# Computer Vision

## Object Measuring Project

# Task 1

Calculate the necessary working space and working distance of camera for your task. Comparing the calculated and the real data of working space and working distance.

# Camera Selection

For this project, I choose an Iphone 8 as a camera with the following parameters:

- Focal Length = 4mm
- Sensor size = 4.8 x 3.6 mm

## Step 1: Select desired FOV

Our group want the FOV of the camera to be 500 × 280

$$\Rightarrow \text{FOV (Diagonal)} = \sqrt{500^2 + 280^2} = 573.06\ (mm)$$

Apply the following formula to calculate Working Distance of the camera

Focal Length x FOV = Working Distance x Sensor Size

$$\Rightarrow \text{Working Distance} = \frac{\text{Focal Length} \times \text{FOV (Diagonal)}}{\text{Sensor Size (Diagonal)}} = 382.04\ (mm)$$

## Step 2: Place the camera

Position the camera parallel to the working space and at a distance of 382 (mm)

## Step 3: Measuring the actual FOV

When the phone has been fixed, we begin to measure the actual FOV get the following parameters:

- FOV (Width) = 482 (mm)
- FOV (Height) = 270 (mm)
- FOV (Diagonal) = 552.47 (mm)

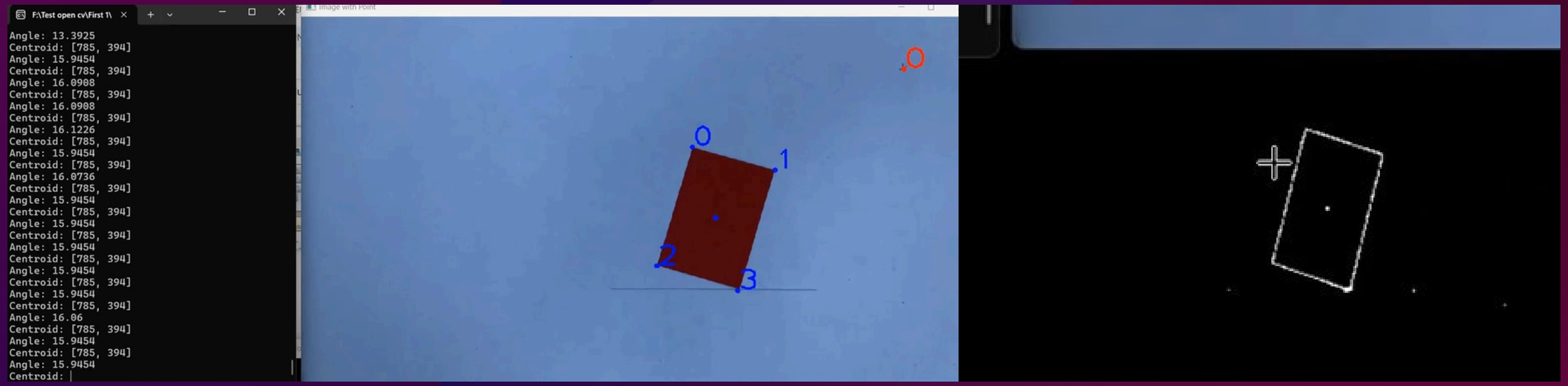
## Step 4: Comparing

Comparing between the actual FOV and our desired FOV from the beginning

| | Desired FOV (mm) | Actual FOV (mm) | Error (%) |
|---|---|---|---|
| Height | 500 | 482 | 3.6 |
| Width | 280 | 270 | 3.57 |
| Diagonal | 573.06 | 552.47 | 3.7 |

# Task 2

Chose a shaped object. Calculate pixel coordinate and angle of object.




