Master Mathematical Analysis and Applications
Course M1 - S2

Computer vision

Image formation and Camera Calibration

Week 9-10

Mohammed Hachama
hachamam@gmail.com
http://sites.google.com/site/hachamam/

University of Khemis Miliana

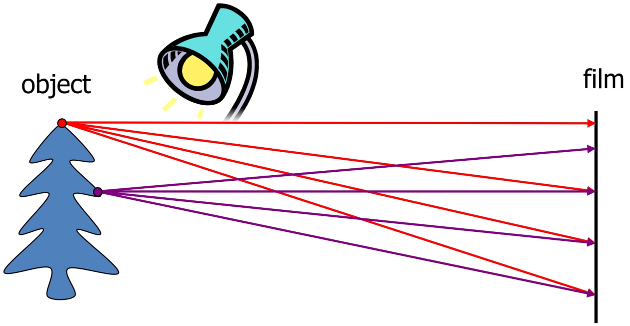
-March 2020-

Plan

Image formation

Image Formation

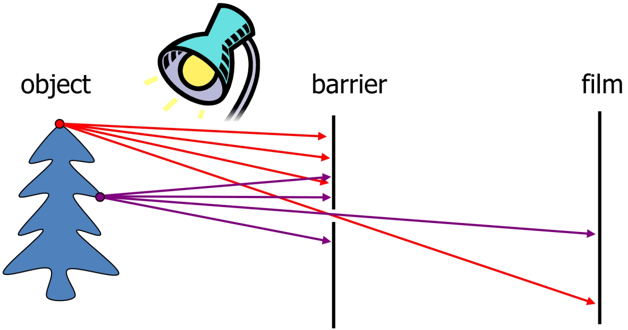
- How to form an image



- Place a piece of film in front of an object → Do we get a reasonable image ?

Image Formation

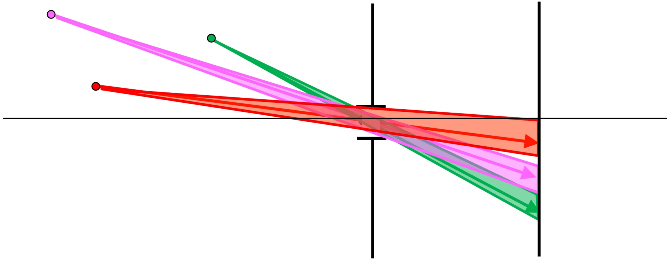
- How to form an image



- Add a barrier to block off most of the rays
  - This reduces blurring
  - The opening is known as the aperture

Image Formation

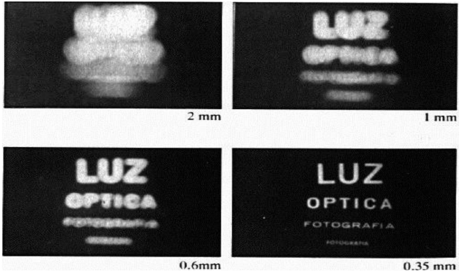
- Effects of the Aperture Size



- In an ideal pinhole, only one ray of light reaches each point on the film the image can be very dim
- Making aperture bigger makes the image blurry

Image Formation

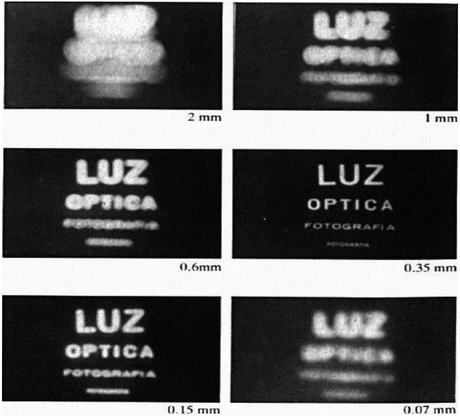
- Effects of the Aperture Size



- In an ideal pinhole, only one ray of light reaches each point on the film the image can be very dim
- Making aperture bigger makes the image blurry

