Camera Calibration

Jenn-Jier James Lien (連 震 杰)
Professor

Computer Science and Information Engineering
National Cheng Kung University

(O) (06) 2757575 ext. 62540
jjlien@csie.ncku.edu.tw
http://robotics.csie.ncku.edu.tw

Major Issues

1. **Camera Calibration for:**
   1) Intrinsic Parameters (and Lens Distortion Parameters) and
   2) Extrinsic Parameters

2. **Optimization Based on Homogenous Matrix**

3. **Why do we want to know Camera Modeling and Camera Calibration?**

Local optimization (0 moment): 1 and 2 =>
Global optimization: 3 => 1) LM 2) EM 3) SGD (Stochastic Gradient Decent)

Solution/Optimization of Homogenous Matrix

1. **Close Form Solution:**
   $Ax = 0$ => A^tA = Covariance Matrix =SVD $=UWU^T$: Smallest eigenvalue >0 => eigenvector
2. **Pseudo Inverse:**
   $Ax = b$, x=? Or A=?
3. **Sum of Squared Difference: (max likelihood – exponential term)**
   min $E = \sum [Ax - b]^2$

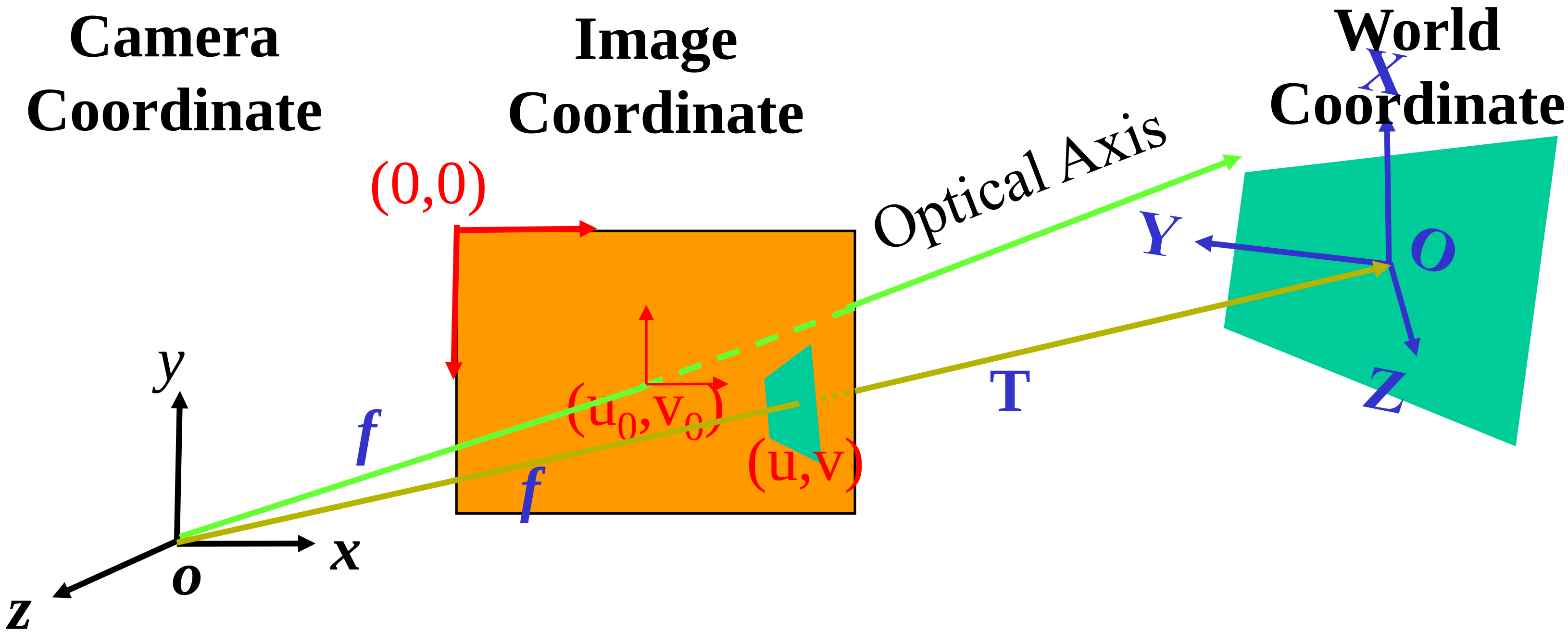
   3.1 $Ax = b'$: estimation value. b: ground truth, $E = \sum [b - b']^2$
   a. Initial value estimation => Pseudo Inverse (linear approach)
   b. L-M (Levenberg-Marquardt Algorithm: non-linear approach)
   b.1 First order Taylor series expansion
   b.2 2nd order Taylor series expansion (sensitive to noise)

   3.2 $Ax = b'$: estimation value. b'': estimation value, $E = \sum [b'' - b']^2$
   a. EM (Expected-Maximization), initial b = average value

Machine learning for prediction

4. **Lagrange Approach (outlier) with constraint**
   min $E = \sum [Ax - b]^2 + \lambda (x^2+y^2)^2$

Pinhole Camera Model: World ⇔ Camera ⇔ Image Coordinates



$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{pmatrix} R_{11} & R_{21} & R_{31} \\ R_{12} & R_{22} & R_{32} \\ R_{13} & R_{23} & R_{33} \end{pmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + \begin{bmatrix} T_1 \\ T_2 \\ T_3 \end{bmatrix}$$

Extrinsic Parameters

$$\lambda \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{pmatrix} \alpha & \gamma & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{pmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

Intrinsic Parameters

Camera Parameters: Intrinsic + Extrinsic Parameters

$$\lambda \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = \begin{pmatrix} \alpha & \gamma & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} R_{11} & R_{21} & R_{31} & T_1 \\ R_{12} & R_{22} & R_{32} & T_2 \\ R_{13} & R_{23} & R_{33} & T_3 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

$u = -fk_u(x/z) + \gamma\,(y/z) + u_0$

$v = -fk_v(y/z) + v_0$

Scale Factor: λ

Normalization

- **Intrinsic Parameters:**
  - **Scale Factor**: $\alpha = -f\,k_u$, f: focal length
  - **Scale Factor**: $\beta = -f\,k_v$
  - **Aspect Ratio**= $\alpha/\beta = k_u/k_v$
  - **Skew Factor:** γ
  - **Principal Point:** (u_0, v_0)
- **Extrinsic Parameters:**
  - **Rotation: R**
  - **Translation: T**

Pinhole Camera Model:
World ⇔ Camera ⇔ Image Coordinates

Framework: distortion vs undistortion??

Chessboard:棋盤格影像
Chessboard pattern:投影棋盤格

3D camera: 1 chessbd 15 images
3D DLP: 15 images 1 pattern at

$$\min \sum_{n=1}^{N} \left(p_n^u - p'^u_n\right)^2 \qquad s\begin{bmatrix} u_n \\ v_n \\ 1 \end{bmatrix} = \begin{bmatrix} h11 & h12 & h13 & h14 \\ & & & \\ h41 & h42 & h43 & 1 \end{bmatrix} \begin{bmatrix} X_n^w \\ Y_n^w \\ Z_n^w \\ 1 \end{bmatrix}$$

$Z_n^w = 0$, $N_{rc}^c = 48$ $\quad P_n^w, Z_n^{w'} = 0$ $\quad H_{t,3\times3}$

2D reprojection 2D ?? Distortion or undistortion?

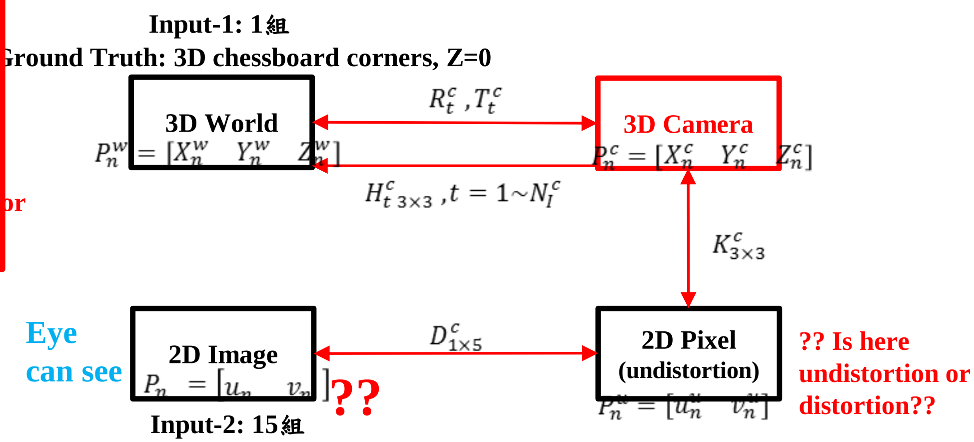
1) From 2D image (distorted image) coordinate (u_n, v_n) to 2D pixel (undistorted) coordinate $P_n^u = [u_n^u \quad v_n^u]$

(1) Radial (barrel) distortion:

$$u_n^u = u_n(1 + k_1 r^2 + k_2 r^4 + k_3 r^6)$$
$$v_n^u = v_n(1 + k_1 r^2 + k_2 r^4 + k_3 r^6)$$

(2) Tangential distortion:

$$u_n^u = u_n + [2p_1 v_n + p_2(r^2 + 2u_n{}^2)]$$
$$v_n^u = v_n + [p_1(r^2 + 2v_n{}^2) + 2p_2 u_n] \qquad \textbf{where } r^2 = u_n{}^2 + v_n{}^2$$



2) Perspective Projection: From 3D camera coordinate projects to 2D image coordinate

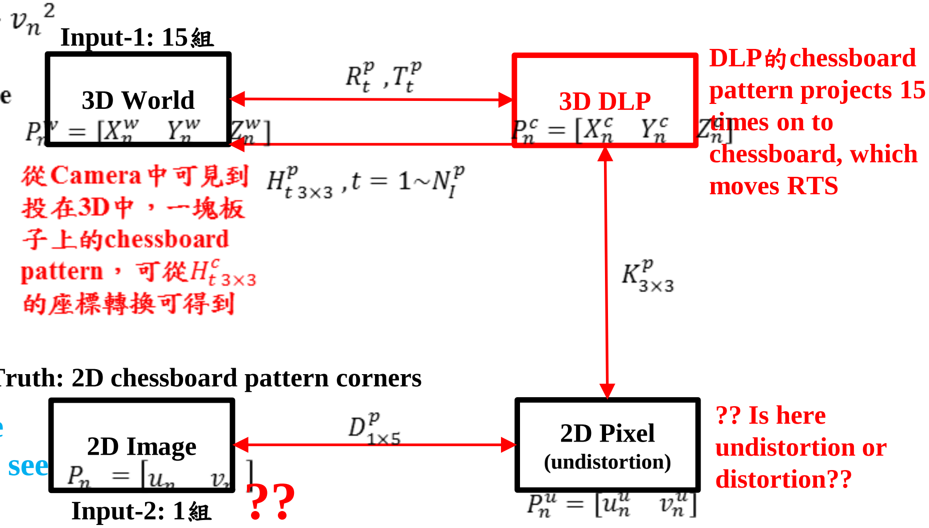
?? If this is correct, then D doesnot calculate included.??

$$s\begin{bmatrix} u_n \\ v_n \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & \gamma = 0 & c_x & 0 \\ 0 & f_y & c_y & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_n^c \\ Y_n^c \\ Z_n^c = 0 \\ 1 \end{bmatrix}$$

3) Affine: From 3D world coordinate to 3D camera coordinate

$$s\begin{bmatrix} X_n^c \\ Y_n^c \\ Z_n^c = 0 \\ 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_n^{w'} \\ Y_n^{w'} \\ Z_n^{w'} = 0 \\ 1 \end{bmatrix}$$

Burn to DLP 裏的 chessboard pattern，以此來投射



4) From 3D world coordinate to 2D coordinate

$$s\begin{bmatrix} u_n \\ v_n \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & \gamma = 0 & c_x & 0 \\ 0 & f_y & c_y & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_n^{w'} \\ Y_n^{w'} \\ Z_n^{w'} = 0 \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} su \\ sv \\ s \end{bmatrix} = \begin{bmatrix} \alpha & \gamma & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + \begin{bmatrix} \alpha & \gamma & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} T_X \\ T_Y \\ T_Z \end{bmatrix}$$

$$\begin{bmatrix} su \\ sv \\ s \end{bmatrix} = \begin{bmatrix} \alpha & \gamma & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & T_X \\ r_{21} & r_{22} & r_{23} & T_Y \\ r_{31} & r_{32} & r_{33} & T_Z \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} su \\ sv \\ s \end{bmatrix} = \begin{bmatrix} \alpha r_{11} + \gamma r_{21} + u_0 r_{31} & \alpha r_{12} + \gamma r_{22} + u_0 r_{32} & \alpha r_{13} + \gamma r_{23} + u_0 r_{33} & \alpha T_X + \gamma T_Y + u_0 T_Z \\ \beta r_{21} + v_0 r_{31} & \beta r_{22} + v_0 r_{32} & \beta r_{23} + v_0 r_{33} & \beta T_Y + v_0 T_Z \\ r_{31} & r_{32} & r_{33} & T_Z \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

Collinear Equation:

$$u = \frac{su}{s} = u_0 + k_u f \, \frac{r_{11}X + r_{12}Y + r_{13}Z + T_X}{r_{31}X + r_{32}Y + r_{33}Z + T_Z} + \overset{=0}{\gamma} \, \frac{r_{21}X + r_{22}Y + r_{23}Z + T_Y}{r_{31}X + r_{32}Y + r_{33}Z + T_Z}$$

$$v = \frac{sv}{s} = v_0 + k_v f \, \frac{r_{21}X + r_{22}Y + r_{23}Z + T_Y}{r_{31}X + r_{32}Y + r_{33}Z + T_Z}$$

$$\begin{bmatrix} su \\ sv \\ s \end{bmatrix} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

$$u = \frac{su}{s} = \frac{p_{11}X + p_{12}Y + p_{13}Z + p_{14}}{p_{31}X + p_{32}Y + p_{33}Z + p_{34}}$$

$$v = \frac{sv}{s} = \frac{p_{21}X + p_{22}Y + p_{23}Z + p_{24}}{p_{31}X + p_{32}Y + p_{33}Z + p_{34}}$$

Intrinsic Parameters:

- **Intrinsic parameters (5) (Image Coordinate ⇔ Camera Coordinate)**
  - Do not depend on the camera location
  - Internal camera parameters
    - Principal point (u_0, v_0)
    - Focal length (f)
    - CCD dimensions/Pixel width and height
    - Lens distortion
- **In many situations it is appropriate to take advantage of a'priori knowledge about the cameras intrinsic parameters**
- **Some common simplifying assumptions**
  - Assume that the skew angle is zero, i.e., $\gamma = 0$.
  - Often the aspect ratio α/β is known. In most digital images, this is engineered to be 1
  - In many cases we assume that the principal point (u_0, v_0) is located in the center of the image

Extrinsic Parameters:

- **Extrinsic parameters (6) (Camera Coordinate ⇔ World Coordinate)**
  - Depend on the camera location. That is, relate the camera's coordinate system to the world coordinate system
    - Translation (3) and rotation (3) parameters in 3D
- **The parameters,** R **and** T**, which capture the relationship between the extrinsic reference frame and the cameras frame, are referred to as extrinsic parameters.**

Parameter Estimation

- **The parameter estimation framework is frequently employed in situations where:**
  - There is no easy way to invert the function f
  - There are multiple noisy measurements which overconstrain the system

Camera Calibration:
2D Image Coordinate ⇔ 3D World Coordinate

project ⇄ reconstruct

- **To achieve this, we need to have a calibrated camera**
  - Calculate the camera parameters that enable us to go from 2D image location to 3D ray in world coordinates
- **The goal of camera calibration procedures: is to produce an estimate for the projection matrix, P, and/or the intrinsic and extrinsic parameters, A, R and T**
- **Procedure:**
  - Given the correspondences between a set of point features in the world and their projections in an image, compute the coefficients of the transformation
- **Points to Note:**
  - It is difficult to recover estimates for some of the parameters because their effects are confounded in the projection matrix.
  - Most calibration procedures do not fully account for the orthogonal constraints on R which leads to imprecision

- **Camera calibration can be viewed as a parameter estimation problem**
- **Common Optimization Techniques**
  - Newton-Raphson optimization
  - Conjugate Gradient Methods
  - Simplex methods
- **Most techniques require a reasonable starting estimate - this can be obtained from simpler, less accurate linear methods**

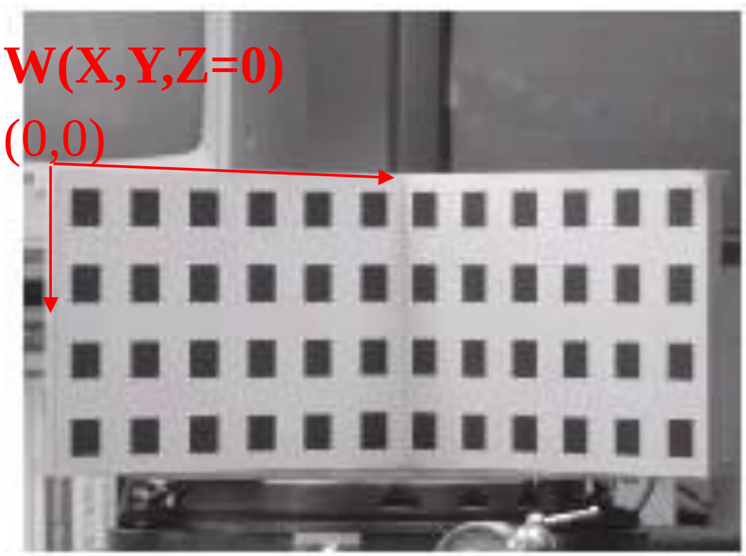
Camera calibration: 3D $\rightarrow$ 2D

Project

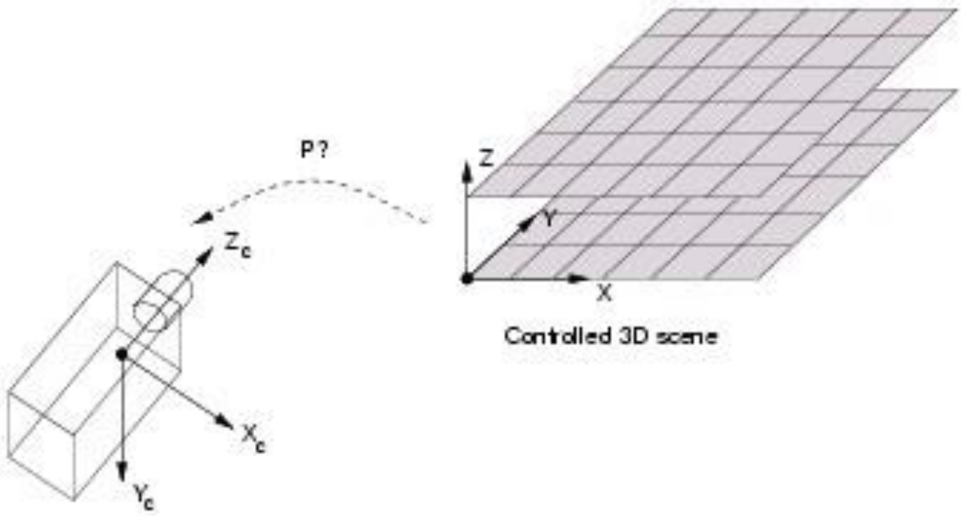
Intrinsic Parameters Extrinsic Parameters

Camera calibration is the name given to the process of discovering the projection matrix (and its decomposition into camera matrix and the position and orientation of the camera) from an image of a controlled scene. For example, we might set up the camera to view a calibrated grid of some sort.

translation



en



For a projective camera we have:

$$\begin{bmatrix} su \\ sv \\ s \end{bmatrix} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

There are 11 parameters to estimate (since the overall scale of P does not matter, we could, for example, set p_{34} to 1).

