# Lecture 8:

# Stereo Reconstruction I
# Projective and Epipolar Geometry

## Contents

1. Stereo Matching

2. Pinhole Camera Model

3. Projective/Homogeneous Coordinates

4. Extrinsic and Intrinsic Parameters

5. Epipolar Geometry

6. Reconstruction of the Depth

7. Summary

© 2007-2009 Andrés Bruhn, Markus Mainberger

---

**Stereo Matching (1)**

# Stereo Matching

### What is the Stereo Problem?

**Given:** Two different views of the same scene ($\rightarrow$ two cameras, same time).
**Wanted:** Displacement field between both views (disparity map).

### Example for Stereo Matching
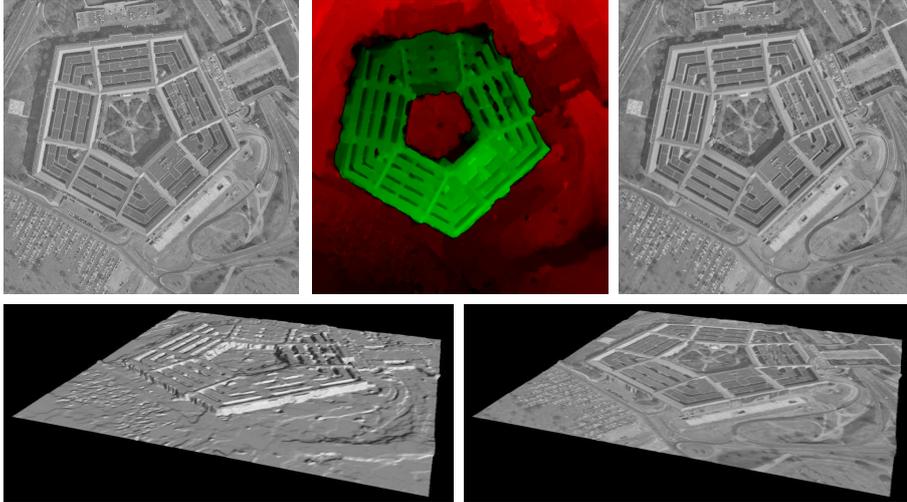
◆ Matching of two views of a human face
(*Slesareva 2005*)



Javier Sánchez stereo pair. **(a) Left:** Left image. **(b) Middle:** Disparity map. **(c) Right:** Right image.

## Stereo Matching (2)

M I
🏛 A
| 1 | 2 |
| 3 | 4 |
| 5 | 6 |
| 7 | 8 |
| 9 | 10 |
| 11 | 12 |
| 13 | 14 |
| 15 | 16 |
| 17 | 18 |
| 19 | 20 |
| 21 | 22 |
| 23 | 24 |
| 25 | 26 |
| 27 | 28 |
| 29 | 30 |
| 31 | 32 |
| 33 | 34 |

### Reconstruction as Final Goal

◆ Reconstruction of the Pentagon
(*Slesareva/Bruhn/Weickert 2005*)



Pentagon image pair (CMU database). **(a) Top Left:** Left frame. **(b) Top Center:** Disparity map. **(c) Top Right:** Right frame. **(d) Bottom Left:** Reconstruction. **(e) Bottom Right:** With texture.

## Stereo Matching (3)

M I
🏛 A
| 1 | 2 |
| 3 | 4 |
| 5 | 6 |
| 7 | 8 |
| 9 | 10 |
| 11 | 12 |
| 13 | 14 |
| 15 | 16 |
| 17 | 18 |
| 19 | 20 |
| 21 | 22 |
| 23 | 24 |
| 25 | 26 |
| 27 | 28 |
| 29 | 30 |
| 31 | 32 |
| 33 | 34 |

### Fields of Application

◆ Face recognition

◆ Free viewpoint stereo

◆ 3-D map generation

◆ Virtual reality

◆ Structure-from-Motion

### Challenges in Stereo Matching

◆ Very large displacements (wide baseline stereo)

◆ Rigid body motion (due to change of viewpoint)

◆ Camera calibration (if position and orientation of cameras is not known)

◆ Multi-view integration

◆ Occlusions

# The Pinhole Camera Model

## How Are 2-D Images Acquired from a 3-D Scene?

◆ *Idea:* Acquisition can be modelled realistically by the **pinhole camera model**

◆ *Assumption:* Perspective projection through a hole of infinitely small diameter
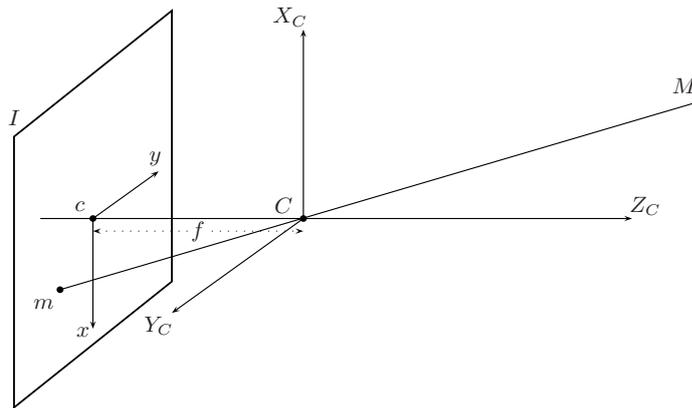
Pinhole camera with image plane behind the optical center. *Author:* M. Mainberger.

## Notation for the Pinhole Camera Model

◆ *Attention:* Two different coordinate systems are used

- 3-D **camera coordinate system** $(X_C, Y_C, Z_C)$ with center $\mathbf{C}$
- 2-D **image coordinate system** $(x, y)$ with center $\mathbf{c}$

◆ *Notation:* The following notation is used for the different geometric entities

$\mathbf{M}$:    3-D scene point with $\mathbf{M} = (X_C, Y_C, Z_C)^\top$
$\mathbf{m}$:    2-D image point with $\mathbf{m} = (x, y)^\top$

$\mathbf{C}$:    **optical center**
$\mathbf{c}$:    **principal point** (intersection between image plane and optical axis)

$I$:    **image plane**
$f$:    **focal length** (distance image plane - optical center $\mathbf{C}$)

**optical axis:**    orthogonal to image plane, passes through $\mathbf{C}$
**optical ray:**    passes trough $\mathbf{M}$ and $\mathbf{C}$

