COMPUTER VISION
MID TERM ASSIGNMENT
SAGAR GHIMIRE

PROJECT TOPIC: VIOLENT AND NON-VIOLENT SITUATION DETECTION

- The purpose of this project is to detect violent or non-violent situations from the images using Computer Vision.
- The datasets are taken from various sources such as Kaggle and Roboflow.com. Some data are images and some are in video form. Different images will be taken from videos for the project.

Resources for dataset:

- 1.Kaggle.com(images and videos)
- 2.Universal roboflow.com(images)

- Task performed
- 1. Enhancement
- 2. Sharpening with sobel
- 3. Segmentation of POI using k-mean and mean shift
- 4. Binarized the segmented
- 5. Multiply the binarized with enhanced and sharpened image
- 6. Calculate the hist of POI for one of the image from the particular class(ready and not ready)
- 7. Compare the hist value with other images POI using
  - i. Chi-squared
  - ii. correlation coefficient

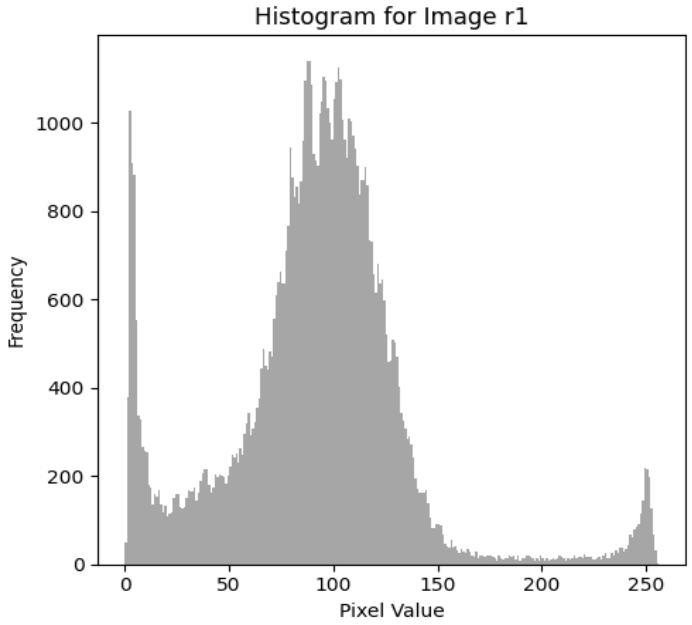
8. CONCLUSION

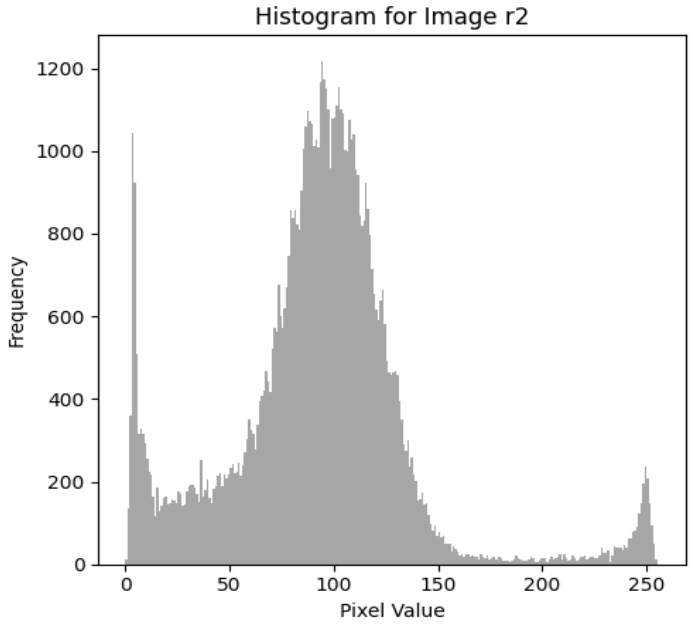
1.Input your project database images, including both good and noisy images.

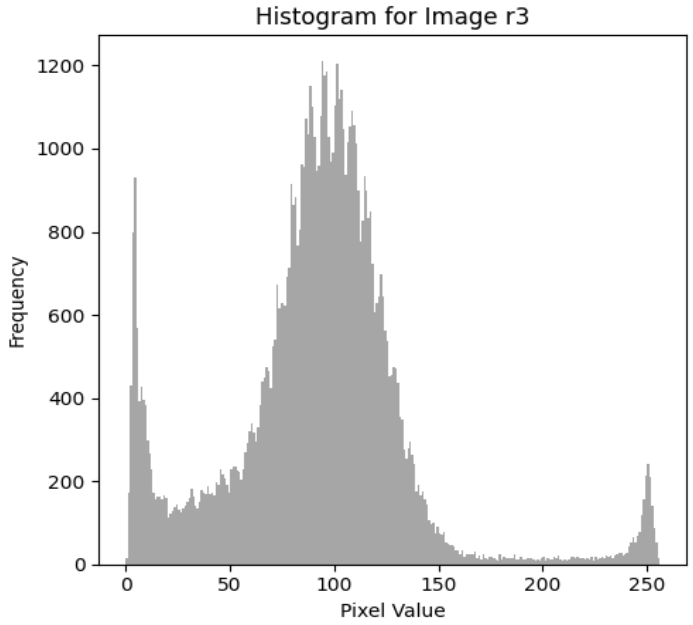
Since, the project asks to work with gray image we are only posting gray image.

In this project we are using histogram-based image recognition or classification method to recognize ready or not ready for gun firing. All the images are taken from project images which are similar with different position of body. Total six images taken.

