Camera Calibration

Dr. Gerhard Roth

Winter 2011

Finding camera parameters (intrinsic)

- Can use the EXIF tag for any digital image
  - Has focal length f in millimeters but not the pixel size
  - But you can get the pixel size from the camera manual
  - There are only a finite number of different pixels sizes because number of sensing element sizes is limited
  - If there is not a lot of image distortion due to optics then this approach is sufficient (only linear calibration)
- Can perform explicit camera calibration
  - Put a calibration pattern in front of the camera
  - Take a number of different pictures of this pattern
  - Now run the calibration algorithm (different types)
  - Result is intrinsic camera parameters (linear and non-linear) and the extrinsic camera parameters of all the images

Explicit camera calibration

- Use a calibration pattern with known geometry
  - In Opencv use a checkerboard
  - Other systems use special targets with known 3d geometry
- Write equations linking co-ordinates of the projected points, and the camera parameters
- From images of the calibration target
  - Intrinsic camera parameters
    - (depend only on camera characteristics)
  - Extrinsic camera parameters
    - (depend only on position camera)
  - In OpenCV the calibration process finds fx, fy, ox, oy, along with the distortion parameters
  - We study a method that does not find the distortion parameters

Calibration using known 3d geometry

- Use a calibration pattern with known 3d geometry (often a box, not planar)
- Write projection equations linking known co-ordinates of a set of 3-D points and their projections and solve for camera parameters
- Given a set of one or more images of the calibration pattern estimate
  - Intrinsic camera parameters
    - (depend only on camera characteristics)
  - Extrinsic camera parameters
    - (depend only on position camera)
- We do not estimate distortion parameters

Estimating camera parameters

- Projection matrix

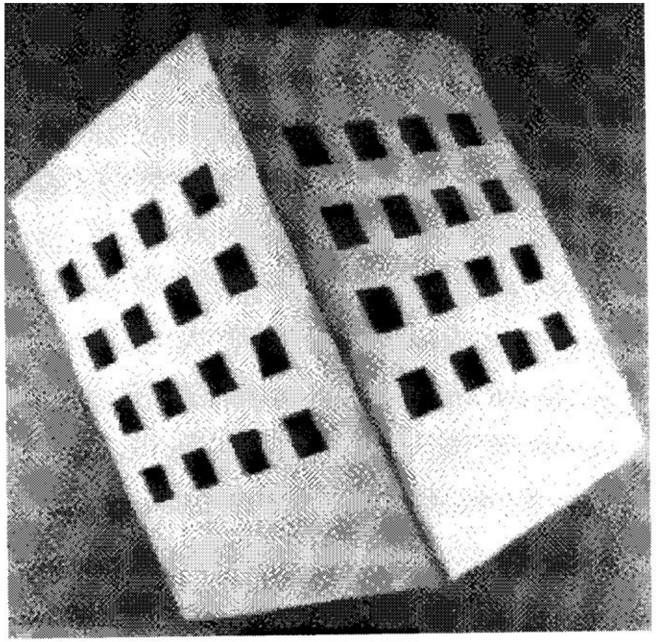
Calibration pattern

(u_1, v_1) — (X_1, Y_1, Z_1)

(u_2, v_2) — (X_2, Y_2, Z_2)

(u_3, v_3) — (X_3, Y_3, Z_3)

(u_N, v_N) — (X_N, Y_N, Z_N)



Camera parameters

- Intrinsic parameters (K matrix)
  - There are 5 intrinsic parameters
  - Focal length f
  - Pixel size in x and y directions, sx and sy
  - Principal point ox, oy
- But they are not independent
  - Focal length fx = f / sx and fy = f / sy
  - Principal point ox, oy
  - This makes four intrinsic parameters
- Extrinsic parameters [R| T]
  - Rotation matrix and translation vector of camera
  - Relations camera position to a known frame
  - [R|T] are the intrinsic parameters
- Projection matrix
  - 3 by 4 matrix P =K [R | T] is called projection matrix

Projection Equations

Projective Space

- Add fourth coordinate
  - $P_w = (X_w, Y_w, Z_w, 1)^T$
- Define $(u,v,w)^T$ such that
  - $u/w = x_{im}$, $v/w = y_{im}$

$$\begin{pmatrix} x_{im} \\ y_{im} \end{pmatrix} = \begin{pmatrix} u/w \\ v/w \end{pmatrix}$$

$$\begin{pmatrix} u \\ v \\ w \end{pmatrix} = \mathbf{M}_{\mathbf{int}}\mathbf{M}_{\mathbf{ext}} \begin{pmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{pmatrix}$$

3x4 Matrix E_{ext}

- Only extrinsic parameters
- World to camera

$$\mathbf{M}_{ext} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & T_x \\ r_{21} & r_{22} & r_{23} & T_y \\ r_{31} & r_{32} & r_{33} & T_z \end{bmatrix} = \begin{bmatrix} \mathbf{R}_1^T & T_x \\ \mathbf{R}_2^T & T_y \\ \mathbf{R}_3^T & T_z \end{bmatrix}$$

