Lab 6 Pre-Lab: Basic Computer Vision

1.104 Team
Department of Civil & Environmental Engineering
Massachusetts Institute of Technology

1 Introduction

In this prelab, we will explore the theoretical foundations of camera models and calibration. Understanding these concepts is crucial for many computer vision applications. Camera calibration is a prerequisite step for tasks such as 3D reconstruction, augmented reality, and computer vision-based measurement systems. By the end of this prelab, you should have a good understanding of the pinhole camera model, camera parameters, and the mathematics behind camera calibration. **There are no questions in this prelab and you do not need to submit anything.**

2 Homogeneous Coordinates in Computer Vision

Before diving into camera models, it's important to understand homogeneous coordinates, which are fundamental to projective geometry and computer vision.

2.1 Why Homogeneous Coordinates?

In computer vision, we often need to represent geometric transformations such as translations, rotations, scaling, and perspective projections. While rotations and scaling can be represented using matrix multiplication in standard Cartesian coordinates, translations require addition:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} \\ r_{21} & r_{22} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} \tag{1}$$

This creates an inconsistency in how we represent and combine transformations. Homogeneous coordinates solve this problem by allowing all transformations, including translations, to be represented as matrix multiplications.

2.2 Definition of Homogeneous Coordinates

A point (x, y) in 2D Euclidean space is represented in homogeneous coordinates as (wx, wy, w) where $w \neq 0$. Typically, we normalize these coordinates by setting $w = 1$, giving us $(x, y, 1)$.

Similarly, a point (X, Y, Z) in 3D Euclidean space is represented in homogeneous coordinates as (wX, wY, wZ, w) where $w \neq 0$, or more commonly as $(X, Y, Z, 1)$ after normalization.

The key insights of homogeneous coordinates are:

- All points (wx, wy, w) with the same ratio of coordinates represent the same 2D point.
- Points with $w = 0$ represent points at infinity, corresponding to directions.

- The extra coordinate allows us to represent projective transformations as linear transformations.

2.3 Transformation Matrices in Homogeneous Coordinates

Using homogeneous coordinates, we can represent various transformations as matrices:

Translation:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{2}$$

Rotation:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{3}$$

Scaling:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & 0 \\ 0 & s_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{4}$$

Perspective Projection:

$$\begin{bmatrix} x' \\ y' \\ w' \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{5}$$

To convert back to Euclidean coordinates, we divide by the last component: $(x', y') = (x'/w', y'/w')$.

2.4 Advantages in Computer Vision

In computer vision, homogeneous coordinates offer several advantages:

- **Uniform transformation representation**: All transformations (including projections) can be represented as matrix multiplications.
- **Composition of transformations**: Multiple transformations can be combined by multiplying their matrices.
- **Representation of projective geometry**: Points at infinity can be handled, which is crucial for perspective projections.
- **Mathematical elegance**: Many computer vision algorithms become simpler and more elegant when formulated using homogeneous coordinates. Essentially, homogeneous coordinates is closely related to differential geometry since the transformation is on the manifold rather than the Euclidean space.

When you see a "1" appended to coordinates in computer vision equations (like in the pinhole camera model), it indicates that homogeneous coordinates are being used. This allows for the linear representation of projective transformations, which is essential for camera modeling.

3 The Pinhole Camera Model

3.1 Historical Context

The pinhole camera, also known as the camera obscura (Latin for "dark chamber"), is one of the earliest known imaging devices. The concept dates back to ancient China and Greece, with scholars like Aristotle describing the phenomena of light passing through a small aperture to create an inverted image on an opposite surface. This simple physical principle forms the basis of modern camera models in computer vision.

3.2 Mathematical Formulation

The pinhole camera model describes the mathematical relationship between the 3D world and its 2D projection onto an image plane. This model assumes that all light rays pass through a single point (the pinhole) and projects onto an image plane.