Ready images

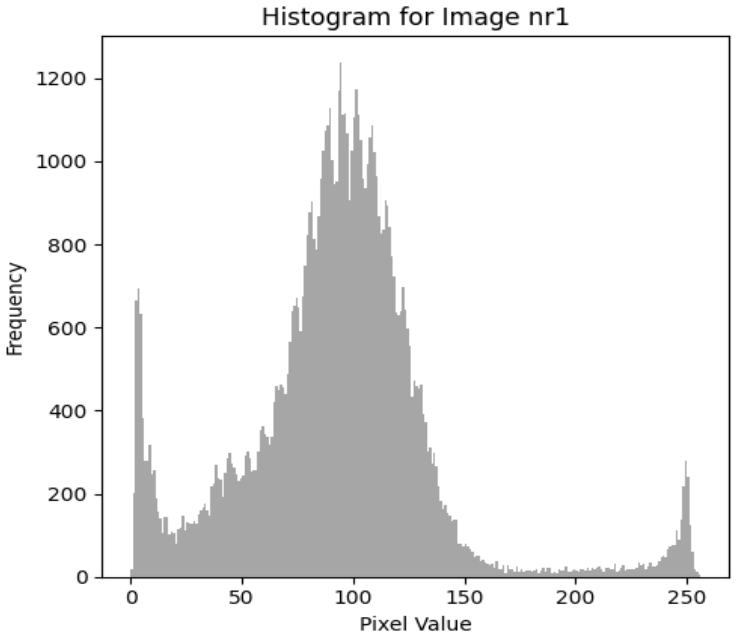


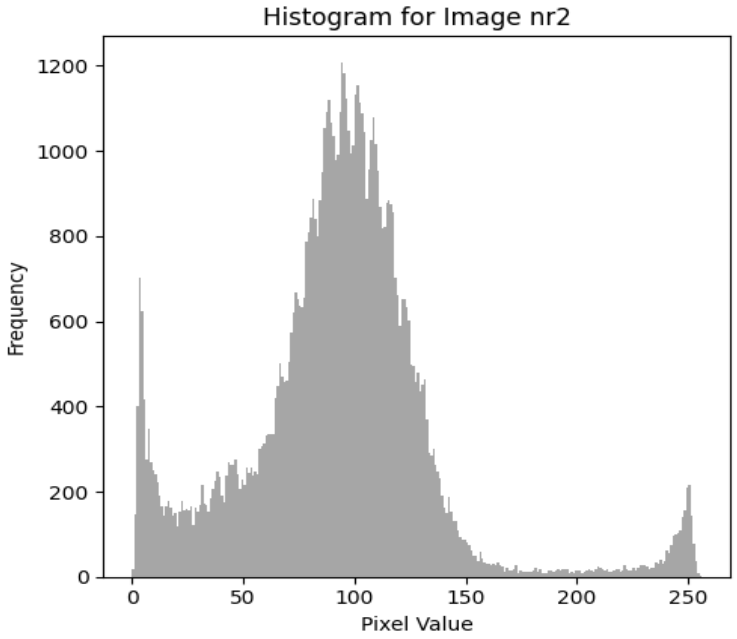


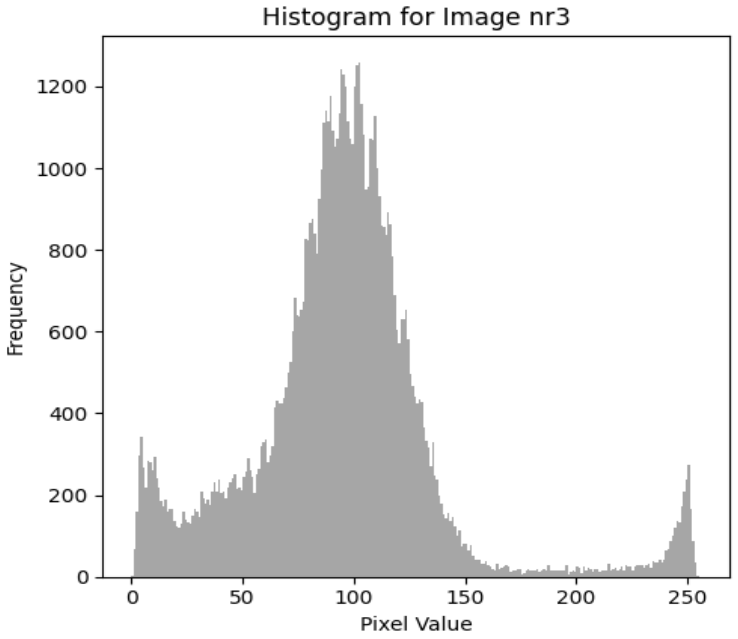


Histogram for ready images

Not ready







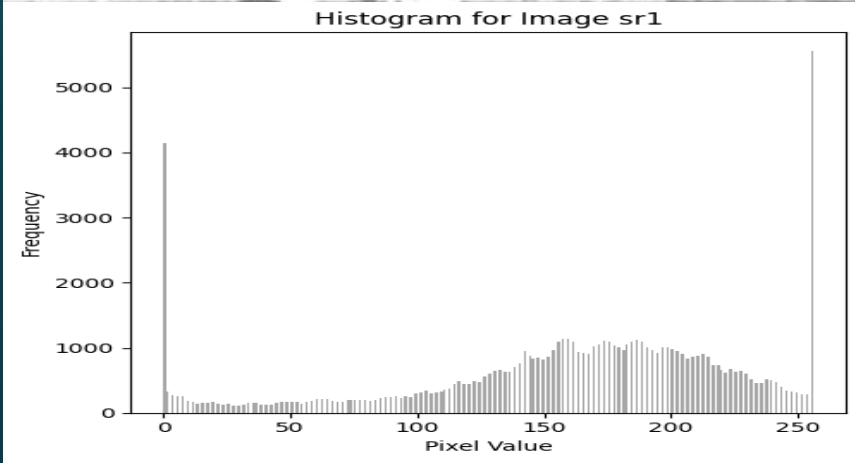
Hist for not ready

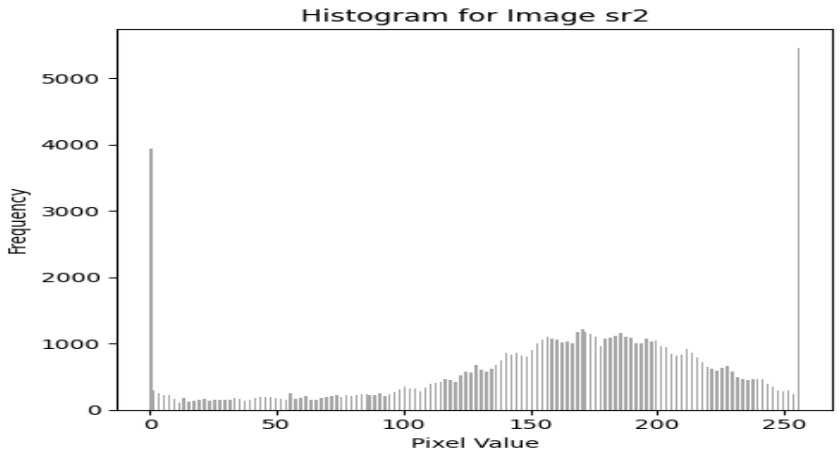
2.Apply image restoration and enhancement methods to improve the quality of the images. Evaluate and choose the best methods based on their effectiveness.

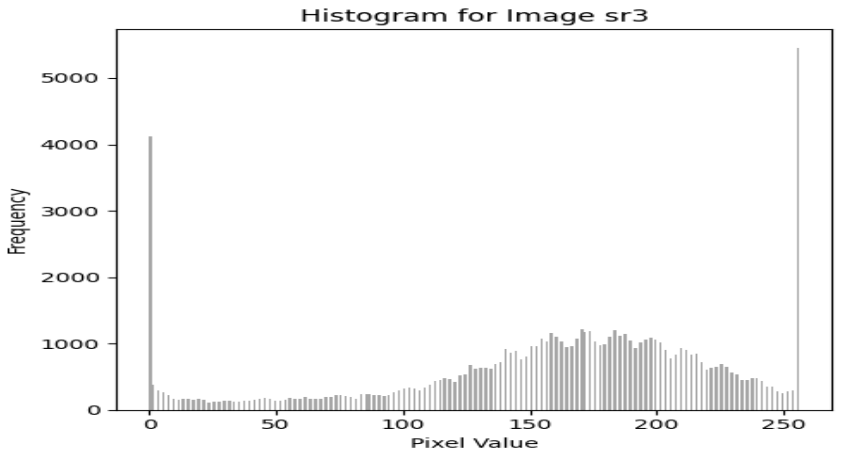
Now, we are going for image sharpening and enhancement process, since the image of slightly less quality we will apply percentile based and histogram-based methods to enhance the images.

First applied bit plane slicing to see if there is any noisy plane, but all the 8 plane slices had some info of the image, so we dropped that idea.