Each point we observe gives us a pair of equations. Setting p_{34} to 1 we obtain:

$$u_i = \frac{su_i}{s} = \frac{p_{11}X_i + p_{12}Y_i + p_{13}Z_i + p_{14}}{p_{31}X_i + p_{32}Y_i + p_{33}Z_i + 1}$$

$$v_i = \frac{sv_i}{s} = \frac{p_{21}X_i + p_{22}Y_i + p_{23}Z_i + p_{24}}{p_{31}X_i + p_{32}Y_i + p_{33}Z_i + 1}$$

Known 3D relative position at world coordinate

Since we are observing a calibrated scene, we know X_i, Y_i, and Z_i, and we observe the pixel coordinates $\hat{u}_i$ *and* $\hat{v}_i$ in the image. The equations above can be rearranged to give two linear equations in the unknown projection matrix parameters.

Estimated/measured corresponding 2D position (u_i, v_i) at image coordinate by corner detection

Since there are 11 unknowns, we need to observe at least 6 points to calibrate the camera. The equations can be solved using linear least squares. Note how the use of the projective camera has linearized the problem.

The linear solution is, however, only approximate and should ideally be used as the starting point for non-linear minimization: i.e. finding the parameters of the projection matrix that minimize the errors between measured image points, (u_i, v_i) and projected (or modelled) image positions, $((\hat{u}_i, \hat{v}_i)$:

$$\min_{\mathbf{P}} \sum_i ((u_i - \hat{u}_i)^2 + (v_i - \hat{v}_i)^2)$$

$2(\ )^{1/2}$, $(\ )^2$

Assume Z=0

Having obtained the projection matrix, it is possible to decompose it into the camera calibration matrix and the orientation and position of the camera (if necessary):

$$\mathrm{P}_{ps} = \mathbf{K}[\mathrm{R}|\mathbf{T}]$$

addition

Standard matrix techniques exist for decomposing the 3×3 sub-matrix into the product of an upper triangular matrix, $\mathbf{K}$, and a rotation (orthogonal) matrix $\mathbf{R}$ (known as QR decompositon).

The translation vector or position of the camera can then be obtained by:

$$\mathbf{T} = \mathbf{K}^{-1}(p_{14}, p_{24}, p_{34})^T$$

Min Error/Energy/cost function/

SSD

⇔Registration

⇔Optical flow

Non-Linear minimization

⇔Linear minimization by

Taylor expansion

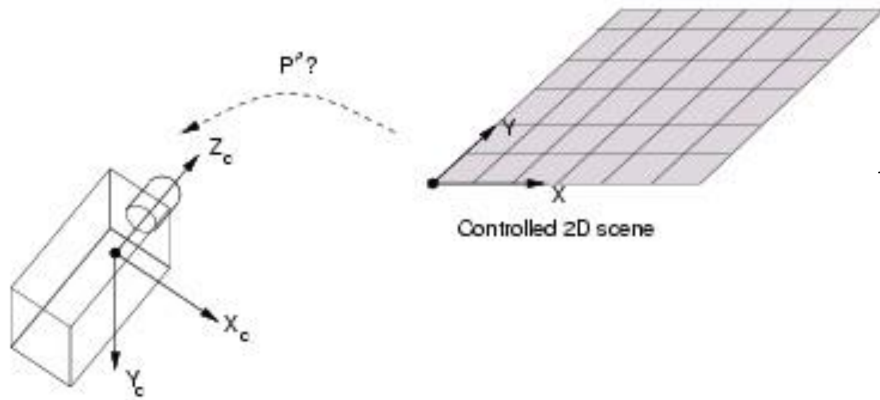
$$P_{ps\ 3x4} = K_{3x3}[R_{3x3} \mid T_{3x1}] = K_{3x3}R_{3x3} + K_{3x3}T_{3x1} = K_{3x3}(R_{3x3} + T_{3x3})$$

$$\begin{bmatrix} su \\ sv \\ s \end{bmatrix} = \begin{bmatrix} \alpha_u & 0 & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & T_x \\ r_{21} & r_{22} & r_{23} & T_y \\ r_{31} & r_{32} & r_{33} & T_z \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} su \\ sv \\ s \end{bmatrix} = \begin{bmatrix} \alpha_u & 0 & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} + \begin{bmatrix} \alpha_u & 0 & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} T_x \\ T_Y \\ T_Z \end{bmatrix}$$

Camera calibration: 2D → 2D

To calibrate the camera for viewing planar scenes, we could set up the camera to view some sort of calibrated planar grid.



Assume Z=0

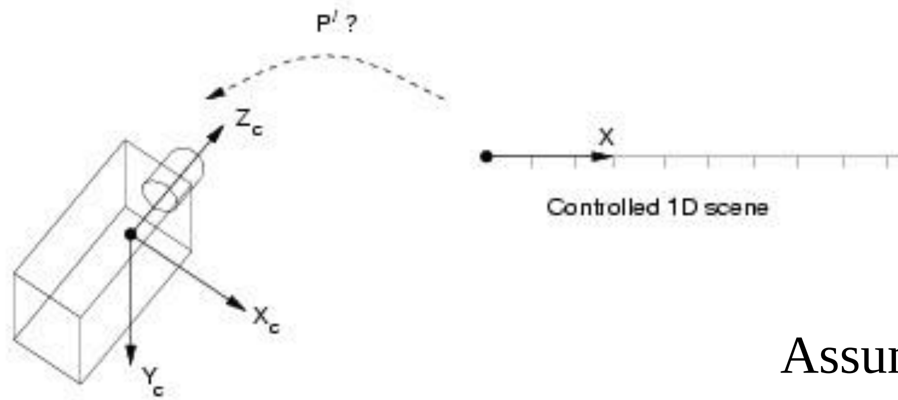
For a plane to plane projectivity, we have

$$\begin{bmatrix} su \\ sv \\ s \end{bmatrix} = \begin{bmatrix} p_{11} & p_{12} & p_{13} \\ p_{21} & p_{22} & p_{23} \\ p_{31} & p_{32} & p_{33} \end{bmatrix} \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix}$$

There are 8 parameters to estimate (if we set, for example, p_{33} to 1), and each observed point gives us a pair of linear equations, so we need to observe at least 4 points. Again, we use linear least squares to solve for the elements of P^p.

Camera calibration: 1D → 1D

Finally, we consider the calibration of a camera viewing a line. This is accomplished by viewing a line with some markings at known positions.



Assume Y=Z=0

For a projective camera we have

$$\begin{bmatrix} su \\ sv \\ s \end{bmatrix} = \begin{bmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \\ p_{31} & p_{32} \end{bmatrix} \begin{bmatrix} X \\ 1 \end{bmatrix}$$

There are 5 parameters to estimate (if we set, for example, p_{32} to 1), and each observed point gives us a pair of linear equations, so we need to observe at least 3 points. Again, we use linear least squares to solve for the elements of P^l.

Recovery of world position

With a calibrated camera, we can attempt to recover the world position of image features.

1D case (line to line): given u, we can uniquely determine the position of the point on the line.

$$\begin{bmatrix} su \\ s \end{bmatrix} = \begin{bmatrix} p_{11} & p_{12} \\ p_{31} & 1 \end{bmatrix} \begin{bmatrix} X \\ 1 \end{bmatrix}$$

$$\begin{bmatrix} X \\ 1 \end{bmatrix} = \begin{bmatrix} P_{11} & P_{12} \\ P_{31} & 1 \end{bmatrix}^{-1} \begin{bmatrix} su \\ s \end{bmatrix} = \frac{\begin{bmatrix} 1 & -P_{12} \\ -P_{31} & P_{11} \end{bmatrix}}{P_{11} - P_{12}P_{31}} \begin{bmatrix} su \\ s \end{bmatrix}$$

$$\Leftrightarrow u = \frac{su}{s} = \frac{p_{11}X + p_{12}}{p_{31}X + 1}$$

$$\begin{bmatrix} \lambda X \\ \lambda \end{bmatrix} \rightarrow \frac{X}{1}$$

$$\Leftrightarrow X = \frac{p_{12} - u}{p_{31}u - p_{11}}$$

2D case (plane to plane): given u and v, we can uniquely determine the position of the point on the world plane. For a plane to plane projectivity, we have

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} p_{11} & p_{12} & p_{13} \\ p_{21} & p_{22} & p_{23} \\ p_{31} & p_{32} & 1 \end{bmatrix} \begin{bmatrix} \lambda X \\ \lambda Y \\ \lambda \end{bmatrix}$$

$$\Leftrightarrow \begin{bmatrix} \lambda X \\ \lambda Y \\ \lambda \end{bmatrix} = \begin{bmatrix} p_{11} & p_{12} & p_{13} \\ p_{21} & p_{22} & p_{23} \\ p_{31} & p_{32} & 1 \end{bmatrix}^{-1} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} p_{11}^{i} & p_{12}^{i} & p_{13}^{i} \\ p_{21}^{i} & p_{22}^{i} & p_{23}^{i} \\ p_{31}^{i} & p_{32}^{i} & p_{33}^{i} \end{bmatrix} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}$$

Z=k

at different distances

$$\Leftrightarrow X = \frac{p_{11}^i u + p_{12}^i v + p_{13}^i}{p_{31}^i u + p_{32}^i v + p_{33}^i}, \quad Y = \frac{p_{21}^i u + p_{22}^i v + p_{23}^i}{p_{31}^i u + p_{32}^i v + p_{33}^i}$$

Jenn-Jier James Lien

3D case (3D world to image plane): given u and v, we cannot uniquely determine the position of the point in the world.

$$\begin{bmatrix} su \\ sv \\ s \end{bmatrix} = \begin{bmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

$$\Rightarrow u = \frac{su}{s} = \frac{p_{11}X + p_{12}Y + p_{13}Z + p_{14}}{p_{31}X + p_{32}Y + p_{33}Z + p_{34}}$$

$$\Rightarrow v = \frac{sv}{s} = \frac{p_{21}X + p_{22}Y + p_{23}Z + p_{24}}{p_{31}X + p_{32}Y + p_{33}Z + p_{34}}$$

Each observed image point (u, v) gives us two equations in three unknowns (X, Y, Z). These equations define a line (ie. a ray) in space, on which the world point must lie.

For general 3D scene interpretation, we need to use more than one view. Later in the course we will take a detailed look at stereo vision and structure from motion.

$$\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \begin{bmatrix} p_{11}{}^{i} & p_{12}{}^{i} & p_{13}{}^{i} \\ p_{21}{}^{i} & p_{22}{}^{i} & p_{23}{}^{i} \\ p_{31}{}^{i} & p_{32}{}^{i} & p_{33}{}^{i} \\ p_{41}{}^{i} & p_{42}{}^{i} & p_{43}{}^{i} \end{bmatrix} \begin{bmatrix} su \\ sv \\ s \end{bmatrix}$$

$$X = \frac{X}{1} = \frac{p_{11}{}^{i}u + p_{12}{}^{i}v + p_{13}{}^{i}}{p_{41}{}^{i}u + p_{42}{}^{i}v + p_{43}{}^{i}}$$

$$Y = \frac{Y}{1} = \frac{p_{21}{}^{i}u + p_{22}{}^{i}v + p_{23}{}^{i}}{p_{41}{}^{i}u + p_{42}{}^{i}v + p_{43}{}^{i}}$$

$$Z = \frac{Z}{1} = \frac{p_{31}{}^{i}u + p_{32}{}^{i}v + p_{33}{}^{i}}{p_{41}{}^{i}u + p_{42}{}^{i}v + p_{43}{}^{i}}$$

Summary:

❑ Calibration Procedure:

1) Print a pattern and attach it to a planar surface.
2) Take a few (15~20) images of the model plane under different orientations by moving either the plane or the camera.
3) Detect the feature points (corner points) in the images.

---

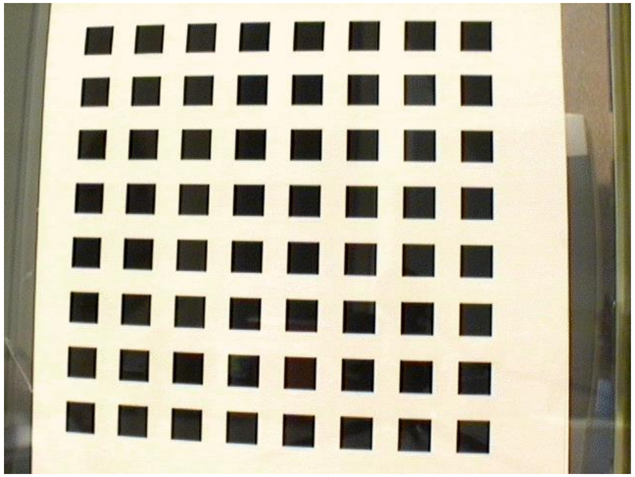
4) Estimate the five intrinsic parameters and all the extrinsic parameters using the closed-form solution.
5) Estimate the coefficients of the radial distortion by solving the linear least-squares – Pseudo Inverse

$$k = \left(D^T D\right)^{-1} D^T d$$

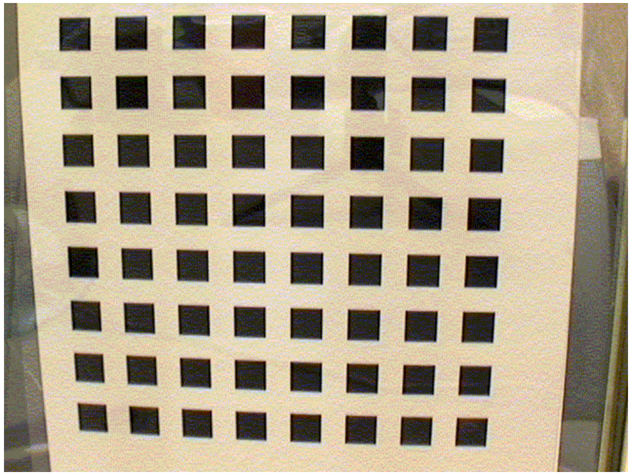
6) Refine all parameters by minimizing MSE

$$\sum_{i=1}^{n} \sum_{j=1}^{m} \left\| m_{ij} - \breve{m}\left(A, k_1, k_2 R_i, t_i, M_j\right) \right\|^2$$

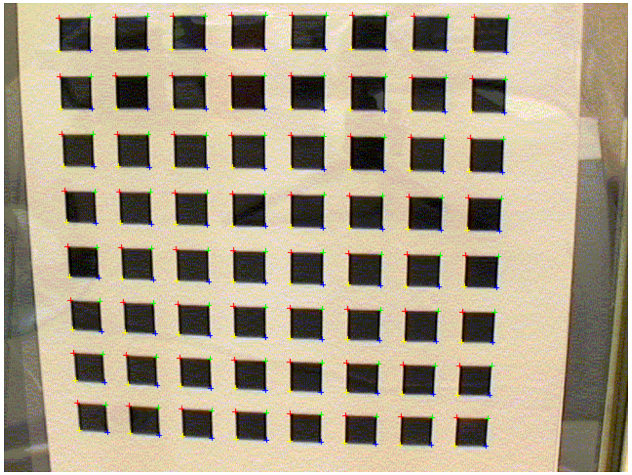
Computation of Camera Calibration



Original planar pattern



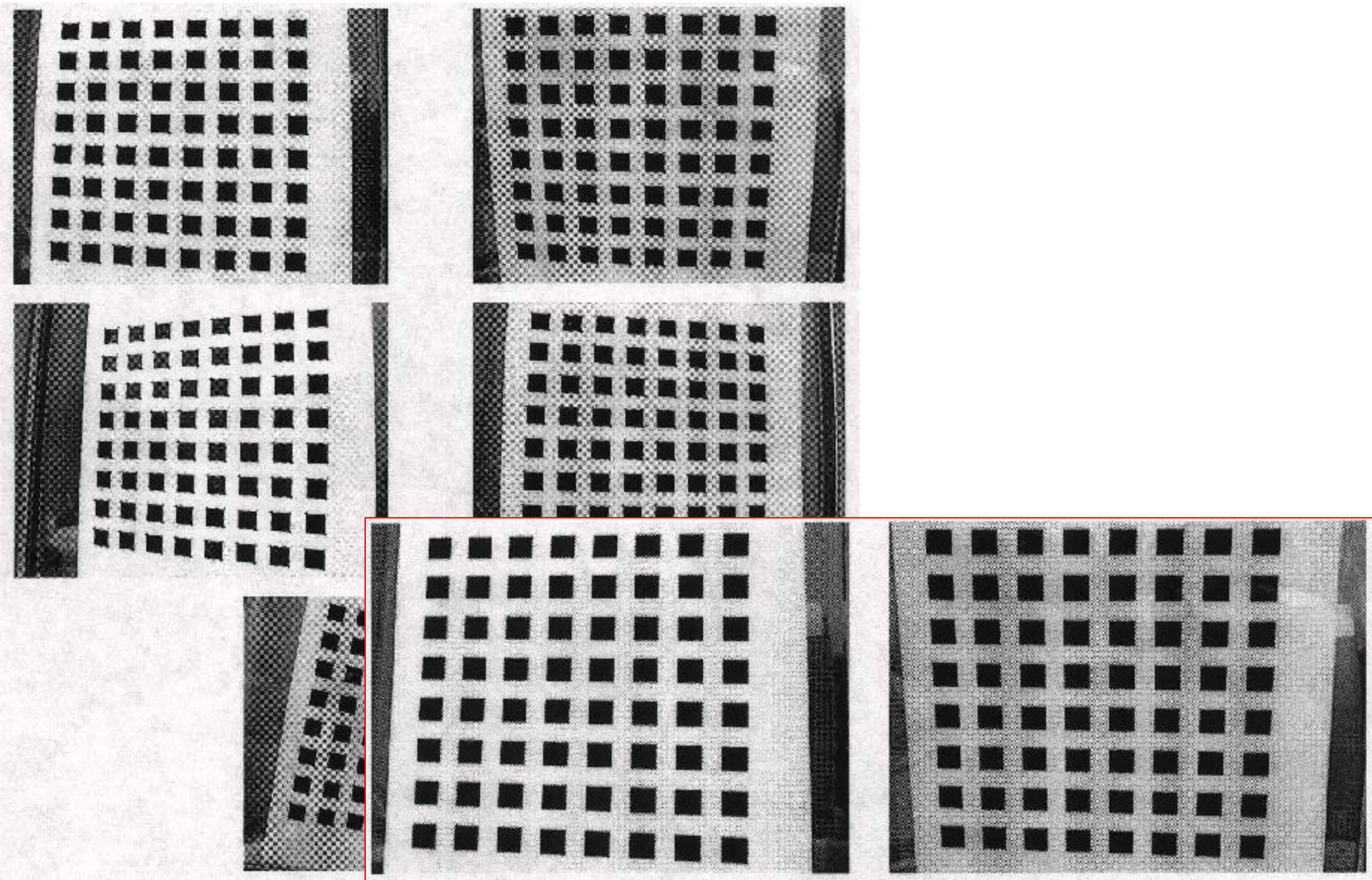
Corrected for radial lens distortion



Edges

- Grab images of this pattern at least two random orientations. The motion/orientation does not need to be known.
- Closed-form solution then non-linear refinement
- Result: Camera intrinsic and extrinsic parameters

Real Data:



Basic Equations

Notation:

?

$$s\tilde{m} = \boxed{A\,[R\ t]}\ \tilde{M}$$

$$m = [u, v]^T \qquad \tilde{m} = [u, v, 1]^T$$

$$M = [X, Y, Z]^T \qquad \tilde{M} = [X, Y, Z, 1]^T$$

$$s\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha & \gamma & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & T_X \\ r_{21} & r_{22} & r_{23} & T_Y \\ r_{31} & r_{32} & r_{33} & T_Z \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

Z=?