## The Pinhole Camera Model (3)

**The Simplified Pinhole Camera Model**

◆ *Problem:* Objects appear upside down (different orientation of $x, y$ and $X_C, Y_C$)

◆ *Idea:* Use **virtual image plane** located between object and optical center
$\rightarrow$ Orientation of objects is preserved during the image acquisition process
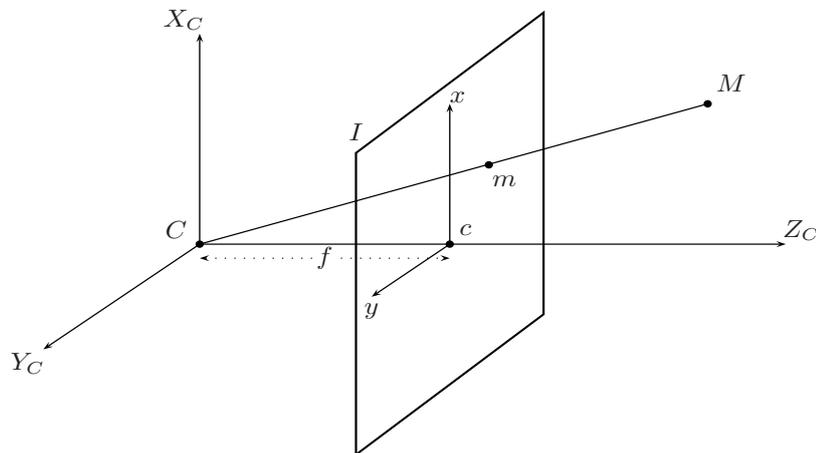


Pinhole camera with image plane in front of optical center. *Author:* M. Mainberger.

## The Pinhole Camera Model (4)

**Geometric Properties of the Simplified Pinhole Camera Model**

◆ *Properties:* The relation between the 3-D point $\mathbf{M} = (X_C, Y_C, Z_C)^\top$ and its projection $\mathbf{m} = (x, y)^\top$ is described by the **theorem of intersecting lines**

$$\frac{x}{X_C} = \frac{y}{Y_C} = \frac{f}{Z_C} \ .$$

For the coordinates of the projected point $m$ on the image plane this implies

$$x = \frac{f \cdot X_C}{Z_C} \quad \text{and} \quad y = \frac{f \cdot Y_C}{Z_C} \ .$$

◆ *Consequence:* All points on the corresponding optical ray $\lambda(X_C, Y_C, Z_C)^\top$ are projected on the same image point $\mathbf{m}$. **The original depth information is lost!**

**To overcome this depth ambiguity we need at least one additional view!**

**Projective/Homogeneous Coordinates (1)**

M I
A

| 1 | 2 |
| 3 | 4 |
| 5 | 6 |
| 7 | 8 |
| 9 | 10 |
| 11 | 12 |
| 13 | 14 |
| 15 | 16 |
| 17 | 18 |
| 19 | 20 |
| 21 | 22 |
| 23 | 24 |
| 25 | 26 |
| 27 | 28 |
| 29 | 30 |
| 31 | 32 |
| 33 | 34 |

# Projective/Homogeneous Coordinates

## How Can We Express these Nonlinear Projective Relations in a Simpler Way?

◆ *Idea:* Use homogeneous/projective coordinates that allow to describe nonlinear projective transformations in terms of **linear mappings** (matrix operations)

◆ *Definition:* The transformation from Euclidean to projective coordinates reads

$$\mathbf{x} = (x_1, ..., x_n)^\top \mapsto \tilde{\mathbf{x}} = (x_1, ..., x_n, 1)^\top$$

while the backtransform from projective to Euclidean coordinates is given by

$$\tilde{\mathbf{x}} = (\tilde{x}_1, ..., \tilde{x}_n, \tilde{x}_{n+1})^\top \mapsto \mathbf{x} = \left( \frac{\tilde{x}_1}{\tilde{x}_{n+1}}, ..., \frac{\tilde{x}_n}{\tilde{x}_{n+1}} \right)^\top .$$

◆ *Ambiguity:* All projective coordinates that are a multiple of each other, i.e. where $\tilde{\mathbf{a}} = \lambda \tilde{\mathbf{b}}$ for $\lambda \neq 0$, denote the **same point**. This reflects the depth ambiguity.

**Projective/Homogeneous Coordinates (2)**

M I
A

| 1 | 2 |
| 3 | 4 |
| 5 | 6 |
| 7 | 8 |
| 9 | 10 |
| 11 | 12 |
| 13 | 14 |
| 15 | 16 |
| 17 | 18 |
| 19 | 20 |
| 21 | 22 |
| 23 | 24 |
| 25 | 26 |
| 27 | 28 |
| 29 | 30 |
| 31 | 32 |
| 33 | 34 |

## Properties of Projective/Homogeneous Coordinates

◆ *Points at Infinity:* Points in projective coordinates for which hold $x_{n+1} = 0$ have no Euclidean equivalent. They are located at infinity. This can be seen from

$$\lim_{x_{n+1} \to 0} \left( \frac{x_1}{x_{n+1}}, ..., \frac{x_n}{x_{n+1}} \right)^\top = (\infty, ..., \infty)^\top .$$

◆ *Further Properties:* Properties of Euclidean and projective coordinates in 2-D

