Computer Vision

Introduction

Discord Link in Description

Course / Tutorial Overview

- **Basics of Image Processing**
  - Color Models
  - Noise and Smoothing
  - Histograms
  - Geometric Operations
- **Binary Image Processing**
  - Thresholding
  - Mathematical Morphology
  - Connectivity
- **3D Vision**
  - Camera Model
  - Epipolar Geometry
  - 3D Position with stereo vision

- **Edge Image Processing**
  - Edge Detection
  - Contour Following
  - Hough Transform
- **Features**
  - Moravec, Harris, FAST
  - SIFT
- **Recognition**
  - Hough
  - SIFT
  - Template and Chamfer Matching
  - Robust Object Detection

What is Computer Vision?

- Computer Vision is about understanding images
    - Greyscale, Color or Multi-spectral
    - Snapshots or video sequences
    - Static or moving camera
    - Static or dynamic scene
    - Calibrated / un-calibrated camera

- Computer Vision aims to extract useful information from images
    - Inspection purposes
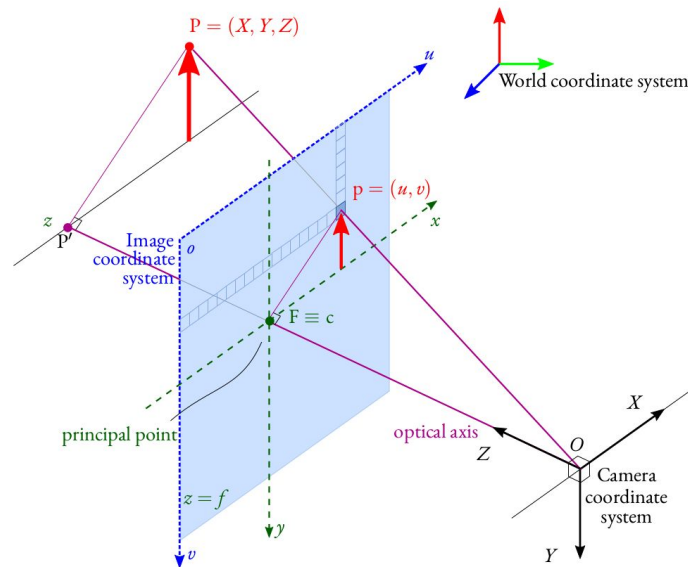   - Analysis purposes
    - Control purposes

Motivation for Computer Vision
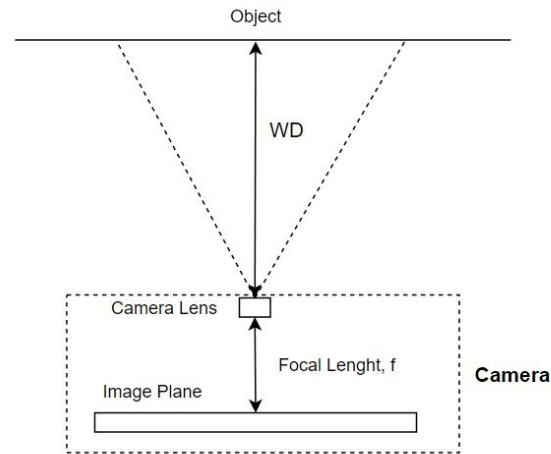
Pinhole Model for Cameras

$$\begin{bmatrix} i.w \\ j.w \\ w \end{bmatrix} = \begin{bmatrix} f_i & 0 & c_i \\ 0 & f_j & c_j \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

- Image Plane

- Focal Length

- Location of the optical centre

Camera Specifications

- Sensor type
  - CMOS/CDD
- Sensor Size (d)
- Resolution (pixels)
- Lens Focal Length (f)
- Lens Aperture Range