Percentile based contrast stretching per_min=5, per_max=**93** **ready**

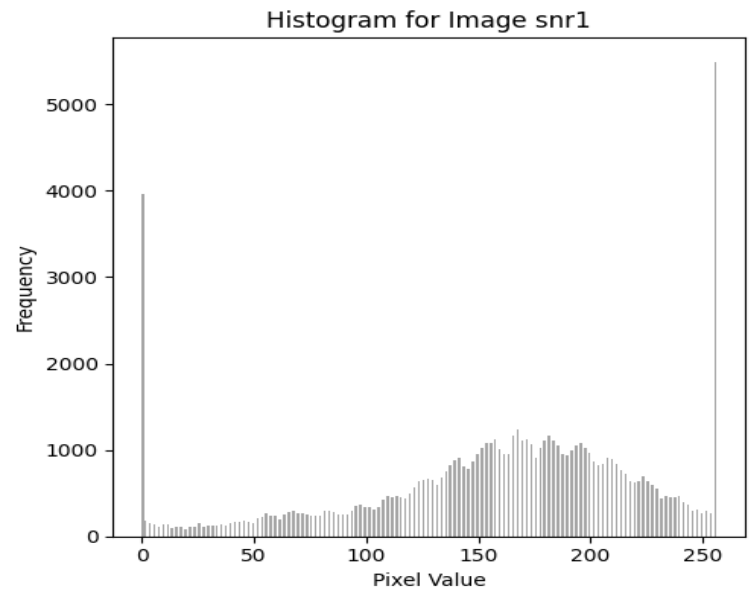


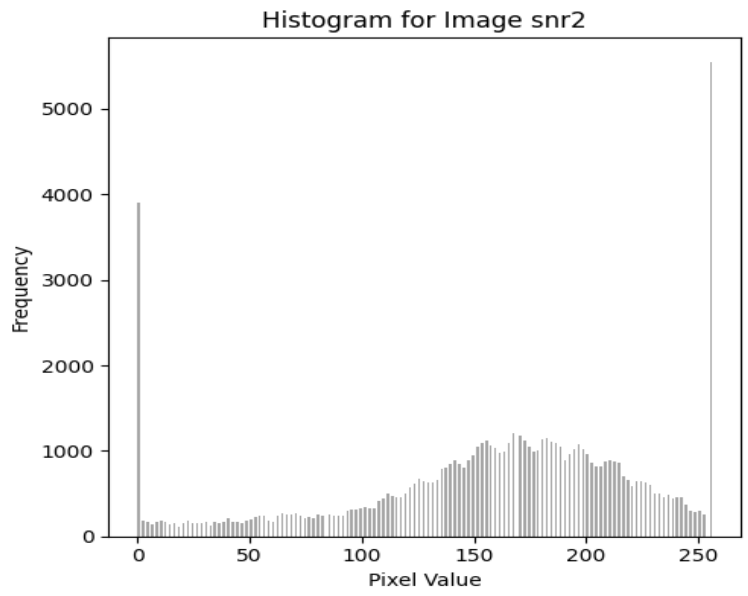


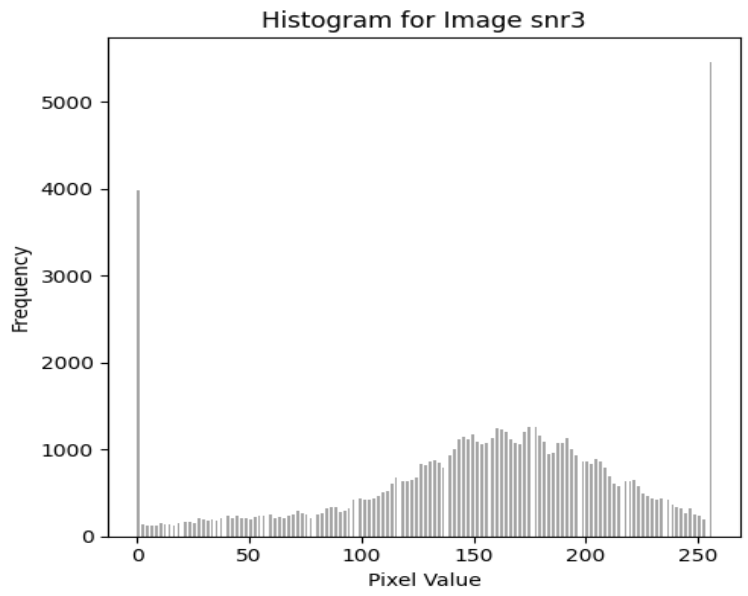


Not ready

Percentile based image stretching

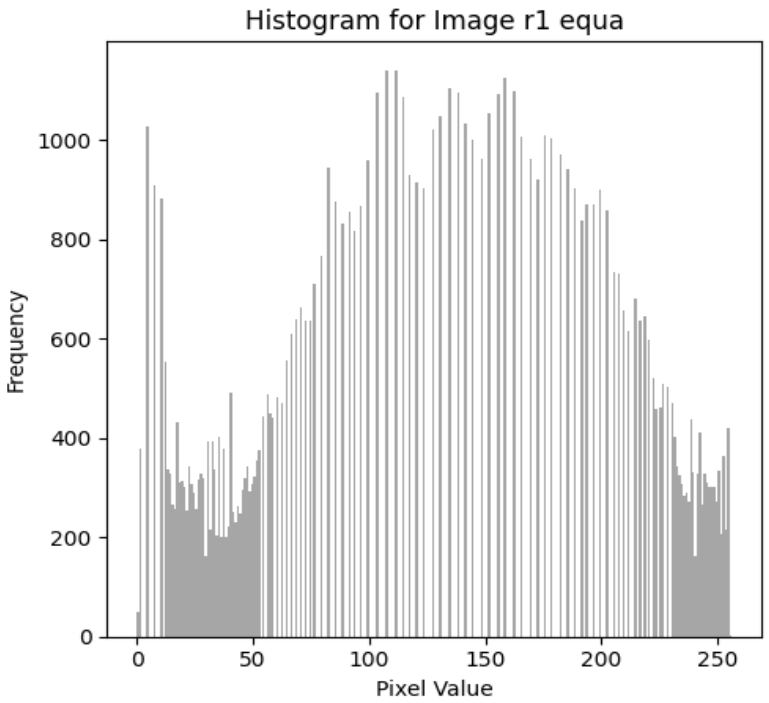


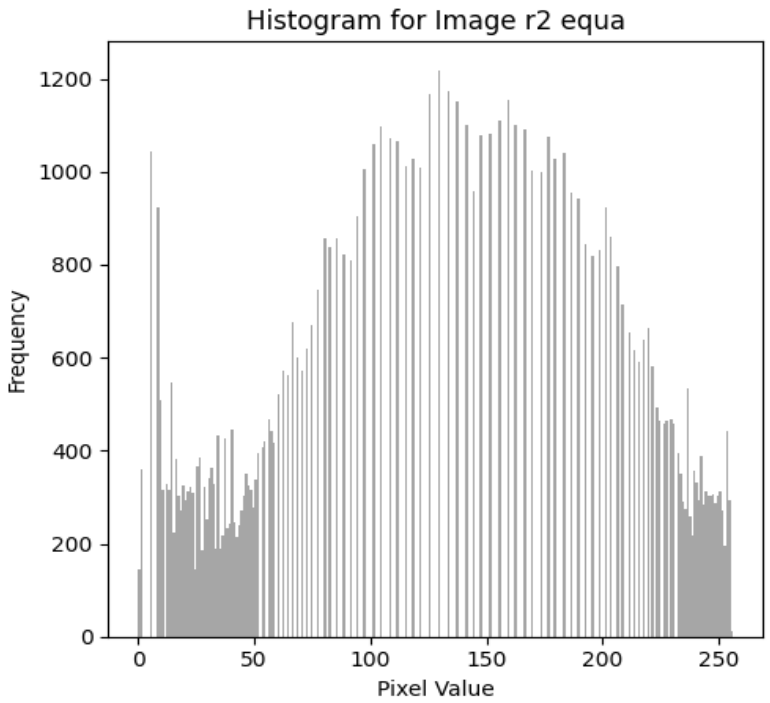


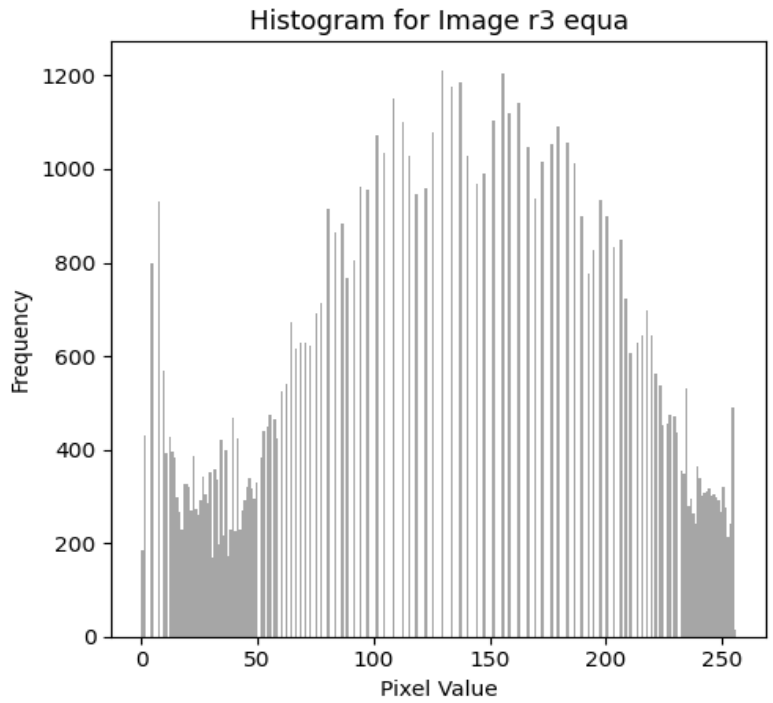


Histogram based image enhancement

Ready 1 ready 2 ready 3

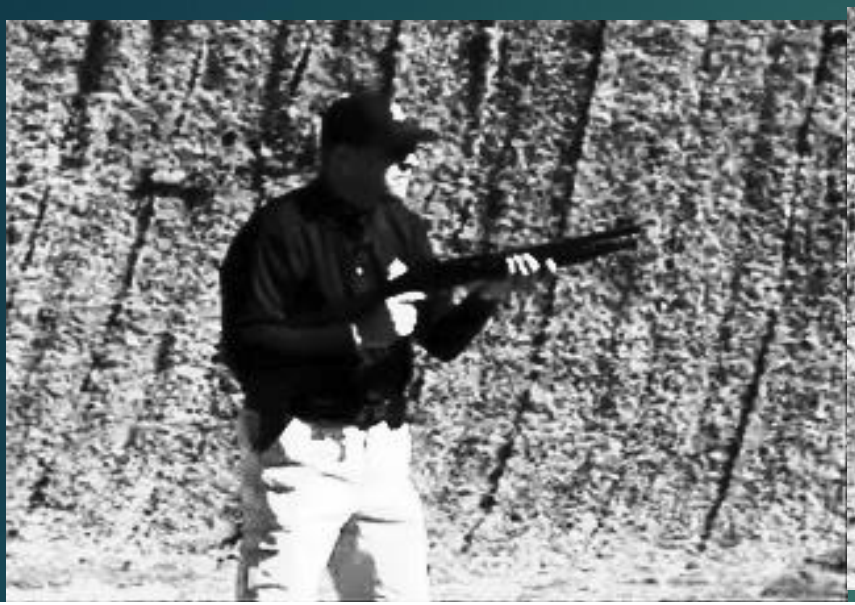