| Euclidean coordinates $\mathbb{R}^2$ | Projective coordinates $\mathbb{P}^2$ | Properties |
|---|---|---|
| Point $\mathbf{m} = (x, y)^\top$ | Point $\tilde{\mathbf{m}} = (\tilde{x}, \tilde{y}, \tilde{w})^\top$ | $x = \tilde{x}/\tilde{w}, \quad y = \tilde{y}/\tilde{w}$ |
| Line $ax + by + c = 0$ | Line $\tilde{a}\tilde{x} + \tilde{b}\tilde{y} + \tilde{c} = 0$ | $\tilde{a} = a, \quad \tilde{b} = b, \quad \tilde{c} = c$ |
| Point $\mathbf{m} = (x, y)^\top$ on line $\mathbf{l} = (a, b, c)^\top$ | Point $\tilde{\mathbf{m}} = (\tilde{x}, \tilde{y}, \tilde{z})^\top$ on line $\tilde{\mathbf{l}} = (\tilde{a}, \tilde{b}, \tilde{c})^\top$ | $\tilde{\mathbf{m}} \cdot \mathbf{l} = 0$ |
| Line $\mathbf{l}$ through $\mathbf{m}_1$ and $\mathbf{m}_2$ | Line $\tilde{\mathbf{l}}$ through $\tilde{\mathbf{m}}_1$ and $\tilde{\mathbf{m}}_2$ | $\mathbf{m}_1 \times \mathbf{m}_2 = \mathbf{l}$ $\Rightarrow \tilde{\mathbf{m}}_1 \times \tilde{\mathbf{m}}_2 = \mathbf{l}$ |
| Intersection of lines $\mathbf{l}_1$ and $\mathbf{l}_2$ in point $\mathbf{m}$ | Intersection of lines $\tilde{\mathbf{l}}_1$ $\tilde{\mathbf{l}}_2$ and $\tilde{\mathbf{l}}_2$ in point $\tilde{\mathbf{m}}$ | $\tilde{\mathbf{l}}_1 \times \tilde{\mathbf{l}}_2 = \tilde{\mathbf{m}}$ (parallel lines intersect at infinity) |

**Projective/Homogeneous Coordinates (3)**

M I
1 2
3 4
5 6
7 8
9 10
11 12
13 14
15 16
17 18
19 20
21 22
23 24
25 26
27 28
29 30
31 32
33 34

**How Are These Coordinates Used in Practice?**

◆ *Example*: Let us consider the simplified pinhole camera model. Using the projective coordinates $\tilde{\mathbf{m}} = (x, y, 1)^\top$ for $\mathbf{m}$ we obtain the **linear form**

$$\lambda \, \tilde{\mathbf{m}} = \lambda \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \underbrace{\begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}}_{\text{projection matrix } P_0} \begin{pmatrix} X_C \\ Y_C \\ Z_C \\ 1 \end{pmatrix} = P_0 \, \tilde{\mathbf{M}} \, .$$

where $\lambda = Z_C$ (knowing that $x = \dfrac{f \cdot X_C}{Z_C}$ and $y = \dfrac{f \cdot Y_C}{Z_C}$).

◆ *Properties:* The $3 \times 4$ projection matrix $P_0$ has rank 3 and is thus not invertible.

By integrating information on the pose and orientation of the camera as well as on its internal characteristics this matrix can become significantly more complex.
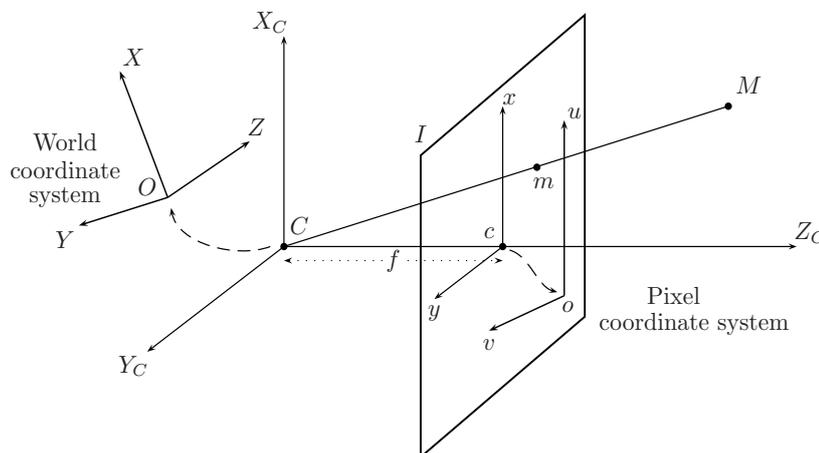
**This information is given in terms of the extrinsic and intrinsic parameters**

**Extrinsic and Intrinsic Parameters (1)**

M I
1 2
3 4
5 6
7 8
9 10
11 12
13 14
15 16
17 18
19 20
21 22
23 24
25 26
27 28
29 30
31 32
33 34

# Extrinsic and Intrinsic Parameters

**How Can We Integrate this Information?**

◆ *Idea:* Introduce world coordinate system and pixel coordinate system



Pinhole camera with world and pixel coordinate system. *Author:* M. Mainberger.

**Extrinsic and Intrinsic Parameters (2)**

M I
A

1 2
3 4
5 6
7 8
9 10
11 12
13 14
15 16
17 18
19 20
21 22
23 24
25 26
27 28
29 30
31 32
33 34

## Extrinsic Parameters

◆ *Idea:* The extrinsic parameters describes the **position and orientation** of the camera coordinate system relative to the world coordinate system

◆ *Position:* In Euclidean coordinates given by **translation vector** $\mathbf{t} = (t_1, t_2, t_3)^\top$. In projective coordinates this translation can be written as the $4 \times 4$ matrix

$$T = \left( \begin{array}{ccc|c} 1 & 0 & 0 & t_1 \\ 0 & 1 & 0 & t_2 \\ 0 & 0 & 1 & t_3 \\ \hline 0 & 0 & 0 & 1 \end{array} \right).$$

◆ *Orientation:* In Euclidean coordinates given by the $3 \times 3$ **rotation matrix** $\mathcal{R}$ with entries $r_{ij}$. In projective coordinates this rotation matrix yields the $4 \times 4$ matrix

$$R = \left( \begin{array}{ccc|c} r_{11} & r_{12} & r_{13} & 0 \\ r_{21} & r_{22} & r_{23} & 0 \\ r_{31} & r_{32} & r_{33} & 0 \\ \hline 0 & 0 & 0 & 1 \end{array} \right).$$

**Extrinsic and Intrinsic Parameters (3)**

M I
A

1 2
3 4
5 6
7 8
9 10
11 12
13 14
15 16
17 18
19 20
21 22
23 24
25 26
27 28
29 30
31 32
33 34

## Extrinsic Parameters

◆ *Example:* The transformation matrix for a rotation of $\phi_Z$ degrees around the $Z$-axis of the world coordinate system is for instance given by

$$R_Z = \left( \begin{array}{cc|c|c} \cos \phi_Z & -\sin \phi_Z & 0 & 0 \\ \sin \phi_Z & \cos \phi_Z & 0 & 0 \\ \hline 0 & 0 & 1 & 0 \\ \hline 0 & 0 & 0 & 1 \end{array} \right).$$

◆ *Extrinsic Matrix:* The overall matrix that transforms world coordinates to camera coordinates ist thus given by the concatenation of $T$ and $R$ which reads

$$A_{\text{ext}} = TR = \left( \begin{array}{ccc|c} 1 & 0 & 0 & t_1 \\ 0 & 1 & 0 & t_2 \\ 0 & 0 & 1 & t_3 \\ \hline 0 & 0 & 0 & 1 \end{array} \right) \left( \begin{array}{ccc|c} r_{11} & r_{12} & r_{13} & 0 \\ r_{21} & r_{22} & r_{23} & 0 \\ r_{31} & r_{32} & r_{33} & 0 \\ \hline 0 & 0 & 0 & 1 \end{array} \right) = \left( \begin{array}{ccc|c} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \\ \hline 0 & 0 & 0 & 1 \end{array} \right).$$

◆ *Attention:* In general the matrices $T$ and $R$ **do not commute**, i.e. $TR \neq RT$.