Image Formation

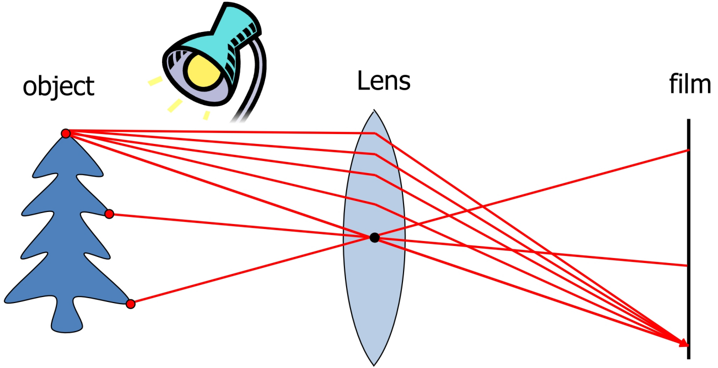
- Effects of the Aperture Size



- Why not make the aperture as small as possible ?
    - Less light gets through (must increase the exposure)
    - Diffraction effects...

Image Formation

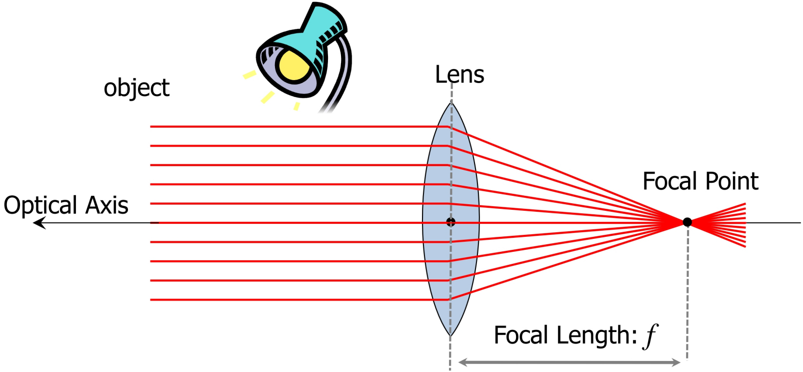
- Image formation using a converging lens



- A lens focuses light onto the film
- Rays passing through the Optical Center are not deviated

Image Formation

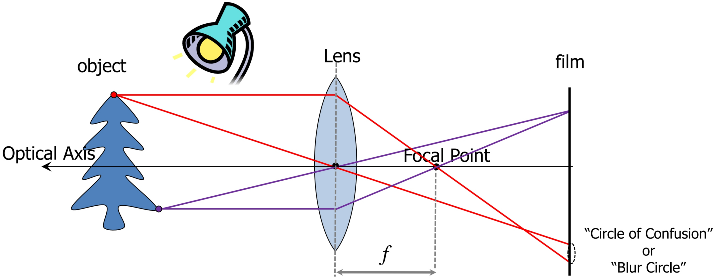
- Image formation using a converging lens



- All rays parallel to the Optical Axis converge at the Focal Point.

Image Formation

- In focus



- There is a specific distance from the lens, at which world points are "in focus" in the image
- Other points project to a "blur circle" in the image