## Step 1: Preprocessing

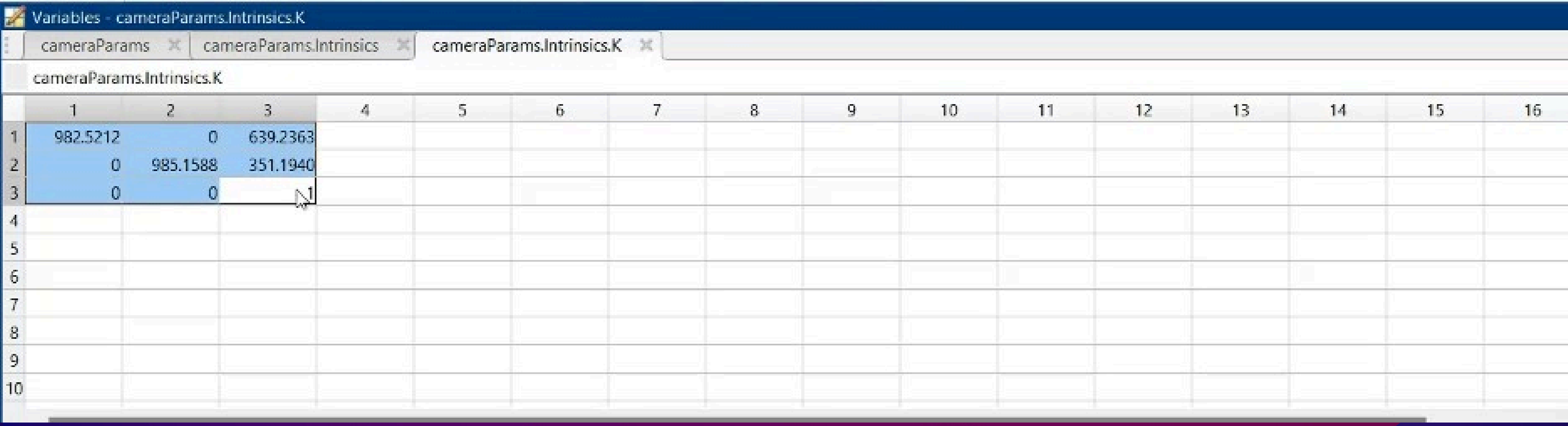
Convert from RGB to Gray Scale ⟶ Gaussian blur ⟶
Canny Edge Detection ⟶ Dilate ⟶ Erode

## Step 2: Feature Extraction

Find Contours ⟶ Calculate Moments ⟶ Find Contour Center
⟶ Calculate Contour Angle

# Task 3

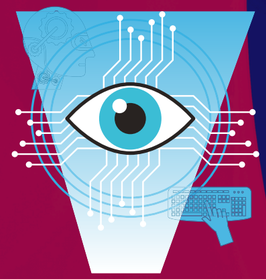
Perform camera calibration and calculate the real coordinate of object in world coordinate.




- Open Camera Calibrator app on matlab ⟶ Input 20 images of different orientation ⟶ Enter size of the checkerboard square = 19.5 mm
- The app will calculate the Intrinsic, Extrinsic matrix and other parameters from the input images.

# Task 4

Doing image processing to recognize feature of object and then calculate the dimension of object.



# Question 4

## Step 1: Reading frame from the camera

- Transfer image directly from camera to laptop as input frame

## Step 2: Preprocessing and Detecting contours in the frame

- A function is called to preprocess and detect the largest contour with 4 corners in the frame.
- Converting the image to gray scale ⟶ Gaussian blur ⟶ Canny edge detection ⟶ Dilate and Erode ⟶ Finding Contour

## Step 3: Point Reordering

- A function is called so that with the input point vector, it changes the order of the point based on their position.
- This ensures the desired position of the point doesnt's change no matter how the object is rotated.
- The points' order is put on the frame window.

## Step 4: Finding real world coordinates

- A function is used to convert image pixel coordinates to real world coordinates using parameters from the calibration step.

## Step 5: Calculating required values

- Calculates the dimensions of the object with real world coordinates
- Compute the distance from a predefined origin to the centroid in real world coordinate.
- Display these value to the terminal.

# COMPARING ACTUAL AND CALCULATED VALUE

| | Actual Value (mm) | Calculated Value (mm) | Error (%) |
|---|---|---|---|
| **Width** | 90 | 91.0381 | 1.15 |
| **Height** | 60 | 60.8598 | 1.41 |
| **Distance from origin** | 130 | 129.751 | 0.19 |