Object

WD

Camera Lens

Focal Lenght, f   **Camera**

Image Plane

$$\text{FOV} = \frac{d \cdot \text{WD}}{f}$$

Images and Pixels

- Continuous 2D Functions
  - (i,j) or (column,row) or (x,y)
- Discrete Representation
  - Sample
  - Quantise
- Number of Samples
  - Wasted time and computation
- Resolution
  - Quantisation - typically 8 bits
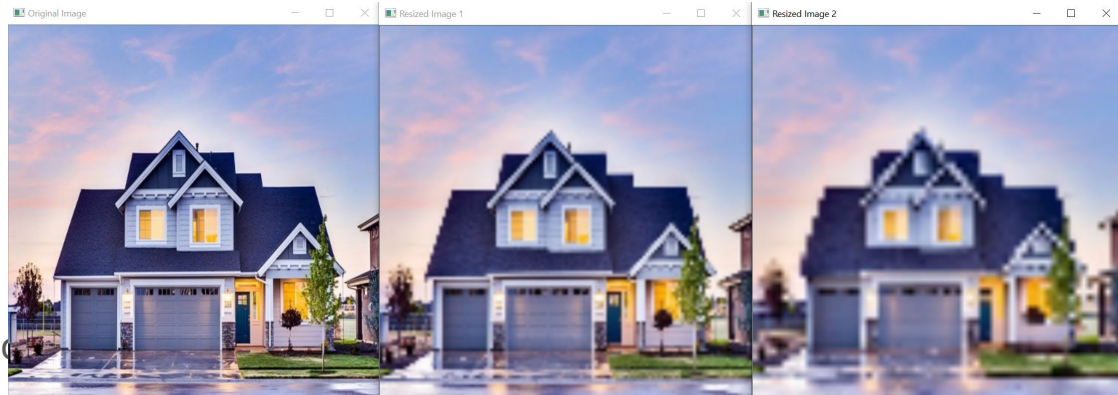

```
resize(image1, resizedImg, Size(image1.cols / 4, image1.rows / 4));
```

Quantisation - Code Example

```
void changeQuantisationGreyImages(Mat& img, int bits) {
    CV_Assert((img.type() == CV_8UC1) && (bits >= 1) && (bits <= 8));
    uchar mask = 0xFF << (8 - bits);
    for (int row = 0; row < img.rows; row++) {
        for (int col = 0; col < img.cols; col++)
        {
            img.at<uchar>(row, col) = img.at<uchar>(row, col) & mask;
        }
    }
}
```
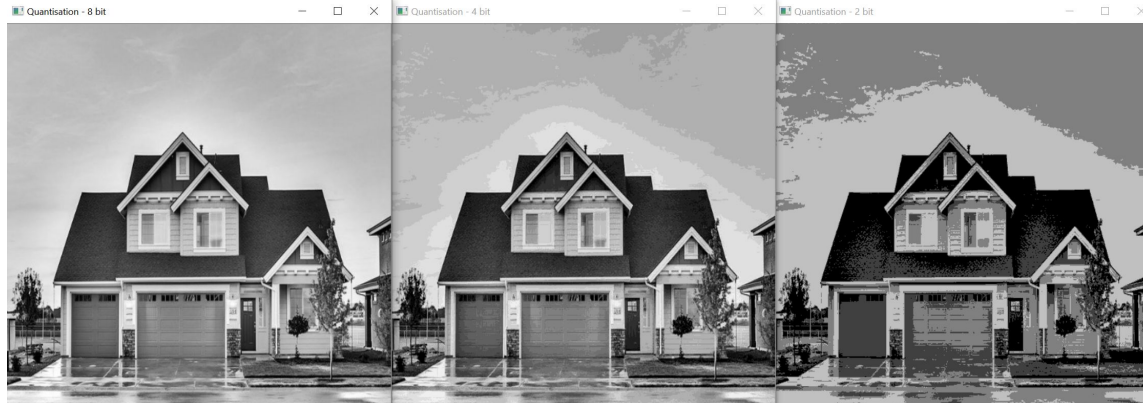
Color Models - Color vs Grayscale

- Luminance only
  - Simple representation
  - Humans can understand

- Color images
  - Multiple channels for colors
  - More complex to process

- Different Color Models
  - CMY Images
  - YUV Images
  - HLS Images
  - HSV Images

Typically operations on grayscale and representation on color images