Image Formation and Representation - Overview

Key Concepts to Master

1. Mathematical Representation of Digital Images

- **Digital image:** 2D matrix I(row, col) or I(x, y)
- **Coordinate system:**
  - Matrix notation: I(row, col) - row increases downward
  - Image notation: I(x, y) - x is horizontal, y is vertical
  - MATLAB uses 1-based indexing: I(1,1) is top-left

2. Sampling and Quantization

Sampling: Converting continuous spatial domain → discrete pixels

- Controls spatial resolution
- Higher sampling = more pixels = finer detail
- Aliasing occurs when sampling rate too low

Quantization: Converting continuous intensity → discrete levels

- Controls intensity resolution (bit depth)
- 8-bit = 256 levels (0-255)
- Lower quantization = posterization effect

3. Pinhole Camera Model

Geometry:

```
3D World Point (X, Y, Z)
      ↓
  Pinhole (optical center)
      ↓
2D Image Point (x, y)
```

Key properties:

- No lens distortion
- Infinite depth of field
- Inverted image
- Smaller aperture → sharper but darker

4. Perspective Projection Equations

Basic equations:

```
x = f * (X / Z)
y = f * (Y / Z)
```

Where:

- (X, Y, Z) = 3D world coordinates
- (x, y) = 2D image coordinates
- f = focal length

- Z = depth (distance from camera)

Key insight: Objects further away (larger Z) appear smaller

5. Homogeneous Coordinates

Why use them?

- Represent both points and lines uniformly
- Enable matrix operations for transformations
- Handle points at infinity

Conversion:

- Cartesian (x, y) → Homogeneous $[x, y, 1]^T$
- Homogeneous $[x, y, w]^T$ → Cartesian (x/w, y/w)

Projection Matrix:

```
[x']   [f  0  0  0] [X]
[y'] = [0  f  0  0] [Y]
[z']   [0  0  1  0] [Z]
                     [1]

Then: x = x'/z', y = y'/z'
```

6. Effects of Perspective Projection

- **Parallel lines converge** to vanishing points
- **Size decreases** with distance
- **Shape distortion** for non-frontal planes
- **Depth ambiguity** - multiple 3D points project to same 2D point

7. Digital Image Types

Binary images:

- Pixels are 0 or 1
- Used for segmentation masks, text

Grayscale images:

- Typically 8-bit: 0 (black) to 255 (white)
- Single intensity value per pixel

Color images:

- RGB: 3 channels (Red, Green, Blue)
- Each channel typically 8-bit
- Other spaces: HSV, YCbCr, LAB

Labeled images:

- Each pixel has integer label
- Used for segmentation (different regions)

MATLAB Quick Reference

```
% Load and display image
img = imread('image.jpg');
imshow(img);

% Get image size
[rows, cols, channels] = size(img);

% Access pixel value
pixel_value = img(row, col);  % Grayscale
pixel_rgb = img(row, col, :);  % Color

% Convert to grayscale
gray_img = rgb2gray(img);

% Image type conversion
img_double = im2double(img);  % [0, 1]
img_uint8 = im2uint8(img);     % [0, 255]

% Perspective projection example
X = 10; Y = 5; Z = 20; f = 50;
x = f * (X / Z);
y = f * (Y / Z);
```

Study Checklist

- ☑ Can convert between coordinate systems
- ☑ Can perform perspective projection calculation by hand
- ☑ Understand homogeneous coordinates representation
- ☑ Know projection matrix form
- ☑ Can explain effects of sampling and quantization
- ☑ Can describe effects of perspective projection

Practice Problems

1. Given 3D point (6, 3, 10) and f=50mm, find image coordinates
2. Convert between matrix and image coordinate systems
3. Explain why parallel railroad tracks appear to converge

Answers

1. x = 50 * (6/10) = **30mm**, y = 50 * (3/10) = **15mm**. Image coordinates: **(30, 15) mm**.
2. Matrix notation I(row, col) maps to image notation I(x, y) by swapping axes: row corresponds to y (vertical, increasing downward) and col corresponds to x (horizontal, increasing rightward). So matrix I(r, c) = image I(c, r). Example: matrix position (3, 5) = image coordinates (x=5, y=3).
3. Perspective projection maps 3D points via x = f(X/Z). Railroad tracks have constant separation in X but increasing Z with distance. As Z grows, the projected x-coordinates f(X/Z) for each rail get closer together, converging toward x = 0 at Z = infinity. That convergence point is the **vanishing point**.

Related Topics

- Lecture 2: Image Formation
- Project 1: Image basics