LAB 1: ERROR ANALYSIS AND ORIENTATION

Evan Chan, Kate Oliver, Kevin Lei

Texas A&M University
College Station, TX 77843, US.

Abstract
This Lab report covers error estimation and propagation. A camera was used to track the position of a rectangle as it slid across a table. The position of the corners of the rectangle were traced and the position data from those points was used to estimate the length, width, and area of the rectangle. Error propagation was then used to find uncertainty in the area measurement, which was then compared to individual measurements. The goal of this lab is to find the uncertainty in measurements taken by the tracking camera.

Keywords: *Error propagation, measurements, uncertainty, vectors*

1. Introduction

The goal of the lab was to determine the area of the given rectangle with data from the tracking camera and determine margins of error within the tracking camera's measurements. By taking a large number of position data sets we can correct for error and get much more precise and accurate measurements. The data the camera collected was the position vectors of the three corners which allows us to calculate the area. The camera isn't 100% accurate, so we will have to calculate the statistical and propagation errors for the length, width, and area.

The lab computer generated a csv file containing position data for the pink, orange, and green dots on the tile for as long as we ran the program. We ended up with about a thousand measurements, and 632 of those measurements were complete with position data for all three dots.

Since the tile is a square, we determined that the length is the distance between the orange dot and green dot, while the width is the distance between the pink dot and the orange dot.

In order to calculate the area of the square we first need the length and width of the square, we need to subtract the position vectors of one corner from the other to get the length and width (***Equation 1***) . From there we are able to use the distance formula(***Equation 2)*** to find the distance between the two corners.

In order to calculate the error for the area we must understand the different types of errors. Standard error is how much discrepancy the sample data points have from the data points . For standard error, it is the standard deviation, or the average distance of each data point from the average, over the root of the number of data points (***Equation 3***). The propagation of error is used to accurately calculate the error when adding, or multiplying data points. The equation is the square root of the sum of each data point's errors divided by the data point squared (***Equation 4***).

Equation 1: $<x_{(n+1)} - x_n,\ y_{(y+1)} - y_n>$

Equation 2: $d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$

Equation 3: $\delta f = \frac{\sigma f}{\sqrt{n}}$

Equation 4: $\delta f(x,..., y) = \sqrt{(\frac{\partial f}{\partial x}\delta x)^2 +... + (\frac{\partial f}{\partial y}\delta y)^2}$

2. Experimental Procedure

First we set up the experiment to slide a square with three stickers shown below in ***figure 1***, using an air table to reduce friction, a camera to track the stickers, and a laptop to run the program to collect data.

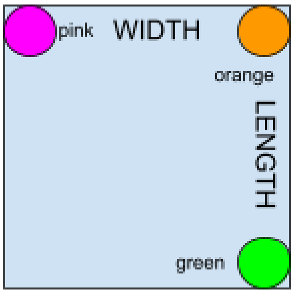

Figure 1: Rectangle with tracking dots, length, and width marked

Next we collected the data into a csv file to later be used in python to calculate the length, width, and area. First we organized the data into an array, skipping cases that were missing data points.

```
# creating new array with complete data only
dataSorted = []
for i in data:
    add = True
    for j in range(len(i)):
        if not i[j]:
            add = False
            break
    if add:
        dataSorted.append(i)
dataSorted = dataSorted[1:]
```

Figure 2: Code used to organize the data

After organizing the data, we calculated the width and length of the square using the distance formula ***(Equation 2)*** from one corner to the other and stored it into an array to later be used to calculate the average, standard deviation, and standard error of the data (***Equation 3***).

```
# pink to orange
widths = []
# calculating and storing the widths into array
for i in dataSorted[1:]:
    width = math.sqrt( (i[4]-i[2])**2 + (i[5]-i[3])**2 )
    widths.append(width)
avgWidth = sum(widths)/len(widths)
print(f"Average width: {avgWidth}")
```

Figure 3: Code used to calculate the width of the square at each case, and the average width

Next we multiplied the average width and length and propagated the calculation to get the propagated area by using the error of the length and width, their averages, and the average area in the propagated error formula (***Equation 4***).

```
#propagated error of area using average l and w
avgArea = avgLength * avgWidth
print(f"Average area using average len * width: {avgArea}")
avgAreaError = avgArea * math.sqrt( (stdErrLengths / avgLength)**2 + (stdErrWidths / avgWidth)**2 )
print(f"Error: {avgAreaError}")
```