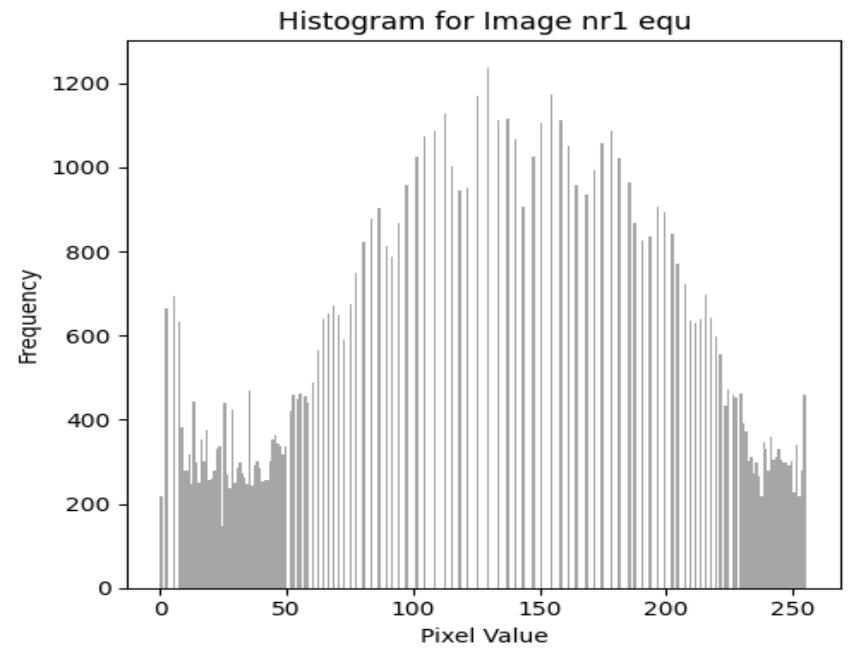


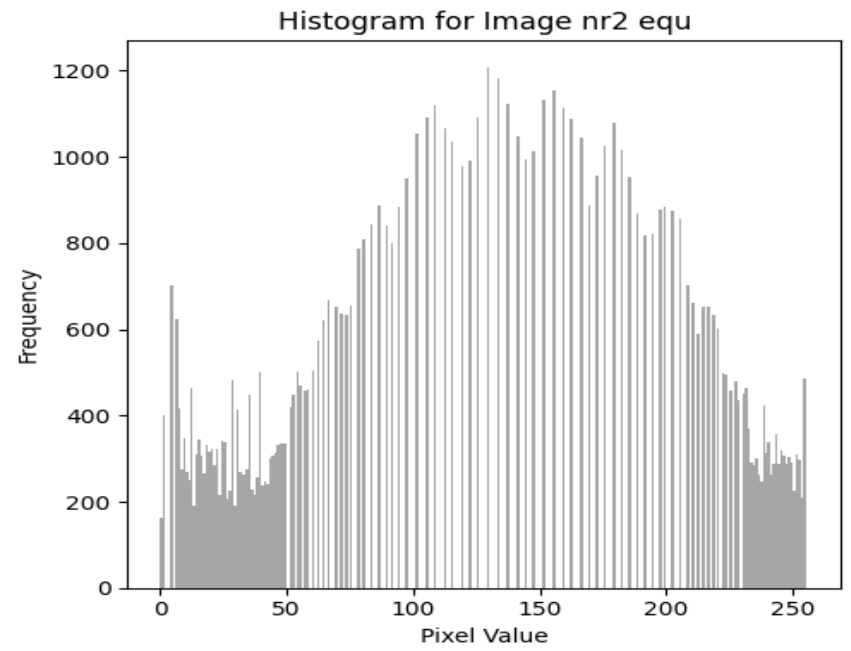


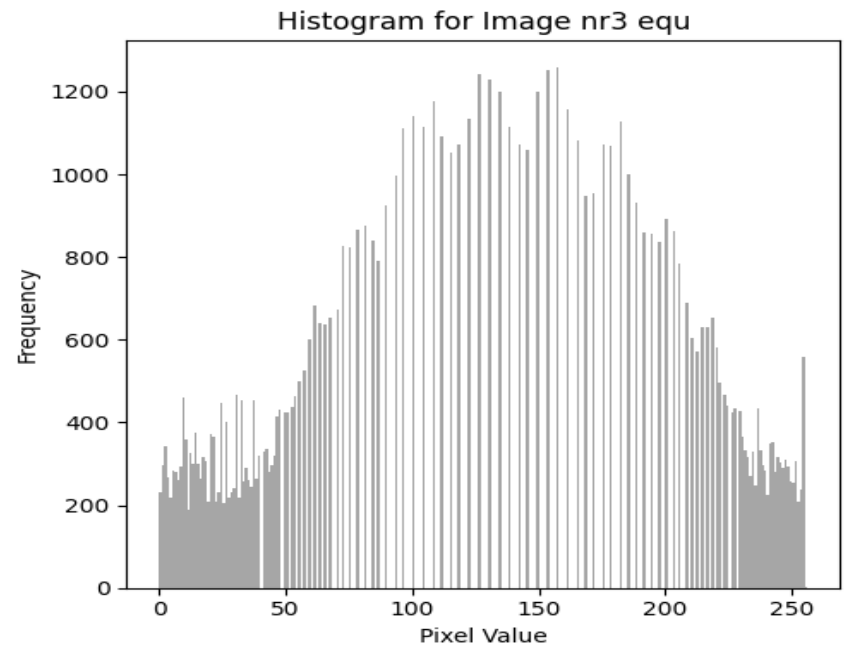
Histogram based for not ready images

Not ready 1 not ready2 not ready 3









NOTE:

Out of percentile based and histogram based, hist based seems to be better than percentile based, so taking histogram based equalization enhancement for sharpening processes.

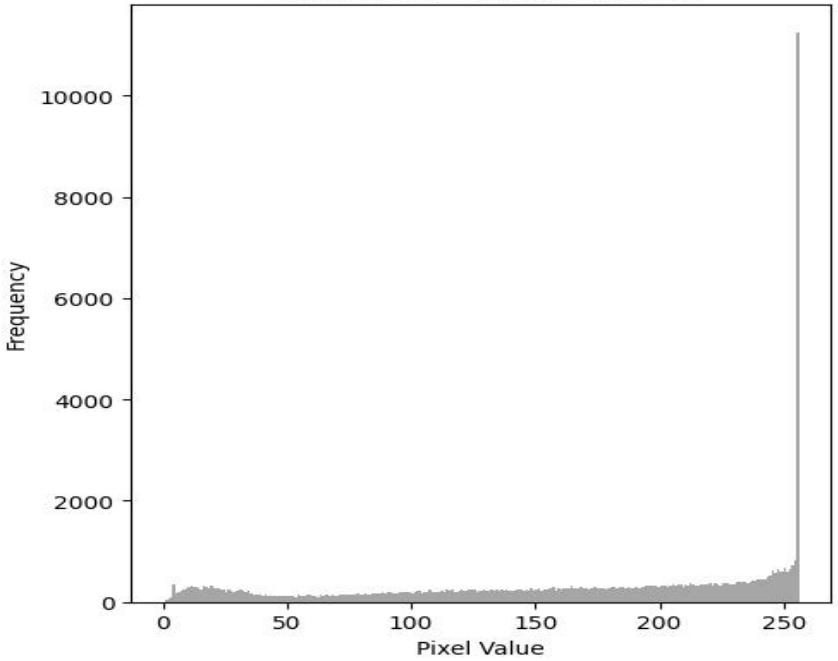
Image sharpening for ready images using sobel method sharpening_strength=0.2