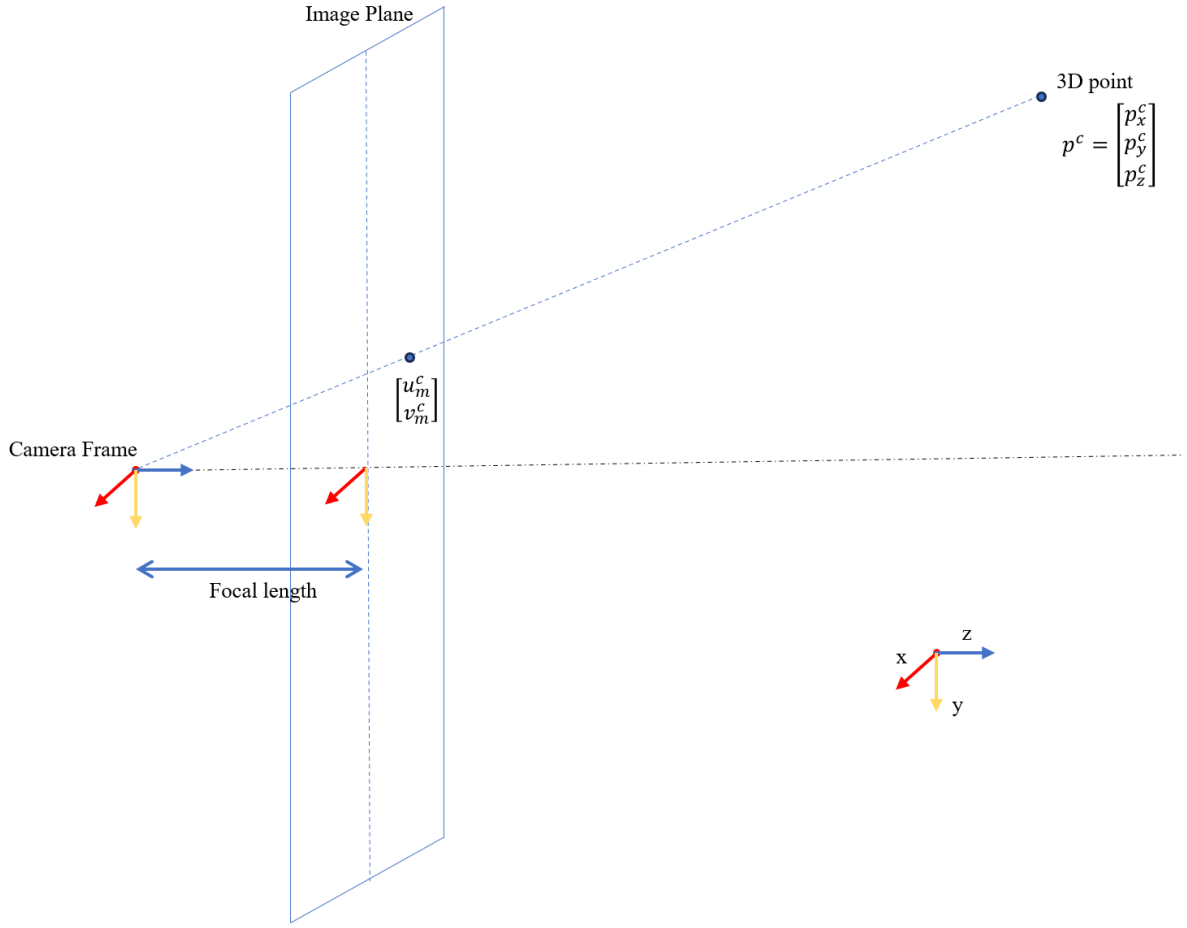

Figure 1: The pinhole camera model: a 3D point $\mathbf{p}^c$ in camera coordinates is projected onto the image plane at point $[u_m^c, v_m^c]$.

In practice, we usually place the image plane in front of the camera center (rather than behind it as in a real pinhole camera) to avoid having to deal with inverted images. This provides an equivalent model that is mathematically more convenient.

3.2.1 Projection Equations

Following the notation in the figure, let's consider a 3D point $\mathbf{p}^c = [p_x^c, p_y^c, p_z^c]$ in the camera coordinate system and its projection $[u_m^c, v_m^c]$ on the image plane. Using similar triangles, we can derive:

$$\frac{u_m^c}{f} = \frac{p_x^c}{p_z^c} \quad \text{and} \quad \frac{v_m^c}{f} = \frac{p_y^c}{p_z^c} \tag{6}$$

Where f is the focal length of the camera (the distance from the camera center to the image plane). Rearranging, we get:

$$u_m^c = f\frac{p_x^c}{p_z^c} \quad \text{and} \quad v_m^c = f\frac{p_y^c}{p_z^c} \tag{7}$$

These equations represent the perspective projection, which maps 3D points to 2D image coordinates. We can express this in homogeneous coordinates as:

$$\begin{bmatrix} u_m^c \\ v_m^c \\ 1 \end{bmatrix} = \frac{1}{p_z^c} \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} p_x^c \\ p_y^c \\ p_z^c \\ 1 \end{bmatrix} \tag{8}$$

3.3 From Idealized to Real Camera Models

The simplified model above assumes that:

- The origin of the image coordinate system is at the principal point (where the optical axis intersects the image plane).
- The pixels are perfectly square.
- There is no lens distortion.