Figure 4: calculating the propagated error of the area

The last calculations we had to make were for the area of the square for each case and the standard error for the average of all of those areas. First we calculated each area and made an array for them, next we used the array to calculate the standard deviation and standard error (***Equation 3***) for the average area.

```
# area from average of sum of areas + uncertainty
areas = []
for i, k in enumerate(length):
    areas.append(k * widths[i-1])
statAvgArea = sum(areas)/len(areas)
print(f"Average of individual areas: {statAvgArea}")

temp = 0
for i in areas:
    temp += (i - statAvgArea)** 2
stdevAreas = math.sqrt( (1/len(areas)) * temp )
print(f"Error: {stdevAreas/math.sqrt(len(areas))}")
```

Figure 5: calculating the standard error of area

3. Results and Analysis

Table 1: Average width, length, and area values with uncertainty

Value	**Measurement**	**Uncertainty**
Width	181.503 pixels	2.116 pixels
Length	182.553 pixels	2.096 pixels
Area (propagated)	33133.980 pixels	21.582 pixels
Area (statistical)	33134.463 pixels	23.430 pixels

After calculating the following width, length, and area values, we found them to be fairly in line with expectations. Since the tile is a square, it made sense that our length and width calculations were almost the same. The area was found using two methods: by multiplying the average length and average width, and by multiplying each length and width measurement and taking the average of all of them. These two methods produced similar but not identical results, which falls in line with our expectations.

We also observed relatively small uncertainty values, which makes sense due to the large size of data that was collected. The accuracy of our uncertainty is confirmed by the histograms shown below. The length and width calculations follow a rough normal distribution which shows that most measurements are close to the mean with a few measurements on either extreme of the distribution.

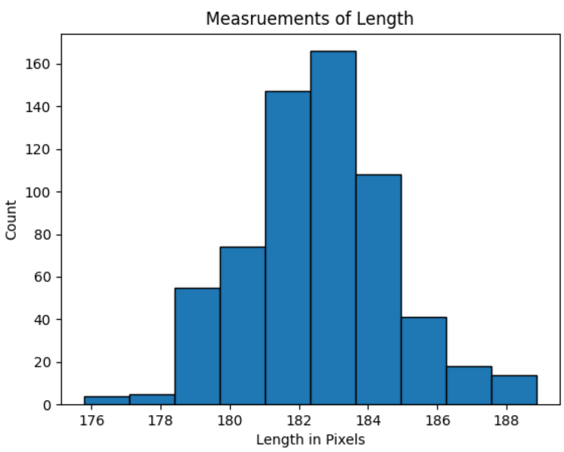

Figure 6: Histogram for measurements of Length

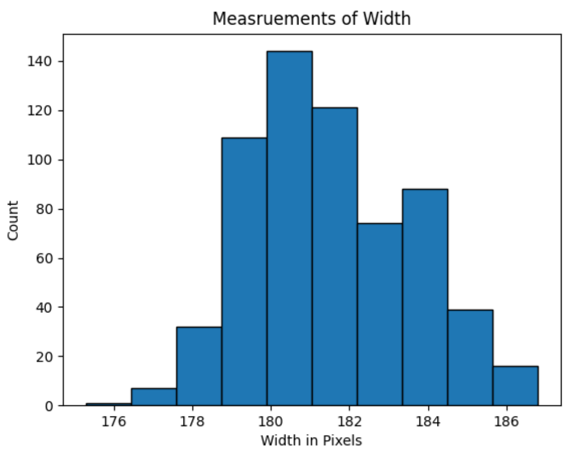

Figure 7: Histogram for measurement of Width

4. Conclusions

In summary, this experiment involved taking many measurements of position data for three colored labels on a floating square tile. This position data was then used to calculate the length and width of the tile using the distance formula (**Equation 2**). Area was then calculated using length and width. Uncertainties were found for all calculations using the appropriate propagation formulas (***Equations 3 & 4***) Plotting the length and width data produced a normal-like histogram matching the calculated means and uncertainties.