Camera Calibration and Image Rectification[1]

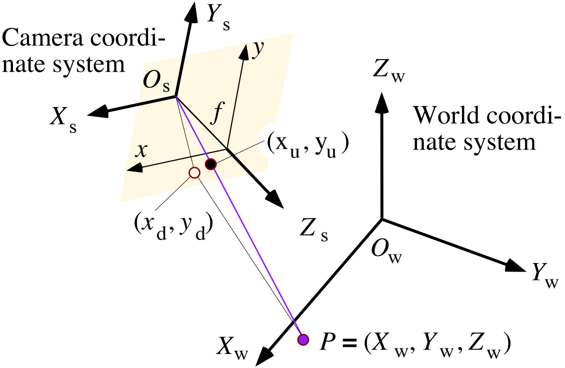
Lecture 14

See Section 6.3 in
Reinhard Klette: Concise Computer Vision
Springer-Verlag, London, 2014

[1]See last slide for copyright information.

Coordinate Systems and Calibration

World coordinates $X_w Y_w Z_w$ and camera coordinates $X_s Y_s Z_s$



We need to specify all the parameters involved

Agenda

1. Camera Calibration Basics
2. Involved Transforms
3. Point Localization
4. Rectification of Stereo Image Pairs
5. Fundamental and Essential Matrix of Stereo Vision

Intrinsic and Extrinsic Parameters

Intrinsic: camera-specific, also called *internal*

Examples: (effective) focal length, dimensions of the sensor matrix, sensor cell size or aspect ratio of sensor height to width, radial distortion parameters, coordinates of the principal point, or the scaling factor

Extrinsic: parameters of one- or multi-camera configuration

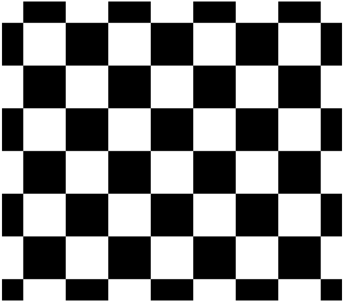
Parameters of the applied affine transforms for identifying *poses* (i.e. location and direction) of cameras in a world coordinate system

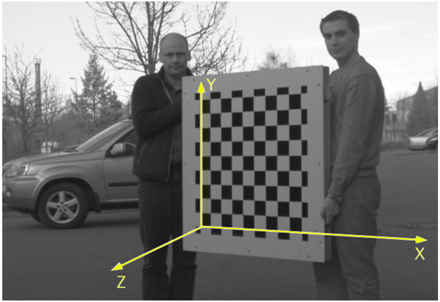
Goal: sufficient background for understanding what is happening *in principle*

No details of any particular calibration method

A User's Perspective on Camera Calibration

Camera-producer specifies some internal parameters (e.g. the physical size of sensor cells), but given data often not accurate enough for computer vision





2D checkerboard pattern used for camera calibration; a portable calibration rig

Where is the world coordinate system?

A Quick Guide

Use geometric pattern on 2D or 3D surfaces for which we are able to measure them very accurately

Calibration rig: either attached to walls or moving in front of the camera system

Used geometric patterns are recorded, localized in images, appearance in the image compared with available measurements about their geometry

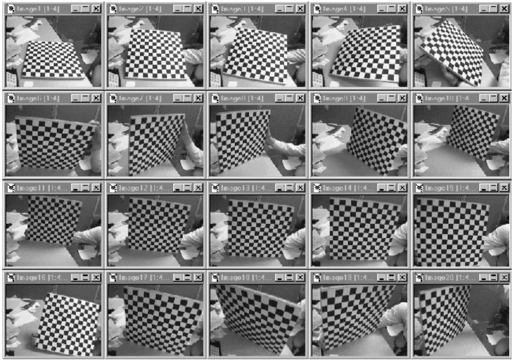
Calibration needs to be redone from time to time

Calibration of a multi-camera system: all cameras need to be exactly time-synchronized

Calibration Software

Calibration software is available online

Example: C sources provided by J.-Y. Bouget



`www.vision.caltech.edu/bouguetj/calib_doc/`
or in the `OpenCV` library

Agenda

List of Transforms

Each camera has its own coordinate system

Calibration rig defines world coordinates at the moment when taking an image

1. Coordinate transform from world (X_w, Y_w, Z_w) into camera coordinates (X_s, Y_s, Z_s)
2. A central projection of (X_s, Y_s, Z_s) into undistorted coordinates (x_u, y_u)
3. Lens distortion maps (x_u, y_u) into distorted coordinates (x_d, y_d)
4. Shift of $x_d y_d$ coordinates by principal point (x_c, y_c), defining sensor coordinates (x_s, y_s)
5. Mapping of sensor coordinates (x_s, y_s) into pixel coordinates (x, y)