Image Formation

- Projective Geometry
- What is preserved : Straight lines are still straight

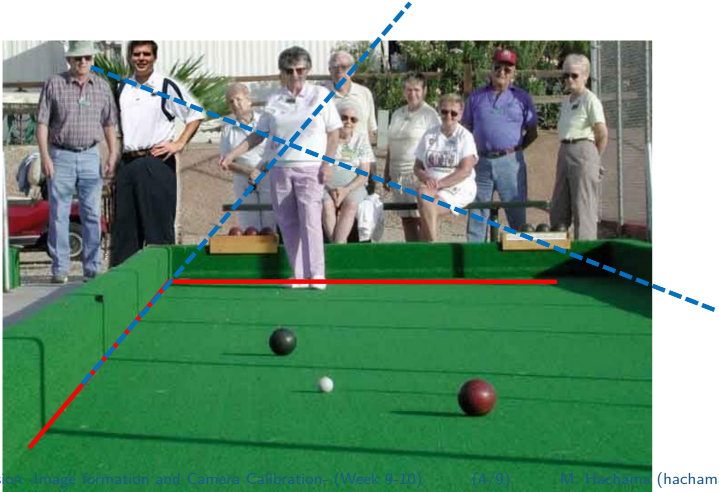


Image Formation

- Projective Geometry
- What is lost ? : Length, Angles

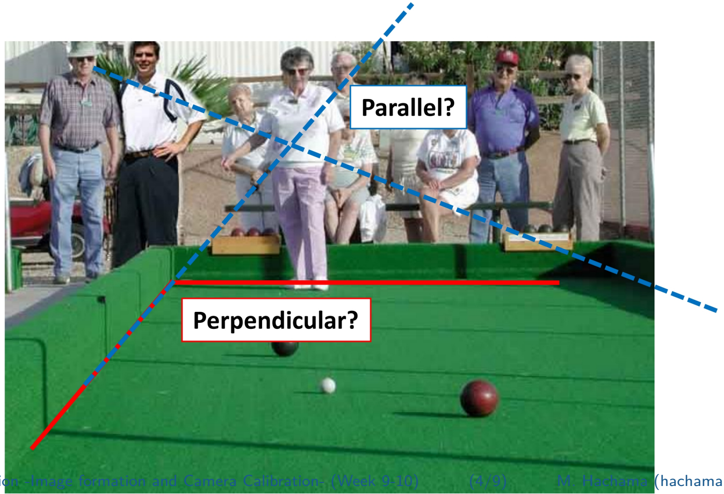


Image Formation

- Vanishing points and lines
- Parallel lines in the world intersect in the image at a “vanishing point”

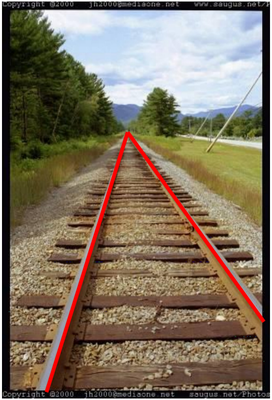


Image Formation

- Vanishing points and lines
- Parallel lines in the world intersect in the image at a "vanishing point"

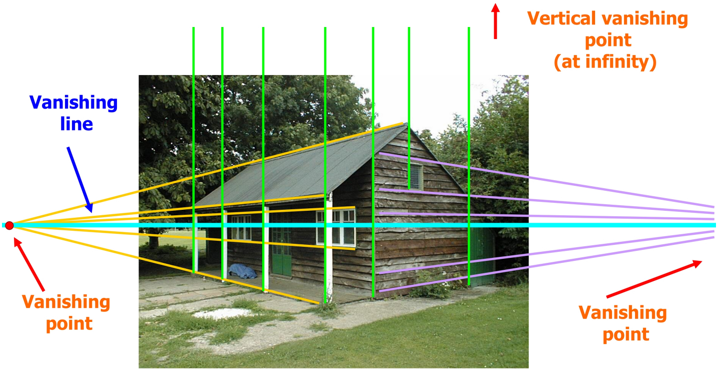
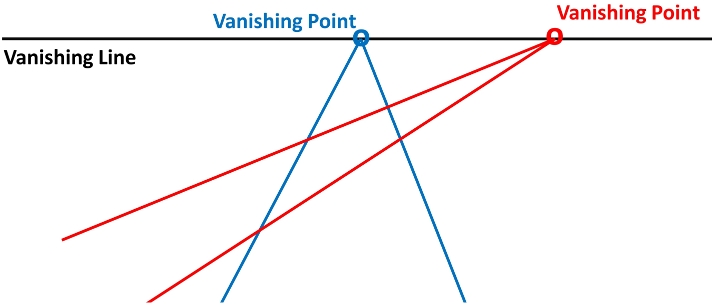