Real cameras, however, deviate from these assumptions. To account for real-world cameras, we need to introduce additional parameters:

3.3.1 Principal Point Offset

The principal point is rarely at the exact center of the image. Let (c_x, c_y) be the coordinates of the principal point in the image coordinate system. Then:

$$u_m^c = f\frac{p_x^c}{p_z^c} + c_x \quad \text{and} \quad v_m^c = f\frac{p_y^c}{p_z^c} + c_y \tag{9}$$

3.3.2 Pixel Aspect Ratio

In many cameras, especially older ones, pixels might not be perfectly square. Different scaling factors f_x and f_y can be used for the x and y directions:

$$u_m^c = f_x\frac{p_x^c}{p_z^c} + c_x \quad \text{and} \quad v_m^c = f_y\frac{p_y^c}{p_z^c} + c_y \tag{10}$$

3.3.3 Skew Factor

In some cases, the x and y axes of the image sensor might not be perfectly perpendicular. This introduces a skew factor s:

$$u_m^c = f_x \frac{p_x^c}{p_z^c} + s\frac{p_y^c}{p_z^c} + c_x \quad \text{and} \quad v_m^c = f_y \frac{p_y^c}{p_z^c} + c_y \tag{11}$$

In modern cameras, the skew factor is typically very close to zero.

4 Camera Parameters

Camera parameters are divided into two categories: intrinsic and extrinsic.

4.1 Intrinsic Parameters

Intrinsic parameters represent the internal optical and geometric characteristics of the camera:

$$K = \begin{bmatrix} f_x & s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \tag{12}$$

Where:

- f_x and f_y are the focal lengths in pixel units along the x and y axes
- s is the skew factor (typically near zero for modern cameras)
- (c_x, c_y) is the principal point, usually near the center of the image

The intrinsic parameters are specific to a camera and remain constant as long as the camera's internal settings (like focal length for zoom lenses) don't change.

4.2 Extrinsic Parameters

Extrinsic parameters define the camera's position and orientation in the world coordinate system. They transform points from *world coordinates* ($\mathbf{p}^w$) to *camera coordinates* ($\mathbf{p}^c$):

$$\mathbf{p}^c = R(\mathbf{p}^w - \mathbf{t}) \tag{13}$$

Or in homogeneous coordinates:

$$\begin{bmatrix} p_x^c \\ p_y^c \\ p_z^c \\ 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & -R\mathbf{t} \\ r_{21} & r_{22} & r_{23} & -R\mathbf{t} \\ r_{31} & r_{32} & r_{33} & -R\mathbf{t} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p_x^w \\ p_y^w \\ p_z^w \\ 1 \end{bmatrix} \tag{14}$$

Often, the extrinsic parameters are expressed in a more compact form:

$$[R|t] = \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \tag{15}$$

Where:

- R is a 3×3 rotation matrix describing the camera's orientation relative to the world
- t is a 3×1 translation vector describing the camera's position in world coordinates

The extrinsic parameters change whenever the camera moves relative to the world coordinate system.

4.3 The Complete Camera Model

Combining intrinsic and extrinsic parameters, we get the complete camera projection matrix P:

$$P = K[R|t] \tag{16}$$

Then, the projection of a 3D point in world coordinates $\mathbf{p}^w = [p_x^w, p_y^w, p_z^w, 1]^T$ to a 2D image point $[u_m^c, v_m^c, 1]^T$ is:

$$\lambda \begin{bmatrix} u_m^c \\ v_m^c \\ 1 \end{bmatrix} = P \begin{bmatrix} p_x^w \\ p_y^w \\ p_z^w \\ 1 \end{bmatrix} \tag{17}$$

Where λ is a scaling factor equal to the depth of the point in camera coordinates (p_z^c).