Sharpened ready Image 1

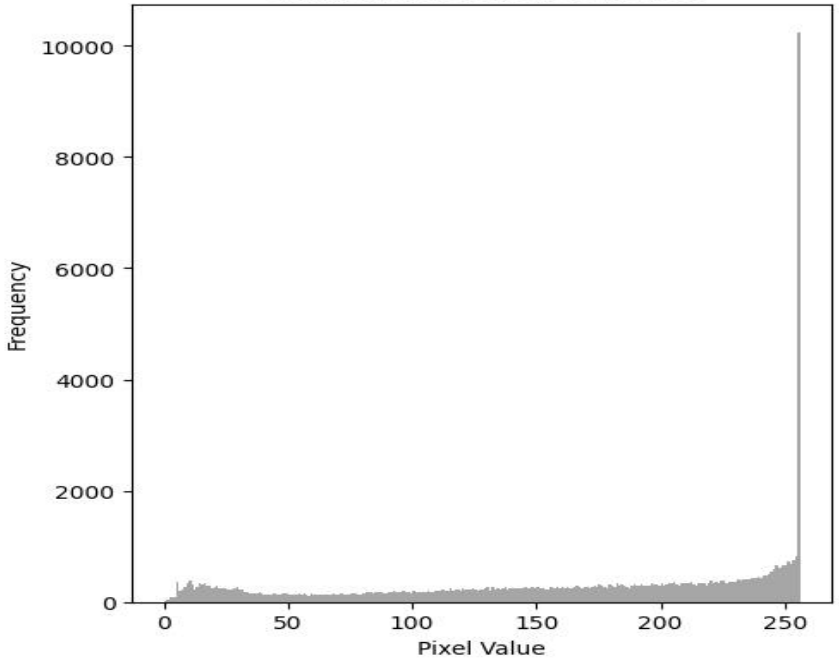
Sharpened ready Image 2

Sharpened ready Image 3

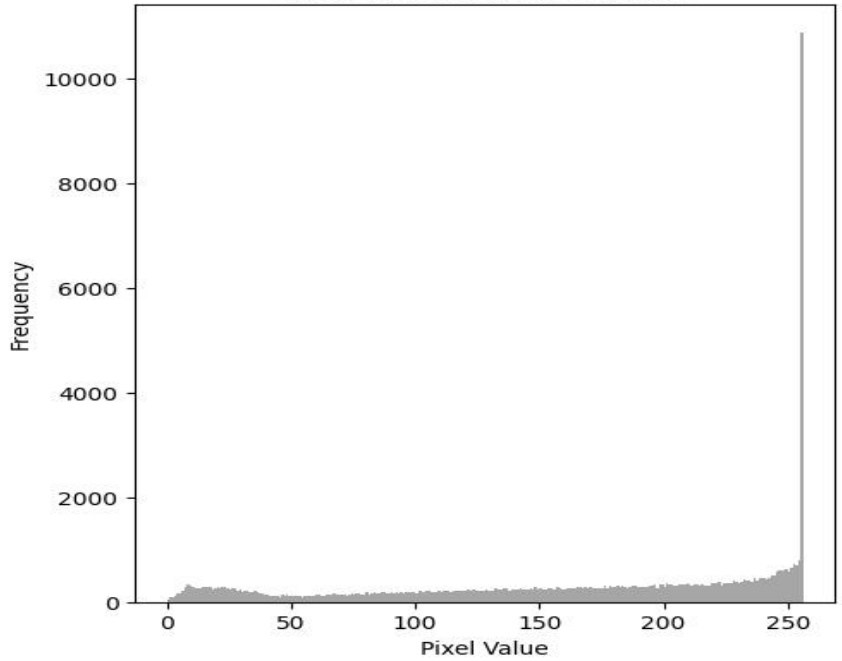
Sharpened ready Histogram 1



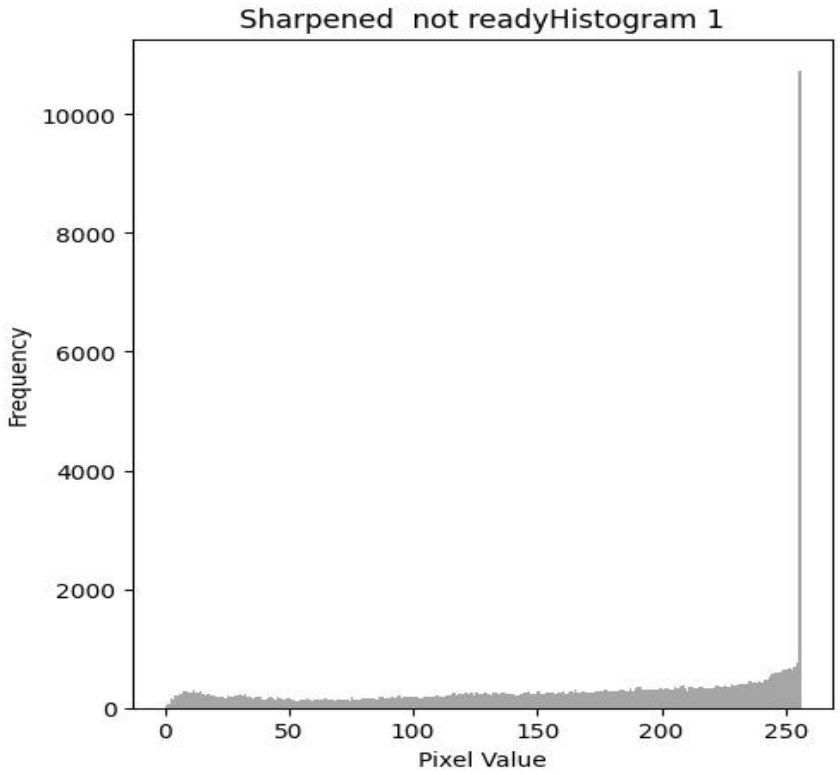
Sharpened ready Histogram 2

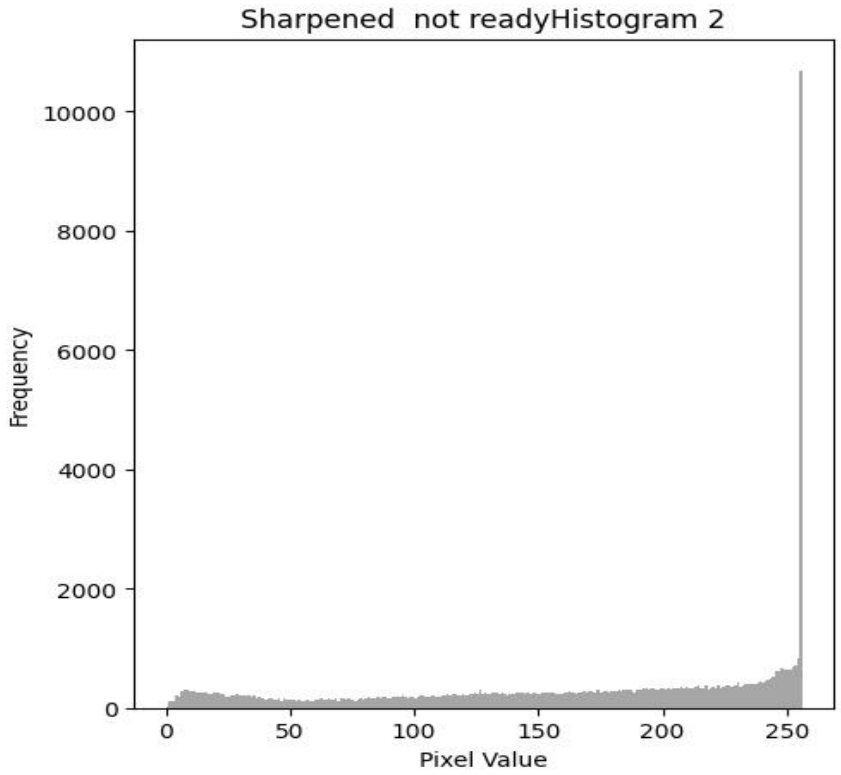


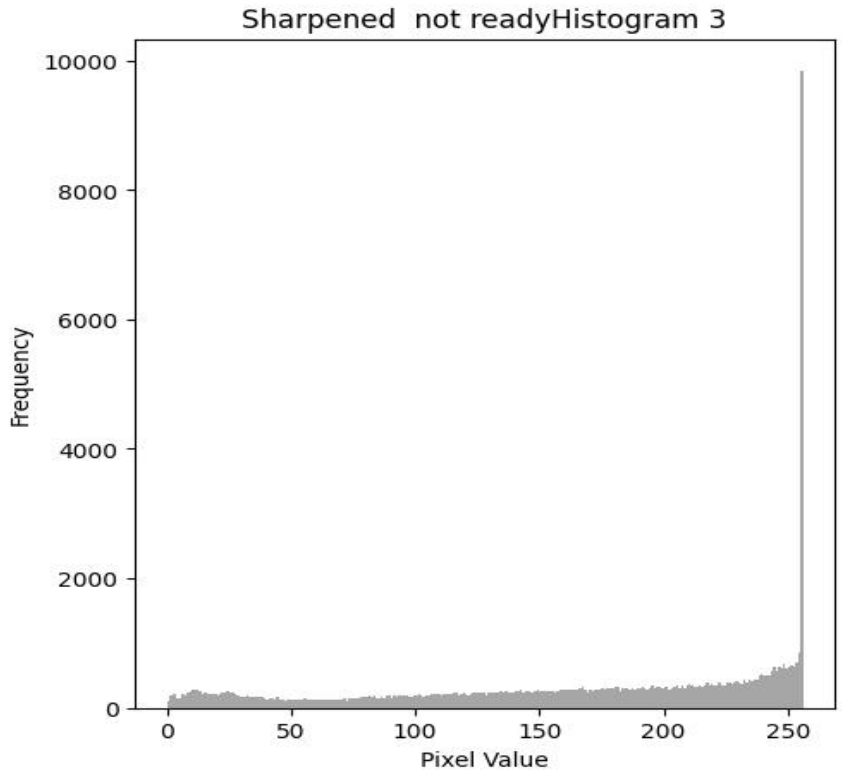
Sharpened ready Histogram 3



Sharpening for not ready images using sobel sharpening_strength=0.2







3. Apply image segmentation methods to separate the objects of interest from the background. Evaluate and choose the best method based on its accuracy.

Ready

(x,y)=(130,50)
Square size=90

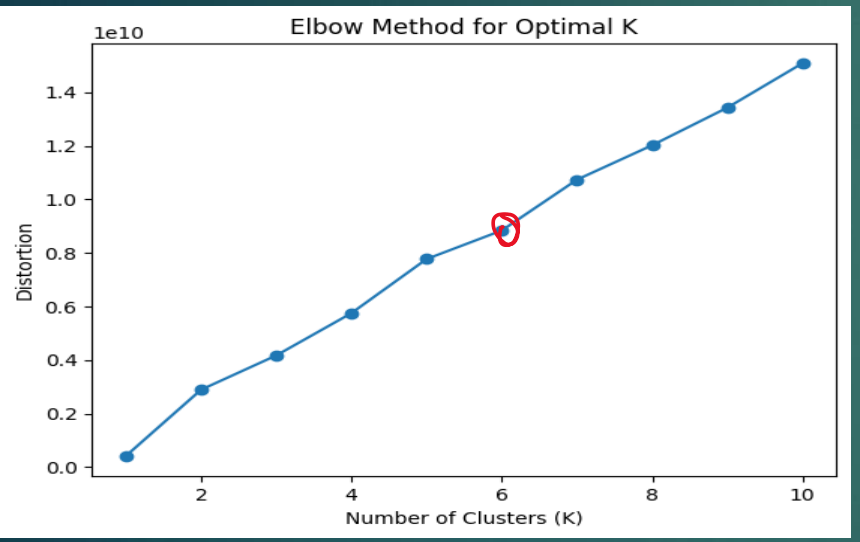
Not ready

First applied the square box to show the POI of interest now using 2 types of segmentation method

1. k-mean clustering
2. Mean shift filtering

K-mean clustering(elbow method

For this taking image 1_ready as a best image to find the k value



From image we can see that there is a bend at 6 making an elbow, taking 6 as k values.

Applying k mean clustering to POI

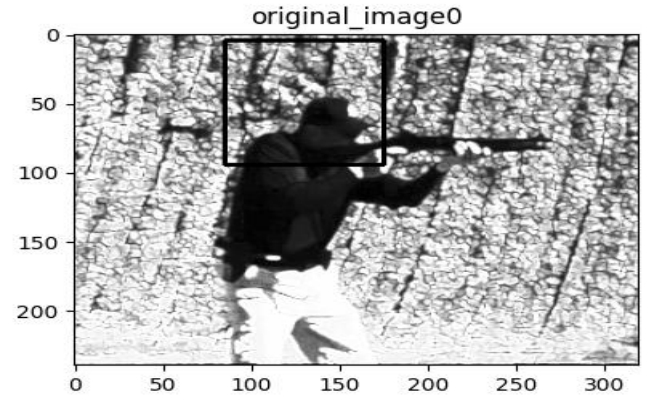
K-mean clustering segmented images all 6 images

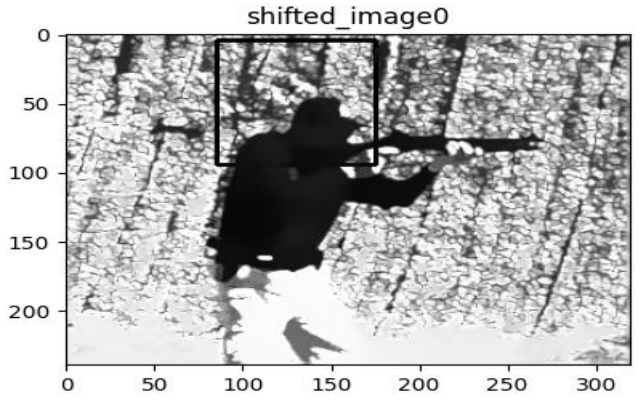
```
K_kmeans = 6  # Number of clusters
_, labels_kmeans, centers_kmeans = cv2.kmeans(Z_kmeans, K_kmeans, None, criteria_kmeans, 10, cv2.KMEANS_RANDOM_CENTERS)
segmented_kmeans_roi_3 = centers_kmeans[labels_kmeans.flatten()].reshape(result_image_3.shape)
```