There are total 2mn equations => overdetermined.

where *m*: Total points on the model plane.

n: Different orientation images of the model plane.

Homography Between the Model Plane and its Image:

Assume the model plane is on Z=0

$$s\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \overset{?}{\boxed{A\begin{bmatrix} r_1 & r_2 & r_3 & t \end{bmatrix}}} \begin{bmatrix} X \\ Y \\ 0 \\ 1 \end{bmatrix} = \overset{?}{\boxed{A\begin{bmatrix} r_1 & r_2 & t \end{bmatrix}}} \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix}$$

$$s\tilde{m} = \overset{?}{\boxed{H}}\tilde{M} \quad with \;\; H = A\begin{bmatrix} r_1 & r_2 & t \end{bmatrix} = \begin{bmatrix} h_1 & h_2 & h_3 \end{bmatrix}$$

$$\tilde{M} = [X, Y, 1]^T$$

$$M = [X, Y]^T$$

2.2 Homography Introduction

- **Assume: Model plane Z = 0 in world coordinate system**

?

$$s\tilde{m} = \boxed{A[R\ t]}\widetilde{M}$$

? ?

$$\Rightarrow s\begin{bmatrix}u\\v\\1\end{bmatrix} = \boxed{A[r_1\ r_2\ r_3\ t]}\begin{bmatrix}X\\Y\\0\\1\end{bmatrix} = \boxed{A[r_1\ r_2\ t]}\begin{bmatrix}X\\Y\\1\end{bmatrix}$$

- ? Homography $H = A[r_1\ r_2\ t]\left(= [h_1\ h_2\ h_3] = \begin{bmatrix}\bar{h}_1^T\\\bar{h}_2^T\\\bar{h}_3^T\end{bmatrix}\right)$

$$\Rightarrow s\tilde{m} = H\widetilde{M}$$

?

$$\Rightarrow s\begin{bmatrix}u\\v\\1\end{bmatrix} = \begin{bmatrix}\bar{h}_1^T\\\bar{h}_2^T\\\bar{h}_3^T\end{bmatrix}\begin{bmatrix}X\\Y\\1\end{bmatrix} = \boxed{\begin{bmatrix}\bar{h}_{11} & \bar{h}_{12} & \bar{h}_{13}\\\bar{h}_{21} & \bar{h}_{22} & \bar{h}_{23}\\\bar{h}_{31} & \bar{h}_{32} & \bar{h}_{33}\end{bmatrix}}\begin{bmatrix}X\\Y\\1\end{bmatrix}$$

$$\Rightarrow \begin{cases} su = \bar{h}_{11}X + \bar{h}_{12}Y + \bar{h}_{13}\\ sv = \bar{h}_{21}X + \bar{h}_{22}Y + \bar{h}_{23}\\ s = \bar{h}_{31}X + \bar{h}_{32}Y + \bar{h}_{33}\end{cases}$$

$$\Rightarrow \begin{cases} u = \dfrac{\bar{h}_{11}X + \bar{h}_{12}Y + \bar{h}_{13}}{\bar{h}_{31}X + \bar{h}_{32}Y + \bar{h}_{33}} = \dfrac{\bar{h}_1^T\widetilde{M}}{\bar{h}_3^T\widetilde{M}}\\[2ex] v = \dfrac{\bar{h}_{21}X + \bar{h}_{22}Y + \bar{h}_{23}}{\bar{h}_{31}X + \bar{h}_{32}Y + \bar{h}_{33}} = \dfrac{\bar{h}_2^T\widetilde{M}}{\bar{h}_3^T\widetilde{M}}\end{cases} \Longrightarrow \hat{m}_i = \begin{bmatrix}u\\v\end{bmatrix} = \frac{1}{\bar{h}_3^T\widetilde{M}}\begin{bmatrix}\bar{h}_1^T\widetilde{M}\\\bar{h}_2^T\widetilde{M}\end{bmatrix}$$

$$\boldsymbol{m} - \widehat{\boldsymbol{m}} = \begin{matrix} \boldsymbol{u} - \dfrac{\overline{\boldsymbol{h}_1}\ \widetilde{\boldsymbol{M}_1}}{\overline{\boldsymbol{h}_3}\ \widetilde{\boldsymbol{M}_3}} = \boldsymbol{0}\\[2ex] \boldsymbol{v} - \dfrac{\overline{\boldsymbol{h}_2}\ \widetilde{\boldsymbol{M}_2}}{\overline{\boldsymbol{h}_3}\ \widetilde{\boldsymbol{M}_3}} = \boldsymbol{0}\end{matrix}$$

2.2 Homography Estimation (1/2)

Estimation of the Homography Between the Model Plane and its Image -

$$\boxed{\min \sum_i (m_i - \hat{m}_i)^T \Lambda_{m_i}^{-1} (m_i - \hat{m}_i)}$$

$$where \quad \hat{m}_i = \frac{1}{\bar{h}_3^T M_i} \begin{bmatrix} \bar{h}_1^T M_i \\ \bar{h}_2^T M_i \end{bmatrix} \quad with\ \bar{h}_i, the\ ith\ row\ of\ H$$

$$\Lambda_{m_i} = \sigma^2 I \quad \text{for all } i$$

$$\min_H \sum_i \|m_i - \hat{m}_i\|^2 \longrightarrow \min \sum_{i=1}^{n} \sum_{j=1}^{m} \left\| m_{ij} - \hat{m}(A, R_i, t_i, M_j) \right\|^2$$

$$Let \quad x = [\bar{h}_1^T, \bar{h}_2^T, \bar{h}_3^T]^T \ ?$$

$$\overset{?}{L}x = \begin{bmatrix} \tilde{M}^T & 0^T & -u\tilde{M}^T \\ 0^T & \tilde{M}^T & -v\tilde{M}^T \end{bmatrix} \overset{?}{\begin{bmatrix} \bar{h}_1 \\ \bar{h}_2 \\ \bar{h}_3 \end{bmatrix}} = 0 \Longleftarrow \boldsymbol{m} - \hat{\boldsymbol{m}} = \begin{matrix} \boldsymbol{u} - \dfrac{\overline{\boldsymbol{h}_1}\,\widetilde{\boldsymbol{M}_1}}{\overline{\boldsymbol{h}_3}\,\widetilde{\boldsymbol{M}_3}} = \boldsymbol{0} \\ \boldsymbol{v} - \dfrac{\overline{\boldsymbol{h}_2}\,\widetilde{\boldsymbol{M}_2}}{\overline{\boldsymbol{h}_3}\,\widetilde{\boldsymbol{M}_3}} = \boldsymbol{0} \end{matrix}$$

Closed-Form Solution

Covariance Matrix $L^T L$

Jenn-Jier James Lien

2.2 Homography Estimation (2/2)

1) **Before applying LM Algorithm to solve H, we need an initial guess:**

$$\begin{cases} u = \dfrac{\bar{h}_{11}X + \bar{h}_{12}Y + \bar{h}_{13}}{\bar{h}_{31}X + \bar{h}_{32}Y + \bar{h}_{33}} \\ v = \dfrac{\bar{h}_{21}X + \bar{h}_{22}Y + \bar{h}_{23}}{\bar{h}_{31}X + \bar{h}_{32}Y + \bar{h}_{33}} \end{cases}$$

$$\Rightarrow \begin{cases} \bar{h}_{11}X + \bar{h}_{12}Y + \bar{h}_{13} - 1u\bar{h}_{31}X - u\bar{h}_{32}Y - u\bar{h}_{33} = 0 \\ \bar{h}_{21}X + \bar{h}_{22}Y + \bar{h}_{23} - v\bar{h}_{31}X - v\bar{h}_{32}Y - v\bar{h}_{33} = 0 \end{cases}$$

$$\hat{m}_i = \begin{bmatrix} u \\ v \end{bmatrix} = \frac{1}{\bar{h}_3^T \tilde{M}} \begin{bmatrix} \bar{h}_1^T \tilde{M} \\ \bar{h}_2^T \tilde{M} \end{bmatrix}$$

$$\Rightarrow \begin{bmatrix} X & Y & 1 & 0 & 0 & 0 & -uX & -uY & -u \\ 0 & 0 & 0 & X & Y & 1 & -vX & -vY & -v \end{bmatrix} \begin{bmatrix} \bar{h}_{11} \\ \bar{h}_{12} \\ \bar{h}_{13} \\ \bar{h}_{21} \\ \bar{h}_{22} \\ \bar{h}_{23} \\ \bar{h}_{31} \\ \bar{h}_{32} \\ \bar{h}_{33} \end{bmatrix} = \begin{bmatrix} \tilde{M}^T & 0^T & -u\tilde{M}^T \\ 0^T & \tilde{M}^T & -v\tilde{M}^T \end{bmatrix} \begin{bmatrix} \bar{h}_1^T \\ \bar{h}_2^T \\ \bar{h}_3^T \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$\overset{?}{L}x = \begin{bmatrix} \tilde{M}^T & 0^T & -u\tilde{M}^T \\ 0^T & \tilde{M}^T & -v\tilde{M}^T \end{bmatrix} \begin{bmatrix} \overset{?}{\bar{h}_1} \\ \bar{h}_2 \\ \bar{h}_3 \end{bmatrix} = 0$$

When we are given n points, we have n above equations, which can be written in matrix equation as $\mathbf{Lx} = \mathbf{0}$, where $\mathbf{L}$ is a $2n \times 9$ matrix. As $\mathbf{x}$ is defined up to a scale factor, the solution is well known to be the right singular vector of $\mathbf{L}$ associated with the smallest singular value (or equivalently, the eigenvector of $\mathbf{L}^T\mathbf{L}$ associated with the smallest eigenvalue).

2.3 Constraints on the Intrinsic Parameters

$$H = [h_1 \quad h_2 \quad h_3] = \bar{\lambda} A [r_1 \quad r_2 \quad t]$$

2. Knowing H allows us to obtain constraints on the intrinsic parameters:

$$\begin{cases} h_1{}^T A^{-T} A^{-1} h_2 = 0 \\ h_1{}^T A^{-T} A^{-1} h_1 = h_2{}^T A^{-T} A^{-1} h_2 \end{cases}$$

➢ $H = [h_1\ h_2\ h_3] = \lambda A [r_1\ r_2\ t]$

$$\Rightarrow \begin{cases} r_1 = \dfrac{1}{\lambda} A^{-1} h_1 \\ r_2 = \dfrac{1}{\lambda} A^{-1} h_2 \end{cases}$$

1. $\because r_1 \perp r_2$

$\therefore r_1 \cdot r_2 = r_1{}^T r_2 = (A^{-1} h_1)^T (A^{-1} h_2) = h_1{}^T A^{-T} A^{-1} h_2 = 0$

2.$\because \|\mathrm{r}_1\| = \|\mathrm{r}_2\| = 1$

$\therefore r_1{}^T r_1 = r_2{}^T r_2$

$\Rightarrow (A^{-1} h_1)^T (A^{-1} h_1) = (A^{-1} h_2)^T (A^{-1} h_2)$

$\Rightarrow h_1{}^T A^{-T} A^{-1} h_1 = h_2{}^T A^{-T} A^{-1} h_2$

Geometric Interpretation:

$$\begin{bmatrix} r_3 \\ r_3^T t \end{bmatrix}^T \begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix} = 0$$

$$x_\infty = a\begin{bmatrix} r_1 \\ 0 \end{bmatrix} + b\begin{bmatrix} r_2 \\ 0 \end{bmatrix} = \begin{bmatrix} ar_1 + br_2 \\ 0 \end{bmatrix}$$

$$x_\infty{}^T x_\infty = 0$$

$$(ar_1 + br_2)^T (ar_1 + br_2) = 0, \quad or \quad a^2 + b^2 = 0$$

$$b = \pm a\, i, \text{ where } i^2 = -1$$

$$x_\infty = a\begin{bmatrix} r_1 + ir_2 \\ 0 \end{bmatrix}$$

$$\tilde{m}_\infty = A(r_1 \pm i r_2) = h_1 \pm i h_2$$

$$(h_1 \pm i h_2)^T A^{-T} A^{-1} (h_1 \pm i h_2) = 0$$

3.0 Solving Camera Calibration

3.1 Closed-Form Solution

$$Vb = 0$$

3.2 Maximum Likelihood Estimation

$$\min \sum_{i=1}^{n} \sum_{j=1}^{m} \left\| m_{ij} - \hat{m}\left(A, R_i, t_i, M_j\right) \right\|^2$$

3.1 Closed-Form Solution:

$$B = \lambda A^{-T}A^{-1} \equiv \lambda \begin{bmatrix} B_{11} & B_{12} & B_{13} \\ B_{21} & B_{22} & B_{23} \\ B_{31} & B_{32} & B_{33} \end{bmatrix} = \lambda \begin{bmatrix} \frac{1}{\alpha^2} & -\frac{\gamma}{\alpha^2\beta} & \frac{v_0\gamma - u_0\beta}{\alpha^2\beta} \\ -\frac{\gamma}{\alpha^2\beta} & \frac{\gamma^2}{\alpha^2\beta^2}+\frac{1}{\beta^2} & -\frac{\gamma(v_0\gamma-u_0\beta)}{\alpha^2\beta^2}-\frac{v_0}{\beta^2} \\ \frac{v_0\gamma-u_0\beta}{\alpha^2\beta} & -\frac{\gamma(v_0\gamma-u_0\beta)}{\alpha^2\beta^2}-\frac{v_0}{\beta^2} & \frac{(v_0\gamma-u_0\beta)^2}{\alpha^2\beta^2}+\frac{v_0^2}{\beta^2}+1 \end{bmatrix}$$

$$b = [B_{11}, B_{12}, B_{22}, B_{13}, B_{23}, B_{33}]^T$$

Let the ith column vector of H be $h_i = [h_{i1}, h_{i2}, h_{i3}]^T$

$$\boxed{h_i^T B h_j = v_{ij}^T \overset{?}{b}}$$

$$H = [h_1 \quad h_2 \quad h_3] = \bar{\lambda} A [r_1 \quad r_2 \quad t]$$

$$v_{ij} = [h_{i1}h_{j1}, h_{i1}h_{j2} + h_{i2}h_{j1}, h_{i2}h_{j2}, h_{i3}h_{j1} + h_{i1}h_{j3}, h_{i3}h_{j2} + h_{i2}h_{j3}, h_{i3}h_{j3}]^T$$

$$\boxed{\begin{bmatrix} v_{12}^T \\ (v_{11} - v_{22})^T \end{bmatrix} \overset{?}{b} = 0 \qquad V\overset{?}{b} = 0}$$

$$\xrightarrow{\text{Covariance Matrix } V^TV} \quad \overset{?}{b} = [B_{11}, B_{12}, B_{22}, B_{13}, B_{23}, B_{33}]^T$$

3.1 Closed Form Solution (1/2)

- $B = A^{-T}A^{-1} = \begin{bmatrix} \frac{1}{\alpha^2} & -\frac{\gamma}{\alpha^2\beta} & \frac{v_0\gamma - u_0\beta}{\alpha^2\beta} \\ -\frac{\gamma}{\alpha^2\beta} & \frac{\gamma^2}{\alpha^2\beta^2} + \frac{1}{\beta^2} & -\frac{\gamma(v_0\gamma - u_0\beta)}{\alpha^2\beta^2} - \frac{v_0}{\beta^2} \\ \frac{v_0\gamma - u_0\beta}{\alpha^2\beta} & -\frac{\gamma(v_0\gamma - u_0\beta)}{\alpha^2\beta^2} - \frac{v_0}{\beta^2} & \frac{(v_0\gamma - u_0\beta)^2}{\alpha^2\beta^2} + \frac{{v_0}^2}{\beta^2} + 1 \end{bmatrix}$
  - Define: $b = [B_{11}B_{12}B_{22}B_{13}B_{23}B_{33}]^T$

