Humanoid Robotics

Camera Parameters

Maren Bennewitz
Goal of This Chapter

- Understand the mapping from the world coordinate system to the sensor coordinate system of the robot
- Know how such transformations can be described by matrices
- Know which camera parameters influence the mapping
- Understand why calibration of camera parameters is needed
Camera Calibration

- A camera projects 3D world points onto the 2D image plane

- **Calibration**: Find the internal quantities of the camera that affect this process
  - Image center
  - Focal length (camera constant)
  - Lens distortion parameters
Why is Calibration Needed?

- Camera production errors
- Cheap lenses

Precise calibration is required for
- 3D interpretation of images
- Re-construction of world models
- Robot interaction with the world (hand-eye coordination)
Three Assumptions Made for the Pinhole Camera Model

1. All rays from the object intersect in a single point
2. All image points lie on a plane
3. The ray from the object point to the image point is a straight line

Often these assumptions do not hold and lead to imperfect images
Lens vs. Pinhole

- A lens is only an approximation of the pinhole camera model
- The corresponding points on the object and in the image, and the center of the lens typically do not lie on one line
- The further away a beam passes the center of the lens, the larger the error
Coordinate Frames

1. World coordinate frame
2. Camera coordinate frame
3. Image coordinate frame
4. Sensor coordinate frame
Coordinate Frames

1. World coordinate frame $S_o$
   written as: $[X, Y, Z]^T$

2. Camera coordinate frame $S_k$
   written as: $[^kX, ^kY, ^kZ]^T$

3. Image coordinate frame $S_c$
   written as: $[^cX, ^cY]^T$

4. Sensor coordinate frame $S_s$
   written as: $[^sX, ^sY]^T$
Transformation

We want to compute the mapping

\[
\begin{bmatrix}
  s\bar{x} \\
  s\bar{y} \\
  1
\end{bmatrix} = sH_c^cH_k^kH_o^o
\begin{bmatrix}
  X \\
  Y \\
  Z \\
  1
\end{bmatrix}
\]

in the sensor frame to image to sensor to image to camera to world in the world frame
Visualization

Image courtesy: Förstner
From the World to the Sensor

- From the World to the Sensor
- Ideal projection (3D to 2D)
- Image to sensor frame (2D)
- Deviation from the linear model (2D)

world to camera frame (3D)
Extrinsic & Intrinsic Parameters

- **Extrinsic parameters** describe the pose of the camera in the world
- **Intrinsic parameters** describe the mapping of the scene in front of the camera to the pixels in the final image (sensor)
Extrinsic Parameters

- Pose of the camera with respect to the world
- Invertible transformation

How many parameters are needed?

6 parameters: 3 for the position + 3 for the orientation
Extrinsic Parameters

- Point $\mathcal{P}$ with coordinates in world coordinates
  \[ X_P = [X_P, Y_P, Z_P]^T \]
- Origin of the camera frame
  \[ X_O = [X_O, Y_O, Z_O]^T \]
Extrinsic Parameters

- Point $\mathcal{P}$ with coordinates in world coordinates

$$X_P = [X_P, Y_P, Z_P]^T$$

- Origin of the camera frame

$$X_O = [X_O, Y_O, Z_O]^T$$

In the following, we will see why H.C. are the better choice for describing transformations compared to Euclidean coordinates
Transformation

- **Translation** between the origin of the world frame and the camera frame

\[ X_O = [X_O, Y_O, Z_O]^\top \]

- **Rotation** $R$ from the frame $S_o$ to $S_k$

- In Euclidian coordinates this yields

\[ ^kX_P = R(X_P - X_O) \]
Transformation in H.C.

