Homework Spring 2025

Applied Computer Vision (CMU-Africa)

Release Date: Wednesday, January 22nd, 2025 CAT
DUE: **Sunday February 2^{nd}, 2025, 11:59 PM CAT**

1. **Collaboration policy:** You are permitted to collaborate with fellow students on discussing homework problems; however, the submitted work must be composed in your own words and should not be copied from any external sources. It is essential to document the individuals with whom you discussed the assignment by listing their names at the top of the first page of your homework solutions. While we highly recommend using LaTeX to type your report, it is not mandatory.

2. **Submitting your work:** Assignment will be submitted through Canvas. Please ensure that you include all the necessary files to reproduce your output. If you believe a specific environment is required to run your code, kindly submit a readme file detailing the procedures for setting up this environment.

3. **Getting Help**: Please use office hours and Piazza to ask any questions related to this assignment.

4. **Refrain from using ChatGPT**: These problems are easy and can be easily solved with the use of generative AI's like ChatGPT. However, you will not learn by using these tools. Not learning will impact your performance outside of the classroom and you will not reflect . Ask questions on Piazza about anything you don't understand, the TA's and instructors will respond to you as fast as possible. Moreover, you will not have access to internet during the exam.

Learning Objectives

Understand Camera Projection Principles

- Gain insight into the mathematical and geometric concepts of projecting 3D points onto 2D planes.
- Learn how image plane positioning and orientation influence projection results.

Explore the Impact of Camera Parameters

- Experiment with different focal lengths (f) and understand their effect on the field of view and projection distortions.
- Investigate the impact of tilted image planes on projection and how it alters the perspective.

Develop Analytical Skills in Projection Techniques

- Analyze the differences between projections on standard and tilted image planes.
- Compare and interpret how changes in image plane properties and focal length affect the final projections.

Apply Visualization to Reinforce Concepts

- Use plotting tools to visualize the 2D projections for different configurations.
- Build a stronger connection between theoretical calculations and practical graphical outputs.

Image Formation

In this section, we will explore the process through which a camera captures images. Figure 1 below provides an overview of an image sensing pipeline. We will simulate certain aspects of this pipeline, with Part 1 concentrating on the optical component. In Part 2, you will implement one of the demosaic algorithms as part of the Image Signal Processor. For additional information, you can refer to Chapter Two in the textbook Computer Vision Algorithms and Applications Second Edition Richard Szeliski

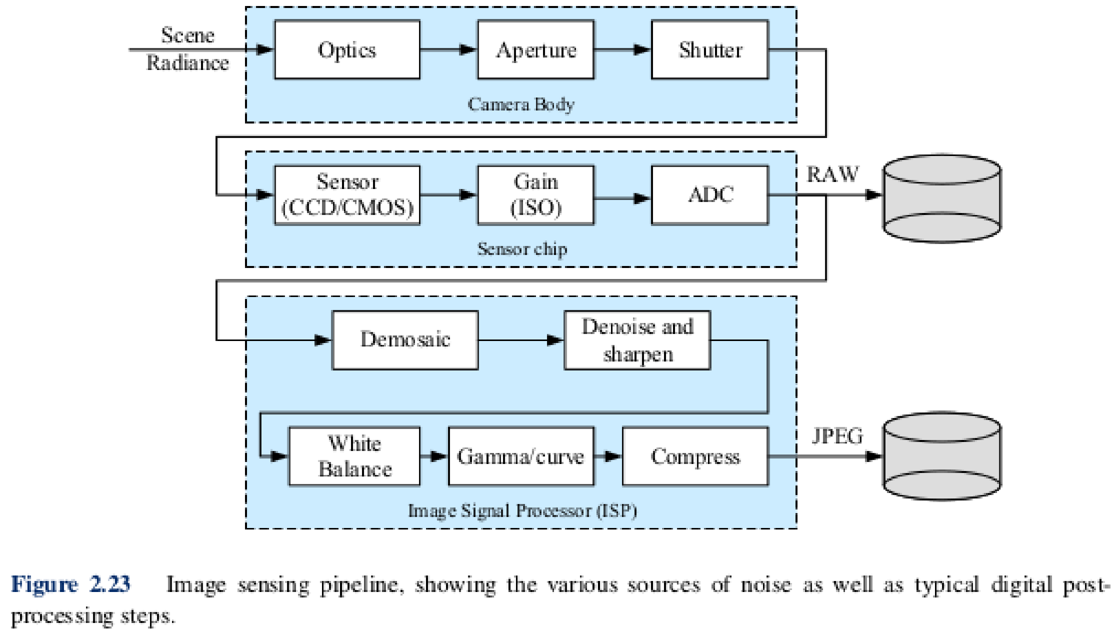
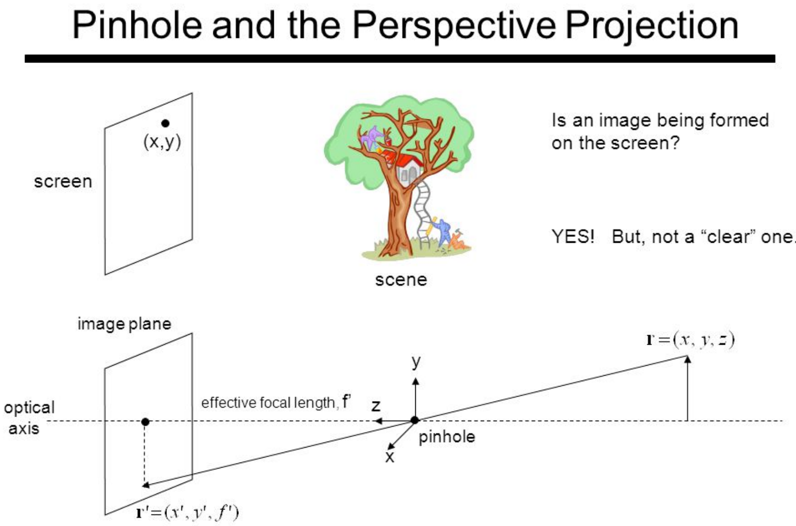