    $v_{ij} = [h_{i1}h_{j1}, h_{i1}h_{j2} + h_{i2}h_{j1}, h_{i2}h_{j2}, h_{i3}h_{j1} + h_{i1}h_{j3}, h_{i3}h_{j2} + h_{i2}h_{j3}, h_{i3}h_{j3}]^T$

  $\Rightarrow h_i^T B h_j = v_{ij}^T b$

  - $\begin{cases} {h_1}^T A^{-T} A^{-1} h_2 = 0 \\ {h_1}^T A^{-T} A^{-1} h_1 = {h_2}^T A^{-T} A^{-1} h_2 \end{cases}$ can be rewritten as $\begin{bmatrix} v_{12}^T \\ (v_{11} - v_{22})^T \end{bmatrix}$? $\mathrm{b} = 0$

- **6 variables to be solved, each image stacks 2 equations. (*n* images of model plane are obsearved)**
  - $n \geq 3$: get a unique solution b.
  - $n = 2$: impose skewless constraint $\gamma = 0$??still missing one equation
  - $n = 1$: assume u_0, v_0 are known and $\gamma = 0$

3.1 Closed Form Solution (2/2)

- **After b is estimated, intrinsic parameter A can be computed by $B = \lambda A^{-T}A^{-1}$:**

$$v_0 = \frac{(B_{12}B_{13} - B_{11}B_{23})}{(B_{11}B_{22} - B_{12}^2)}$$

$$\lambda = B_{33} - \frac{[B_{13}^2 + v_o(B_{12}B_{13} - B_{11}B_{23})]}{B_{11}}$$

$$\alpha = \sqrt{\lambda/B_{11}}$$

$$\beta = \sqrt{\lambda B_{11}/(B_{11}B_{22} - B_{12}^2)}$$

$$\gamma = -\frac{B_{12}\alpha^2\beta}{\lambda}$$

$$u_0 = \frac{\gamma v_0}{\beta} - B_{13}\alpha^2/\lambda$$

- **After A is computed, r_1, r_2, r_3, t can be computed by $[h_1 h_2 h_3] = A[r_1 r_2 t]$**

$$r_1 = \sigma A^{-1}h_1$$

$$r_2 = \sigma A^{-1}h_2$$

$$r_3 = r_1 \times r_2$$

$$t = \sigma A^{-1}h_3$$

Where $\sigma = \frac{1}{\|A^{-1}h_1\|} = \frac{1}{\|A^{-1}h_2\|}$

- **Better results can be obtained by the method described in Appendix C**

Extraction of the Intrinsic Parameters from Matrix B -

Compute camera intrinsic matrix A

Known

$$B = \lambda A^{-T} A^{-1} \longleftarrow \mathbf{A} \longleftarrow b = \left[B_{11}, B_{12}, B_{22}, B_{13}, B_{23}, B_{33}\right]^T$$

$$v_0 = \left(B_{12}B_{13} - B_{11}B_{23}\right) / \left(B_{11}B_{22} - B_{12}^2\right)$$

$$\lambda = B_{33} - \left[B_{13}^2 + v_0\left(B_{12}B_{13} - B_{11}B_{23}\right)\right] / B_{11}$$

$$\alpha = \sqrt{\lambda / B_{11}}$$

$$\beta = \sqrt{\lambda B_{11} / \left(B_{12}^2 - B_{11}B_{22}\right)}$$

$$\gamma = -B_{12}\alpha^2\beta / \lambda$$

$$u_0 = \gamma v_0 / \beta - B_{13}\alpha^2 / \lambda$$

Extrinsic Parameters -

Known intrinsic matrix A, compute extrinsic parameters

$$r_1 = \lambda A^{-1} h_1$$

$$r_2 = \lambda A^{-1} h_2$$

$$r_3 = r_1 * r_2$$

$$t = \lambda A^{-1} h_3$$

$$\lambda = 1 \Big/ \left\| A^{-1} h_1 \right\| = 1 \Big/ \left\| A^{-1} h_2 \right\|$$

$$R = \left[r_1, r_2, r_3 \right]$$

C. Approximating a 3x3 Orthogonal Matrix by a Rotation Matrix -

$$\min_R \|R - Q\|_F^2 \qquad \textit{subject to} \quad R^T R = I$$

$$\|R - Q\|_F^2 = trace\left((R-Q)^T (R-Q)\right) = 3 + trace\left(Q^T Q\right) - 2trace\left(R^T Q\right)$$

Let $Q=USV^T$, where $S = diag(\sigma_1, \sigma_2, \sigma_3)$

Define an orthogonal matrix $Z=V^TR^TU$ => $R=UZV^T$

$$trace\left(R^T Q\right) = trace\left(R^T USV^T\right) = trace\left(V^T R^T US\right) = trace\left(ZS\right) \sum_{i=1}^{3} z_{ii}\sigma_i \le \sum_{i=1}^{3} \sigma_i$$

$Z=I$

$$R = UV^T$$

Experiment Results

- **Application to Image Based Modeling**

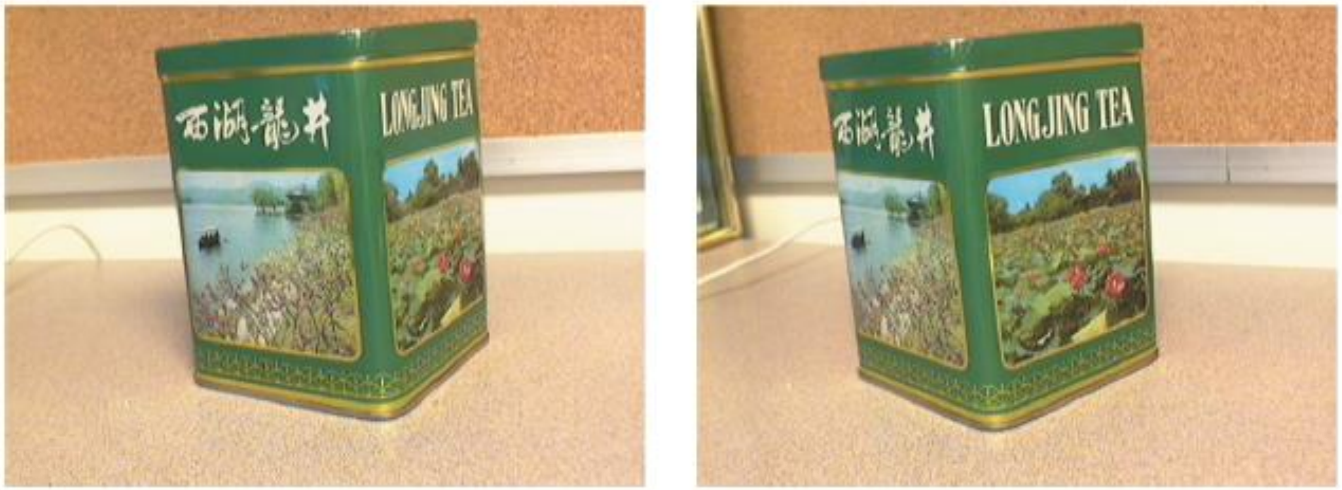


Figure 6: Two images of a tea tin

Figure 7: Three rendered views of the reconstructed tea tin

3.2 Maximum Likelihood Estimation Using Levenberg-Marquardt Algorithm

$$\min \sum_{i=1}^{n} \sum_{j=1}^{m} \left\| m_{ij} - \hat{m}\left(A, R_i, t_i, M_j\right) \right\|^2$$

Step 1: Linear Minimization

1.1 Closed-form Solution

or

1.2

1.2.1 $\frac{\partial E}{\partial A} = 0, \frac{\partial E}{\partial R_i} = 0, \frac{\partial E}{\partial t_i} = 0, \frac{\partial E}{\partial M_j} = 0$

1.2.2 By first order Taylor series expansion individually

Step 2: Nonlinear Minimization

3.3 Dealing with Radial Distortion:

- **It is also possible to calibrate for non-linear camera effects such as radial distortion and there are situations where it is important to estimate these parameters**
- **Note how straight lines are distorted into curves by radial distortion in this image**
- **Corresponds to a dilation of the image**
- **It is most pronounced in optical systems with a wide field of view**

1) Radial Distortion

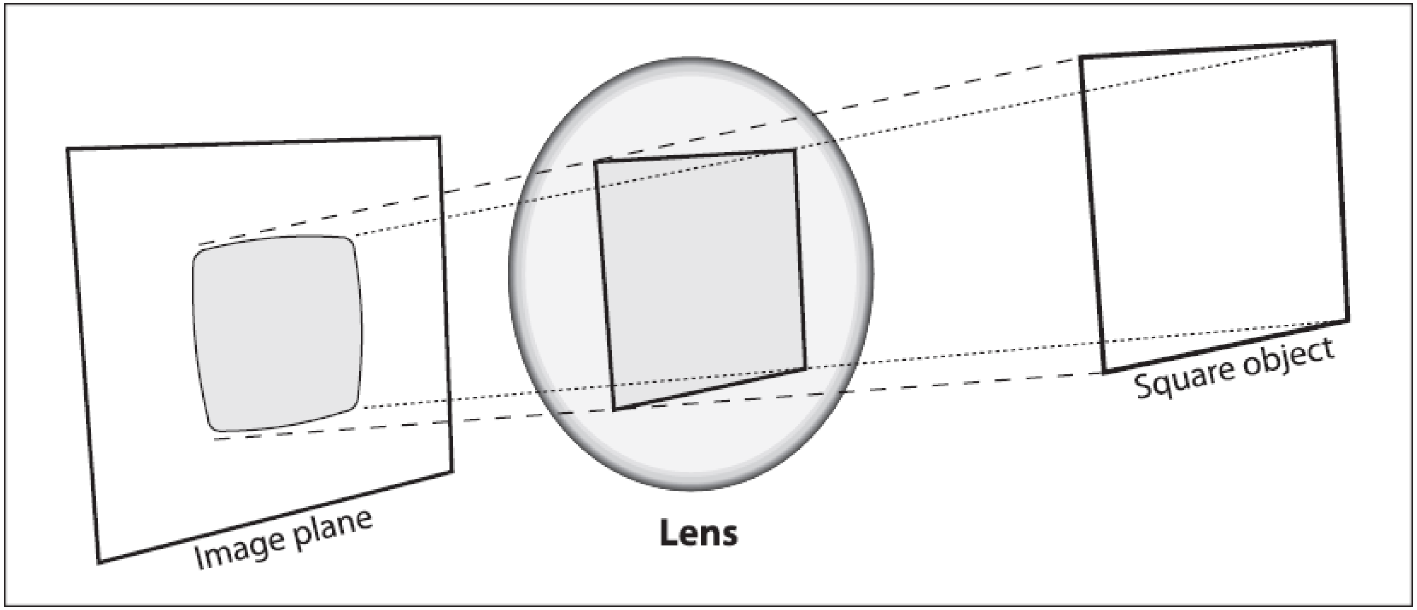


Figure 11-3. Radial distortion: rays farther from the center of a simple lens are bent too much compared to rays that pass closer to the center; thus, the sides of a square appear to bow out on the image plane (this is also known as barrel distortion)

$$x_{corrected} = x(1 + k_1 r^2 + k_2 r^4 + k_3 r^6)$$

$$y_{corrected} = y(1 + k_1 r^2 + k_2 r^4 + k_3 r^6)$$

Here, (x, y) is the original location (on the imager) of the distorted point and $(x_{corrected}, y_{corrected})$ is the new location as a result of the correction. Figure 11-4 shows displacements of a rectangular grid that are due to radial distortion. External points on a front-facing rectangular grid are increasingly displaced inward as the radial distance from the optical center increases.

2) Tangential Distortion

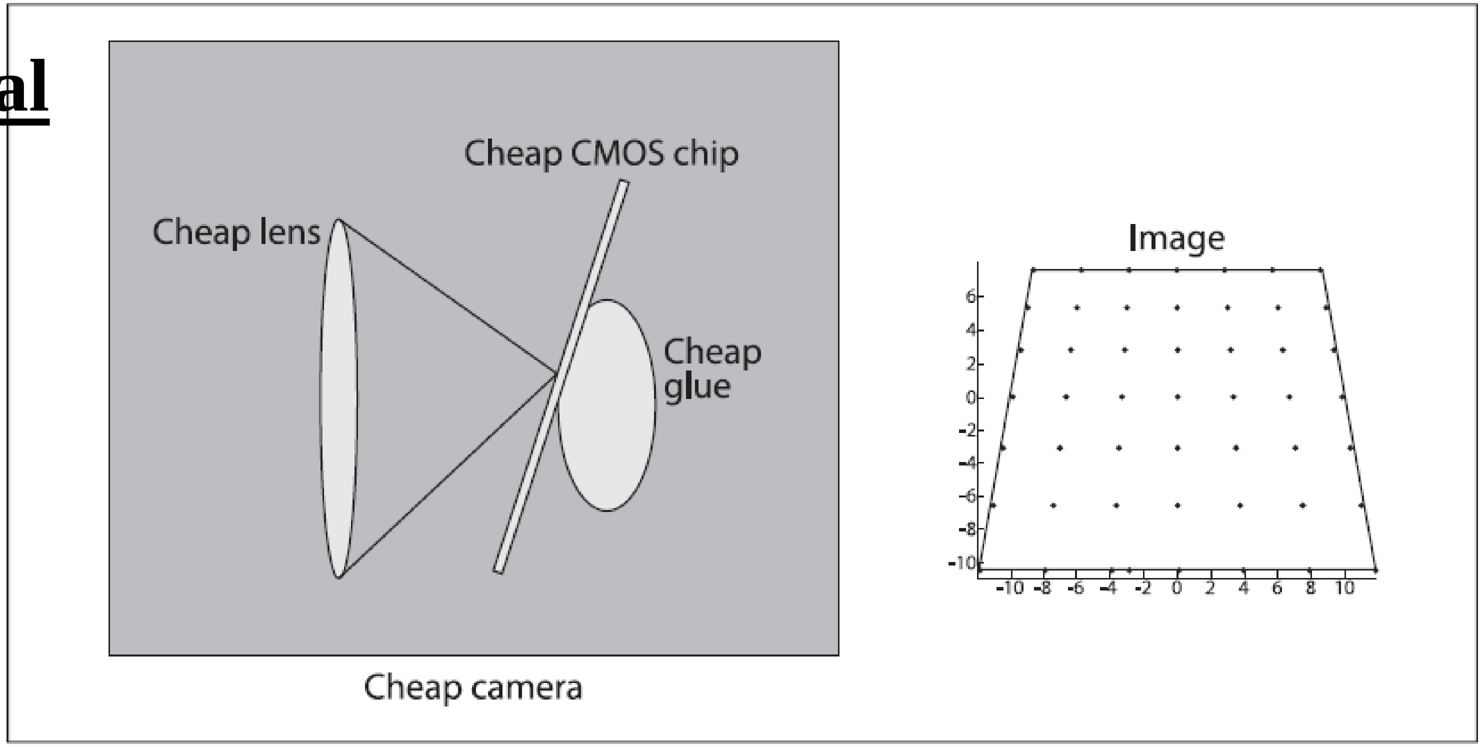


Figure 11-5. Tangential distortion results when the lens is not fully parallel to the image plane; in cheap cameras, this can happen when the imager is glued to the back of the camera (image courtesy of Sebastian Thrun)

The second-largest common distortion is *tangential distortion*. This distortion is due to manufacturing defects resulting from the lens not being exactly parallel to the imaging plane; see Figure 11-5.

Tangential distortion is minimally characterized by two additional parameters, p_1 and p_2, such that:*

$$x_{\text{corrected}} = x + [2p_1 y + p_2(r^2 + 2x^2)]$$

$$y_{\text{corrected}} = y + [p_1(r^2 + 2y^2) + 2p_2 x]$$

Thus in total there are five distortion coefficients that we require. Because all five are necessary in most of the OpenCV routines that use them, they are typically bundled into one *distortion vector*; this is just a 5-by-1 matrix containing k_1, k_2, p_1, p_2, and k_3

$$\begin{pmatrix} X_c \\ Y_c \\ Z_c \end{pmatrix} = \begin{pmatrix} R_{11} & R_{21} & R_{31} \\ R_{12} & R_{22} & R_{32} \\ R_{13} & R_{23} & R_{33} \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} + \begin{pmatrix} T_1 \\ T_2 \\ T_3 \end{pmatrix} \quad \Rightarrow \quad \begin{bmatrix} x \\ y \end{bmatrix} = \frac{f}{Z_c} \begin{bmatrix} X_c \\ Y_c \end{bmatrix}$$

$$\begin{bmatrix} \breve{x} \\ \breve{y} \end{bmatrix} = \left(1 + k_1 r^2 + k_2 r^4\right) \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} 2k_3 xy + k_4(r^2 + 2x^2) \\ k_3(r^2 + 2y^2) + 2k_4 xy \end{bmatrix}$$

radial distortion tangential distortion

$(x, y): distortion-free \quad (\breve{x}, \breve{y}): distortion \quad r^2 = x^2 + y^2$

$$\breve{x} = x\left[1 + k_1\left(x^2 + y^2\right) + k_2\left(x^2 + y^2\right)^2\right]$$

$$\breve{y} = y\left[1 + k_1\left(x^2 + y^2\right) + k_2\left(x^2 + y^2\right)^2\right]$$

$$\begin{bmatrix} \breve{u} \\ \breve{v} \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha & \gamma & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \breve{x} \\ \breve{y} \\ 1 \end{bmatrix}$$

$$\breve{u} = u_0 + \alpha\breve{x} + \gamma\breve{y}$$

$$\breve{v} = v_0 + \beta\breve{y}$$

1. Estimating Radial Distortion by Alternation – (slow convergence)

$$\begin{bmatrix} \breve{u} \\ \breve{v} \\ 1 \end{bmatrix} = \begin{bmatrix} \alpha & \gamma & u_0 \\ 0 & \beta & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \breve{x} \\ \breve{y} \\ 1 \end{bmatrix}$$

$$\min \sum_{i=1}^{n} \sum_{j=1}^{m} \left\| m_{ij} - \hat{m}\left(A, R_i, t_i, M_j\right) \right\|^2$$

Observe at image   Known (computation)

$$\breve{x} = x + x\left[k_1\left(x^2 + y^2\right) + k_2\left(x^2 + y^2\right)^2\right]$$

$$\breve{y} = y + y\left[k_1\left(x^2 + y^2\right) + k_2\left(x^2 + y^2\right)^2\right]$$

$$\begin{bmatrix} x\left(x^2 + y^2\right) & x\left(x^2 + y^2\right)^2 \\ y\left(x^2 + y^2\right) & y\left(x^2 + y^2\right)^2 \end{bmatrix} \begin{bmatrix} k_1 \\ k_2 \end{bmatrix} = \begin{bmatrix} \breve{x} - x \\ \breve{y} - y \end{bmatrix}$$