- In Euclidian coordinates: \( kX_P = R(X_P - X_O) \)
- Expressed in Homogeneous Coord.

\[
\begin{bmatrix}
  kX_P \\
  1
\end{bmatrix}
= \begin{bmatrix}
  R & 0 \\
  0^T & 1
\end{bmatrix}
\begin{bmatrix}
  I_3 \\
  0^T \\
  1
\end{bmatrix}
\begin{bmatrix}
  -X_O \\
  1
\end{bmatrix}
\begin{bmatrix}
  X_P \\
  1
\end{bmatrix}
= \begin{bmatrix}
  R \\
  0^T \\
  1
\end{bmatrix}
\begin{bmatrix}
  X_P \\
  1
\end{bmatrix}
\]

- or written as

\[
kX_P = kH \cdot X_P \quad \text{with} \quad kH = \begin{bmatrix}
  R \\
  0^T \\
  -RX_O \\
  1
\end{bmatrix}
\]
Intrinsic Parameters

- The process of projecting points from the camera frame to the sensor frame
- Invertible transformations:
  - image plane to sensor frame
  - model deviations
- Not directly invertible: projection
Ideal Perspective Projection

We split up the mapping into 3 steps

1. Ideal perspective projection to the image plane
2. Shifting to the sensor coordinate frame (pixel)
3. Compensation for the fact that the two previous mappings are idealized
Image Coordinate System

Physically motivated coordinate system:
c > 0

rotation by 180 deg

Image courtesy: Förstner
Camera Constant $c$

- Distance between the center of projection $O$ and the principal point $\mathcal{H}$
- Value is computed as part of the camera calibration
- Here coordinate system with $c < 0$

Image courtesy: Förstner
Ideal Perspective Projection

Through the intercept theorem, we obtain for the point $\overline{P}$ in the image plane the coordinates $[^{c}x_{\overline{P}}, ^{c}y_{\overline{P}}]$

$$^{c}x_{\overline{P}} := k \overline{X_{P}} = \frac{kX_{P}}{kZ_{P}}$$

$$^{c}y_{\overline{P}} := k \overline{Y_{P}} = \frac{kY_{P}}{kZ_{P}}$$

coordinates wrt. camera frame (3D)

coordinates in image frame (2D)

[in Euclidean coordinates]
In Homogenous Coordinates

We can express that in H.C. as:

\[
^c x_P = \begin{bmatrix}
^c u_P \\
^c v_P \\
^c w_P \\
1
\end{bmatrix} = \begin{bmatrix}
c & 0 & 0 & 0 \\
0 & c & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix} \begin{bmatrix}
k X_P \\
k Y_P \\
k Z_P \\
1
\end{bmatrix}
\]
Verify the Transformation

- Before, we derived

\[ c x_{\mathcal{P}} = c \frac{kX_{\mathcal{P}}}{kZ_{\mathcal{P}}} \quad c y_{\mathcal{P}} = c \frac{kY_{\mathcal{P}}}{kZ_{\mathcal{P}}} \]

- Now, we have to verify

\[
\begin{bmatrix}
  c x_{\mathcal{P}} \\
  c y_{\mathcal{P}} \\
  1
\end{bmatrix}
= \begin{bmatrix}
  c & 0 & 0 & 0 \\
  0 & c & 0 & 0 \\
  0 & 0 & 1 & 0
\end{bmatrix}
\begin{bmatrix}
  kX_{\mathcal{P}} \\
  kY_{\mathcal{P}} \\
  kZ_{\mathcal{P}} \\
  1
\end{bmatrix}
\]
In Homogenous Coordinates

- Thus, we can write for any point

\[ \begin{align*} 
{^c}_P x_{P} &= {^c}_P P_k {^k}X_P 
\end{align*} \]

- with

\[ {^c}_P P_k = \begin{bmatrix} 
  c & 0 & 0 & 0 \\
  0 & c & 0 & 0 \\
  0 & 0 & 1 & 0 \\
 0 & 0 & 0 & 1 \\
\end{bmatrix} \]

- This defines the projection from a point in the camera frame into the image frame
Assuming an Ideal Camera