Figure 1: Image Sensinge Pipeline source:Computer Vision Algorithms and Applications Second Edition Richard Szeliski

Part I (Pinhole Camera) (70 Points)

A pinhole camera is a simple optical device consisting of a light-tight box or chamber with a small aperture (or pinhole) on one side and a photosensitive surface on the opposite side. Light entering through the pinhole projects an inverted image of the scene outside onto the photosensitive surface, illustrating the fundamental principles of optical projection in a camera without the need for a lens. As the pinhole camera involves sensors, we will examine an earlier version of such a camera known as the Camera Obscura. This camera operates on the principle of perspective projection, as depicted in Figure 2. Using the equations, we can determine the projected positions of objects in the world onto the image as they pass through the pinhole. In this context, we assume that every object in the world has a reflected ray passing through the pinhole, thus neglecting the position of the light source. For simplicity, we will treat the objects in the world as points, described by their (x, y, z) coordinates.



$$\frac{\mathbf{r}'}{f'} = \frac{\mathbf{r}}{z} \quad \Rightarrow \quad \frac{x'}{f'} = \frac{x}{z} \qquad \frac{y'}{f'} = \frac{y}{z}$$

Figure 2: Pinhole and Perspective Projection

You are provided with a list of 3D points (blue and black points) in a NumPy pickle file. Your task is to project these points onto various image planes using the pinhole camera model. You will experiment with different camera positions, orientations, and focal lengths while exploring the use of homogeneous coordinates for a more generalized representation.

1. Step 1: Define the parameters of the camera and image plane.

   - Choose the parameters for your pinhole camera:
     - Camera Center $C = (C_x, C_y, C_z)$
     - Focal length f (distance from the camera center to the image plane)
     - Orientation of the optical axis (aligned with or deviating from the $Z-$axis.
     - Position and equation of the image plane.

   Tasks:

   (a) Write down the equation of your image plane (both custom plane equation ($ax + by + cz + d = 0$) and constant ($Z = Z_0$)

   (b) Experiment with at least two different image planes
      - A standard plane perpendicular to the Z-axis.
      - A rotated or slanted plane

2. Step 2: Trace Rays from 3D points to the Image Plane

   For a given 3D point $P = (X, Y, Z)$, write the parametric equation of the ray passing through the camera center C.
   Tasks:

(a) Derive the general ray equation

(b) Calculate where this ray intersects the image plane

3. Step 3: Project Points to the Image Plane Find the intersection point of the ray with the image plane. Use the plane equation and the ray equation to calculate the point of intersection.
Tasks:

(a) Compute the intersection points $I = (I_x, I_y, Z_0)$

(b) Repeat this process for all given 3D points and image planes

4. Step 4: Map to 2D Image Coordinates

- Map the intersection points on the image plane to 2D image coordinates (u, v) based on:
  - Principal point (image center) (c_x, c_y) in the image plane.
  - Pixel scaling factors s_x and s_y

  Tasks

  (a) Use the following equations for mapping
  $u = s_x(I_x - O_x) + c_x, \quad v = s_y(I_y - O_y) + c_y$
  Assuming $O_x = O_y = 0 \;\; s_x = s_y = 1$
  $u = s_x(I_x - O_x) + c_x = 1 \cdot (I_x - 0) + c_x = I_x + c_x$
  $v = s_y(I_y - O_y) + c_y = 1 \cdot (I_y - 0) + c_y = I_y + c_y$
  calculate (u, v)

Experimentation Tasks

Projection with Standard Image Plane

Use $Z = Z_0$ as your image plane. Experiment with different focal lengths f (e.g., $f = 10\,, 50\,, 100\,$). Plot the 2D projections for all points on this plane.