3x3 Matrix E_{int}

- Only intrinsic parameters
- Camera to frame

$$\mathbf{M}_{\text{int}} = \begin{bmatrix} -f_x & 0 & o_x \\ 0 & -f_y & o_y \\ 0 & 0 & 1 \end{bmatrix}$$

Simple Matrix Product! Projective Matrix

$M = M_{int} M_{ext}$

- $(X_w, Y_w, Z_w)^T$ -> $(x_{im}, y_{im})^T$
- Linear Transform from projective space to projective plane
- M defined up to a scale factor – 11 independent entries

Two different calibration methods

- Both use a set of 3d points and 2d projections
- Direct approach (called Tsai method)
  - Write projection equations in terms of all the parameters
    - That is all the unknown intrinsic and extrinsic parameters
  - Solve for these parameters using non-linear equations
- Projection matrix approach
  - Compute the projection matrix (the 3x4 matrix M)

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

  - Compute camera parameters as closed-form functions of M

•Two different calibration methods

- Both approaches work with same data
  - Projection matrix approach is simpler to explain than the direct approach
- Direct approach requires an extra step
  - There are also other calibration methods
- But all calibration methods
  - Use patterns with know geometry or shape
  - Take multiple views of theses patterns
  - Match the information across the different views
- Perform some mathematics to calculate the intrinsic and extrinsic camera parameters
- We look at simplified case of only one view!

Estimating the projection matrix

World – Frame Transform

- Drop “im” and “w”
- N pairs (xi,yi) <-> (Xi,Yi,Zi)

$$x_i = \frac{u_i}{w_i} = \frac{m_{11}X_i + m_{12}Y_i + m_{13}Z_i + m_{14}}{m_{31}X_i + m_{32}Y_i + m_{33}Z_i + m_{34}}$$

$$y_i = \frac{u_i}{w_i} = \frac{m_{21}X_i + m_{22}Y_i + m_{23}Z_i + m_{24}}{m_{31}X_i + m_{32}Y_i + m_{33}Z_i + m_{34}}$$

Linear equations of m

- 2N equations, 11 independent variables
- N >=6 , SVD => m up to a unknown scale

$$\mathbf{Am} = \mathbf{0}$$

$$\mathbf{A} = \begin{bmatrix} X_1 & Y_1 & Z_1 & 1 & 0 & 0 & 0 & 0 & -x_1X_1 & -x_1Y_1 & -x_1Z_1 & -x_1 \\ 0 & 0 & 0 & 0 & X_1 & Y_1 & Z_1 & 1 & -y_1X_1 & -y_1Y_1 & -y_1Y_1 & -y_1 \\ . & . & . & . & . & . & & & & & & \end{bmatrix}$$

$$\mathbf{m} = \begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} & m_{21} & m_{22} & m_{23} & m_{24} & m_{31} & m_{32} & m_{33} & m_{34} \end{bmatrix}^T$$

Homogeneous System

- M linear equations of form A**x** = 0
- If we have a given solution x1, s.t. Ax1 = 0 then c * x1 is also a solution A(c* x1) = 0
- Need to add a constraint on **x**,
  - Basically make **x** a unit vector $x^T x = 1$
- Can prove that the solution is the eigenvector corresponding to the single zero eigenvalue of that matrix $A^T A$
  - This can be computed using eigenvector of SVD routine
  - Then finding the zero eigenvalue (actually smallest)
  - Returning the associated eigenvector

Decompose projection matrix

- 3x4 Projection Matrix M computed previously
  - Both intrinsic (4) and extrinsic (6) – 10 parameters

$$\mathbf{M} = \begin{bmatrix} -f_x r_{11} + o_x r_{31} & -f_x r_{12} + o_x r_{32} & -f_x r_{13} + o_x r_{33} & -f_x T_x + o_x T_z \\ -f_y r_{21} + o_y r_{31} & -f_y r_{22} + o_y r_{32} & -f_y r_{23} + + o_y r_{33} & -f_y T_y + o_y T_z \\ r_{31} & r_{32} & r_{33} & T_z \end{bmatrix}$$

From M^ to parameters (p134-135)

- Find scale $|\gamma|$ by using unit vector R_3^T
- Determine T_z and sign of γ from m_{34} (i.e. q_{43})
- Obtain R_3^T
- Find (Ox, Oy) by dot products of Rows q1. q3, q2.q3, using the orthogonal constraints of R
- Determine fx and fy from q1 and q2 All the rests: R_1^T, R_2^T, Tx, Ty

