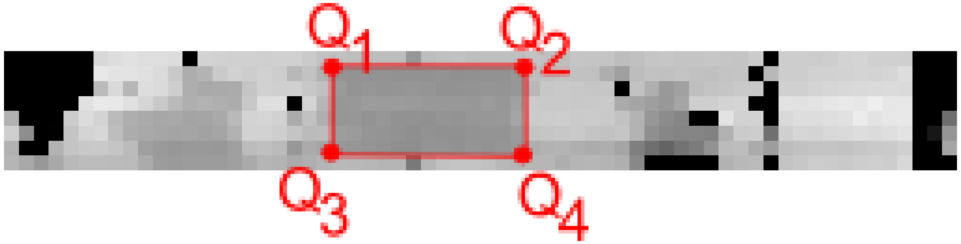
$$Q_3$$

$$Q_4$$

# Multi-sensor calibration: Share the same interpretation

3D-Camera and 2D-Camera



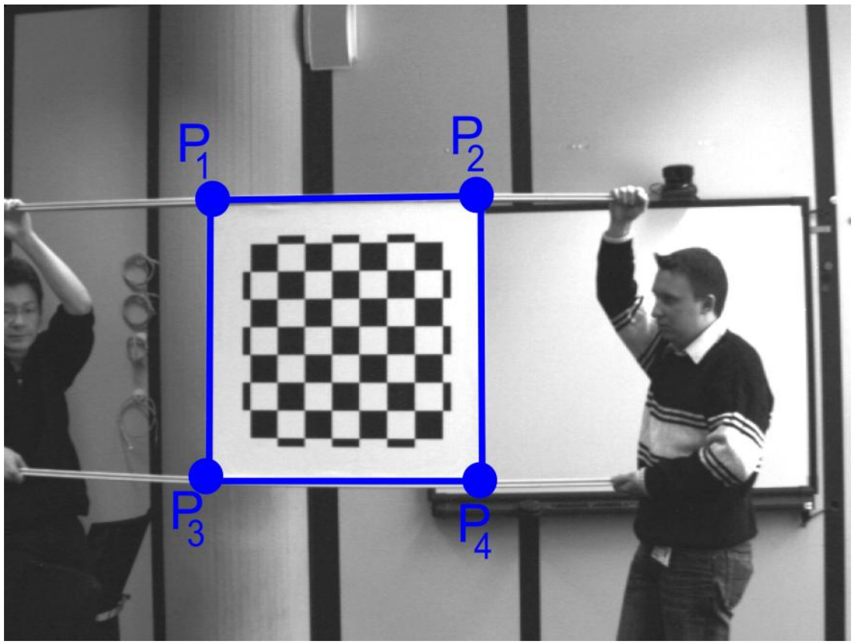

*Relative Position of 2D-Camera to 3D-Camera*:

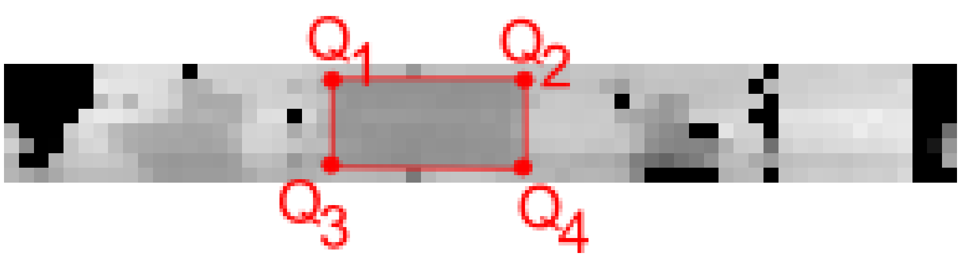
Determine

$$\arg\min_{(R,t)} \sum_{1}^{4} \left\| RQ_i + t - P_i \right\|^2$$

# Multi-sensor calibration: Share the same interpretation

3D-Camera and 2D-Camera




But:

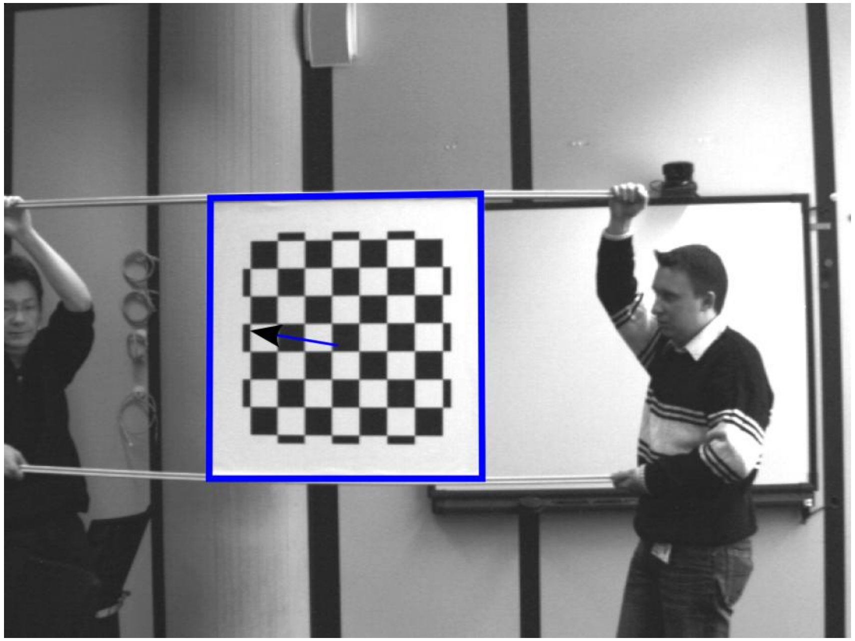
Acquisition of 3D point information by 3D-Camera is not reliable!

- Low Resolution
- Noise
- Difficult Feature Extraction in 3D

→ Weak Correspondencies!

# Multi-sensor calibration: Share the same interpretation

3D-Camera and 2D-Camera



*Shared Scene Interpretation*:

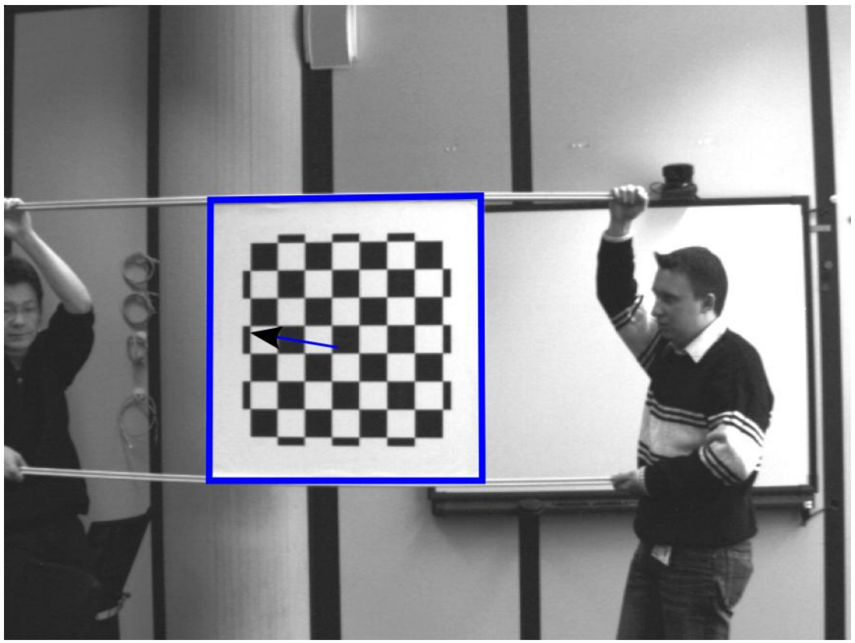
More reliable: Plane information

Idea: Represent plane by point and vector

Calibration: Determine Rotation R and translation t that maps 3D plane to 3D plane

# Multi-sensor calibration: Share the same interpretation

3D-Camera and 2D-Camera



*Plane matching*:

Input:

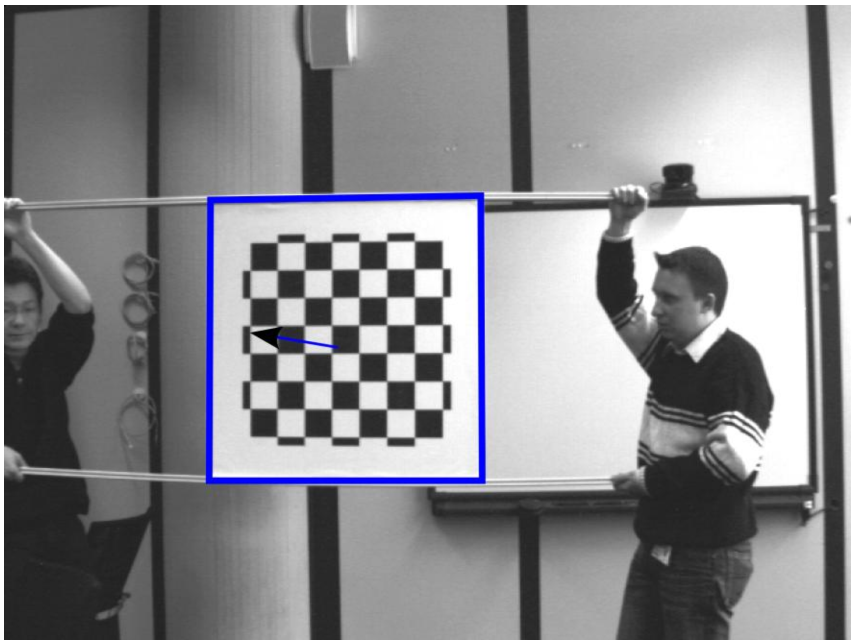
- Normal $n_{3d}$, distance $d_{3d}$ of plane obtained by 3D-Camera
- Normal $n_{2d}$, distance $d_{2d}$ of plane obtained by 3D-Camera

Output: $(R_0,t_0)$ minimizing

$$\sum_{planes} w_n \left\langle Rn_{3d}, n_{2d} \right\rangle^2 + w_d \left| d_{3d} + t - d_{2d} \right|^2$$

# Multi-sensor calibration: Share the same interpretation

3D-Camera and 2D-Camera



*Plane matching*:

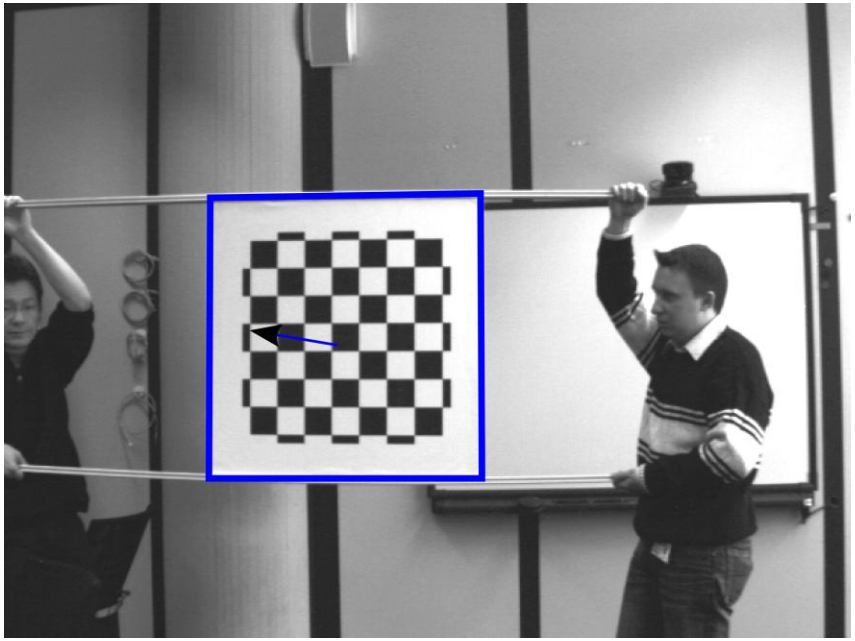
Attention:

If one of the weights $w_n$ or $w_d$ is zero, the problem looses uniqueness. In that case we obtain rather a constraint than an optimization problem!

$$\sum_{planes} w_n \left\langle R n_{3d}, n_{2d} \right\rangle^2 + w_d \left| d_{3d} + t - d_{2d} \right|^2$$

