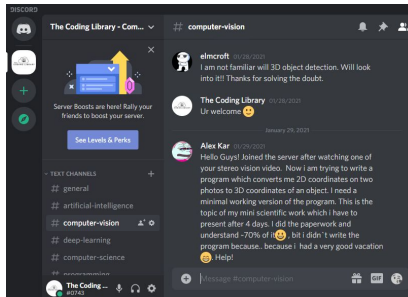




Computer Vision

Stereo Vision Geometry and Code Example



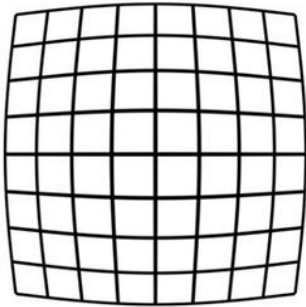
Discord Link in Description

Recap from Stereo Vision Video

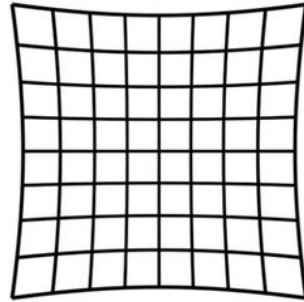
- 3D Sensors for Stereo Vision
- How to estimate the distance using stereo vision?
 - Calibrate the cameras (Intrinsic and extrinsic calibration)
 - Create an epipolar scheme using epipolar geometry
 - Build a disparity and a depth map
- Introduction to Disparity and Depth Maps
- Epipolar Geometry