Dk=d, where k=[k_1, k_2]^T

$$k = \left(D^T D\right)^{-1} D^T d \xrightarrow[\text{until convergence}]{\text{Refine the estimate of parameters}}$$

3.3 Dealing with Radial Distortion:

Let (u, v) be the ideal (nonobservable distortion-free) pixel image coordinates, and $(\breve{u}, \breve{v})$ the corresponding real observed image coordinates. The ideal points are the projection of the model points according to the pinhole model. Similarly, (x, y) and $(\breve{x}, \breve{y})$ are the ideal (distortion-free) and real (distorted) normalized image coordinates. We have [2, 25]

$$\breve{x} = x + x[k_1(x^2 + y^2) + k_2(x^2 + y^2)^2]$$
$$\breve{y} = y + y[k_1(x^2 + y^2) + k_2(x^2 + y^2)^2] ,$$

where k_1 and k_2 are the coefficients of the radial distortion. The center of the radial distortion is the same as the principal point. From* $\breve{u} = u_0 + \alpha\breve{x} + \gamma\breve{y}$ and $\breve{v} = v_0 + \beta\breve{y}$ and assuming $\gamma = 0$, we have

$$\breve{u} = u + (u - u_0)[k_1(x^2 + y^2) + k_2(x^2 + y^2)^2] \tag{11}$$
$$\breve{v} = v + (v - v_0)[k_1(x^2 + y^2) + k_2(x^2 + y^2)^2] . \tag{12}$$

Estimating Radial Distortion by Alternation. As the radial distortion is expected to be small, one would expect to estimate the other five intrinsic parameters, using the technique described in Sect. 3.2, reasonable well by simply ignoring distortion. One strategy is then to estimate k_1 and k_2 after having estimated the other parameters, which will give us the ideal pixel coordinates (u, v). Then, from (11) and (12), we have two equations for each point in each image:

$$\begin{bmatrix} (u-u_0)(x^2+y^2) & (u-u_0)(x^2+y^2)^2 \\ (v-v_0)(x^2+y^2) & (v-v_0)(x^2+y^2)^2 \end{bmatrix} \begin{bmatrix} k_1 \\ k_2 \end{bmatrix} = \begin{bmatrix} \breve{u}-u \\ \breve{v}-v \end{bmatrix} .$$

Given m points in n images, we can stack all equations together to obtain in total $2mn$ equations, or in matrix form as $\mathbf{Dk} = \mathbf{d}$, where $\mathbf{k} = [k_1, k_2]^T$. The linear least-squares solution is given by

$$\mathbf{k} = (\mathbf{D}^T\mathbf{D})^{-1}\mathbf{D}^T\mathbf{d} . \tag{13}$$

Once k_1 and k_2 are estimated, one can refine the estimate of the other parameters by solving (10) with $\hat{\mathbf{m}}(\mathbf{A}, \mathbf{R}_i, \mathbf{t}_i, \mathtt{M}_j)$ replaced by (11) and (12). We can alternate these two procedures until convergence.

Complete Maximum Likelihood Estimation. Experimentally, we found the convergence of the above alternation technique is slow. A natural extension to (10) is then to estimate the complete set of parameters by minimizing the following functional:

$$\sum_{i=1}^{n}\sum_{j=1}^{m}\|\mathbf{m}_{ij}-\breve{\mathbf{m}}(\mathbf{A},k_1,k_2,\mathbf{R}_i,\mathbf{t}_i,\mathtt{M}_j)\|^2\,, \tag{14}$$

$$\sum_{i=1}^{n}\sum_{j=1}^{m}\left\|m_{ij}-\breve{m}\left(A,k_1,k_2R_i,t_i,M_j\right)\right\|^2$$

where $\breve{\mathbf{m}}(\mathbf{A},k1,k2,\mathbf{R}_i,\mathbf{t}_i,\mathtt{M}_j)$ is the projection of point $\mathtt{M}_j$ in image i according to equation (2), followed by distortion according to (11) and (12). This is a nonlinear minimization problem, which is solved with the Levenberg-Marquardt Algorithm as implemented in `Minpack` [18]. A rotation is again parameterized by a 3-vector $\mathbf{r}$, as in Sect. 3.2. An initial guess of $\mathbf{A}$ and $\{\mathbf{R}_i,\mathbf{t}_i|i=1..n\}$ can be obtained using the technique described in Sect. 3.1 or in Sect. 3.2. An initial guess of k_1 and k_2 can be obtained with the technique described in the last paragraph, or simply by setting them to 0.

3.4 Summary:

- **Calibration Procedure:**
  1) Print a pattern and attach it to a planar surface.
  2) Take a few (15~20) images of the model plane under different orientations by moving either the plane or the camera.
  3) Detect the feature points (corner points) in the images.

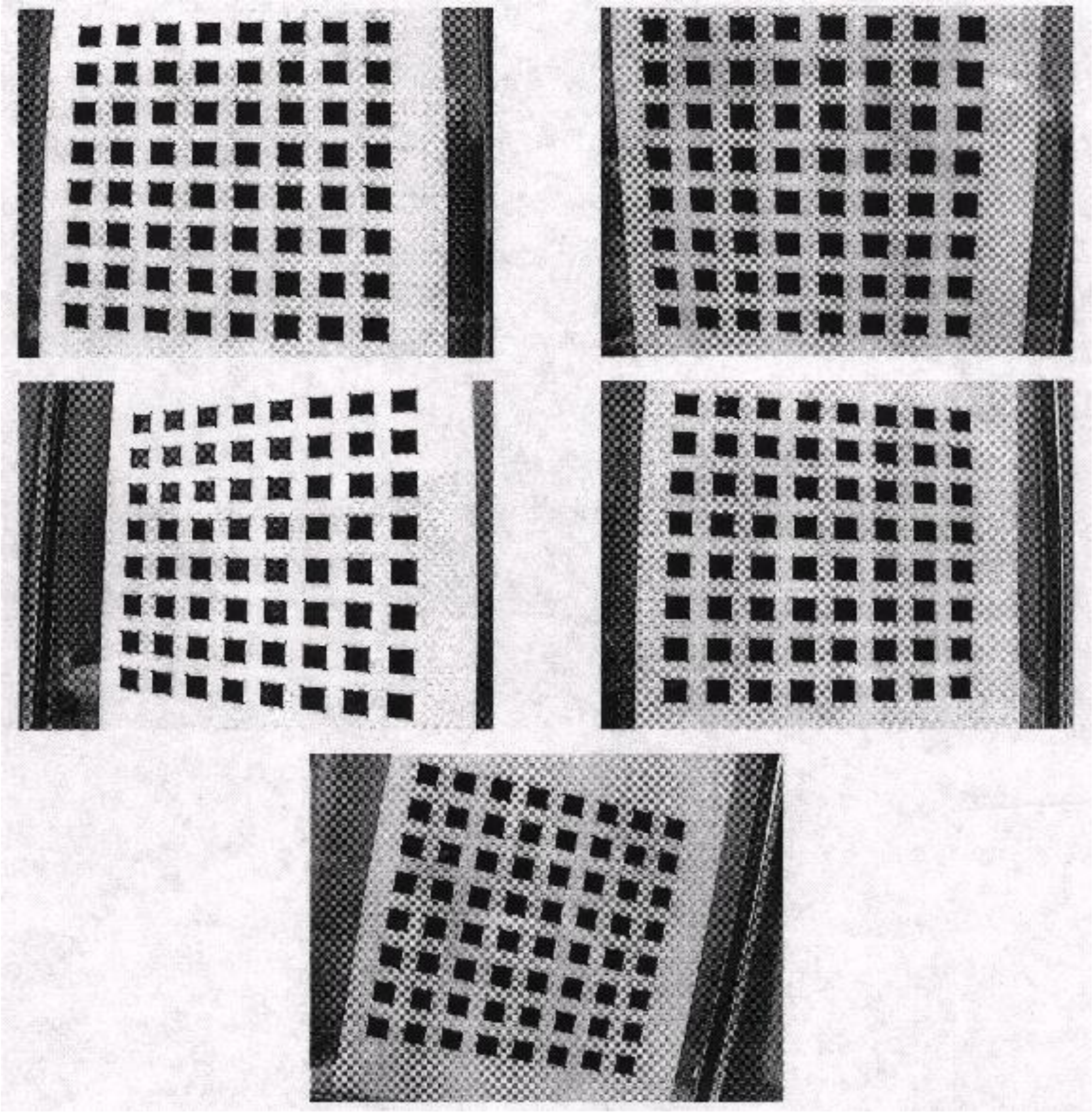
  ---

  4) Estimate the five intrinsic parameters and all the extrinsic parameters using the closed-form solution.
  5) Estimate the coefficients of the radial distortion by solving the linear least-squares – Pseudo Inverse

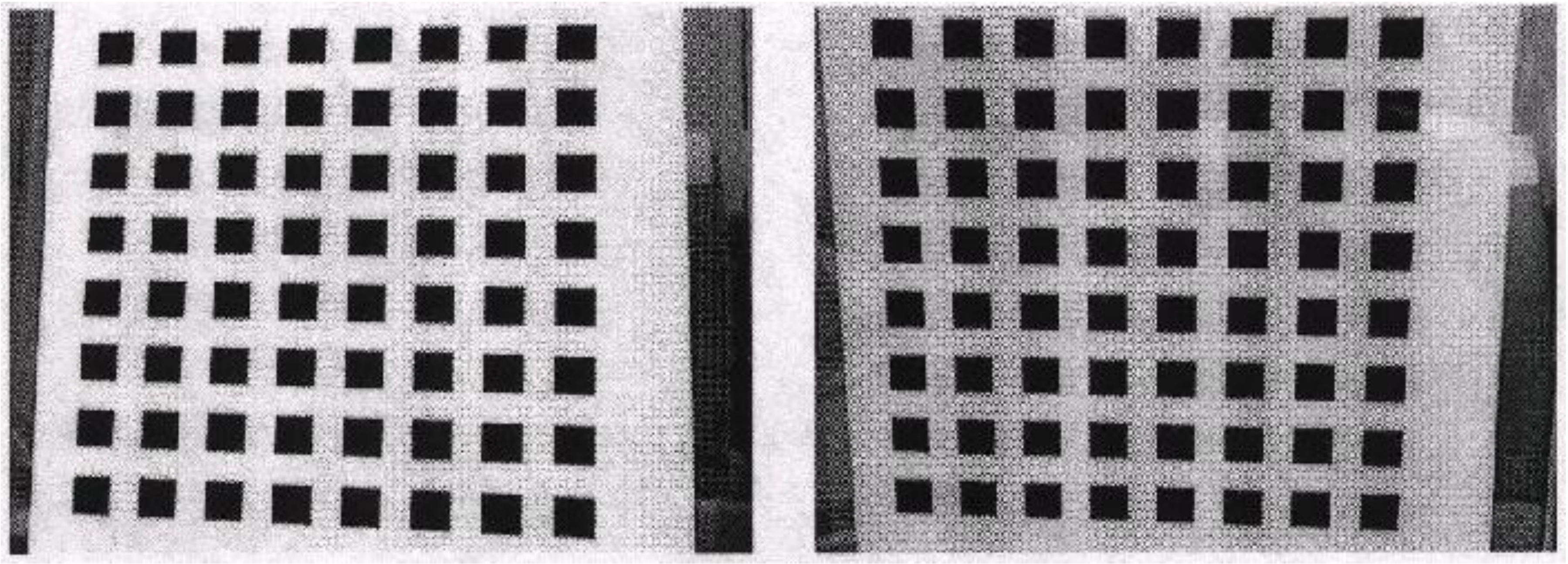
  $$k = \left(D^T D\right)^{-1} D^T d$$

  6) Refine all parameters by minimizing MSE

  $$\sum_{i=1}^{n}\sum_{j=1}^{m}\left\| m_{ij} - \breve{m}\left(A, k_1, k_2 R_i, t_i, M_j\right)\right\|^2$$



Five images of a model plane, together with the extracted corners (indicated by cross)



First and second images after having corrected radial distortion

Degenerate Configurations

$$\begin{cases} h_1^T A^{-T} A^{-1} h_2 = 0 \\ h_1^T A^{-T} A^{-1} h_2 = h_2^T A^{-T} A^{-1} h_2 \end{cases}$$

- **Because above equations are derived from the properties of the rotation matrix, if R_2 is not independent of R_1, then image 2 does not provide additional constraints.**
- **If a plane undergoes a pure translation, then $R_2=R_1$ and image 2 is not helpful for camera calibration.**

❑Proposition:

If the model plane at the second position is parallel to its first position then the second homography does not provide additional constraints.

$$R_1 \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} = R_2$$

$$h_1^{(2)} = \lambda^{(2)}\left(Ar_1^{(1)}\cos\theta + Ar_2^{(1)}\sin\theta\right) = \frac{\lambda^{(2)}}{\lambda^{(1)}}\left(h_1^{(1)}\cos\theta + h_2^{(1)}\sin\theta\right)$$

$$h_2^{(2)} = \lambda^{(2)}\left(-Ar_1^{(1)}\sin\theta + Ar_2^{(1)}\cos\theta\right) = \frac{\lambda^{(2)}}{\lambda^{(1)}}\left(-h_1^{(1)}\sin\theta + h_2^{(1)}\cos\theta\right)$$

$$h_1^{(2)T} A^{-T} A^{-1} h_2^{(2)} = \frac{\lambda^{(2)}}{\lambda^{(1)}}\left[\left(\cos^2\theta - \sin^2\theta\right)\left(h_1^{(1)T} A^{-T} A^{-1} h_2^{(1)}\right) - \cos\theta\sin\theta\left(h_1^{(1)T} A^{-T} A^{-1} h_1^{(1)} - h_2^{(1)T} A^{-T} A^{-1} h_2^{(1)}\right)\right]$$

Camera Calibration Under Known Pure Translation -

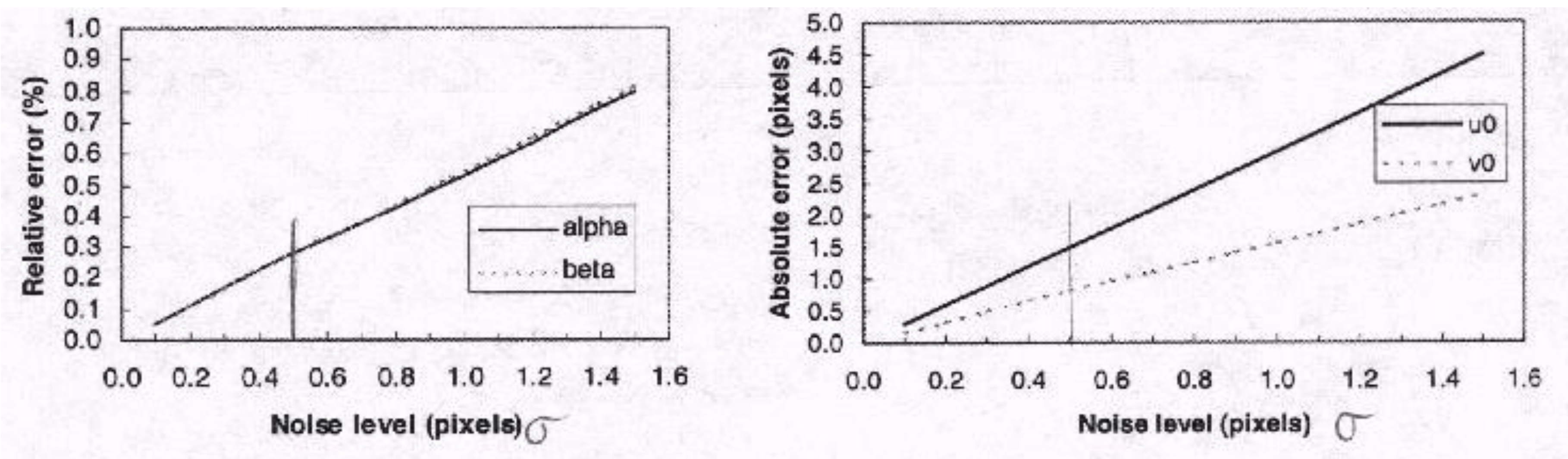
$$t = \lambda A^{-1} h_3, \qquad \text{where} \quad \lambda = 1 \big/ \left\| A^{-1} h_1 \right\|$$

$$t^{(ij)} = t^{(i)} - t^{(j)} = A^{-1}\left(\lambda^{(i)} h_3^{(i)} - \lambda^{(j)} h_3^{(j)}\right)$$

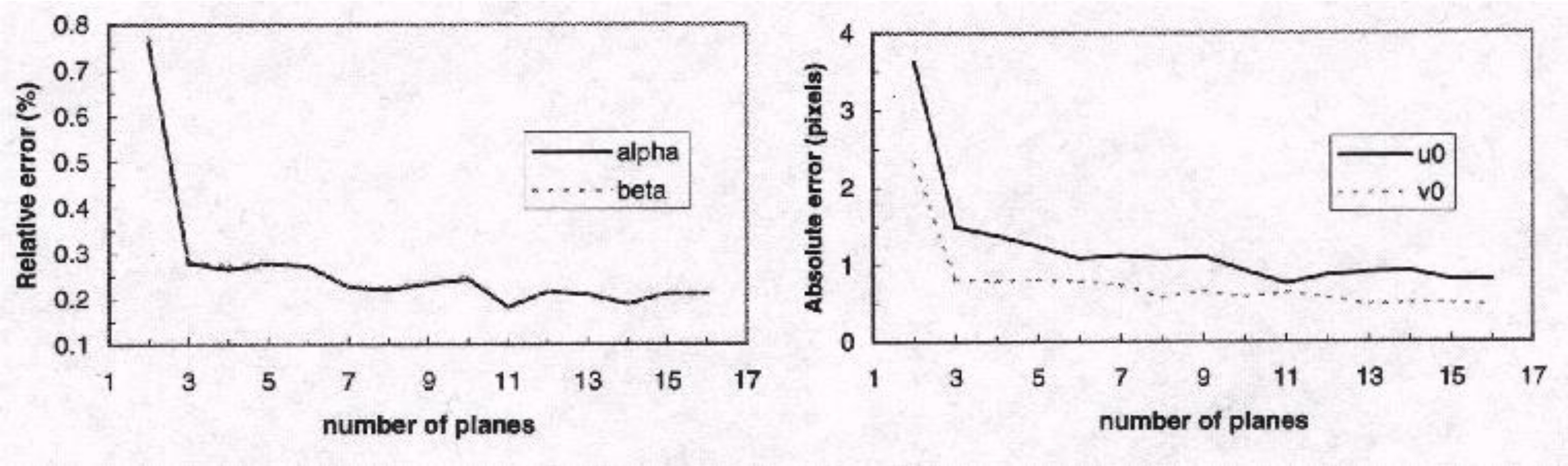
Experimental Results

Computer Simulations:

- **Performance w.r.t. the noise level**
  - The error in u_0 is larger than that in v_0. The main reason is that there are less data in the u direction than in the u direction.