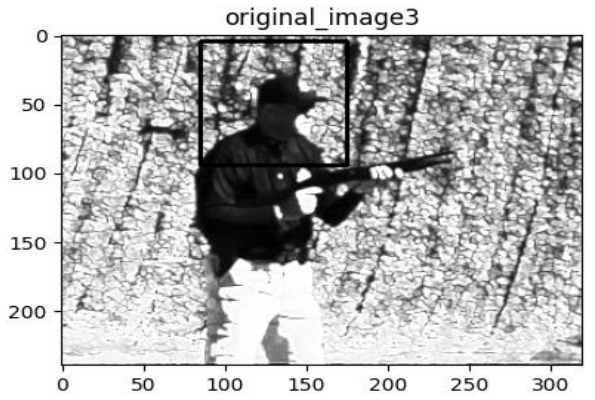
Ready

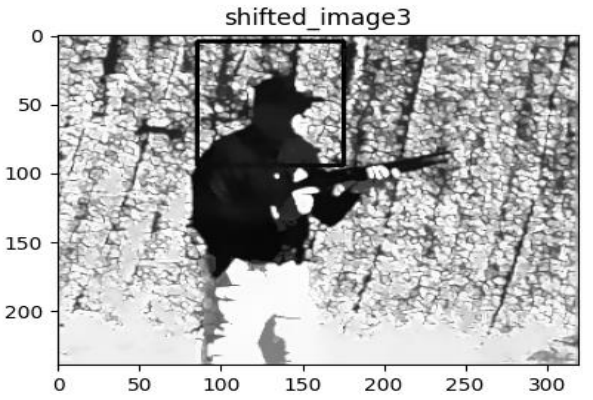
Not ready

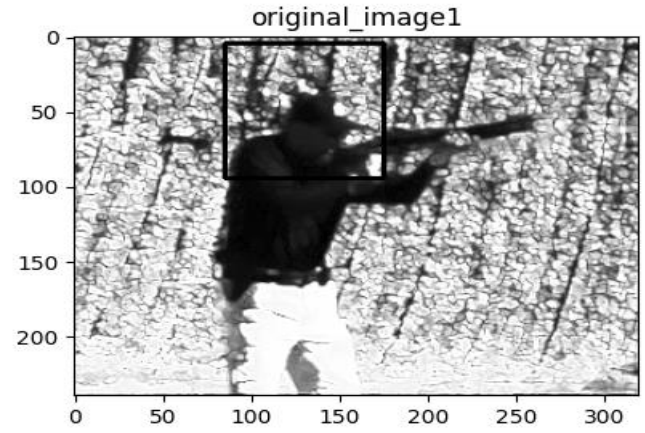
Applying second segmentation method which is Mean shift filter