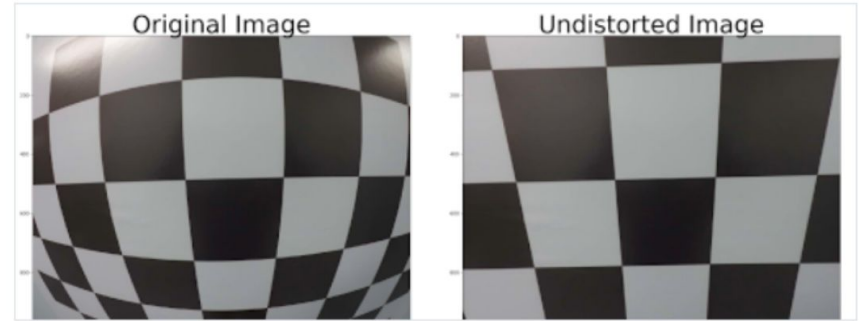
Camera Calibration - Distortion



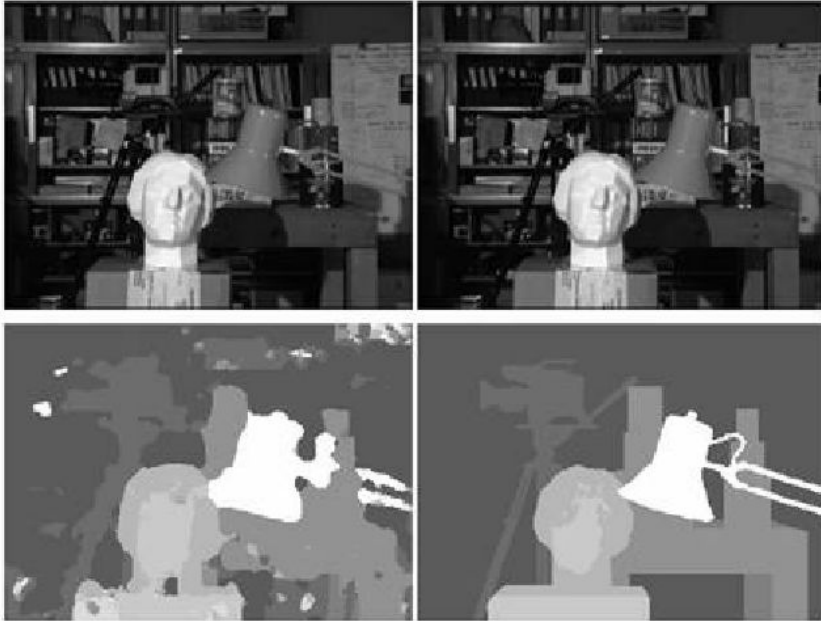
Barrel Distortion



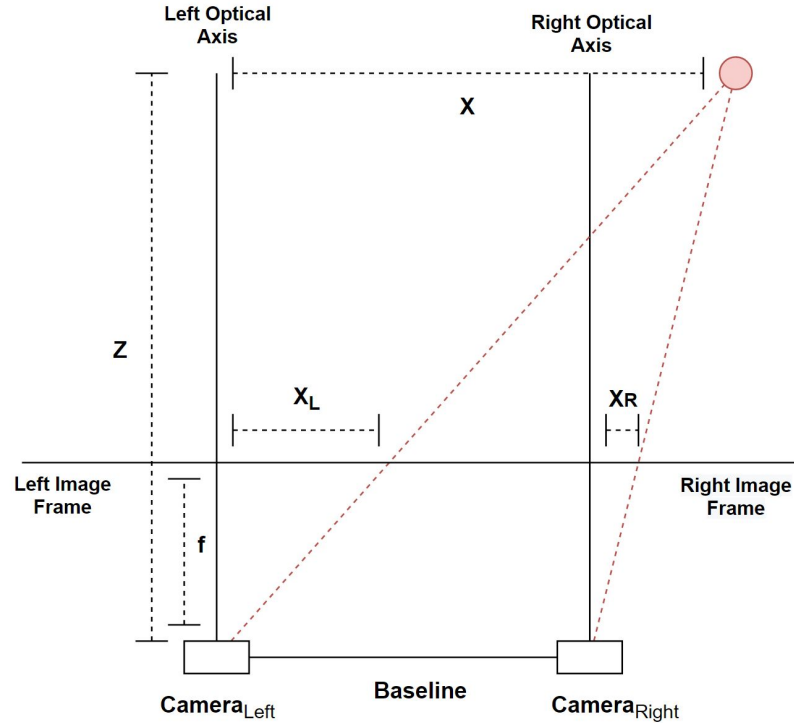
Pincushion Distortion



What are Disparity and Depth Maps?



Multi View Geometry - Disparity



$$\frac{Z}{f} = \frac{X}{x_L}$$

$$\frac{Z}{f} = \frac{X-b}{x_R}$$

$$x_L = \frac{X}{Z} \cdot f$$

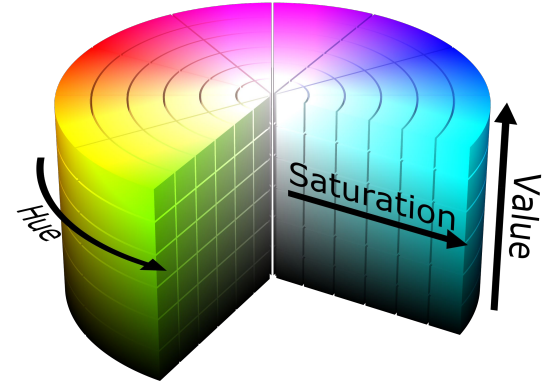
$$x_R = \frac{X-b}{Z} \cdot f$$

$$Disparity = x_L - x_R$$

Depth Estimation - Code Example



```
float StereoVision::find_depth(Point circleLeft, Point circleRight, Mat& leftFrame, Mat& rightFrame) {  
    int focal_pixels = 0;  
  
    if (rightFrame.cols == leftFrame.cols) {  
        // Convert focal length f from [mm] to [pixel]  
        focal_pixels = (rightFrame.cols * 0.5) / tan(alpha * 0.5 * CV_PI / 180.0);  
    }  
    else {  
        cout << "Left and Right Camera frames do not have the same pixel width" << endl;  
    }  
  
    int xLeft = circleLeft.x;  
    int xRight = circleRight.x;  
  
    // Calculate the disparity  
    int disparity = xLeft - xRight;  
  
    // Calculate the Depth Z  
    float zDepth = (baseline * float(focal_pixels)) / float(disparity);    // Depth in [cm]  
  
    return abs(zDepth);  
}
```



Neural Networks - Depth Estimation