Image Formation

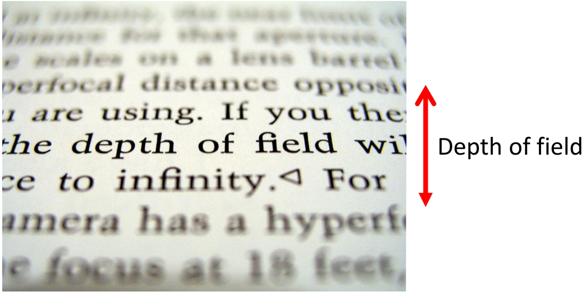
- Vanishing points and lines
- Parallel planes in the world intersect in the image at a "vanishing line"



Camera parameters

Focus and depth of field

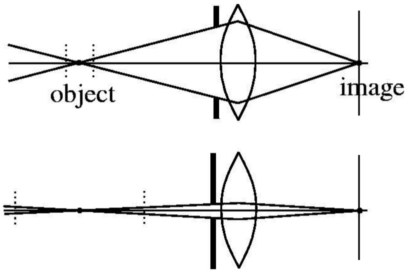
- Depth of field (DOF) : the distance between the nearest and farthest objects in a scene that appear acceptably sharp.
- Although a lens can precisely focus at only one distance at a time, the decrease in sharpness is gradual on each side of the focused distance.



Camera parameters

Focus and depth of field

- How does the aperture affect the depth of field ?

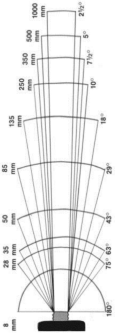


- A smaller aperture increases the depth of field but reduces the amount of light into the camera

Camera parameters

Field of view

- Angular measure of portion of 3d space seen by the camera
- Smaller FOV = larger Focal Length