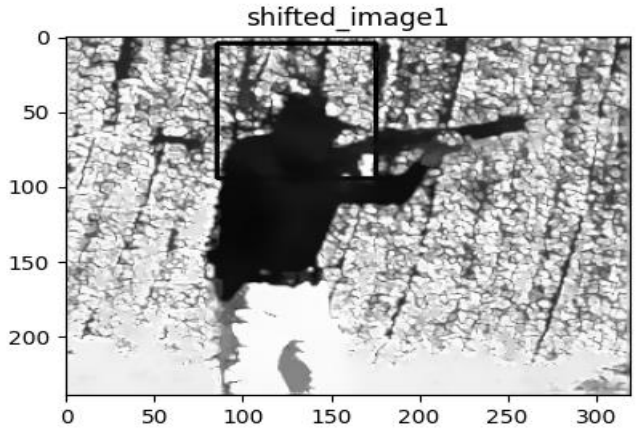


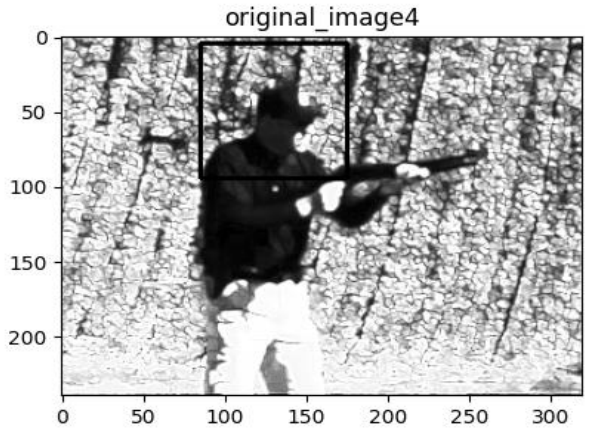


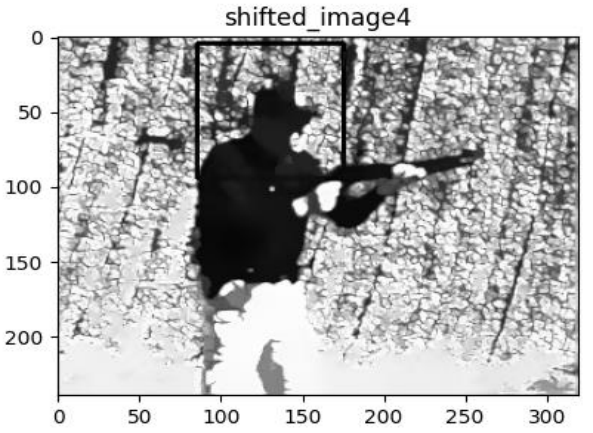


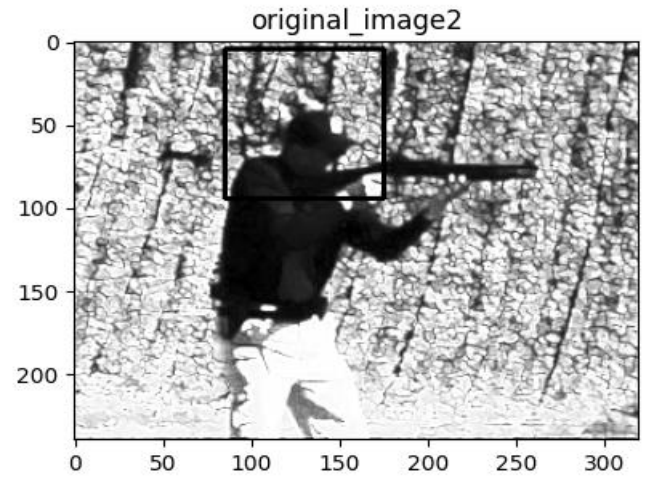


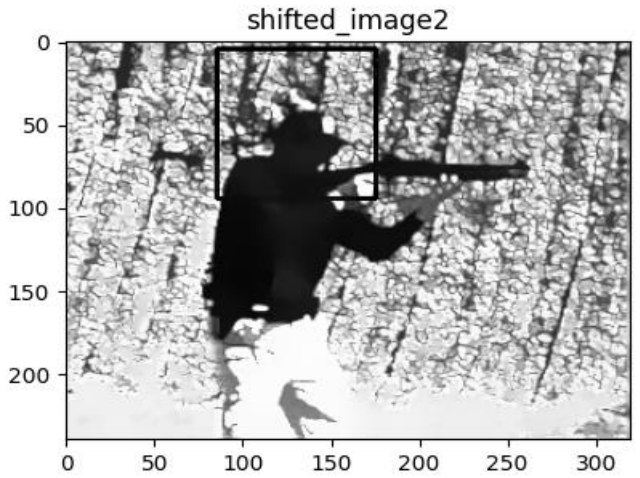


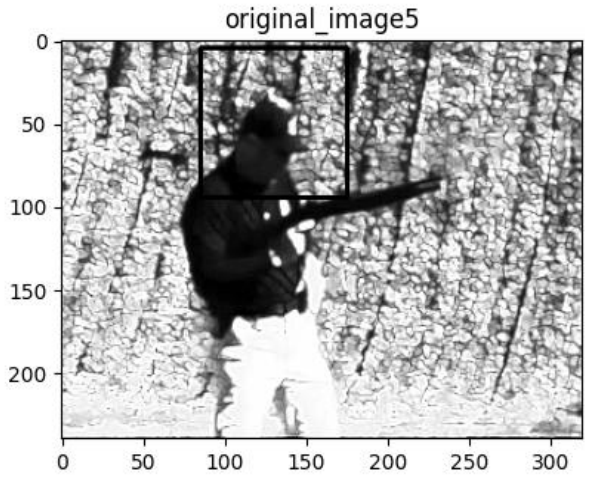


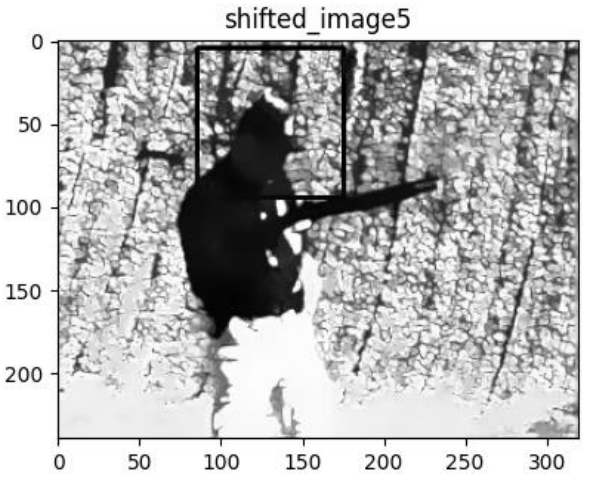










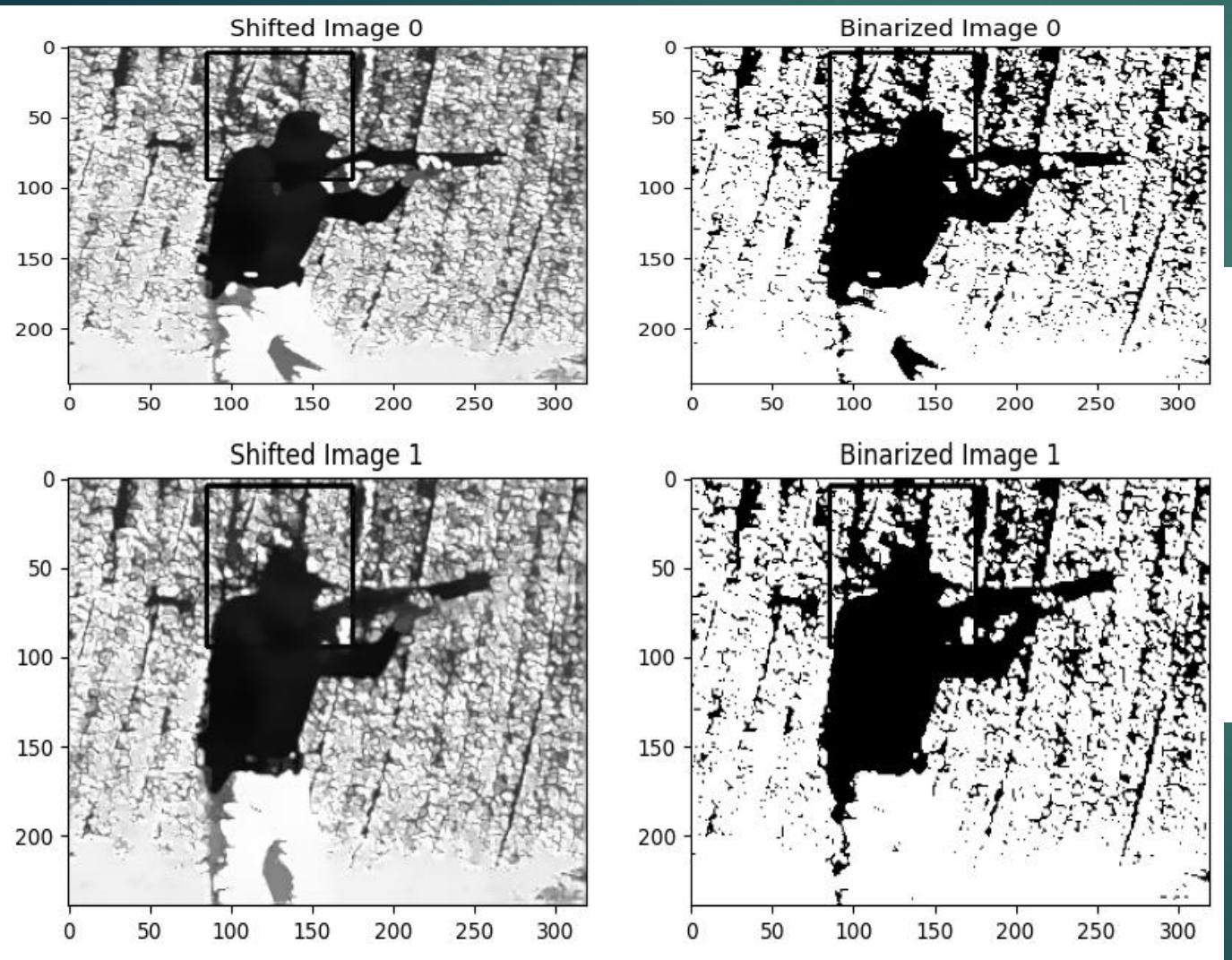


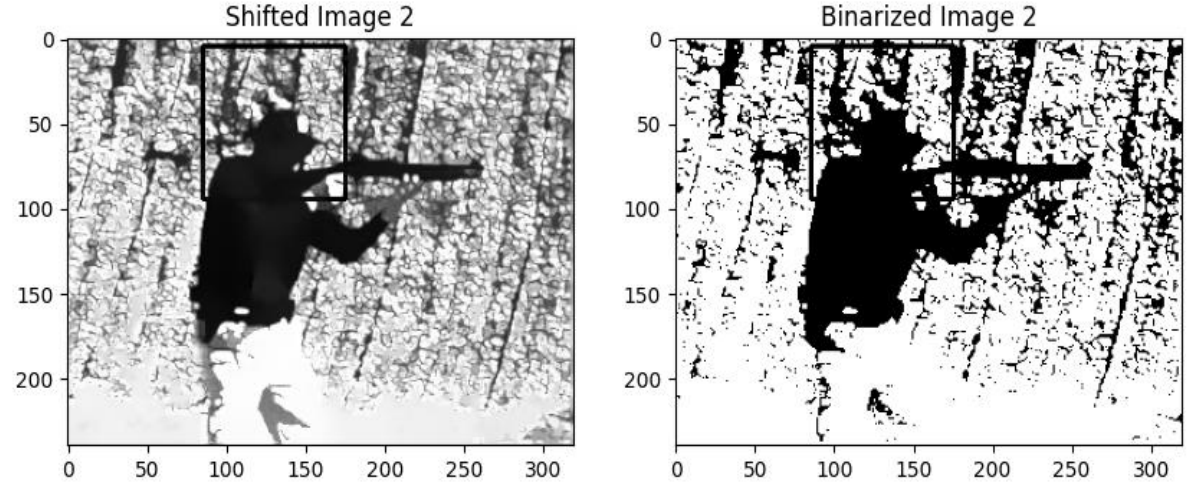
NOTE :Taking mean shifted segmented images as it seems to have more detail about the POI

4.Multiply the best-enhanced image with a binarized image to obtain a segmented image.

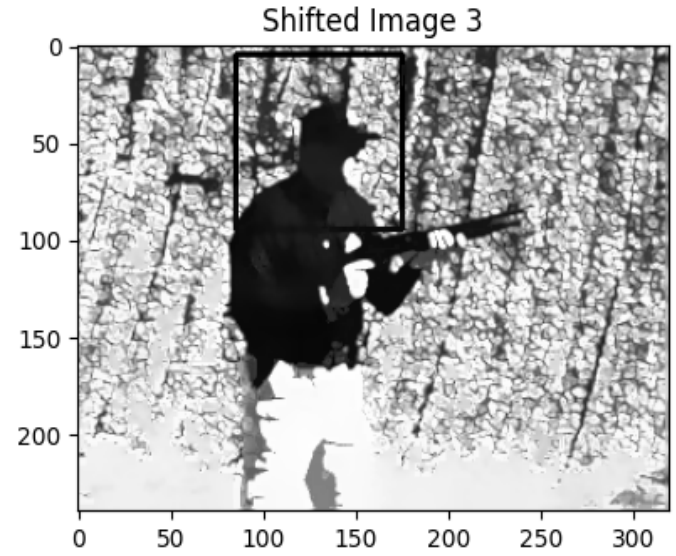
```
# Apply a thresholding method (adjust the threshold value as needed)
_, binary_image = cv2.threshold(shifted_image_gray, 127, 255, cv2.THRESH_BINARY)
```

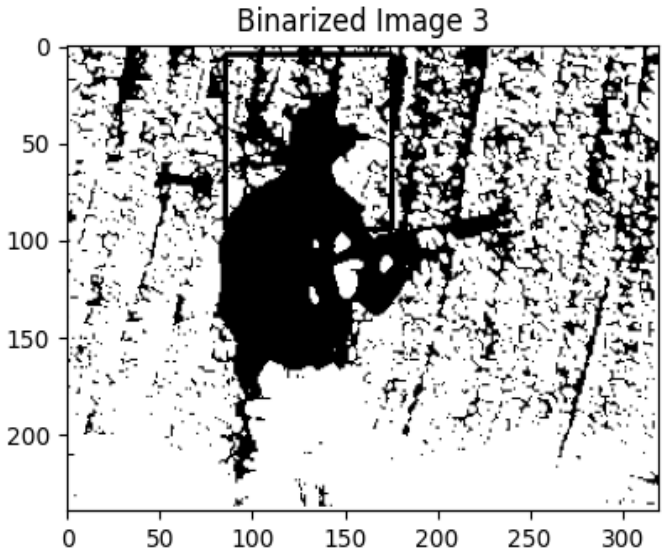
Shifted image means mean shifted segmented