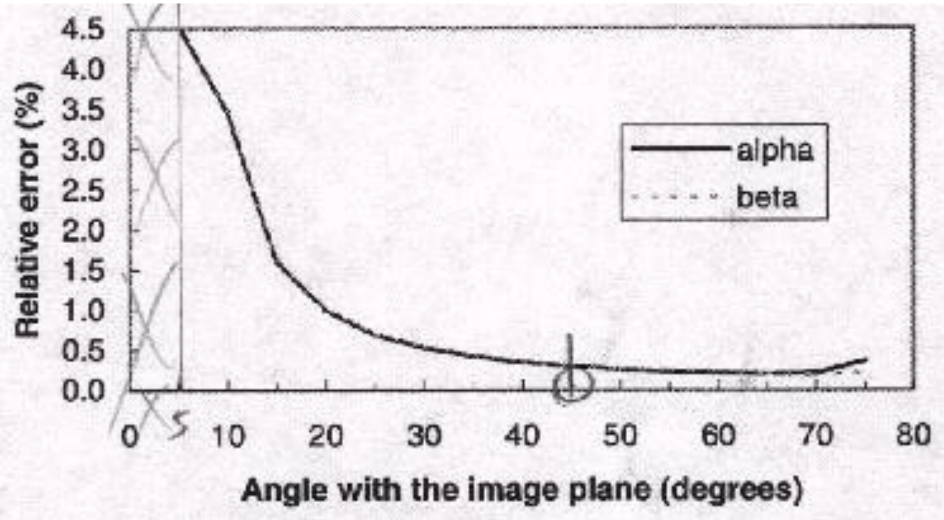


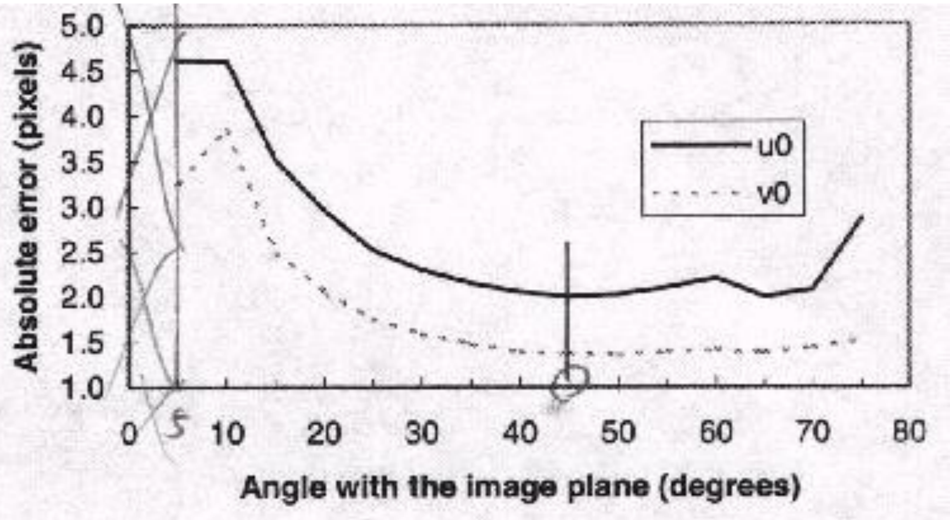
- **Performance w.r.t. the number of planes**
  - The errors decrease when more images are used. From 2 to 3, the errors decrease significantly.



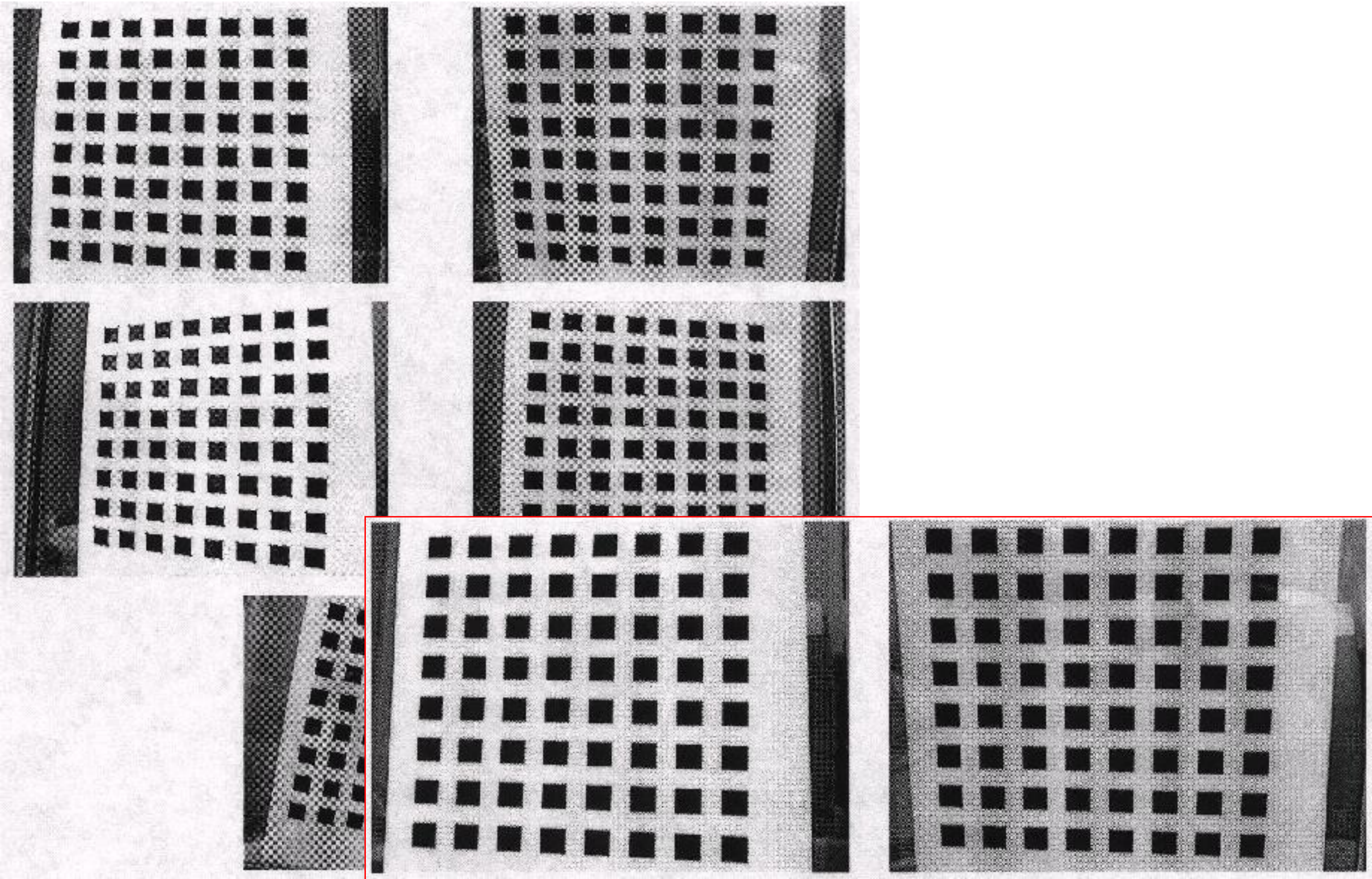
❑ Performance w.r.t the orientation of the model plane

- When Theta=45^0, 40% of the trials failed because the planes are almost parallel each other (degenerate configuration) and the result shown has excluded those trials.
- Best performance seems to be achieved with an angle around 45^0.
- In practice, when the angle increases, foreshortening makes the corner detection less precise, but this is not considered in this experiment.





Real Data:



nb	2 images			3 images			4 images			5 images		
	initial	final	σ	initial	final	σ	initial	final	σ	initial	final	σ
α	825.59	830.47	4.74	917.65	830.80	2.06	876.62	831.81	1.56	877.16	832.50	1.41
β	825.26	830.24	4.85	920.53	830.69	2.10	876.22	831.82	1.55	876.80	832.53	1.38
γ	0	0	0	2.2956	0.1676	0.109	0.0658	0.2867	0.095	0.1752	0.2045	0.078
u_0	295.79	307.03	1.37	277.09	305.77	1.45	301.31	304.53	0.86	301.04	303.96	0.71
v_0	217.69	206.55	0.93	223.36	206.42	1.00	220.06	206.79	0.78	220.41	206.59	0.66
k_1	0.161	−0.227	0.006	0.128	−0.229	0.006	0.145	−0.229	0.005	0.136	−0.228	0.003
k_2	−1.955	0.194	0.032	−1.986	0.196	0.034	−2.089	0.195	0.028	−2.042	0.190	0.025
RMS	0.761	0.295		0.987	0.393		0.927	0.361		0.881	0.335	

- **The uncertainty of the final estimate decreases with the number of images**
- **The inconsistency for k_1 and k_2 between the closed-form solution and the MLE**
  - The reason is that for the closed-form solution, camera intrinsic parameters are estimated assuming no distortion, and the predicted outer points lie closer to the image center than the detected ones.
  - The aspect ratio $a = \alpha/\beta = k_u/k_v$ is very close to 1. That is, the pixels are square.

Sensitivity with Respect to Model Imprecision:

- It is possible that there is some imprecision on the 2D model pattern if we print it on a normal printer or the pattern is not on a flat surface.

The Tsai Camera Model

- **Describe a camera as a pinhole projector combined with radial lens distortion and is completely defined by 6 intrinsic parameters and 6 extrinsic parameters.**
  - 6 intrinsic parameters:
    - 1 focal length
    - 2 image center
    - 1 aspect ratio
    - 2 lens distortion
  - 6 extrinsic parameters:
    - 3 3D rotation
    - 3 3D translation

$$\begin{pmatrix} X_c \\ Y_c \\ Z_c \end{pmatrix} = \begin{pmatrix} R_{11} & R_{21} & R_{31} \\ R_{12} & R_{22} & R_{32} \\ R_{13} & R_{23} & R_{33} \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} + \begin{pmatrix} T_x \\ T_y \\ T_z \end{pmatrix}$$

$$\begin{bmatrix} x_u \\ y_u \end{bmatrix} = \frac{f}{Z_c} \begin{bmatrix} X_c \\ Y_c \end{bmatrix}$$

$$\begin{bmatrix} x_d \\ y_d \end{bmatrix} = (1 + k_1 r^2) \begin{bmatrix} x_u \\ y_u \end{bmatrix}$$

$$r^2 = x_u^2 + y_u^2$$

$$\begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} a & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_d \\ y_d \end{bmatrix} + \begin{bmatrix} u_0 \\ v_0 \end{bmatrix}$$

Model degenerate:

If no lens distortion: $k_1=0$

a unity aspect ratio: $a=1$

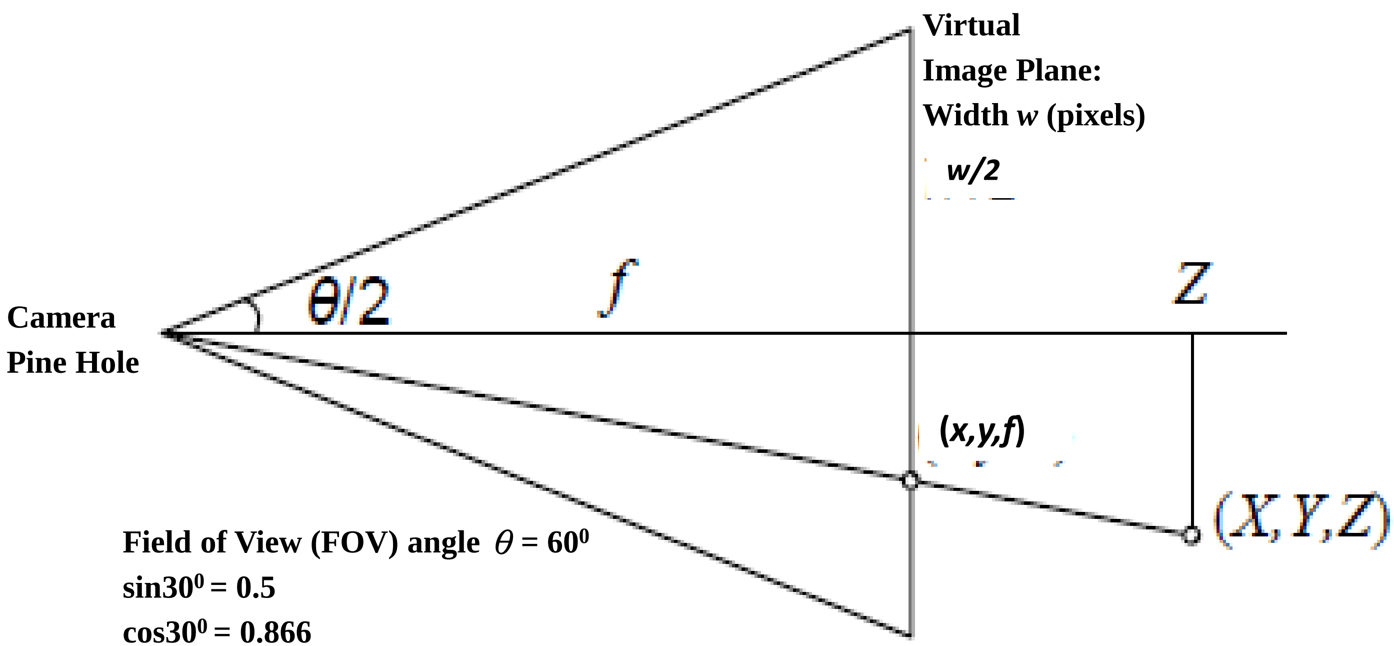
an image center: $(0,0)$

no rotation: $R=I$

no translation along the Z axis: $Tz=0$

then ….

Find Focal Length



Field of View (FOV) angle $\theta = 60^0$
$\sin 30^0 = 0.5$
$\cos 30^0 = 0.866$

References

1. R.Y. Tsai, "An Efficient and Accurate Camera Calibration Technique for 3D Machine Vision," Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Miami Beach, FL, pp. 364-374, 1986.

Best Performance

2. R.Y. Tsai, "A Versatile Camera Calibration Technique for High-Accuracy 3D Machine Vision Metrology Using Off-the-Shelf TV Cameras and Lenses," IEEE Journal of Robotics and Automation, Vol. RA-3, NO. 4, pp. 323-344, August 1987.
3. Z. Zhang, "A Flexible New Technique for Camera Calibration," IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 22, No. 11, pp. 1330-1334, 2000.

A Flexible New Technique for Camera Calibration

Technical Report

MSR-TR-98-71

Zhengyou Zhang
Microsoft Corp., One Microsoft Way, Redmond WA 98052-6399, USA

Shao-Hong Li
ilkn0214@gmail.com

Content

1. Motivation

- **Find Intrinsic Parameters and extrinsic parameters of the camera: Procedure -**

1. H: Estimate homography H using LM algorithm
2. A: Solve intrinsic parameters using close-form solution
3. R, t: Calculate extrinsic parameters by homography and intrinsic parameters
4. Deal with radial distortions

- $s\tilde{m} = A[R\ t]\tilde{M}$

2.1 Camera Model

- **Extrinsic matrix $[R\ t]$: 3D world coordinate to 3D camera coordinate**

$$M_c = \begin{bmatrix} x \\ y \\ z \end{bmatrix} = [R\ t]\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = [R\ t]\widetilde{M}$$

- **Intrinsic matrix A: 3D camera coordinate to 2D pixel coordinate**

From pic. 2.1, $\frac{f}{Z} = \frac{u}{X} = \frac{v}{Y}$

$$\Rightarrow \begin{cases} u(pixel) = \dfrac{f(inch)X(inch)}{Z(inch)} \\ v(pixel) = \dfrac{f(inch)Y(inch)}{Z(inch)} \end{cases}$$ 單位錯誤

$$\Rightarrow \begin{cases} u = \dfrac{fX}{Z} \cdot k_u(pixel/inch) \\ v = \dfrac{fY}{Z} \cdot k_v(pixel/inch) \end{cases}, k_u, k_v \text{ are scale factors}$$

Translation

$$\Rightarrow \begin{cases} u = \dfrac{fX}{Z} \cdot k_u + u_0 \\ v = \dfrac{fY}{Z} \cdot k_v + v_0 \end{cases}$$

Skew (u-axis and v-axis are not orthogonal. Pic. 2.2)

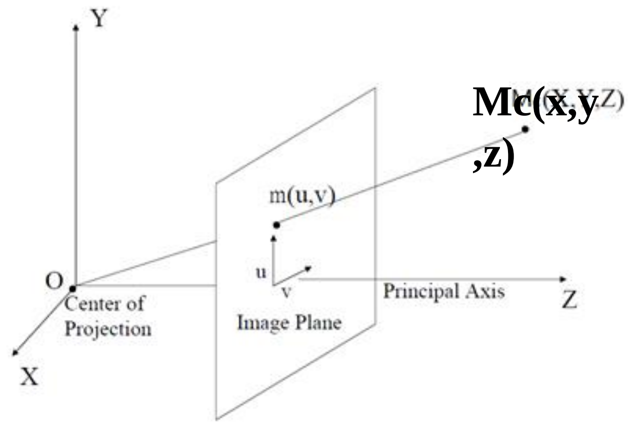
M: **3D point in world coordinate system(WCS)** $M = [X, Y, Z]^T$