28 mm lens, 65.5° × 46.4°

50 mm lens, 39.6° × 27.0°

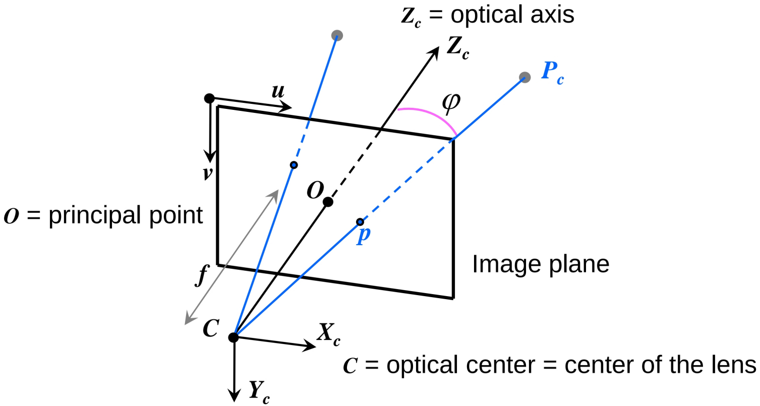
70 mm lens, 28.9° × 19.5°

210 mm lens, 9.8° × 6.5°

From World to Pixel coordinates

Perspective camera model

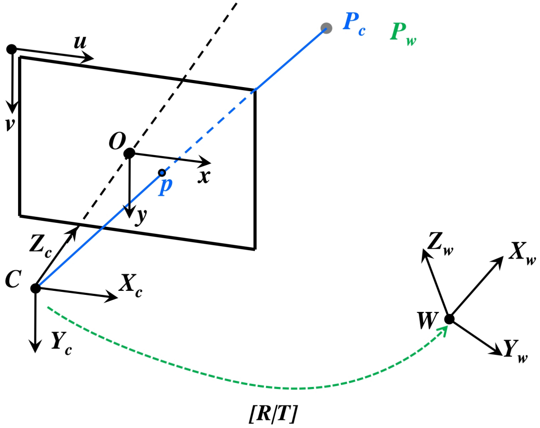
- Camera and world coordinates



From World to Pixel coordinates

Perspective camera model

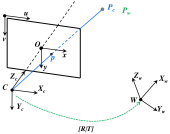
- Camera and world coordinates



From World to Pixel coordinates

Perspective camera model

- From the Camera frame to the image plane



- 1. Convert world point P_w to camera point P_c (**Extrinsic** params. $[R|T]$)
- 2. Convert P_c to image-plane coordinates (x, y) (**Intrinsic** params. f, ...)
- 3. Convert P_c to (discretised) pixel coordinates (u, v)

From World to Pixel coordinates

Perspective camera model

- 1. Convert world point P_w to camera point P_c

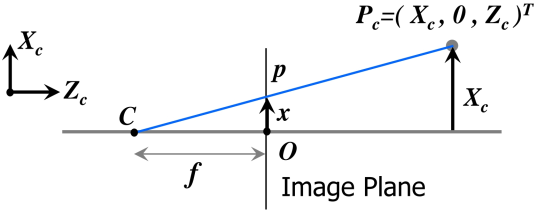
$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = \underbrace{\begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}}_{R} \begin{bmatrix} X_w \\ Y_w \\ Z_w \end{bmatrix} + \underbrace{\begin{bmatrix} t_1 \\ t_2 \\ t_3 \end{bmatrix}}_{T}$$

$$\begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & |t_1 \\ r_{21} & r_{22} & r_{23} & |t_2 \\ r_{31} & r_{32} & r_{33} & |t_3 \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} = [R|T] \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix}$$

From World to Pixel coordinates

Perspective camera model

- 2. Convert P_c to image-plane coordinates (x, y)

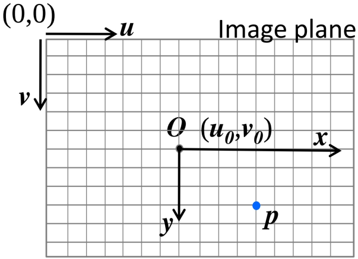


- $P_c = (X_c, 0, Z_c)^T$ projects to $p = (x, y)$ onto the image plane
- From similar triangles : $\frac{x}{f} = \frac{X_c}{Z_c} \implies x = \frac{fX_c}{Z_c}$
- In the general case : $\frac{y}{f} = \frac{Y_c}{Z_c} \implies y = \frac{fY_c}{Z_c}$

From World to Pixel coordinates

Perspective camera model

- 3. Convert P_c to (discretised) pixel coordinates (u, v)



- The pixel coords of the camera optical center : $O = (u_0, v_0)$
- Scale factors k_u, k_v for the pixel-size in both dimensions

$$u = u_0 + k_u x \implies u = u_0 + \frac{k_u f X_c}{Z_C}, \quad v = v_0 + \frac{k_v f Y_c}{Z_C}$$

From World to Pixel coordinates

Perspective camera model

- Homogeneous coordinates :

$$\begin{bmatrix} \lambda u \\ \lambda v \\ \lambda \end{bmatrix} = \begin{bmatrix} k_u f & 0 & u_0 \\ 0 & k_v f & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix} = \underbrace{\begin{bmatrix} \alpha_u & 0 & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}}_{K} \begin{bmatrix} X_c \\ Y_c \\ Z_c \end{bmatrix}$$

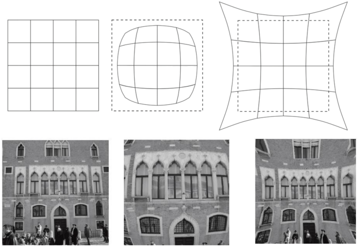
- α_u and α_v are the Focal length in pixels and K is the "Calibration matrix" or "Matrix of Intrinsic Parameters".