4.4 Lens Distortion

Real camera lenses introduce various types of distortion that are not accounted for in the linear pinhole model. The two main types are:

4.4.1 Radial Distortion

Radial distortion causes straight lines to appear curved. It occurs because light rays bend differently depending on their distance from the optical center. The distortion is modeled by:

$$\begin{aligned} u_{distorted}^c &= u_{undistorted}^c(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) \\ v_{distorted}^c &= v_{undistorted}^c(1 + k_1 r^2 + k_2 r^4 + k_3 r^6) \end{aligned} \tag{18}$$

Where:

- $(u_{undistorted}^c, v_{undistorted}^c)$ are the undistorted image coordinates
- $(u_{distorted}^c, v_{distorted}^c)$ are the distorted image coordinates
- $r^2 = (u_{undistorted}^c)^2 + (v_{undistorted}^c)^2$ is the squared distance from the optical center
- k_1, k_2, k_3 are the radial distortion coefficients

4.4.2 Tangential Distortion

Tangential distortion occurs when the lens is not perfectly parallel to the image plane. It is modeled by:

$$\begin{aligned} u^c_{distorted} &= u^c_{undistorted} + [2p_1 u^c_{undistorted} v^c_{undistorted} + p_2(r^2 + 2(u^c_{undistorted})^2)] \\ v^c_{distorted} &= v^c_{undistorted} + [p_1(r^2 + 2(v^c_{undistorted})^2) + 2p_2 u^c_{undistorted} v^c_{undistorted}] \end{aligned} \tag{19}$$

Where p_1 and p_2 are the tangential distortion coefficients.

5 Camera Calibration Methods: Zhang's Method

5.1 Overview

Camera calibration is the process of estimating the intrinsic and extrinsic parameters of a camera. There are several methods for camera calibration.

Zhang's method, which we will use in our lab, requires the camera to observe a planar pattern (like a checkerboard) from multiple viewpoints. The key advantages of this method are:

- It doesn't require expensive 3D calibration objects
- The calibration pattern can be printed on a regular printer
- It can achieve high accuracy with a sufficient number of views
- It's robust to noise and can handle both intrinsic and lens distortion parameters

Zhang's method works by establishing constraints on the intrinsic parameters from the homographies between the model plane and its images. Each view of the planar pattern provides two constraints on the intrinsic parameters. By taking multiple views (at least three), we can solve for all the parameters.

5.2 Mathematical Formulation of Zhang's Method

Let's explore the mathematics behind Zhang's method in more detail.

5.2.1 Homography Between the Model Plane and its Image

Consider a model plane with a world coordinate system where $Z = 0$. Let's denote a point on this plane as $\mathbf{M} = [X, Y, 0, 1]^T$. Its image under the perspective projection is $\mathbf{m} = [u, v, 1]^T$.

The relationship between $\mathbf{M}$ and $\mathbf{m}$ is:

$$s\mathbf{m} = P\mathbf{M} = K[R|t]\mathbf{M} = K[\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3, \mathbf{t}] \begin{bmatrix} X \\ Y \\ 0 \\ 1 \end{bmatrix} = K[\mathbf{r}_1, \mathbf{r}_2, \mathbf{t}] \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} \tag{20}$$

Where $\mathbf{r}_i$ is the i-th column of the rotation matrix R. We define the homography H as:

$$H = K[\mathbf{r}_1, \mathbf{r}_2, \mathbf{t}] \tag{21}$$

Thus:

$$s\mathbf{m} = H \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} \tag{22}$$

5.2.2 Constraints on the Intrinsic Parameters

Given a homography $H = [h_1, h_2, h_3]$, we have:

$$\begin{aligned} h_1 &= \lambda K \mathbf{r}_1 \\ h_2 &= \lambda K \mathbf{r}_2 \\ h_3 &= \lambda K \mathbf{t} \end{aligned} \tag{23}$$

Since $\mathbf{r}_1$ and $\mathbf{r}_2$ are orthonormal (i.e., $\mathbf{r}_1^T \mathbf{r}_2 = 0$ and $\mathbf{r}_1^T \mathbf{r}_1 = \mathbf{r}_2^T \mathbf{r}_2 = 1$), we get the following constraints:

$$\begin{aligned} h_1^T K^{-T} K^{-1} h_2 &= 0 \\ h_1^T K^{-T} K^{-1} h_1 &= h_2^T K^{-T} K^{-1} h_2 \end{aligned} \tag{24}$$

Each view of the planar pattern gives us two such constraints on the intrinsic parameters. With at least three views, we can solve for all five intrinsic parameters (assuming zero skew).