M_c: **3D point in camera coordinate system** $= [x, y, z]^T$

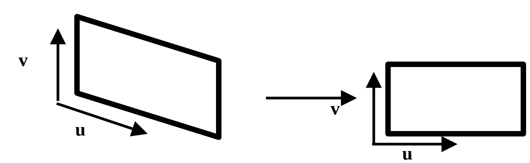
m: **2D (image) point.** $m = [u, v]^T$

$\widetilde{m}$: **Augmented vector (add an extra dimension).**

$\widetilde{m} = [u, v, 1]^T$, $\widetilde{M} = [X, Y, Z, 1]^T$



Pic. 2.1



Pic. 2.2

2.2 Homography Introduction

- **Assume: Model plane Z = 0 in world coordinate system**

$$s\tilde{m} = A[R\ t]\tilde{M}$$

$$\Rightarrow s\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = A[r_1\ r_2\ r_3\ t]\begin{bmatrix} X \\ Y \\ 0 \\ 1 \end{bmatrix} = A[r_1\ r_2\ t]\begin{bmatrix} X \\ Y \\ 1 \end{bmatrix}$$

- Homography $H = A[r_1\ r_2\ t]\left(= [h_1\ h_2\ h_3] = \begin{bmatrix} \bar{h}_1^T \\ \bar{h}_2^T \\ \bar{h}_3^T \end{bmatrix} \right)$

$$\Rightarrow s\tilde{m} = H\tilde{M}$$

$$\Rightarrow s\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} \bar{h}_1^T \\ \bar{h}_2^T \\ \bar{h}_3^T \end{bmatrix}\begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} = \begin{bmatrix} \bar{h}_{11} & \bar{h}_{12} & \bar{h}_{13} \\ \bar{h}_{21} & \bar{h}_{22} & \bar{h}_{23} \\ \bar{h}_{31} & \bar{h}_{32} & \bar{h}_{33} \end{bmatrix}\begin{bmatrix} X \\ Y \\ 1 \end{bmatrix}$$

$$\Rightarrow \begin{cases} su = \bar{h}_{11}X + \bar{h}_{12}Y + \bar{h}_{13} \\ sv = \bar{h}_{21}X + \bar{h}_{22}Y + \bar{h}_{23} \\ s = \bar{h}_{31}X + \bar{h}_{32}Y + \bar{h}_{33} \end{cases}$$

$$\Rightarrow \begin{cases} u = \dfrac{\bar{h}_{11}X + \bar{h}_{12}Y + \bar{h}_{13}}{\bar{h}_{31}X + \bar{h}_{32}Y + \bar{h}_{33}} = \dfrac{\bar{h}_1^T\tilde{M}}{\bar{h}_3^T\tilde{M}} \\ v = \dfrac{\bar{h}_{21}X + \bar{h}_{22}Y + \bar{h}_{23}}{\bar{h}_{31}X + \bar{h}_{32}Y + \bar{h}_{33}} = \dfrac{\bar{h}_2^T\tilde{M}}{\bar{h}_3^T\tilde{M}} \end{cases} \Rightarrow \hat{m}_i = \begin{bmatrix} u \\ v \end{bmatrix} = \frac{1}{\bar{h}_3^T\tilde{M}}\begin{bmatrix} \bar{h}_1^T\tilde{M} \\ \bar{h}_2^T\tilde{M} \end{bmatrix}$$

2.3 Homography Estimation (1/3)

1. Estimate Homography (H) Between the 3D Model Plane and its 2D Image:

M_i: **The 3D model points. (known)**
m_i: **The 2D image points. (known)**
$\hat{m}_i$: **Expected image points under the homography H.**

$$\hat{m}_i = \begin{bmatrix} u \\ v \end{bmatrix} = \frac{1}{\bar{h}_3^T \tilde{M}} \begin{bmatrix} \bar{h}_1^T \tilde{M} \\ \bar{h}_2^T \tilde{M} \end{bmatrix}$$

H can be calculated by $\min_H \sum_i \|m_i - \hat{m}_i\|^2$

- Assume 1: m_i is corrupted by Gaussian noise with mean 0 and covariance matrix Λ_{m_i}.

⇒**Probability Density Function (PDF) as likilhood probablity:**

? ?

$$P(\,?\,|\,?\,) = \frac{1}{(2\pi)^{N/2} \left|\Lambda_{m_i}\right|^{1/2}} exp\left(-\frac{1}{2}(m_i - \hat{m}_i)^T \Lambda_{m_i}^{-1} (m_i - \hat{m}_i)\right)$$

? ?

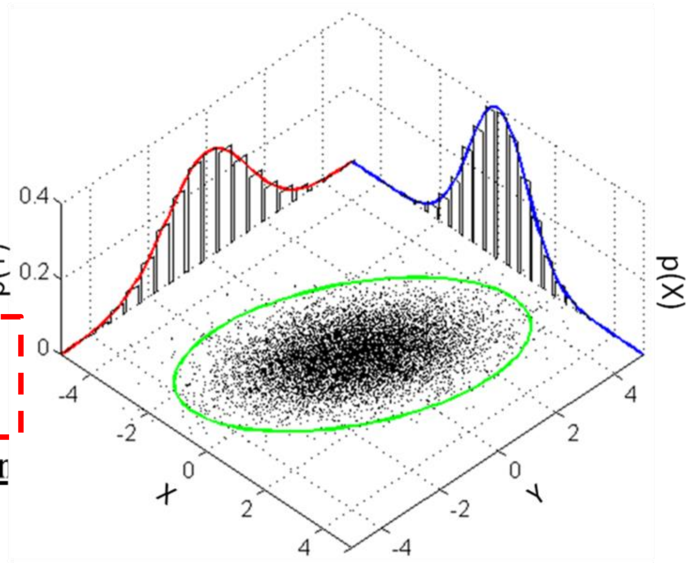
N: dimension of m_i, $N = 2$ in our case

⇒ Maximum likelihood estimation of H: $\min_H \sum_i (m_i - \hat{m}_i)^T \Lambda_{m_i}^{-1} (m_i - \hat{m}_i)$

?

- Assume 2: for all i, $\Lambda_{m_i} = \sigma^2 I$.

$$\Rightarrow \min_H \sum_i (m_i - \hat{m}_i)^T \Lambda_{m_i}^{-1} (m_i - \hat{m}_i) = \min_H \sum_i \|m_i - \hat{m}_i\|^2$$



- **We can obtain H by applying Levenberg-Marqardt algorithm**

1) An initial guess using close form solution
2) Levenberg-Marqardt algorithm

2.3 Homography Estimation (2/3)

1) **Before applying LM Algorithm to solve H, we need an initial guess:**

$$\begin{cases} u = \dfrac{\bar{h}_{11}X + \bar{h}_{12}Y + \bar{h}_{13}}{\bar{h}_{31}X + \bar{h}_{32}Y + \bar{h}_{33}} \\ v = \dfrac{\bar{h}_{21}X + \bar{h}_{22}Y + \bar{h}_{23}}{\bar{h}_{31}X + \bar{h}_{32}Y + \bar{h}_{33}} \end{cases}$$

$$\Rightarrow \begin{cases} \bar{h}_{11}X + \bar{h}_{12}Y + \bar{h}_{13} - 1u\bar{h}_{31}X - u\bar{h}_{32}Y - u\bar{h}_{33} = 0 \\ \bar{h}_{21}X + \bar{h}_{22}Y + \bar{h}_{23} - v\bar{h}_{31}X - v\bar{h}_{32}Y - v\bar{h}_{33} = 0 \end{cases}$$

$$\hat{m}_i = \begin{bmatrix} u \\ v \end{bmatrix} = \frac{1}{\bar{h}_3^T \widetilde{M}} \begin{bmatrix} \bar{h}_1^T \widetilde{M} \\ \bar{h}_2^T \widetilde{M} \end{bmatrix}$$

$$\Rightarrow \begin{bmatrix} X & Y & 1 & 0 & 0 & 0 & -uX & -uY & -u \\ 0 & 0 & 0 & X & Y & 1 & -vX & -vY & -v \end{bmatrix} \begin{bmatrix} \bar{h}_{11} \\ \bar{h}_{12} \\ \bar{h}_{13} \\ \bar{h}_{21} \\ \bar{h}_{22} \\ \bar{h}_{23} \\ \bar{h}_{31} \\ \bar{h}_{32} \\ \bar{h}_{33} \end{bmatrix} = \begin{bmatrix} \widetilde{M}^T & 0^T & -u\widetilde{M}^T \\ 0^T & \widetilde{M}^T & -v\widetilde{M}^T \end{bmatrix} \begin{bmatrix} \bar{h}_1^T \\ \bar{h}_2^T \\ \bar{h}_3^T \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

?

2.3 Homography Estimation (3/3)

2) LM Algorithm to solve H

Algorithm 3.16. Levenberg–Marquardt method

begin
  $k := 0; \quad \nu := 2; \quad \mathbf{x} := \mathbf{x}_0$
  $\mathbf{A} := \mathbf{J}(\mathbf{x})^\top \mathbf{J}(\mathbf{x}); \quad \mathbf{g} := \mathbf{J}(\mathbf{x})^\top \mathbf{f}(\mathbf{x})$
  $\mathit{found} := (\|\mathbf{g}\|_\infty \le \varepsilon_1); \quad \mu := \tau * \max\{a_{ii}\}$
  while (**not** *found*) **and** $(k < k_{\max})$
    $k := k+1; \quad$ Solve $(\mathbf{A} + \mu\mathbf{I})\mathbf{h}_{\text{lm}} = -\mathbf{g}$
    if $\|\mathbf{h}_{\text{lm}}\| \le \varepsilon_2(\|\mathbf{x}\| + \varepsilon_2)$
      $\mathit{found} :=$ **true**
    else
      $\mathbf{x}_{\text{new}} := \mathbf{x} + \mathbf{h}_{\text{lm}}$
      $\varrho := (F(\mathbf{x}) - F(\mathbf{x}_{\text{new}}))/(L(\mathbf{0}) - L(\mathbf{h}_{\text{lm}}))$
      if $\varrho > 0$ {step acceptable}
        $\mathbf{x} := \mathbf{x}_{\text{new}}$
        $\mathbf{A} := \mathbf{J}(\mathbf{x})^\top \mathbf{J}(\mathbf{x}); \quad \mathbf{g} := \mathbf{J}(\mathbf{x})^\top \mathbf{f}(\mathbf{x})$
        $\mathit{found} := (\|\mathbf{g}\|_\infty \le \varepsilon_1)$
        $\mu := \mu * \max\{\frac{1}{3}, 1 - (2\varrho - 1)^3\}; \quad \nu := 2$
      else
        $\mu := \mu * \nu; \quad \nu := 2 * \nu$
end

Reference: Methods for Non-Linear Least Squares Problems (2nd ed.) Kaj Madsen, Hans Bruun Nielsen, Ole Tingleff. Type, Lecture note. Year, 2004 pp. 60

2.4 Constraints on the Intrinsic Parameters

2. Knowing H allows us to obtain constraints on the intrinsic parameters:

$$\boxed{\begin{cases} h_1{}^T A^{-T} A^{-1} h_2 = 0 \\ h_1{}^T A^{-T} A^{-1} h_1 = h_2{}^T A^{-T} A^{-1} h_2 \end{cases}}$$

➢ $H = [h_1\ h_2\ h_3] = \lambda A[r_1\ r_2\ t]$

$$\Rightarrow \begin{cases} r_1 = \dfrac{1}{\lambda} A^{-1} h_1 \\ r_2 = \dfrac{1}{\lambda} A^{-1} h_2 \end{cases}$$

1. $\because r_1 \perp r_2$

$\therefore r_1 \cdot r_2 = r_1{}^T r_2 = (A^{-1} h_1)^T (A^{-1} h_2) = h_1{}^T A^{-T} A^{-1} h_2 = 0$

2. $\because \|\mathrm{r}_1\| = \|\mathrm{r}_2\| = 1$

$\therefore r_1{}^T r_1 = r_2{}^T r_2$

$\Rightarrow (A^{-1} h_1)^T (A^{-1} h_1) = (A^{-1} h_2)^T (A^{-1} h_2)$

$\Rightarrow h_1{}^T A^{-T} A^{-1} h_1 = h_2{}^T A^{-T} A^{-1} h_2$

2.5 Absolute Conic

- **Assume: Model plane Z = 0 in world coordinate system(WCS)**
- **Points on model plane under homogeneous coordinate system:**

$$\begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix} \equiv [r_1\ r_2\ r_3\ t] \begin{bmatrix} X \\ Y \\ 0 \\ 1 \end{bmatrix}$$

- **Circle on the model plane with central $(x_0, y_0, 0,1)^T$ and radius r:**

$$\begin{cases} (x - x_0 w)^2 + (y - y_{0w})^2 = r^2 w^2 \\ z = 0 \end{cases}$$

- **The circle intersects infinity plane at point** $P = \begin{bmatrix} w_p \\ y_P \\ 0 \\ 0 \end{bmatrix}$**: (happens when r is infinity)**

$$\begin{cases} \left(x_p - w x_0\right)^2 + (y_P - w y_0)^2 = w^2 r^2 \\ z_P = 0 \\ w_P = 0 \end{cases}$$

$$\Rightarrow {x_P}^2 + {y_P}^2 = 0$$

$$\Rightarrow P^T P = 0$$

- **Let m be the image of point P**

$$\Rightarrow m = ARP$$

$$\Rightarrow P = R^{-1} A^{-1} m$$

$$\because P^T P = 0$$

2.6 Geometric Interpretation

- ❑ **Assume: Model plane Z = 0 in world coordinate system(WCS)**
- ❑ **Model plane in camera coordinate:**

$$s\begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix} = \begin{bmatrix} r_{11} & r_{21} & r_{31} & t_1 \\ r_{12} & r_{22} & r_{32} & t_2 \\ r_{13} & r_{23} & r_{33} & t_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} X \\ Y \\ 0 \\ 1 \end{bmatrix}$$

Homogeneous coordinate: $\begin{cases} points\ at\ infinity : w = 0 \\ \qquad else \qquad : w = 1 \end{cases}$

$$\Rightarrow \begin{cases} x = (r_{11}X + r_{21}Y + t_1)/w \\ y = (r_{12}X + r_{22}Y + t_2)/w \\ z = (r_{13}X + r_{23}Y + t_3)/w \\ w = 0\ or\ 1 \end{cases}$$

$$\Rightarrow \begin{bmatrix} r_3 \\ -r_3^T t \end{bmatrix}^T \begin{bmatrix} x \\ y \\ z \\ w \end{bmatrix} = 0$$

❑ **Model plane in camera coordinate intersects the infinity plane ($w = 0$) at a line.**

Knowing that $\begin{bmatrix} r_1 \\ 0 \end{bmatrix}$ **and** $\begin{bmatrix} r_2 \\ 0 \end{bmatrix}$ **are two points on the line ($\because r_1 \perp r_3$ and $r_2 \perp r_3$), the line can be described as:**

$$x_\infty = a\begin{bmatrix} r_1 \\ 0 \end{bmatrix} + \mathrm{b}\begin{bmatrix} r_2 \\ 0 \end{bmatrix} = \begin{bmatrix} ar_1 + br_2 \\ 0 \end{bmatrix}$$

❑ **By the definition of absolute conic,** $x_\infty^T x_\infty = 0$

$$\Rightarrow a^2 + b^2 = 0$$

$$\Rightarrow b = \pm ai$$

$$\Rightarrow x_\infty = a\begin{bmatrix} r_1 \pm ir_2 \\ 0 \end{bmatrix}$$

3.1 Closed Form Solution (1/2)

- $B = A^{-T}A^{-1} = \begin{bmatrix} \frac{1}{\alpha^2} & -\frac{\gamma}{\alpha^2\beta} & \frac{v_0\gamma-u_0\beta}{\alpha^2\beta} \\ -\frac{\gamma}{\alpha^2\beta} & \frac{\gamma^2}{\alpha^2\beta^2}+\frac{1}{\beta^2} & -\frac{\gamma(v_0\gamma-u_0\beta)}{\alpha^2\beta^2}-\frac{v_0}{\beta^2} \\ \frac{v_0\gamma-u_0\beta}{\alpha^2\beta} & -\frac{\gamma(v_0\gamma-u_0\beta)}{\alpha^2\beta^2}-\frac{v_0}{\beta^2} & \frac{(v_0\gamma-u_0\beta)^2}{\alpha^2\beta^2}+\frac{{v_0}^2}{\beta^2}+1 \end{bmatrix}$
  - Define: $b = [B_{11}B_{12}B_{22}B_{13}B_{23}B_{33}]^T$

    $v_{ij} = [h_{i1}h_{j1}, h_{i1}h_{j2} + h_{i2}h_{j1}, h_{i2}h_{j2}, h_{i3}h_{j1} + h_{i1}h_{j3}, h_{i3}h_{j2} + h_{i2}h_{j3}, h_{i3}h_{j3}]^T$

    $\Rightarrow h_i^T B h_j = v_{ij}^T b$
  - $\begin{cases} {h_1}^T A^{-T} A^{-1} h_2 = 0 \\ {h_1}^T A^{-T} A^{-1} h_1 = {h_2}^T A^{-T} A^{-1} h_2 \end{cases}$ can be rewritten as $\begin{bmatrix} v_{12}^T \\ (v_{11} - v_{22})^T \end{bmatrix}$ $\mathrm{b} = 0$
- **6 variables to be solved, each image stacks 2 equations. (*n* images of model plane are obsearved)**
  - $n \geq 3$: get a unique solution b.
  - $n = 2$: impose skewless constraint $\gamma = 0$??still missing one equation
  - $n = 1$: assume u_0, v_0 are known and $\gamma = 0$

3.1 Closed Form Solution (2/2)

- **After b is estimated, intrinsic parameter A can be computed by $B = \lambda A^{-T} A^{-1}$:**

$$v_0 = \frac{(B_{12}B_{13} - B_{11}B_{23})}{(B_{11}B_{22} - B_{12}^2)}$$

$$\lambda = B_{33} - \frac{[B_{13}^2 + v_o(B_{12}B_{13} - B_{11}B_{23})]}{B_{11}}$$

$$\alpha = \sqrt{\lambda / B_{11}}$$

$$\beta = \sqrt{\lambda B_{11} / (B_{11}B_{22} - B_{12}^2)}$$

$$\gamma = -\frac{B_{12}\alpha^2\beta}{\lambda}$$

$$u_0 = \frac{\gamma v_0}{\beta} - B_{13}\alpha^2/\lambda$$

- **After A is computed, r_1, r_2, r_3, t can be computed by $[h_1 h_2 h_3] = A[r_1 r_2 t]$**

$$r_1 = \sigma A^{-1} h_1$$

$$r_2 = \sigma A^{-1} h_2$$

$$r_3 = r_1 \times r_2$$

$$t = \sigma A^{-1} h_3$$

Where $\sigma = \frac{1}{\|A^{-1}h_1\|} = \frac{1}{\|A^{-1}h_2\|}$

- **Better results can be obtained by the method described in Appendix C**

3.3 Dealing with Radial Distortion

- **Distortion-free pixel image coordinate**(u, v)
- **Observed pixel image coordinate**$(\breve{u}, \breve{v})$
- **Distortion-free normalized image coordinate**(x, y)
- **Observed normalized image coordinate** $(\breve{x}, \breve{y})$
- **Assume the center of radial distortion remain the same as the principle point:**

$$\breve{x} = x + x[k_1(x^2 + y^2) + k_2(x^2 + y^2)^2]$$
$$\breve{y} = y + y[k_1(x^2 + y^2) + k_2(x^2 + y^2)^2]$$

