Analysis

The code for the first function has a O(1) time complexity.

The item contains a for loop that runs through the 0- number of barcodes in the fil. This allows it to iterate between a dataset of 100 images. This will leave the time complexity at. Otherwise, the algorithm also contains a function that runs through the dataset with a time complexity of n. Since it iterates through the loop and calls the first function multiple times. Each time it calls the function it has a complexity of n. Then it makes the whole function have a time complexity of O(1)\*O(n) which by the principle of Big O notation takes on the item O(n)

We found by using 4 projections

P1: 0

P2:45

P3:90

P4:135

We found the hit ratio to be 61%

The hit ratio can be increased to upward or 68% by using up to 7 projections

Graphical user interface, text

Description automatically generated

0,30,…up to 180 degrees can allow us to obtain at least a 68 percent hit ratio

If we use up to 13 projections The hit ratio unfortunately reduces to 66 percent. This may be due to to many projections

Text

Description automatically generated

Lets try 10 projections

Another decrease to 64 %



Thus when investigating the affect of number of projectijons on hit ratio we feel like the most optimal number of projection s for a given hit ratio is 7.

Cropped to 6 by 25 with a 19 by 19 np matrix

A picture containing icon

Description automatically generated

Cropped to 6 by 26 with a 20 by 20 np matrix

Text

Description automatically generated

Interestingly enough the hit ratio for the 6 by 26 was higher than the 6 by 25

We feel like this might be due to the small difference in area allowing for an inverse result that allows the hit ratio to be slightly higher.

The optimal cropping and size of np matrix is 6 by 26 cropping and 20 by 20 np matrix.

After this investigation we have concluded that a

6:26 and a 20 20 matrix be used in our project.