Radial Lens Distortion

A (simplified) rule: Given lens-distorted image point $p_d = (x_d, y_d)$, we obtain corresponding undistorted image point $p_u = (x_u, y_u)$ as follows:

$$
\begin{aligned}
x_u &= c_x + (x_d - c_x)(1 + \kappa_1 r_d^2 + \kappa_2 r_d^4 + e_x) \\
y_u &= c_y + (y_d - c_y)(1 + \kappa_1 r_d^2 + \kappa_2 r_d^4 + e_y)
\end{aligned}
$$

for principal point (c_x, c_y) and $r_d = \sqrt{(x_d - c_x)^2 + (y_d - c_y)^2}$

Errors e_x and e_y assumed to be zero

Experimental evidence: just κ_1 and κ_2 corrects more than 90% of the radial distortion

If only κ_1 then precision of about 0.1 sensor cells

Lens distortion calibration: may be separated from remaining calibration processes

Designing a Calibration Method

Define the set of parameters to be calibrated, and a corresponding camera model

Given: Points (X_w, Y_w, Z_w) on calibration rig or *calibration marks* in the 3D scene, known by their physically-measured world coordinates

For (X_w, Y_w, Z_w) identify corresponding point (x, y) (subpixel accuracy) in the image

Example: 100 points (X_w, Y_w, Z_w) define 100 sets of identical equations where parameters such as c_x, c_y, f, r_{11} to r_{33}, and t_1, t_2 and t_3 appear as unknowns

Thus: overdetermined equational system; need a "clever" optimization procedure

Possible Refinements in Used Models

Difference between a focal length f_x in x-direction and a focal length f_y in y-direction

Edge length e_x and e_y of the sensor cells in the used sensor matrix in the camera

Also a shearing factor s for evaluating orthogonality of recorded images

Common: Transition from world to camera coordinates in homogenous coordinates and in pixel units

General Procedure

1. Known positions (X_w, Y_w, Z_w) in the world
2. Related to identifiable locations (x, y) in recorded images
3. Equations defining the camera model contain X_w, Y_w, Z_w, x, and y as known values, and intrinsic or extrinsic camera parameters as unknowns
4. Resulting equational system (necessarily non-linear due to central projection or radial distortion) needs to be solved for specified unknowns
5. Over-determined situations provide for stability of a used numeric solution scheme

Agenda

Manufacturing a Calibration Board

Rigid board wearing a black and white checkerboard pattern, e.g. printed onto paper or by using equally-sized black squares glued onto a rigid board

At least 7×7 squares

Squares large enough such that 10×10 pixels at least in image

Example: effective focal length f only needed to estimate size $a \times a$ cm for each square assuming a distance of b m between camera and board

Specify parameter a as a function of arguments f and b

Localizing Corners in the Checkerboard

Calibration marks are the corners of the squares

Can be identified by approximating intersection points of grid lines

Thus corners with subpixel accuracy

Example: 10 vertical and 10 horizontal grid lines

$10 + 10$ peaks in $d\alpha$ Hough space for detecting line segments

Requires that lens distortion has been removed prior to applying the Hough-space method

Localizing Calibration Marks

Calibration pattern defined by marks such as circular or square dots

Identify an image region S of pixels as the area which shows a calibration mark

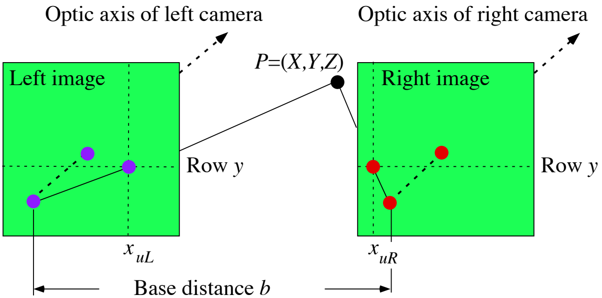
Position of mark identified at subpixel accuracy by calculating the centroid of this region

Agenda

Geometric Rectification

Consider a two-camera recording system (common for stereo vision)

Two cameras need to be mapped into image pairs as recorded in canonical stereo geometry



Multi-Camera Systems

Often more than just two cameras for applying computer vision to complex environments or processes

Then significantly different viewing angles

Multi-Camera Systems

Geometrically rectified images taken at different viewing locations by two cameras installed in a crash-test hall

Cameras i and j

We do not restrict the discussion to just a left and a right camera

General case

Consider Camera i and Camera j, where numbers i and j identify different cameras in a multi-camera system

Camera Model

Considered intrinsic camera parameters of Camera i:

1. Edge lengths e_i^x and e_i^y of camera sensor cells
2. Skew parameter s_i
3. Principal point $\mathbf{c}_i = (c_i^x, c_i^y)$
4. Focal length f_i