M I
🏛 A
1 2
3 4
5 6
7 8
9 10
11 12
13 14
15 16
17 18
19 20
21 22
23 24
25 26
27 28
29 30
31 32
33 34

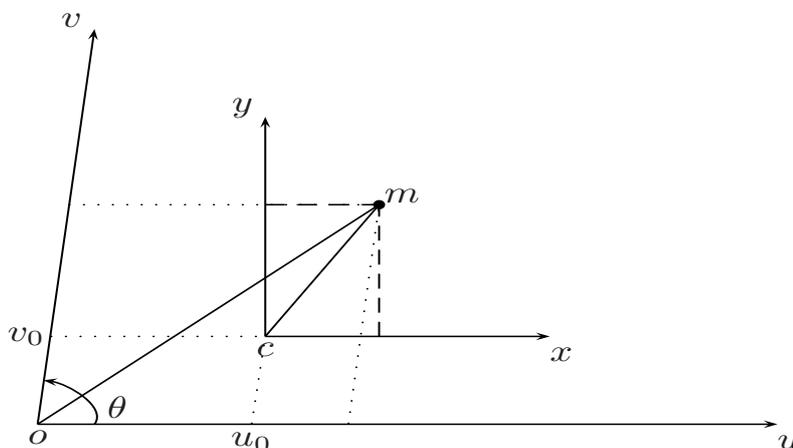**Extrinsic and Intrinsic Parameters (4)**

### Extrinsic Parameters

◆ *Properties:* The matrix $A_{\text{ext}}$ for the extrinsic parameters has rank 4, is thus invertible (in contrast to $P_0$) and offers the following **6 degrees of freedom**

- 3 degrees for the position - the translation vector $\mathbf{t} = (t_1, t_2, t_3)^\top$

- 3 degrees for the rotation - the three 1-D rotation angles $(\phi_X, \phi_Y, \phi_Z)$

◆ *Overall:* The transformation from world coordinates to image coordinates via camera coordinates is thus given by the projective transformation

$$\lambda \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \underbrace{\begin{pmatrix} \mathtt{f} & 0 & 0 & 0 \\ 0 & \mathtt{f} & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}}_{\text{projection matrix } P_0} \underbrace{\left( \begin{array}{ccc|c} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \\ \hline 0 & 0 & 0 & 1 \end{array} \right)}_{\text{extrinsic matrix } A_{\text{ext}}} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

---

**Extrinsic and Intrinsic Parameters (5)**

M I
🏛 A
1 2
3 4
5 6
7 8
9 10
11 12
13 14
15 16
17 18
19 20
21 22
23 24
25 26
27 28
29 30
31 32
33 34

### Intrinsic Parameters

◆ *Idea:* The intrinsic parameters describes the **geometry of the image plane inside the camera** with respect to the image coordinate system



Transformation from image to pixel coordinates. *Author:* M. Mainberger.

**Extrinsic and Intrinsic Parameters (6)**

M I
A

1 2
3 4
5 6
7 8
9 10
11 12
13 14
15 16
17 18
19 20
21 22
23 24
25 26
27 28
29 30
31 32
33 34

## Intrinsic Parameters

◆ *Intrinsic Matrix:* The intrinsic parameters form the $3 \times 3$ intrinsic matrix given by

$$A_{\text{int}} = \begin{pmatrix} k_u & -k_u \cot \theta & u_0 \\ 0 & k_v/\sin \theta & v_0 \\ 0 & 0 & 1 \end{pmatrix}$$

◆ *Properties:* The matrix $A_{\text{int}}$ for the intrinsic parameters has rank 3, is thus invertible (in contrast to $P_0$) and offers the following **5 degrees of freedom**

- 2 degrees for the shifted origin of the image plane - the vector $(u_0, v_0)^\top$

- 2 degrees for pixel dimensions in x- and y-direction - the width/height $k_u, k_v$

- 1 degree for the skewness of the coordinate system - the angle $\theta$

For square unit pixels one obtains the intrinsic parameters $k_u = k_v = 1$ and $\theta = \frac{\pi}{2}$.

**Extrinsic and Intrinsic Parameters (7)**

M I
A

1 2
3 4
5 6
7 8
9 10
11 12
13 14
15 16
17 18
19 20
21 22
23 24
25 26
27 28
29 30
31 32
33 34