This is why in our lab, we need to take multiple photos of the checkerboard pattern from different angles - each image provides two constraints on our camera's intrinsic parameters.

5.2.3 Solving for Camera Parameters

The calibration procedure can be summarized as follows:

1. Capture multiple images of a planar checkerboard pattern from different viewpoints
2. Detect the corner points of the checkerboard in each image
3. Estimate the homography between the model plane and each image
4. Use the constraints from the homographies to estimate the parameters

6 Applications of Camera Calibration

Camera calibration is a fundamental procedure in many computer vision and robotics applications. Here are some important applications:

6.1 3D Reconstruction

To reconstruct a 3D scene from multiple 2D images, you need to know the camera parameters. Camera calibration provides the intrinsic parameters that allow you to triangulate 3D points from their projections in multiple images.

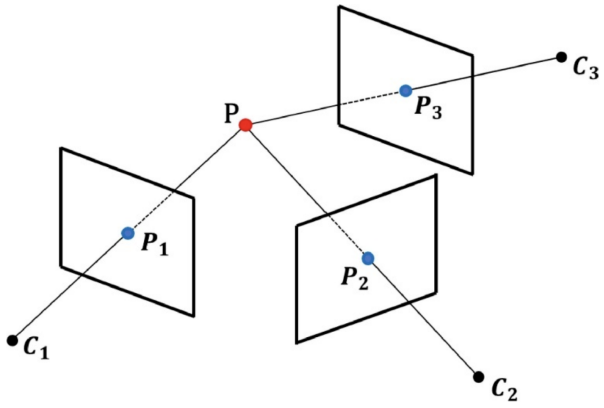

Figure 2: Determining the 3D position of a point from multiple calibrated cameras.

6.2 Visual Odometry and SLAM

Visual odometry is the process of determining the position and orientation of a camera by analyzing the changes between consecutive images. Simultaneous Localization and Mapping (SLAM) goes further by building a map of the environment at the same time. Both require calibrated cameras.

Applications include:

- Autonomous navigation for robots and drones
- Indoor positioning systems
- Advanced driver-assistance systems (ADAS)
- Virtual reality tracking

6.3 Image Rectification

Camera calibration parameters can be used to correct lens distortion and other image aberrations, resulting in more accurate images. Image rectification is widely used in preprocessing for computer vision algorithms, document scanning, and satellite and aerial image processing.

7 Mobile Phone Camera Calibration

Modern smartphones contain highly sophisticated cameras, but they still exhibit lens distortion and other imperfections that can be corrected through calibration. Some specific considerations for mobile phone camera calibration:

- **Rolling Shutter Effects**: Most smartphone cameras use rolling shutters (capturing the image row by row) rather than global shutters. This can introduce distortions when there is motion.
- **Auto-focus and Auto-exposure**: These features can change the intrinsic parameters between shots. It's best to lock these settings during calibration.
- **Wide-angle Lenses**: Many smartphones now have ultra-wide-angle lenses with significant distortion. These require more complex distortion models.
- **Multiple Cameras**: Modern smartphones often have multiple cameras with different parameters. Each camera needs to be calibrated separately.

Despite these challenges, the calibration methods we'll use in the lab (based on Zhang's method) work well for smartphone cameras.

8 Conclusion

Camera calibration is a fundamental process in computer vision that provides the mathematical link between 2D images and the 3D world. Understanding the theory behind camera models and calibration is essential for developing computer vision applications that interact with the real world.

In the upcoming lab session, you will put this theory into practice by calibrating your smartphone camera using a checkerboard pattern. This hands-on experience will solidify your understanding of the concepts covered in this prelab and prepare you for more advanced computer vision tasks.

References

1. Zhang, Z. (2000). A flexible new technique for camera calibration. IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(11), 1330-1334.
2. Hartley, R., & Zisserman, A. (2003). Multiple view geometry in computer vision. Cambridge University Press.
3. Bradski, G., & Kaehler, A. (2008). Learning OpenCV: Computer vision with the OpenCV library. O'Reilly Media, Inc.
4. Szeliski, R. (2010). Computer vision: algorithms and applications. Springer Science & Business Media.
5. Tsai, R. Y. (1987). A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses. IEEE Journal on Robotics and Automation, 3(4), 323-344.
6. Yang, Heng. 2023. "Optimal Control and Estimation." November 14, 2023. https://hankyang.seas.harvard.edu/OptimalControlEstimation/geometric-vision.html.