Assume that lens distortion has been calibrated before

Projection Equation in Homogeneous Coordinates

Projection equation in 4D homogeneous coordinates, mapping a 3D point $P = (X_w, Y_w, Z_w)$ into image coordinates $p_i = (x_i, y_i)$ of the ith camera

$$
\begin{aligned}
k \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} &= \begin{bmatrix} \frac{f_i}{e_i^x} & s_i & c_i^x & 0 \\ 0 & \frac{f_i}{e_i^y} & c_i^y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{R}_i & -\mathbf{R}_i^T \mathbf{t}_i \\ \mathbf{0}^T & 1 \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} \\
&= [\mathbf{K}_i | \mathbf{0}] \cdot \mathbf{A}_i \cdot [X_w, Y_w, Z_w, 1]^T
\end{aligned}
$$

$\mathbf{R}_i$ and $\mathbf{t}_i$ are rotation matrix and translation vector in 3D inhomogeneous world coordinates

$k \neq 0$ is a scaling factor

Intrinsic and Extrinsic Matrix; Camera Matrix

3×3 matrix $\mathbf{K}_i$ of intrinsic camera parameters

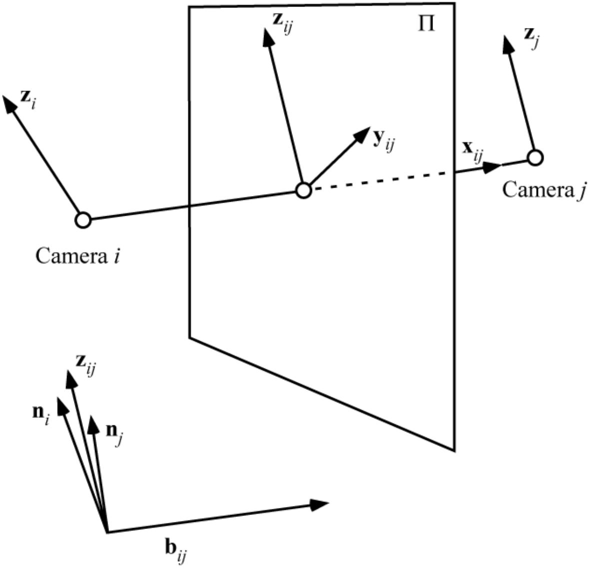
4×4 matrix $\mathbf{A}_i$ of extrinsic parameters

3×4 *camera matrix*

$$\mathbf{C}_i = [\mathbf{K}_i | \mathbf{0}] \cdot \mathbf{A}_i$$

Defined by 11 parameters if we allow for an arbitrary scaling of parameters; otherwise 12

Common Viewing Direction for Cameras i and j



Rectifying Cameras i and j

Common viewing direction replaces given viewing directions

Baseline vector $\mathbf{b}_{ij}$ from center i to center j

Π is plane perpendicular to $\mathbf{b}_{ij}$

Project unit vectors $\mathbf{z}_i^\circ$ and $\mathbf{z}_j^\circ$ of both optic axes into Π

Results in vectors $\mathbf{n}_i$ and $\mathbf{n}_j$

$$\mathbf{n}_i = (\mathbf{b}_{ij} \times \mathbf{z}_i^\circ) \times \mathbf{b}_{ij} \quad \text{and} \quad \mathbf{n}_j = (\mathbf{b}_{ij} \times \mathbf{z}_j^\circ) \times \mathbf{b}_{ij}$$

Use bisector of $\mathbf{n}_i$ and $\mathbf{n}_j$

$$\mathbf{z}_{ij}^\circ = \frac{\mathbf{n}_i + \mathbf{n}_j}{||\mathbf{n}_i + \mathbf{n}_j||_2}$$

Result: This is the unit vector of the common direction

Defining the Tripods

General: For $\mathbf{a}$ and $\mathbf{b}$, $(\mathbf{a}, \mathbf{b}, \mathbf{a} \times \mathbf{b})$ defines a a left-hand tripod

We have: Unit vector $\mathbf{x}^\circ_{ij}$ in direction as $\mathbf{b}_{ij}$
Goal: Unit vector $\mathbf{y}^\circ_{ij}$ defined by ensuring a left-hand 3D Cartesian coordinate system

$$\mathbf{x}^\circ_{ij} = \frac{\mathbf{b}_{ij}}{||\mathbf{b}_{ij}||_2} \quad \text{and} \quad \mathbf{y}^\circ_{ij} = \mathbf{z}_{ij} \times \mathbf{x}^\circ_{ij} = -\mathbf{x}^\circ_{ij} \times \mathbf{z}_{ij}$$