## Intrinsic Parameters

◆ *Overall:* The complete transformation from world to pixel coordinates via camera and image coordinates is thus given by the projective transformation
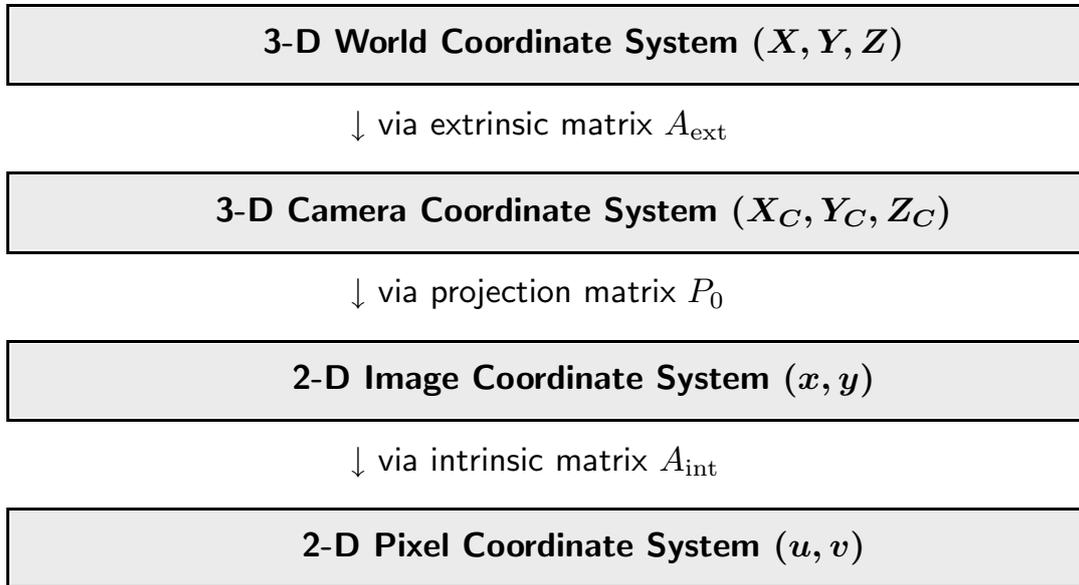
$$w \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \underbrace{\begin{pmatrix} k_u & -k_u \cot \theta & u_0 \\ 0 & k_v/\sin \theta & v_0 \\ 0 & 0 & 1 \end{pmatrix}}_{\text{inrinsic matrix } A_{\text{int}}} \underbrace{\begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}}_{\text{projection matrix } P_0} \underbrace{\begin{pmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \\ 0 & 0 & 0 & 1 \end{pmatrix}}_{\text{extrinsic matrix } A_{\text{ext}}} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

$$= \underbrace{\begin{pmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{pmatrix}}_{\text{full projection matrix } P} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

◆ *Properties:* The $3 \times 4$ full projection matrix $P$ has rank 3, is thus not invertible (due to $P_0$ that is not invertible) and offers the following **11 degrees of freedom**

- 11 degrees for the extrinsic and intrinsic parameters without focal length

- the focal length can be chosen $f = 1$ since $P$ is only defined up to a scale

- 6 or more 3-D/2-D point pairs allow to determine $P \rightarrow$ **calibrated camera**

| M | I |
|---|---|
| 🏛 | A |
| 1 | 2 |
| 3 | 4 |
| 5 | 6 |
| 7 | 8 |
| 9 | 10 |
| 11 | 12 |
| 13 | 14 |
| 15 | 16 |
| 17 | 18 |
| 19 | 20 |
| 21 | 22 |
| 23 | 24 |
| 25 | 26 |
| 27 | 28 |
| 29 | 30 |
| 31 | 32 |
| 33 | 34 |

## From World Coordinates to Pixel Coordinates

◆ *Projective Geometry:* In total we have 4 different coordinate systems

$$\boxed{\textbf{3-D World Coordinate System } (X, Y, Z)}$$

↓ via extrinsic matrix $A_{\text{ext}}$

$$\boxed{\textbf{3-D Camera Coordinate System } (X_C, Y_C, Z_C)}$$

↓ via projection matrix $P_0$

$$\boxed{\textbf{2-D Image Coordinate System } (x, y)}$$

↓ via intrinsic matrix $A_{\text{int}}$

$$\boxed{\textbf{2-D Pixel Coordinate System } (u, v)}$$

---

| M | I |
|---|---|
| 🏛 | A |
| 1 | 2 |
| 3 | 4 |
| 5 | 6 |
| 7 | 8 |
| 9 | 10 |
| 11 | 12 |
| 13 | 14 |
| 15 | 16 |
| 17 | 18 |
| 19 | 20 |
| 21 | 22 |
| 23 | 24 |
| 25 | 26 |
| 27 | 28 |
| 29 | 30 |
| 31 | 32 |
| 33 | 34 |

# Epipolar Geometry

## How Can We Overcome The Depth Ambiguity?

◆ *Idea:* Consider a second camera that provides a different view of the scene

◆ *Example 1:* If the optical axis of both cameras are slightly rotated towards each other, one obtains the following so-called **converging camera setup**
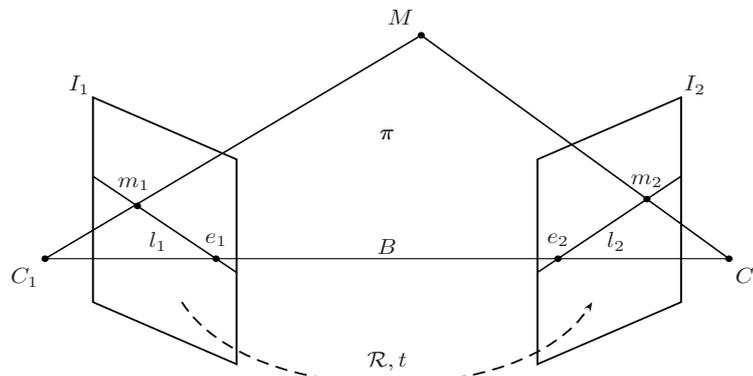


Converging camera setup. *Author:* M. Mainberger.

| M | I |
|---|---|
|   | A |
| 1 | 2 |
| 3 | 4 |
| 5 | 6 |
| 7 | 8 |
| 9 | 10 |
| 11 | 12 |
| 13 | 14 |
| 15 | 16 |
| 17 | 18 |
| 19 | 20 |
| 21 | 22 |
| 23 | 24 |
| 25 | 26 |
| 27 | 28 |
| 29 | 30 |
| 31 | 32 |
| 33 | 34 |

## Epipolar Geometry (2)

### Notation for a System with Two Cameras

◆ *Previous Notation:* Notation for the two cameras separately

$\mathbf{M}$:          3-D scene point with $\mathbf{M} = (X, Y, Z)^\top$

$\mathbf{m}_1, \mathbf{m}_2$:   2-D image points in both images planes with $\mathbf{m}_i = (x_i, y_i)^\top$

$\mathbf{C}_1, \mathbf{C}_2$:   optical centers of both cameras

$I_1, I_2$:      image planes of both cameras

◆ *Additional Notation:* Notation for interaction between the two cameras

$\mathcal{R}, \mathbf{t}$:    rotation/translation between $\mathbf{C}_1/\mathbf{C}_2$ (relative extrinsic parameters)

$B$:          **baseline** (distance between optical centers $\mathbf{C}_1$ and $\mathbf{C}_2$)