- This leads to the mapping using the intrinsic and extrinsic parameters

\[ \mathbf{c}\mathbf{x} = \mathbf{cP}\mathbf{X} \]

- with

\[ \mathbf{cP} = \mathbf{cP}_k \mathbf{H} = \begin{bmatrix} c & 0 & 0 & 0 \\ 0 & c & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{R} \\ \mathbf{0}^T \\ -\mathbf{RX}_O \end{bmatrix} \]

- Transformation from the world frame into the camera frame, followed by the projection into the image frame
Calibration Matrix

- Calibration matrix for the ideal camera:

\[
^{c}K = \begin{bmatrix}
c & 0 & 0 \\
0 & c & 0 \\
0 & 0 & 1
\end{bmatrix}
\]

- We can write the overall mapping as

\[
^{c}P = ^{c}K[R[I_3] - RX_O] = ^{c}K[R[I_3] - X_O]
\]
Notation

We can write the overall mapping as

$$cP = cK[R - RX_O] = cK \begin{bmatrix} l_3 \end{bmatrix} - X_O$$

short for

$$\begin{bmatrix} l_3 \end{bmatrix} - X_O = \begin{bmatrix} 1 & 0 & 0 & -X_O \\ 0 & 1 & 0 & -Y_O \\ 0 & 0 & 1 & -Z_O \end{bmatrix}$$
**Calibration Matrix**

- We have the projection

\[ ^cP = ^cK R \left[ I_3 - X_O \right] \]

- that maps a point to the image frame

\[ ^cX = ^cKR[ I_3 - X_O ]X \]

- and yields for the coordinates of \(^cX\)

\[
\begin{bmatrix}
^c u' \\
^c v' \\
^c w'
\end{bmatrix}
= \begin{bmatrix}
c & 0 & 0 \\
0 & c & 0 \\
0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
r_{11} & r_{12} & r_{13} \\
r_{21} & r_{22} & r_{23} \\
r_{31} & r_{32} & r_{33}
\end{bmatrix}
\begin{bmatrix}
X - X_O \\
Y - Y_O \\
Z - Z_O
\end{bmatrix}
\]
In Euclidian Coordinates

As comparison: image coordinates in Euclidian coordinates

\[
\begin{align*}
  c_x &= c \frac{r_{11}(X - X_O) + r_{12}(Y - Y_O) + r_{13}(Z - Z_O)}{r_{31}(X - X_O) + r_{32}(Y - Y_O) + r_{33}(Z - Z_O)} \\
  c_y &= c \frac{r_{21}(X - X_O) + r_{22}(Y - Y_O) + r_{23}(Z - Z_O)}{r_{31}(X - X_O) + r_{32}(Y - Y_O) + r_{33}(Z - Z_O)}
\end{align*}
\]
Extrinsic & Intrinsic Parameters

Extrinsics

Intrinsics
Mapping to the Sensor Frame

- Next step: mapping from the image plane to the sensor frame
- Assuming linear errors
- Take into account:
  - Location of the principal point in the image plane (offset)
  - Scale difference in x and y based on the chip design
Location of the Principal Point

- The origin of the sensor frame (0,0) is not at the principal point
- Compensate the offset by a shift

\[ sH_c = \begin{bmatrix} 1 & 0 & x_H \\ 0 & 1 & y_H \\ 0 & 0 & 1 \end{bmatrix} \]
Scale Difference

- Scale difference $m$ in x and y

\[
{sH_c} = \begin{bmatrix}
1 & 0 & x_H \\
0 & 1 + m & y_H \\
0 & 0 & 1
\end{bmatrix}
\]