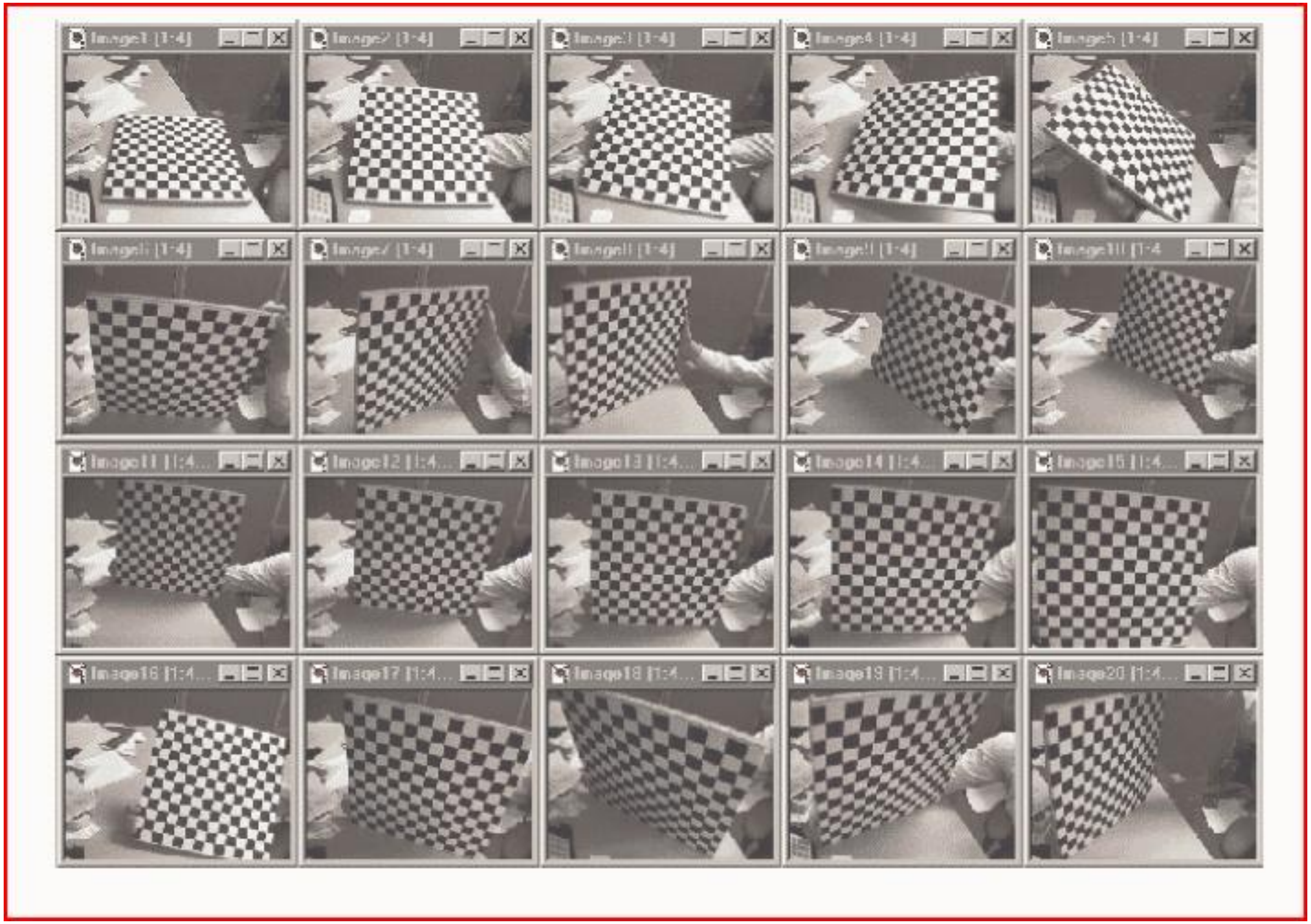
Calibration Summary

- Comparison of methods
  - Direct approach requires extra step to find Ox, Oy
  - Projection approach finds Ox, and Oy at same time
    - Is simpler mathematically than the direct approach
  - Both methods require a refit to find a “valid” R matrix
- There are other calibration methods
  - Zhang approach uses flat plane (implemented in OpenCV)
  - Plane must be flat, but do not need 3D co-ordinates
- But all calibration methods
  - Have some known targets with known 3D geometry or shape
  - Take a number of images of these targets
  - From these measurements calculate the camera partakers
  - Are essential for further processing like reconstruction

Multiple View/Camera Calibration

- Previous math describes the calibration process for a single image
  - We usually take multiple images of the same calibration target (from a variety of different views)
  - Simultaneously find all extrinsic parameters and all the intrinsic parameters of the single camera
- Also calibrate radial distortion using fact that there are straight lines in the pattern
- OpenCV code can do this using a checkerboard pattern
- Zhang’s algorithm is used most in practice

Input set of 2d Calibration Patterns



Final Camera positions and the pattern