$\mathbf{l}_1, \mathbf{l}_2$:   **epipolar lines** (projections of the optical rays)

$\mathbf{e}_1, \mathbf{e}_2$:   **epipoles** (projection of the optical centers)

$\pi$:          **epipolar plane** (plane through $\mathbf{C}_1$, $\mathbf{C}_2$ and $\mathbf{M}$)

## Epipolar Geometry (3)

### Special Setup For Two Cameras

◆ *Idea:* Choose camera setup such that all epipolar lines are horizontal and parallel

◆ *Example 2:* If the optical axis of both cameras are parallel to each other, one obtains the **ortho-parallel camera setup** (only translation of optical centers)
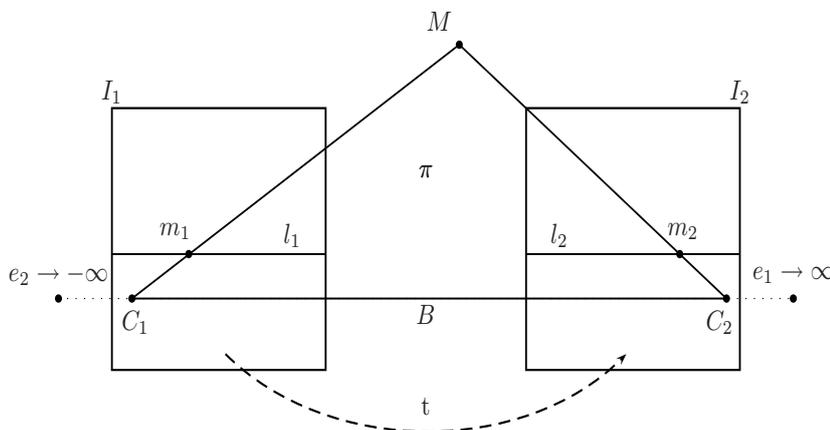


Ortho-parallel camera setup. *Author:* M. Mainberger.

**Epipolar Geometry (4)**

M I
🏰 A
1 2
3 4
5 6
7 8
9 10
11 12
13 14
15 16
17 18
19 20
21 22
23 24
25 26
27 28
29 30
31 32
33 34

## Epipolar Constraint and Fundamental Matrix

◆ *Observation:* The pixel $\mathbf{m}_2$ in the second image that corresponds to the pixel $\mathbf{m}_1$ in the first image cannot lie everywhere. It must fulfil the **epipolar constraint** *(Luong/Faugeras 1993)*

$$\tilde{\mathbf{m}}_2^\top \mathcal{F} \tilde{\mathbf{m}}_1 = 0 \quad \text{with} \quad \mathcal{F} = A_{\text{int}2}^{-\top}[\mathbf{t}]_\times \mathcal{R} \, A_{\text{int}1}^{-1}$$

where $[\mathbf{t}]_\times$ realises the cross product of $\mathbf{t}$ with an argument $\mathbf{m}$ via multiplication

$$\mathbf{t} \times \mathbf{m} = [\mathbf{t}]_\times \mathbf{m} \quad \text{with} \quad [\mathbf{t}]_\times = \begin{bmatrix} t_x \\ t_y \\ t_z \end{bmatrix}_\times = \begin{pmatrix} 0 & -t_z & t_y \\ t_z & 0 & -t_x \\ -t_y & t_x & 0 \end{pmatrix} \, .$$

This constraints states implicitly that optical rays and baseline are **coplanar**.

◆ *Properties:* The $3 \times 3$ matrix $\mathcal{F}$ is called **fundamental matrix**. It has rank 2, is thus not invertible and offers 7 degrees of freedom (-scale,-rank). It is not possible to derive the intrinsic and relative extrinsic parameters from $\mathcal{F}$.

**Epipolar Geometry (5)**

M I
🏰 A
1 2
3 4
5 6
7 8
9 10
11 12
13 14
15 16
17 18
19 20
21 22
23 24
25 26
27 28
29 30
31 32
33 34

## Epipolar Lines and Reduction of Search Space

◆ *Consequence:* If the fundamental matrix is known (**weakly calibrated system**) one knows for each pixel $\mathbf{m}_1$ in the first frame the corresponding epipolar line $\mathbf{l}_2$ in the second frame and vice versa

$$\tilde{\mathbf{m}}_2^\top \mathbf{l}_2 = 0 \qquad \text{with} \qquad \mathbf{l}_2 = \mathcal{F} \tilde{\mathbf{m}}_1 \, ,$$
$$\tilde{\mathbf{m}}_1^\top \mathbf{l}_1 = 0 \qquad \text{with} \qquad \mathbf{l}_1 = \mathcal{F}^\top \tilde{\mathbf{m}}_2 \, .$$

→ **Reduced search space (1-D)** for stereo matching algorithms (**epipolar line**) Ortho-parallel cameras even yield horizontal epipolar lines (search in $x$-direction)

◆ *General Case:* If the fundamental matrix is not known (**uncalibrated system**) one can estimate it from at least 7 point correspondences ($\rightarrow$ 2-D search space)

**How this fundamental matrix can be estimated is topic of the next lecture**

## Epipolar Geometry (6)

M I
A
1 2
3 4
5 6
7 8
9 10
11 12
13 14
15 16
17 18
19 20
21 22
23 24
25 26
27 28
29 30
31 32
33 34

### Relation to the Essential Matrix

◆ *Typical Problem:* The intrinsic parameters of both cameras are known but the relative position and orientation is not known (separate camera calibration)

◆ *Simplified Constraint:* Knowing $A_{\mathrm{int}1}$ and $A_{\mathrm{int}2}$ the epipolar constraint can be formulated in image coordinates via $\tilde{\mathbf{m}}_1' = A_{\mathrm{int}1}\tilde{\mathbf{m}}_1^{-1}$ and $\tilde{\mathbf{m}}_2' = A_{\mathrm{int}2}^{-1}\tilde{\mathbf{m}}_2$ as
*(Longuet-Higgins 1982)*

$$\tilde{\mathbf{m}}_2'^\top \, \mathcal{E} \, \tilde{\mathbf{m}}_1' = 0 \quad \text{with} \quad \mathcal{E} = [\mathbf{t}]_\times \mathcal{R} \; .$$

◆ *Properties:* The antisymmetric $3 \times 3$ matrix $\mathcal{E}$ is called **essential matrix**. It has rank 2, is thus not invertible and offers 5 degrees of freedom (-scale) It holds

$$\mathcal{F} = A_{\mathrm{int}2}^{-\top} \mathcal{E} A_{\mathrm{int}1}^{-1} \; .$$

Using 5 point correspondences or more one can estimate the entries of $\mathcal{E}$. It is even possible to derive the relative extrinsic parameters $\mathcal{R}$ and $\mathbf{t}$ from $\mathcal{E}$!