- Resulting mapping into the sensor frame:

\[
{sX} = {sH_c} {cKR[I_3 | - X_O]} X
\]
Calibration Matrix

The transformation is combined with the calibration matrix:

\[
K = s H_c^c K
\]

\[
= \begin{bmatrix}
1 & 0 & x_H \\
0 & 1 + m & y_H \\
0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
c & 0 & 0 \\
0 & c & 0 \\
0 & 0 & 1
\end{bmatrix}
\]

\[
= \begin{bmatrix}
c & 0 & x_H \\
0 & c(1 + m) & y_H \\
0 & 0 & 1
\end{bmatrix}
\]
**Calibration Matrix**

- The calibration matrix is an affine transformation:

\[ K = \begin{bmatrix} c & 0 & x_H \\ 0 & c(1 + m) & y_H \\ 0 & 0 & 1 \end{bmatrix} \]

- Contains 4 parameters:
  - Camera constant: \( c \)
  - Principal point: \( x_H, y_H \)
  - Scale difference: \( m \)
Non-Linear Errors

- So far, we considered only linear parameters
- The real world is non-linear
Non-Linear Errors

- So far, we considered only **linear** parameters
- The real world is **non-linear**
- Reasons for non-linear errors
  - Imperfect lens
  - Non-planarity of the sensor
  - ...

![Diagram](image.png)
Example

not straight line preserving  rectified image

Image courtesy: Abraham
General Mapping

- Add a last step that covers the non-linear effects
- Location-dependent shift in the sensor coordinate system
- Individual shift for each pixel according to the distance from the image center

\[
\begin{align*}
ax &= sx + \Delta x(x, q) \\
y &= sy + \Delta y(x, q)
\end{align*}
\]
Example: Distortion

- Approximation of the distortion

\[
\begin{align*}
\alpha x &= x(1 + q r^2) \\
\alpha y &= y(1 + q r^2)
\end{align*}
\]

- With \( r \) as the distance to the image center

- The term \( q \) is the additional parameter of the general mapping
General Mapping in H.C.

- General mapping yields
  \[ a\mathbf{x} = a\mathbf{H_s}(\mathbf{x}, q)^s \mathbf{x} \]
  with
  \[ a\mathbf{H_s}(\mathbf{x}, q) = \begin{bmatrix} 1 & 0 & \Delta x(\mathbf{x}, q) \\ 0 & 1 & \Delta y(\mathbf{x}, q) \\ 0 & 0 & 1 \end{bmatrix} \]

- The overall mapping then becomes
  \[ a\mathbf{x} = a\mathbf{H_s}(\mathbf{x}, q) KR[I_3| - X_O]X \]
General Calibration Matrix

- General calibration matrix is obtained by combining the one of the affine transform with the general mapping

\[ \overset{a}{K}(x, q) = \overset{a}{H}_s(x, q) K \]

\[
\begin{bmatrix}
c & 0 & x_H + \Delta x(x, q) \\
0 & c(1 + m) & y_H + \Delta y(x, q) \\
0 & 0 & 1
\end{bmatrix}
\]

- This results in the general projection

\[ \overset{a}{x} = \overset{a}{P}(x, q) X \]

\[ \overset{a}{P}(x, q) = \overset{a}{K}(x, q) R[l| - X_O] \]
Calibrated Camera

- If the intrinsics are unknown, we call the camera uncalibrated.
- If the intrinsics are known, we call the camera calibrated.
- The process of obtaining the intrinsics is called camera calibration.
Camera Calibration

Calculate intrinsic parameters from a series of images

- 2D camera calibration
- 3D camera calibration
- Self-calibration (next lecture)
Summary

- Mapping from the world frame to the sensor frame
- **Extrinsics** = world to camera frame
- **Intrinsics** = camera to sensor frame
- Assumption: Pinhole camera model
- Non-linear model for lens distortion
- We need to know the camera parameters to
  - Map from the world coordinate system to the sensor coordinate system
  - Realize robot interaction with the real world
Literature

- Multiple View Geometry in Computer Vision, R. Hartley and A. Zisserman, Ch. 6

- Slides partially based on Chapter 16 “Camera Extrinsics and Intrinsics”, Photogrammetry I by C. Stachniss