$$\begin{bmatrix} \lambda u \\ \lambda v \\ \lambda \end{bmatrix} = K[R|T] \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix}$$

From World to Pixel coordinates

Perspective camera model

- **Lens distortion** : from the undistorted coordinates (u, v) to the observable distorted coordinates (u_d, v_d)



No distortion Barrel distortion Pincushion

- Simple quadratic model (works for most lenses) :

$$\begin{bmatrix} u_d \\ v_d \end{bmatrix} = (1 + k_1 r^2) \begin{bmatrix} u - u_0 \\ v - v_0 \end{bmatrix} + \begin{bmatrix} u_0 \\ v_0 \end{bmatrix}$$

where $r^2 = (u - u_0)^2 + (v - v_0)^2$.

Camera calibration

Direct Linear Transform (DLT) From 3D objects

- Compute K, R, and T that satisfy the perspective projection equation [1]

$$\tilde{p} = \begin{bmatrix} \tilde{u} \\ \tilde{v} \\ \tilde{w} \end{bmatrix} = \lambda \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = K[R|T] \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} = M \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix}$$

$$\Longrightarrow \begin{bmatrix} \tilde{u} \\ \tilde{v} \\ \tilde{w} \end{bmatrix} = \begin{bmatrix} m_1^T \\ m_2^T \\ m_3^T \end{bmatrix} \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix} = M \begin{bmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{bmatrix}$$

where m_i^T is the i-th row of M.

1. we suppose that radial distortions are already corrected

Direct Linear Transform (DLT) From 3D objects

- For n points, we can stack all these equations into a big matrix : $Q\mathbf{M} = 0$

$$\begin{pmatrix} X_w^1 & Y_w^1 & Z_w^1 & 1 & 0 & 0 & 0 & 0 & -u_1X_w^1 & -u_1Y_w^1 & -u_1Z_w^1 & -u_1 \\ 0 & 0 & 0 & 0 & X_w^1 & Y_w^1 & Z_w^1 & 1 & -v_1X_w^1 & -v_1Y_w^1 & -v_1Z_w^1 & -v_1 \\ & & & & & & \cdots & \cdots & \cdots & & & \\ X_w^n & Y_w^n & Z_w^n & 1 & 0 & 0 & 0 & 0 & -u_nX_w^n & -u_nY_w^n & -u_nZ_w^n & -u_n \\ 0 & 0 & 0 & 0 & X_w^n & Y_w^n & Z_w^n & 1 & -v_nX_w^n & -v_nY_w^n & -v_nZ_w^n & -v_n \end{pmatrix} \begin{pmatrix} m_{11} \\ m_{12} \\ m_{13} \\ m_{14} \\ m_{21} \\ m_{22} \\ m_{23} \\ m_{24} \\ m_{31} \\ m_{32} \\ m_{33} \\ m_{34} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{pmatrix}$$

Direct Linear Transform (DLT) From 3D objects

- For n points, we can stack all these equations into a big matrix : $Q\mathbf{M} = 0$
- Minimal solution
  - $Q_{(2n \times 12)}$ should have rank 11 to have a unique (up to a scale) non-trivial solution M
  - Each 3D-to-2D point correspondence provides 2 independent equations
  - Thus, $5+$ 2 point correspondences are needed (in practice **6 point correspondences** !)

Direct Linear Transform (DLT) From 3D objects

- For n points, we can stack all these equations into a big matrix : $Q\mathbf{M} = 0$
- Over-determined solution
  - $n \geq 6$ points
  - A solution is to minimize $\|QM\|^2$ subject to the constraint $\|M\|^2 = 1$. $\Longrightarrow$ Singular Value Decomposition (SVD) : The solution is the eigenvector corresponding to the smallest eigenvalue of the matrix $Q^T Q$.