---

## Epipolar Geometry (7)

M I
A
1 2
3 4
5 6
7 8
9 10
11 12
13 14
15 16
17 18
19 20
21 22
23 24
25 26
27 28
29 30
31 32
33 34

### Fundamental vs. Essential Matrix

◆ Comparison of the fundamental matrix and the essential matrix

| Fundamental Matrix $\mathcal{F}$ | |
| --- | --- |
| Determinate | $\det(\mathcal{F}) = 0$, due to $\det(\mathcal{E}) = 0$ |
| Rank, D.O.F. | $\mathcal{F}$ is a projective matrix of rank 2 with 7 degrees of freedom |
| Transpose | $\mathcal{F}$ is the fundamental matrix of the cameras $(\mathbf{C}_1, \mathbf{C}_2)$ $\Rightarrow \mathcal{F}^\top$ is the fundamental matrix of the cameras $(\mathbf{C}_2, \mathbf{C}_1)$ |
| Epipolar lines | $\mathbf{l}_2 = \mathcal{F}\tilde{\mathbf{m}}_1$ and $\mathbf{l}_1 = \mathcal{F}\tilde{\mathbf{m}}_2$ are the epipolar lines with respect to $\tilde{\mathbf{m}}_1$ and $\tilde{\mathbf{m}}_2$ |
| Epipoles | $\tilde{\mathbf{e}}_1$ and $\tilde{\mathbf{e}}_2$ epipoles $\Rightarrow \mathcal{F}\tilde{\mathbf{e}}_1 = 0$ and $\mathcal{F}^\top \tilde{\mathbf{e}}_2 = 0$ |

| Essential Matrix $\mathcal{E}$ | |
| --- | --- |
| Determinate | $\det(\mathcal{E}) = \det([t]_\times)\det(\mathcal{R}) = 0$ due to antisymmetry of $[t]_\times$ |
| Rank, D.O.F. | $\mathcal{F}$ is a Euclidean matrix of rank 2 with 5 degrees of freedom |
| Singular values | $\mathcal{E}$ has three eigenvalues, two are equal, the third is zero |

**Reconstruction of the Depth (1)**

M I
A
1 2
3 4
5 6
7 8
9 10
11 12
13 14
15 16
17 18
19 20
21 22
23 24
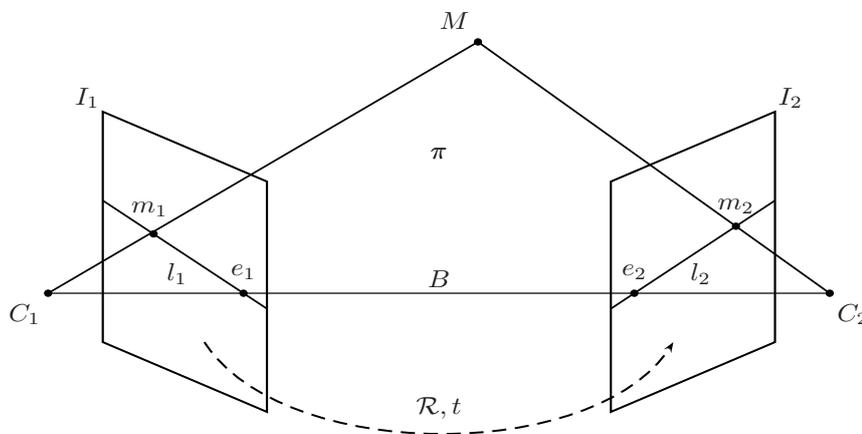25 26
27 28
29 30
31 32
33 34

# Reconstructing the Depth

◆ *Idea:* Recover the depth from the displacement field between $I_1$ and $I_2$

◆ *Example:* If we know the displacement field (disparity) between both frames and we know the intrinsic and relative extrinsic parameters of the system we can

1. specify origin and orientation of world coordinate system relative to both camera systems and compute extrinsic parameters (if not given)
2. compute $\mathbf{m}_2$ by adding the calculated displacement $\mathbf{u}$ to $\mathbf{m}_1$
3. compute the optical ray through $\mathbf{m}_1$ in world coordinates
4. compute the optical ray through $\mathbf{m}_2$ in world coordinates
5. intersect both optical rays $\rightarrow$ one obtains a 3-D point $\mathbf{M}$
6. $Z$-component of $\mathbf{M}$ specifies depth with respect to world coordinates

One often aligns the world coordinate system with the first camera system
$\rightarrow$ computed depth refers to the view of the first camera

**Reconstructing the Depth (2)**

M I
A
1 2
3 4
5 6
7 8
9 10
11 12
13 14
15 16
17 18
19 20
21 22
23 24
25 26
27 28
29 30
31 32
33 34

## Converging Camera Setup

◆ *Illustration:* Intersection of both optical rays yields the original 3-D point



Intersection of optical rays. *Author:* M. Mainberger.

**Reconstructing the Depth (3)**

M I
A

1 2
3 4
5 6
7 8
9 10
11 12
13 14
15 16
17 18
19 20
21 22
23 24
25 26
27 28
29 30
31 32
33 34

## Converging Camera Setup

◆ *Problem:* If displacements are not perfect both optical rays will not intersect

◆ *Solution:* Use closest point to both optical rays as intersection point
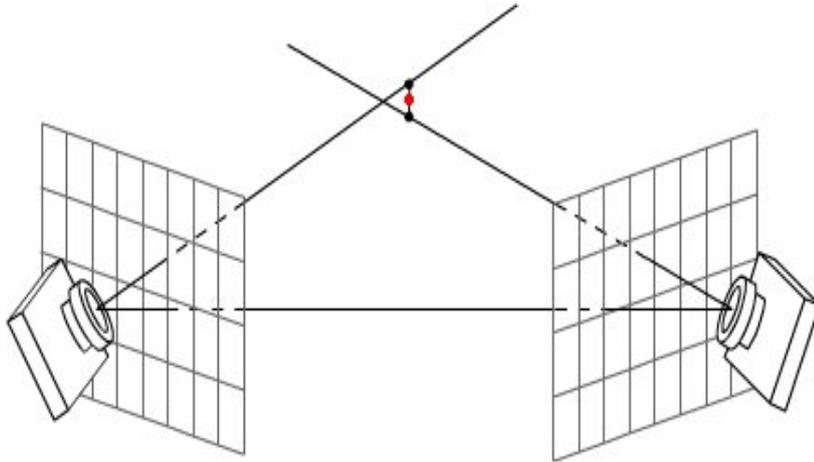


Converging camera setup with non-perfect displacements. *Author:* N. Slesareva.