# Multi-sensor calibration: Share the same interpretation

3D-Camera and 2D-Camera



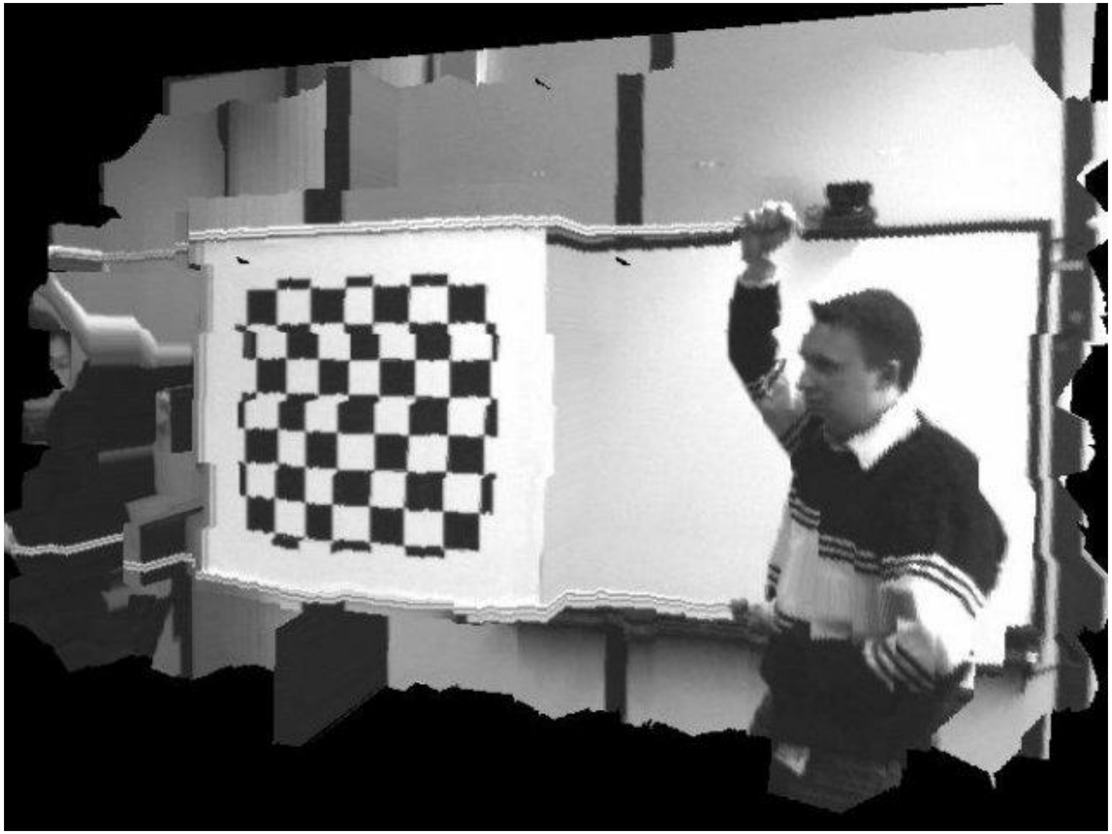
*Algorithm*:

1. Determine initial solution by point matching (closed form!)
2. Refine solution by plane matching (non-linear)

$$\arg\min_{(R,t)} \sum_{planes} w_n \left\langle R n_{3d}, n_{2d} \right\rangle^2 + w_d \left| d_{3d} + t - d_{2d} \right|^2$$

# Multi-sensor calibration: Share the same interpretation

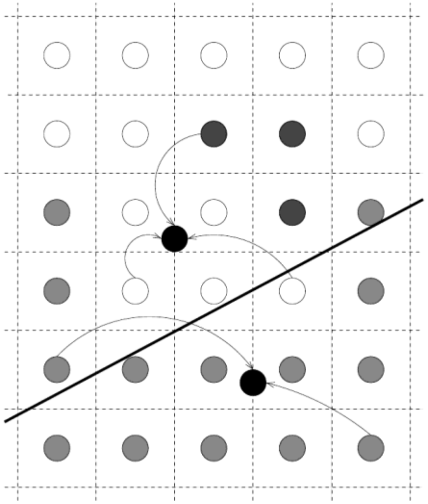
3D-Camera and 2D-Camera: Result



*Application*:

Integrated Depth Image

Depth information is supplied by segmentation of 2D image



# Conclusion

- Multi-Sensor Calibration means sharing the same interpretation of the observed scene
- Shared Interpretation may mean applying a function resulting in an uncommon interpretation by the sensor
- The shared interpretation may only supply a necessary condition (constraint)
- Applications: Multi-sensor systems
  - Calibration of Two Imaging Sensors (Stereo Camera Calibration) modelled as multi-sensor arrays
  - Calibration of an Imaging Sensor to a 3D-Sensor

interactIVe

Accident avoidance by active intervention for Intelligent Vehicles


# Thank you.

Co-funded and supported
by the European Commission

SEVENTH FRAMEWORK
PROGRAMME

Tobias Hanning
University of Passau
Innstr. 43
D-94033 Passau

Partner logo