- **From** $\breve{u} = u_0 + \alpha\breve{x} + \gamma\breve{y}$ **,** $\breve{v} = v_0 + \beta\breve{y}$ **and assuming** $\gamma = 0$

$$\breve{u} = u + (u - u_0)[k_1(x^2 + y^2) + k_2(x^2 + y^2)^2]$$
$$\breve{v} = v + (v - v_0)[k_1(x^2 + y^2) + k_2(x^2 + y^2)^2]$$
$$\Rightarrow \begin{bmatrix} (u - u_0)(x^2 + y^2) & (u - u_0)(x^2 + y^2)^2 \\ (v - v_0)(x^2 + y^2) & (v - v_0)(x^2 + y^2)^2 \end{bmatrix} \begin{bmatrix} k_1 \\ k_2 \end{bmatrix} = \begin{bmatrix} \breve{u} - u \\ \breve{v} - v \end{bmatrix}$$

- **Given *m* points in *n* images, we stack the equations as** $Dk = d$

  Least square solution: $k = (D^T D)^{-1} D^T d$

4. Experiment Results

- **Computer Simulations:**
  - $\alpha = 1250$
  - $\beta = 900$
  - $\gamma = 1.09083(89.95°)$
  - $u_0 = 255$
  - $v_0 = 255$
  - Image Resolution 512 X 512
  - Model Plane: Checker pattern containing 10 X 14 corner points, 18cm X 25cm

4. Experiment Results

- **Performance w.r.t. the noise level**

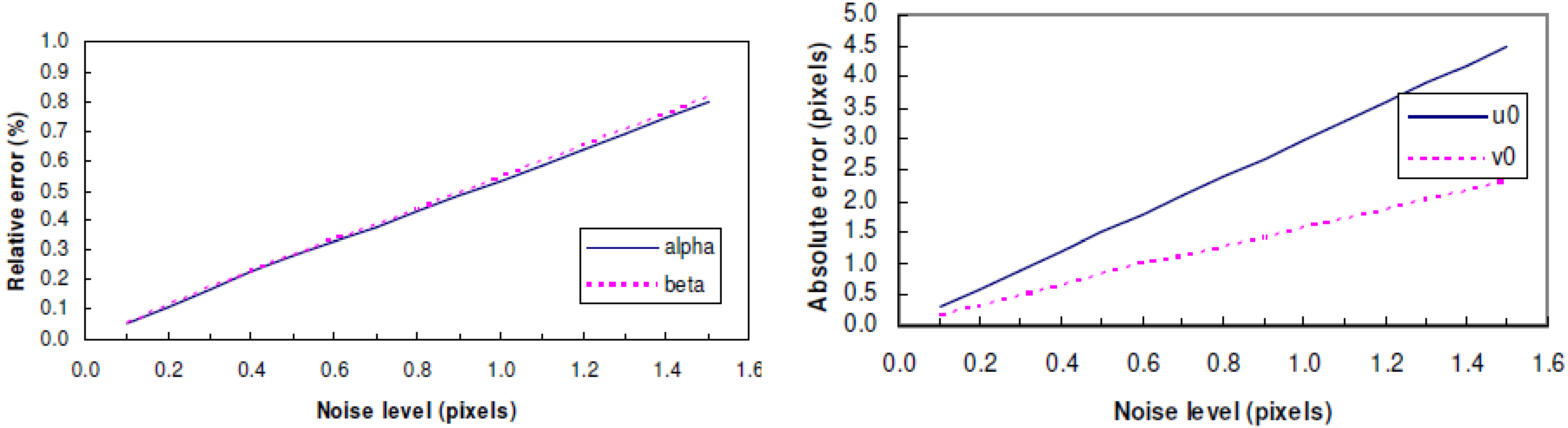


Figure 1: Errors vs. the noise level of the image points

4. Experiment Results

- Performance w.r.t. the number of planes.

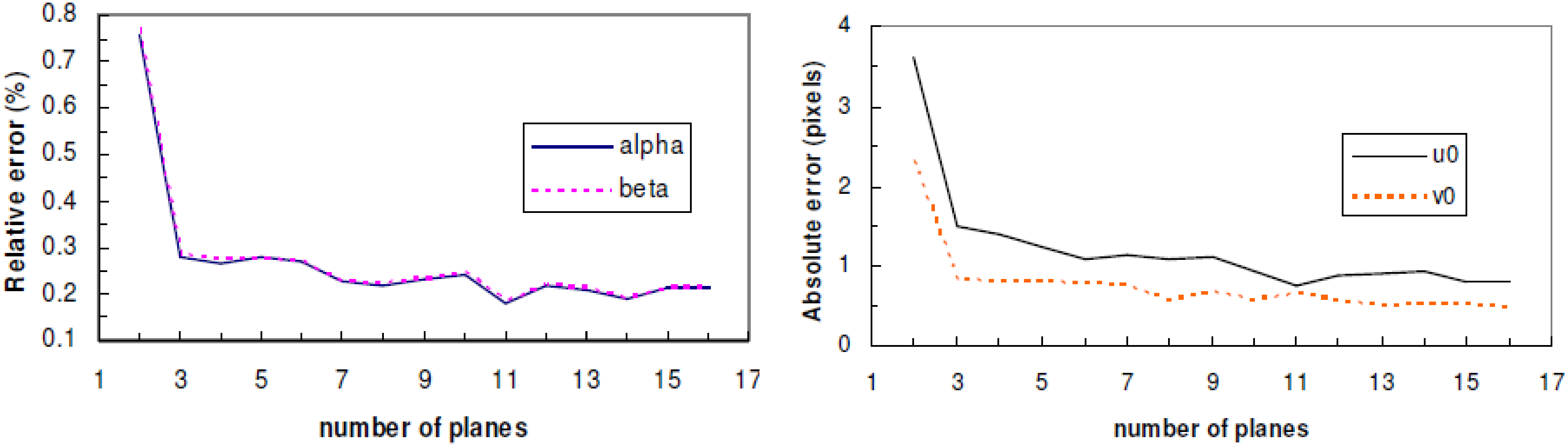


Figure 2: Errors vs. the number of images of the model plane

4. Experiment Results

- Performance w.r.t. the orientation of the model plane.

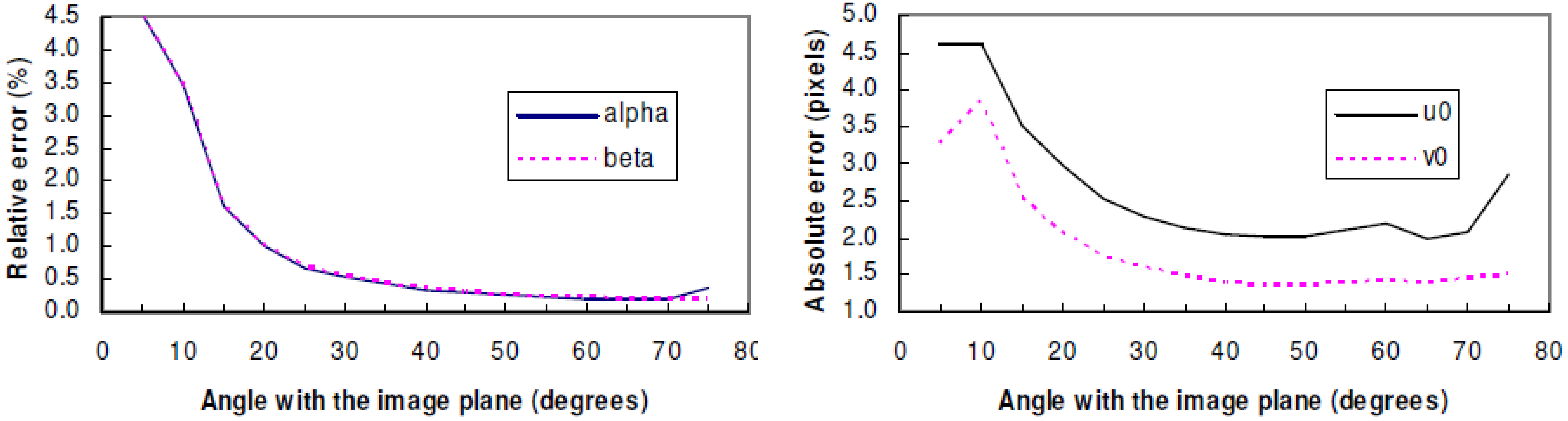


Figure 3: Errors vs. the angle of the model plane w.r.t. the image plane

4. Experiment Results

- **Real Data**

Table 1: Results with real data of 2 through 5 images

nb	2 images			3 images			4 images			5 images		
	initial	final	σ	initial	final	σ	initial	final	σ	initial	final	σ
α	825.59	830.47	4.74	917.65	830.80	2.06	876.62	831.81	1.56	877.16	832.50	1.41
β	825.26	830.24	4.85	920.53	830.69	2.10	876.22	831.82	1.55	876.80	832.53	1.38
γ	0	0	0	2.2956	0.1676	0.109	0.0658	0.2867	0.095	0.1752	0.2045	0.078
u_0	295.79	307.03	1.37	277.09	305.77	1.45	301.31	304.53	0.86	301.04	303.96	0.71
v_0	217.69	206.55	0.93	223.36	206.42	1.00	220.06	206.79	0.78	220.41	206.59	0.66
k_1	0.161	−0.227	0.006	0.128	−0.229	0.006	0.145	−0.229	0.005	0.136	−0.228	0.003
k_2	−1.955	0.194	0.032	−1.986	0.196	0.034	−2.089	0.195	0.028	−2.042	0.190	0.025
RMS	0.761	0.295		0.987	0.393		0.927	0.361		0.881	0.335	

4. Experiment Results

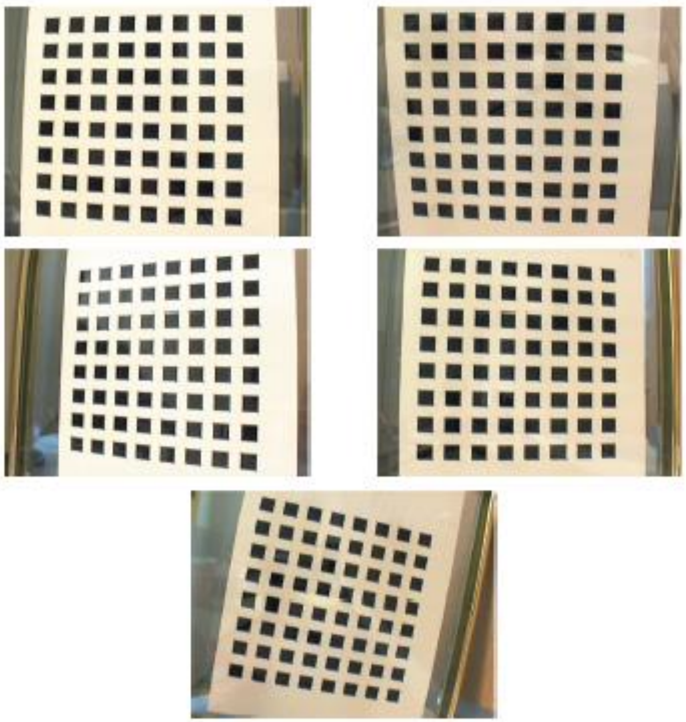


Figure 4: Five images of a model plane, together with the extracted corners (indicated by cross)

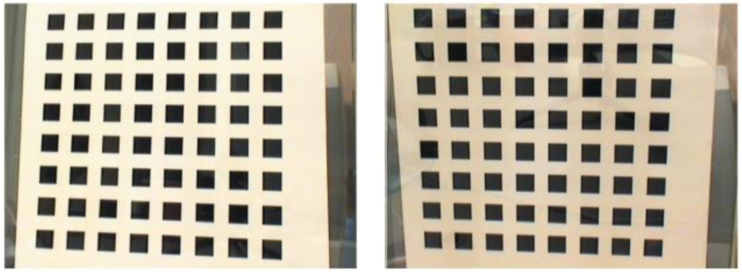


Figure 5: First and second images after having corrected radial distortion

4. Experiment Results

- **Variation of the calibration result**

Table 2: Variation of the calibration results among all quadruples of images

quadruple	(1234)	(1235)	(1245)	(1345)	(2345)	mean	deviation
α	831.81	832.09	837.53	829.69	833.14	832.85	2.90
β	831.82	832.10	837.53	829.91	833.11	832.90	2.84
γ	0.2867	0.1069	0.0611	0.1363	0.1096	0.1401	0.086
u_0	304.53	304.32	304.57	303.95	303.53	304.18	0.44
v_0	206.79	206.23	207.30	207.16	206.33	206.76	0.48
k_1	-0.229	-0.228	-0.230	-0.227	-0.229	-0.229	0.001
k_2	0.195	0.191	0.193	0.179	0.190	0.190	0.006
RMS	0.361	0.357	0.262	0.358	0.334	0.334	0.04

4. Experiment Results

- **Application to Image Based Modeling**

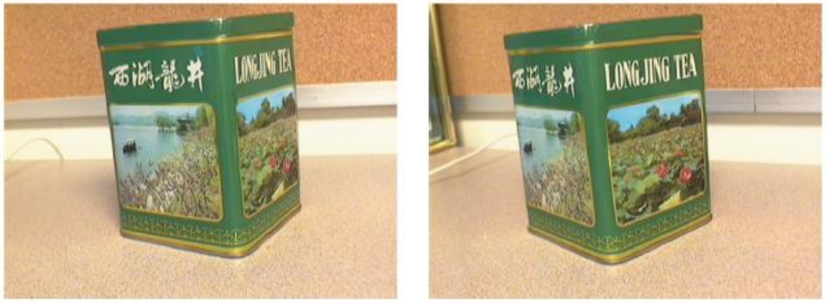


Figure 6: Two images of a tea tin

Figure 7: Three rendered views of the reconstructed tea tin

4. Experiment Results

❑ Sensitivity with Respect to Model Imprecision

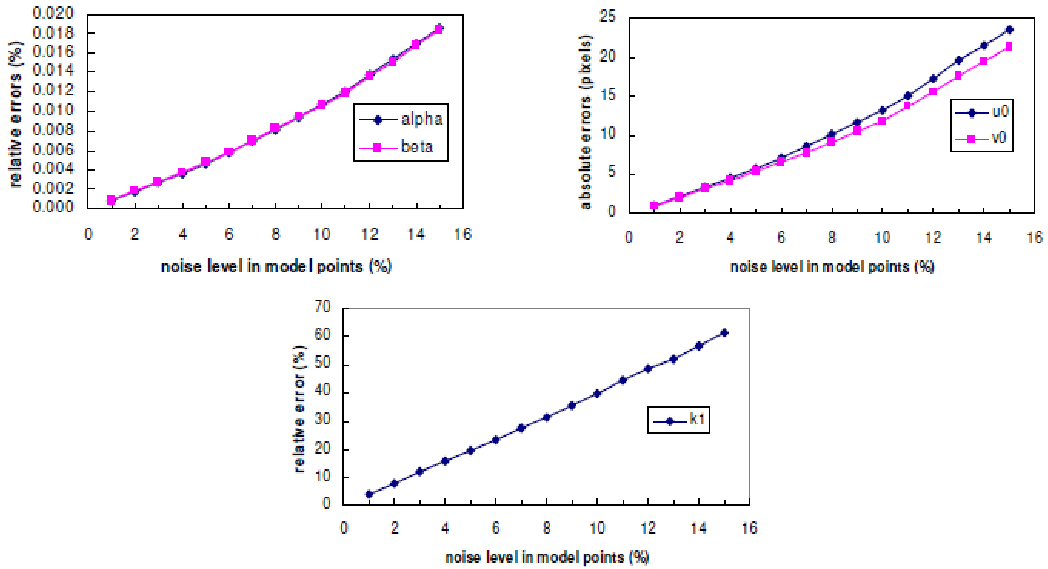


Figure 8: Sensitivity of camera calibration with respect to Gaussian noise in the model points

❑ Systematic non-planarity of the model pattern

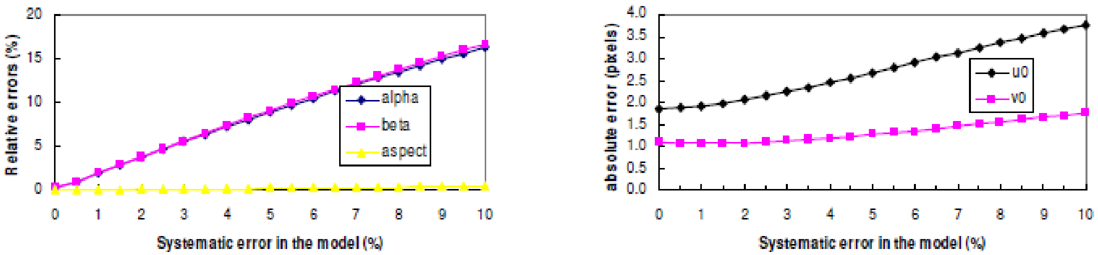


Figure 9: Sensitivity of camera calibration with respect to systematic spherical non-planarity

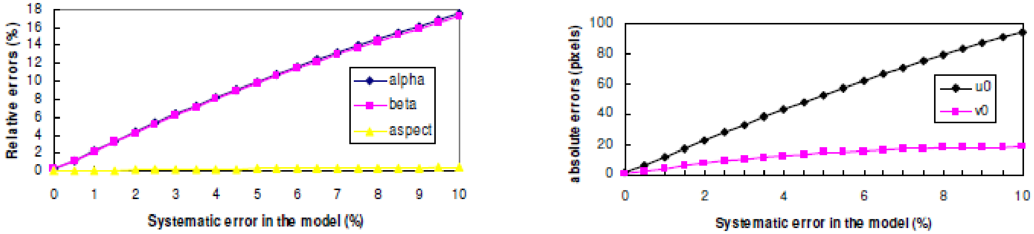


Figure 10: Sensitivity of camera calibration with respect to systematic cylindrical non-planarity