Direct Linear Transform (DLT) From 3D objects

- For n points, we can stack all these equations into a big matrix : $Q\mathbf{M} = 0$

- Once we have the M matrix, we can recover the intrinsic and extrinsic parameters

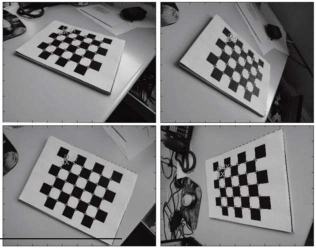
$$M = K(R|T)$$

- We enforce the constraint that R is orthogonal ($R^T R = I$), we can use the QR factorization

Direct Linear Transform (DLT) From planar grids

Direct Linear Transform (DLT) From planar grids [2]

- Compute K, R, and T that satisfy the perspective projection equation [1]
- Use of a planar grid (e.g., a chessboard) and a few images of this shown at different orientations
- We can find relations between points without knowing their $3D$ coordinates.



1. we suppose that radial distortions are already corrected
2. Zhang (1999)

Direct Linear Transform (DLT) From planar grids

- Since the points lie on a plane, we have $Z_w = 0$

$$\begin{bmatrix} \lambda u \\ \lambda v \\ \lambda \end{bmatrix} = \begin{bmatrix} \alpha_u & 0 & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \left[\begin{array}{ccc|c} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{array}\right] \begin{bmatrix} X_w \\ Y_w \\ 0 \\ 1 \end{bmatrix}$$

$$\Longrightarrow \begin{bmatrix} \lambda u \\ \lambda v \\ \lambda \end{bmatrix} = \underbrace{\begin{bmatrix} \alpha_u & 0 & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \left[\begin{array}{cc|c} r_{11} & r_{12} & t_1 \\ r_{21} & r_{22} & t_2 \\ r_{31} & r_{32} & t_3 \end{array}\right]}_{H} \begin{bmatrix} X_w \\ Y_w \\ 1 \end{bmatrix}$$

H is called Homography.

Direct Linear Transform (DLT) From planar grids

- For n points

$$\begin{bmatrix} X_w^1 & Y_w^1 & 1 & 0 & 0 & 0 & -u_1 X_w^1 & -u_1 Y_w^1 & -u_1 \\ 0 & 0 & 0 & X_w^1 & Y_w^1 & 1 & -v_1 X_w^1 & -v_1 Y_w^1 & -v_1 \\ & \dots & & & \dots & & & \dots & \\ X_w^n & Y_w^n & 1 & 0 & 0 & 0 & -u_n X_w^n & -u_n Y_w^n & -u_n \\ 0 & 0 & 0 & X_w^n & Y_w^n & 1 & -v_n X_w^n & -v_n Y_w^n & -v_n \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \\ h_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}$$

$$\Longrightarrow Q.\mathbf{H} = 0$$

Direct Linear Transform (DLT) From planar grids

- Minimal solution
  - $Q_{2n\times 9}$ should have rank 8 to have a unique (up to a scale) non-trivial solution H
  - Each point correspondence provides 2 independent equations
  - Thus, a minimum of 4 non-collinear points is required
- Over-determined solution : $n \geq 4$ points
  - Singular Value Decomposition (SVD)
- Solving for K, R and T : H decomposition

$$\begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} = \begin{bmatrix} \alpha_u & 0 & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & t_1 \\ r_{21} & r_{22} & t_2 \\ r_{31} & r_{32} & t_3 \end{bmatrix}$$

Direct Linear Transform (DLT) From planar grids

- Plane-based self calibration : Zhang's method
  - 1. Images $I_1, ...I_M$ are taken under different views.
  - 2. From each i, estimate the associated homography H_i. Thus, the intrinsic and extrinsic parameters are computed.
  - 3. Refining all parameters (Global optimization)

$$E = \sum_{i=1}^{M} \sum_{j=1}^{N} (u^{i,j} - T(P_W^{i,j}, R^i, T^i, \alpha_u, \alpha_v, u_0, v_0))^2$$