Indeed: It is a left-hand tripod $(\mathbf{x}^\circ_{ij}, \mathbf{z}^\circ_{ij} \times \mathbf{x}^\circ_{ij}, \mathbf{z}^\circ_{ij})$ because

$$\mathbf{x}^\circ_{ij} \times (\mathbf{z}^\circ_{ij} \times \mathbf{x}^\circ_{ij}) = \mathbf{z}^\circ_{ij}(\mathbf{x}^\circ_{ij} \cdot \mathbf{x}^\circ_{ij}) - \mathbf{x}^\circ_{ij}(\mathbf{x}^\circ_{ij} \cdot \mathbf{z}^\circ_{ij}) = \mathbf{z}^\circ_{ij}$$

$(\mathbf{x}^\circ_{ij}, \mathbf{x}^\circ_{ij} \times \mathbf{z}^\circ_{ij}, \mathbf{z}^\circ_{ij})$ would be a right-hand tripod

Producing the Rectified Image Pair

Images of cameras i and j as taken in direction $\mathbf{R}_{ij} = (\mathbf{x}_{ij}\ \mathbf{y}_{ij}\ \mathbf{z}_{ij})^T$, instead of directions $\mathbf{R}_i$ and $\mathbf{R}_j$

Rotation of cameras into their virtual viewing directions

$$\mathbf{R}_i^* = \mathbf{R}_{ij}\mathbf{R}_i^T \quad \text{and} \quad \mathbf{R}_j^* = \mathbf{R}_{ij}\mathbf{R}_j^T$$

General: Rotating camera around projection center about matrix $\mathbf{R}$, then image transformed by a rotation *homography*

$$\mathbf{H} = \mathbf{K} \cdot \mathbf{R} \cdot \mathbf{K}^{-1}$$

where $\mathbf{K}$ is 3×3 matrix of intrinsic parameters of the camera

1. Matrix $\mathbf{K}^{-1}$ transfers pixel coordinates into camera coordinates in world units
2. Matrix $\mathbf{R}$ rotates them into the common plane
3. Matrix $\mathbf{K}$ transfers them back into pixel coordinates

Creating an Identical Twin

Rectified image, pixel by pixel, by using

$$p = \mathbf{H}^{-1}\hat{p}$$

Warping: New value at $\hat{p}$ based on original image values in a neighbourhood of point p

Want to have image of Camera j after rotation homography with respect to parameters of Camera i

$$\mathbf{H}_{ij} = \mathbf{K}_i \cdot \mathbf{R}_j^* \cdot \mathbf{K}_j^{-1}$$

1. Transform by $\mathbf{K}_j^{-1}$ points in the jth image plane into a "normalized" coordinate system
2. $\mathbf{R}_j^*$ performs the desired rotation
3. $\mathbf{K}_i$ transforms rotation result according to parameters of Camera i

Agenda

Fundamental Matrix

Fundamental matrix **F** describes binocular stereo geometry with including the characteristics of the used cameras

Assume a left and a right camera; p_L and p_R be any pair of *corresponding stereo points* in homogeneous coordinates:

$$p_R^T \cdot \mathbf{F} \cdot p_L = 0$$

for a 3×3 matrix **F** of rank 2

Example: $\mathbf{F} \cdot p_L$ defines a line in the image plane of the right camera, the *epipolar line*

Calculating the Fundamental Matrix

In general, seven pairs of corresponding points (in general position) are sufficient to identify matrix $\mathbf{F}$

Defining equation for fundamental matrix:

$$\mathbf{F} = \mathbf{K}_R^{-T} \cdot \mathbf{R}[\mathbf{t}]_\times \cdot \mathbf{K}_L^{-1}$$

for camera matrices $\mathbf{K}_R$ and $\mathbf{K}_L$, and $[\mathbf{t}]_\times$ is the *cross product matrix* of vector $\mathbf{t}$, defined by $[\mathbf{t}]_\times \cdot \mathbf{a} = \mathbf{t} \times \mathbf{a}$, or

$$[\mathbf{t}]_\times = \begin{bmatrix} 0 & -t_3 & t_2 \\ t_3 & 0 & -t_1 \\ -t_2 & t_1 & 0 \end{bmatrix}$$

We go here from pixel coordinates to camera coordinates (possibly in world units), and back to pixel coordinates

Essential Matrix

Essential matrix **E** describes binocular stereo geometry without including the characteristics of the used cameras

Defined by

$$\mathbf{E} = \mathbf{R}[\mathbf{t}]_\times$$

Has five degrees of freedom

Uniquely defined by left and right camera up to scaling

Copyright Information

This slide show was prepared by Reinhard Klette
with kind permission from Springer Science+Business Media B.V.

The slide show can be used freely for presentations.
However, *all the material* is copyrighted.

In case of citation: just cite the book, that's fine.