Projection with a Tilted Image Plane

Define a tilted image plane, such as $ax + by + cz + d = 0$. Choose reasonable values for a, b, c, d and compute the intersections. Plot the 2D projections for all points.

Analysis of Parameters

- How does changing the focal length f impact the projection?
- How does the position and orientation of the image plane affect the results?
- Compare projections for the two image planes and discuss the differences.

Ideal Points

Calculate the distance between the first black point and the first blue point as well as the last blue and black point (you can calculate this by taking the L2 norm of a vector formed by this pair of points. Repeat same for the points just on the image plane just before the pixel mapping.

- What is your observation?
- What is the reason for them being different or them being the same ?

Part II (Image Demosaicing) (30 Points)

Image demosaicing is a crucial process that reconstructs a full-color image from incomplete color information captured by a digital camera's sensor, often using a color filter array (CFA). As image sensors do not directly measure full-color details at each pixel, demosaicing algorithms play a vital role in filling in missing color information. This enhances visual appeal, ensures accurate color representation in the final image, and is critical for overall image quality. The efficiency of demosaicing significantly impacts color accuracy, sharpness, and the reproduction of fine details in digital photographs.

In regions such as Africa, where the cost of obtaining high-quality images is prohibitive, efficient demosaicing algorithms become essential for improving the visual quality of images captured by low-cost cameras. You will implement a simple demosaicing algorithm as part of this process. You will be implementing a straightforward Bilinear Interpolation algorithm to fill the missing color information in a raw image. Please refer strictly to the instructions provided in the attached notebook, as we will not cover all the steps outlined in the referenced paper.

Sample images are provided for your reference; however, it's important to note that we will assess your implementation using other images not included in this assignment. Ensure that your algorithm generalizes well to various images beyond the provided samples.

In your report, consider discussing potential improvements to the demosaicing process. Explore how this process could be enhanced and highlight the limitations of the implemented algorithm. Additionally, provide insights into other demosaicing algorithms, including their advantages and limitations.

For instance, it can be mentioned that deep learning methods have shown promise in improving image quality, but they often require large amounts of labeled data for training. Discuss the constraints posed by data availability in certain scenarios, such as limited resources or specific regions like Africa mentioned earlier. This would provide a comprehensive view of the challenges and potential avenues for advancing demosaicing techniques.

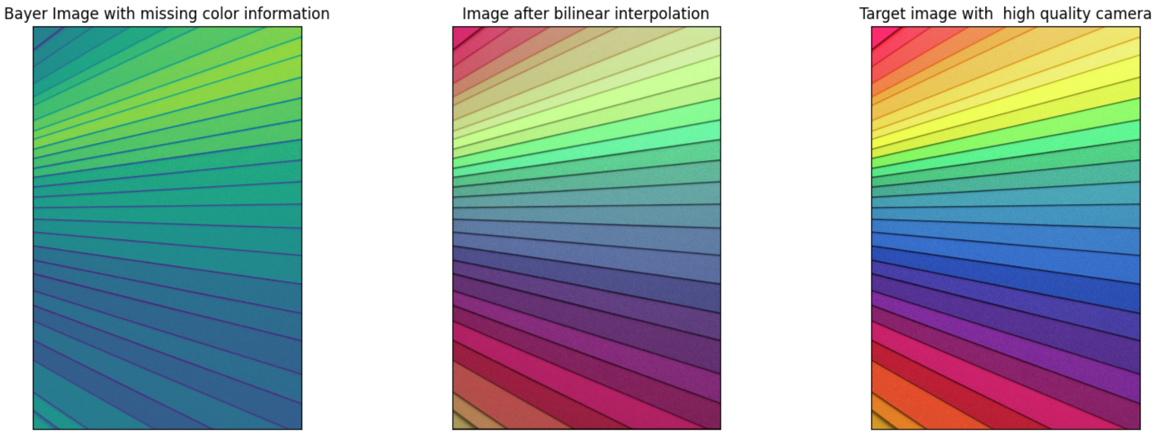


Figure 3: Image Demosaicing Results