EE 8374: Fundamentals of Computer Vision

Final Fall 2018

Mingze Sun

Automated Dimensioning of Boxes using ToF sensors

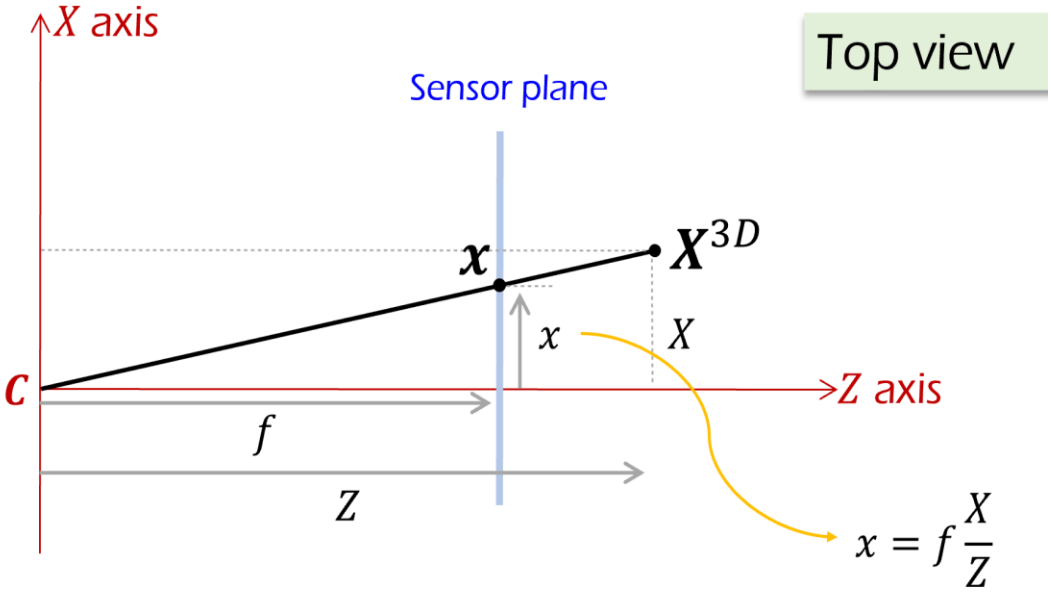
Questions:

1. Explain why the projective model of Eq.(1) is representative of the imaging geometry of Figure-1?

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \frac{1}{Z_w} K \begin{bmatrix} X_w \\ Y_W \\ Z_w \end{bmatrix}$$

Where x,y is pixel coordinate in image, Xw,Yw is world coordinate in 3D, Zw is depth from camera to object and K is camera intrinsic matrix.

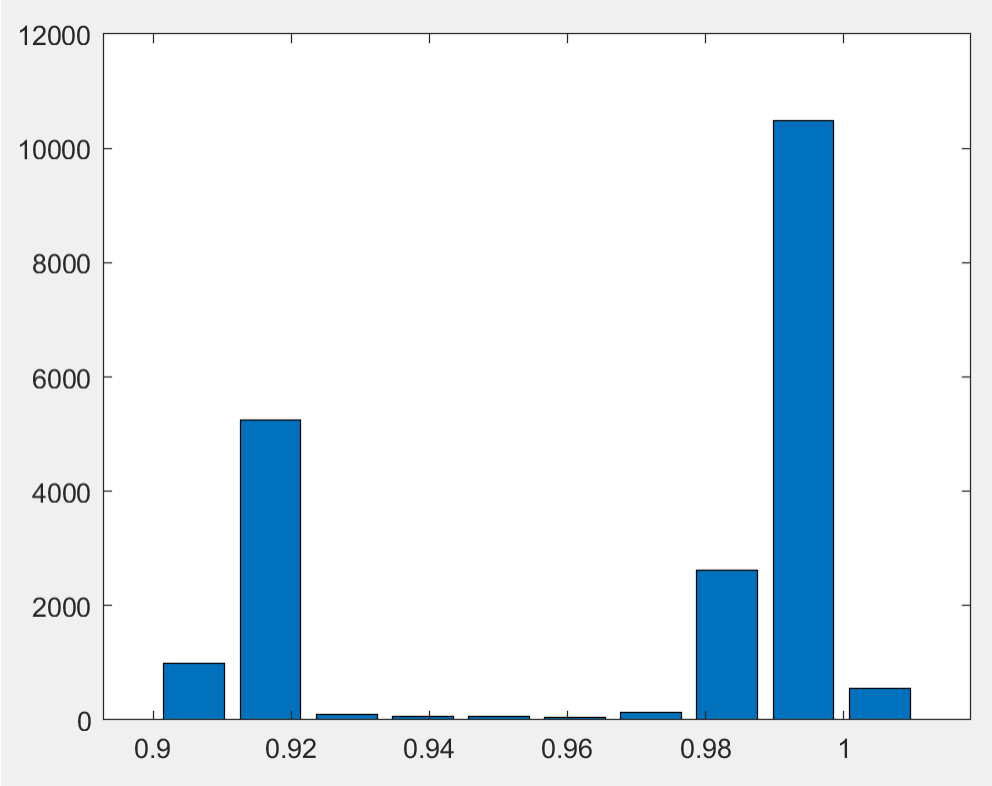
According to camera model in slides:



We know that x,y in image token by camera can be calculate by : $x = f\frac{X}{Z},\ y = f\frac{Y}{Z}$

Because f is in K, and The intrinsic matrix transforms 3D camera coordinates to 2D homogeneous image coordinates. So Eq.(1) is representative of the imaging geometry of Figure-1.

2. How many peaks do you see in the histogram of depths within the region-of-interest?
   Two peaks, which means there are two plains dominate.



3. Does the histogram represent a bimodal distribution?

   No, bimodal distribution is a mathematical theory and can be calculated by equation. But this histogram is influenced artificially because I choose the region of interest, if I change the ROI, the histogram could only represent one peak.

4. What do each of the peaks tell you about the depth Z_w of the box and the depth of the conveyor belt from the camera?
   First peak tell us about the depth of box and the highest peak tell us about conveyor belt because box is closer to camera.
5. What is your estimate of Z_w for points on the box and points on the conveyor belt (express in centimeters)?
   Zw_box = 0.917m = 91.7cm and Zw_conveyor = 0.9942m = 99.42cm
6. What is your estimate of the height of the box (express in centimeters)?
   0.0772m = 7.72cm

7. What is your estimate of the length of the box (express in centimeters)?

   Length = 35.82cm

8. What is your estimate of the width of the box (express in centimeters)?

   Width = 28.61cm

Provide a detailed explanation of your strategy for identifying the corners of the box, and subsequently the dimensions of the box.

I used detectHarrisFeature function to detect corners of the box, however, I found that if I just use the undistort intensity image, it's hard to detect the corners of the box and will return many wrong features. So I first choose region of interest from 80 to 180 in row and 60 to 260 in column, then I found that the depth map become very good for detecting corner, but if I only use depth map, the corner be detected would be a little difference to intensity map's corner, so I add depth map to intensity image and using Harris detect function then choose 4 strongest corners, it shows that I got the four correct corners.

Then I sort the X value of 4 corners, the minimal value represents the column of leftmost corner and the maximum value represents the column of rightmost corner, the I only need to get the row value corresponding these two pixels, I can get the coordinate value of leftmost and rightmost corner. In same way, I sort the Y value and can find the top and bottom corners.

It is worth to mention that in this way, I don't need to figure out the top-left, top-right, bottom-left or bottom-right corner because the length and width will be only calculated by L-T and L-B or R-T and R-B corners. However, I noticed another problem that when the box is parallel to the conveyor, I would get 2 minimal and 2 maximum X or Y values and the coordinate corresponding to these values would mess up. This problem could be fix up by following way: when I get the x value of leftmost pixel and want to find the y corresponding to it, I will add a min function to make sure there is only one least [x,y] pixel be returned. In parallel situation, this pixel will be automatically set to the left-top corner. And when I get the x value of rightmost pixel, I will add a max function to get the y value corresponding to it, and this will automatically set the pixel to right-bottom corner. In same way, when I get the least y value, which means it is a top corner, I will add a max function to make this pixel to a top-right corner and same with bottom-left corner, it will be added a min function.

The next step is to convert pixel coordinate to world coordinate, according to eq(1), pixel matrix is `pixel_L = [corners_L(1),corners_L(2),1]';` and world coordinate matrix is previous setting to `world_L = [1,1,Z_box]'`. because the dimension of box is what I need to measure, so the Zw should be the depth of box. then use the function `world_L = Z_box*inv(IntrinsicMatrix)*pixel_L;` the world

coordinate of the corner will be calculated. In same way, I can get all world coordinate of four corners.

The last step is to calculate the length and width, I use left-top and right-bottom corners to calculate the length and get their mean value. And left-bottom and right-top corners to calculate the width and get their mean value. The final results are 35.82cm and 28.61cm.