**Reconstructing the Depth (4)**

M I
A

1 2
3 4
5 6
7 8
9 10
11 12
13 14
15 16
17 18
19 20
21 22
23 24
25 26
27 28
29 30
31 32
33 34

## Ortho-Parallel Camera Setup

◆ *Depth Recovery:* Optical rays always intersect, depth can be computed directly
For a horizontal displacement $d = x_1 - x_2$ the theorem of intersecting lines yields

$$\frac{Z}{f} = \frac{B \, k_u}{d} \qquad \text{wich gives} \qquad Z = \frac{f \, B \, k_u}{d} \; .$$
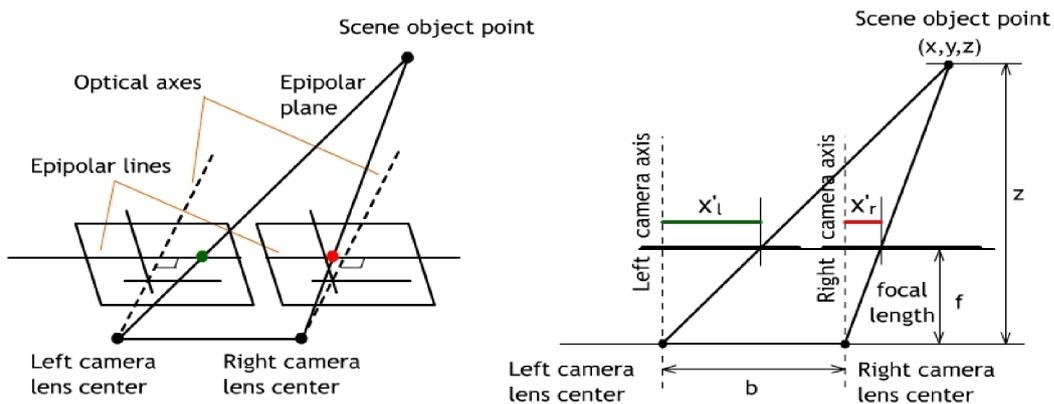


Depth recovery for an ortho-parallel camera setup. *Author:* N. Slesareva.

## Summary (1)

# Summary

- ◆ Projective/homogeneous coordinates offer a useful and compact notation
- ◆ The geometry of one view → **Projective geometry**
  - • can be represented via a single projection matrix
  - • can be decomposed into intrinsic and extrinsic parameters
- ◆ The geometry of two views → **Epipolar geometry**
  - • fundamental matrix as direct relation between two views
  - • essential matrix as Euclidean relation between two cameras
- ◆ The stereo reconstruction problem is only a 1-D search problem
  - • epipolar constraint reduces the search space for stereo matching algorithms from 2-D to a 1-D (search along the epipolar line)

## Summary (2)

### Literature

- ◆ O. Faugeras, Q.-T. Luong:
  *The Geometry of Multiple Images.*
  MIT Press, 2001.
  (*book on projective geometry and stereo reconstruction*)

- ◆ R. Hartley, A. Zisserman:
  *Multiple View Geometry in Computer Vision.*
  Cambridge University Press, 2000.
  (*more practical book on projective geometry and stereo reconstruction*)

- ◆ Y. Ma, S. Soatto, J. Kosecka, S. Sastry:
  *An Invitation to 3-D Vision.*
  Springer, 2004.
  (*more theoretical book on projective geometry and stereo reconstruction*)

- ◆ E. Trucco, A. Verri:
  *Introductory Techniques for 3-D Computer Vision.*
  Prentice Hill, 1998.
  (*introduction to computer vision with chapters on stereo*)

**Assignment 7**

M I
A
1 2
3 4
5 6
7 8
9 10
11 12
13 14
15 16
17 18
19 20
21 22
23 24
25 26
27 28
29 30
31 32
33 34

# Assignment 7

## Programming Exercise 1 (Coarse-to-Fine Warping)

You can download the file copcv09_ex07.tgz from the web page

    http://www.mia.uni-saarland.de/Teaching/copcv09.shtml

To unpack these data, use `tar xzvf copcv09_ex07.tgz`.

1. Take a look a the recursive routine `horn_schunck_warp()` that implements the actual warping strategy. Use the corresponding script and try to follow the implementation step by step.

2. Supplement the routine `horn_schunck_warp_sor.c` with missing code so that it implements the SOR method for the Horn and Schunck method with warping (see Assignment 6). For compiling your programme please use the contained makefile. The compiled programme is then executed by

    `./frontend <input_image1.pgm> <input_image2.pgm> <zoom_ratio> [ground_truth.F]`

   where the integer parameter `zoom_ratio` is in general set to 1. The use of a ground truth file `ground_truth.F` is optional and triggers the computation of the average angular error (AAE).

3. Use the provided image pair `yos1.pgm` and `yos2.pgm` to optimise your results with respect to the average angular error (AAE). Can you obtain better results than the ones given in the script?

**Assignment 7**

M I
A
1 2
3 4
5 6
7 8
9 10
11 12
13 14
15 16
17 18
19 20
21 22
23 24
25 26
27 28
29 30
31 32
33 34

# Assignment 7

## Programming Exercise 2 (Backward Registration II)

Use the same code for the second task.

4. As in Assignment 5 you can write out the motion compensated second frame by pressing $F8$ after computing a displacement field. This frame is then written out as file `frame2_bw.pgm`. Use the Linux command

   `animate <frame1.pgm> <frame2_bw.pgm`

   to compare the first and the motion compensated second frame visually for your best results for the sequences `yos1.pgm` and `yos2.pgm` as well as `rhein1.pgm` and `rhein2.pgm`. Compare your results with the ones you obtain without warping.