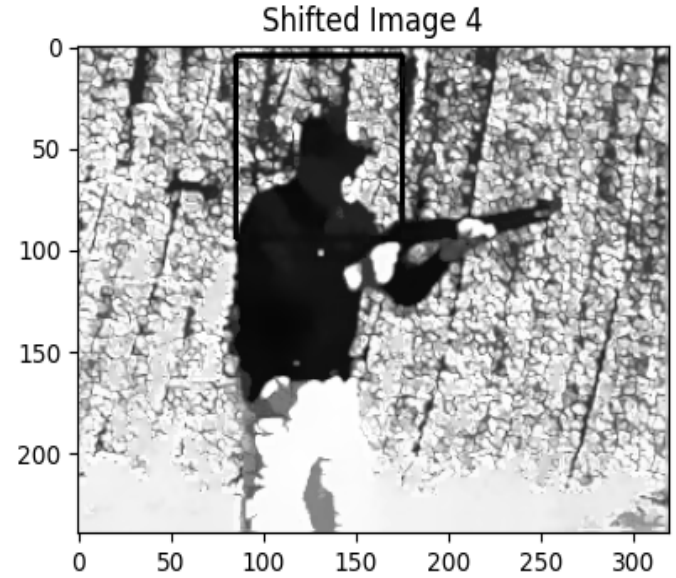


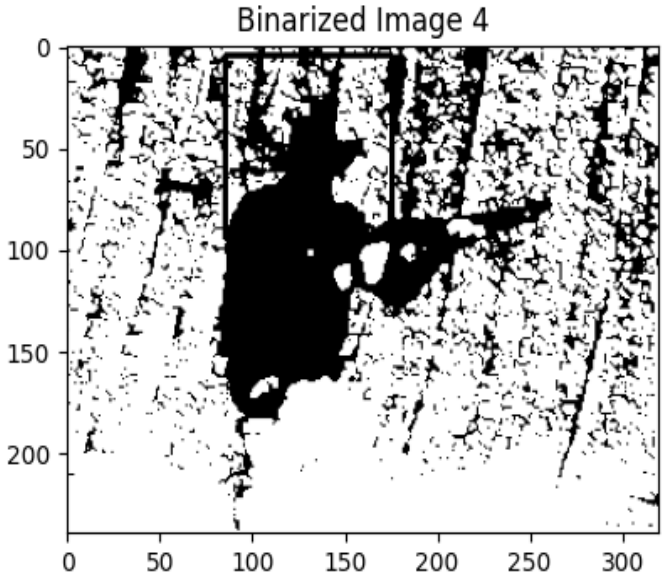


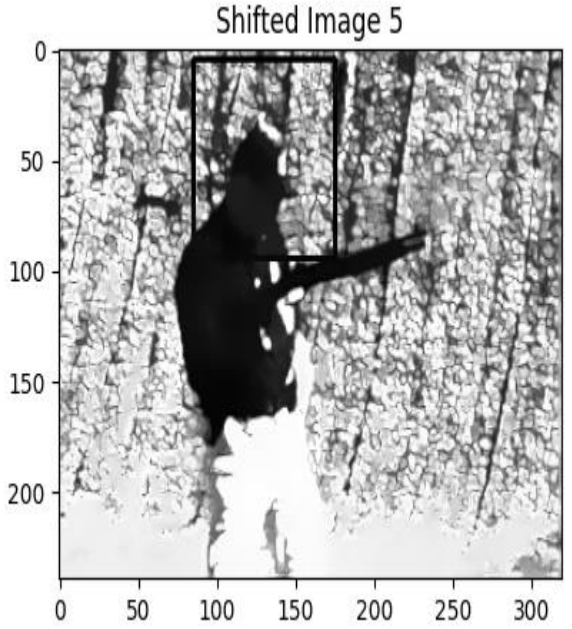
Binary continue……..
Shifted image means
Mean shifted segmented images

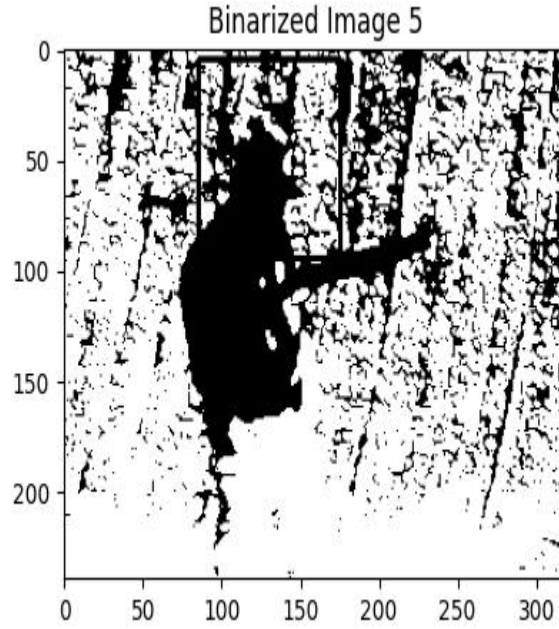




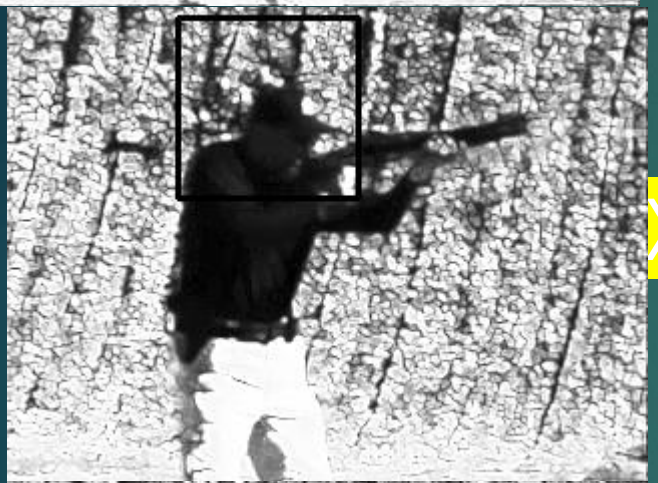


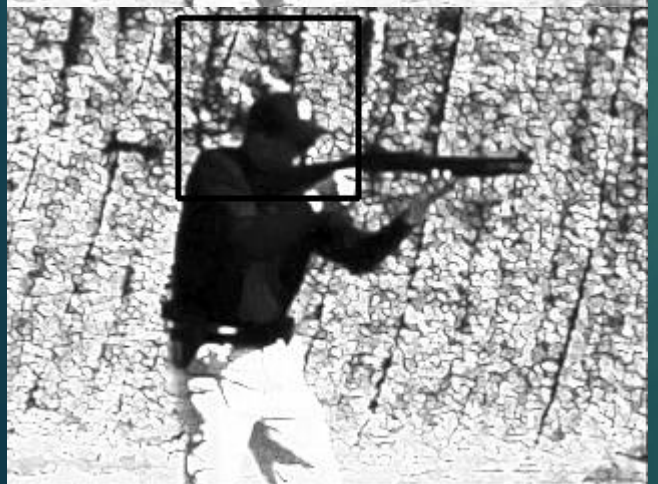






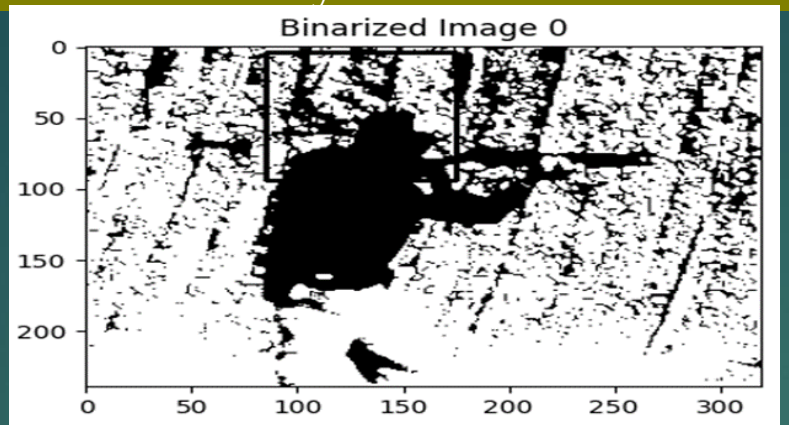
Enh & Sharpened images(original)

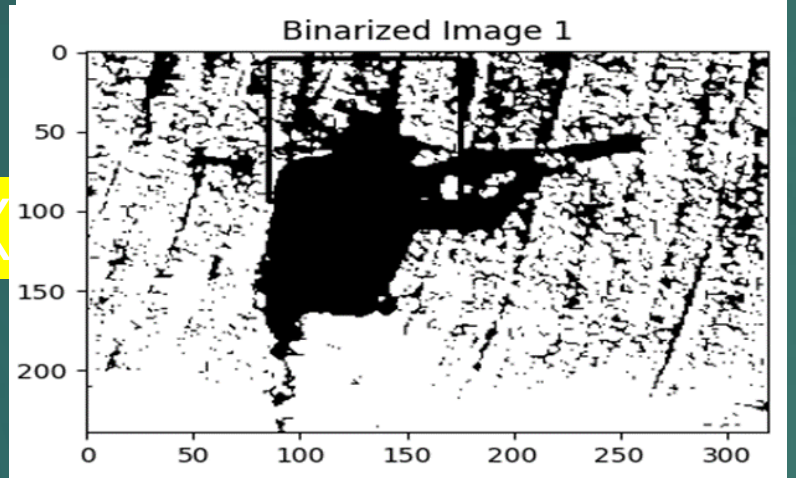


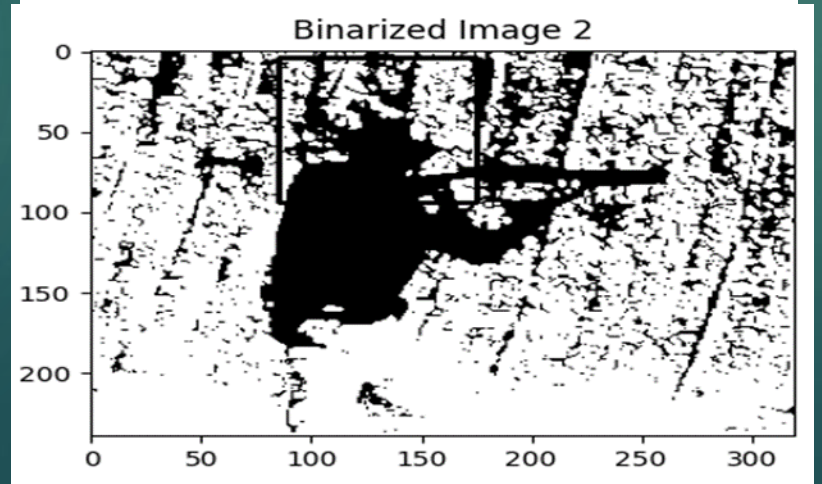


X

Binary







=

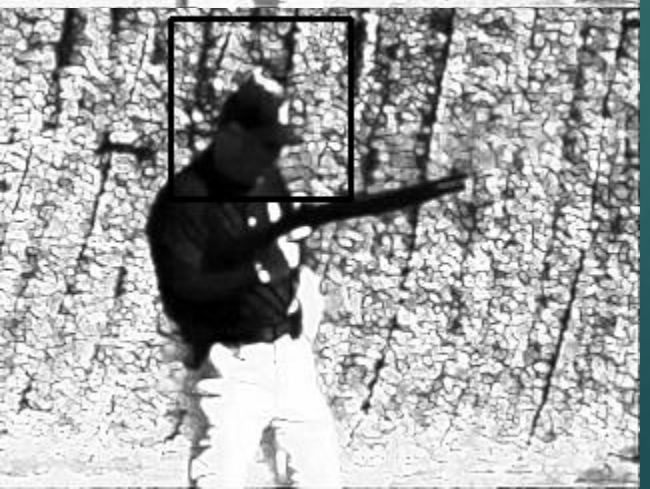
Result

1

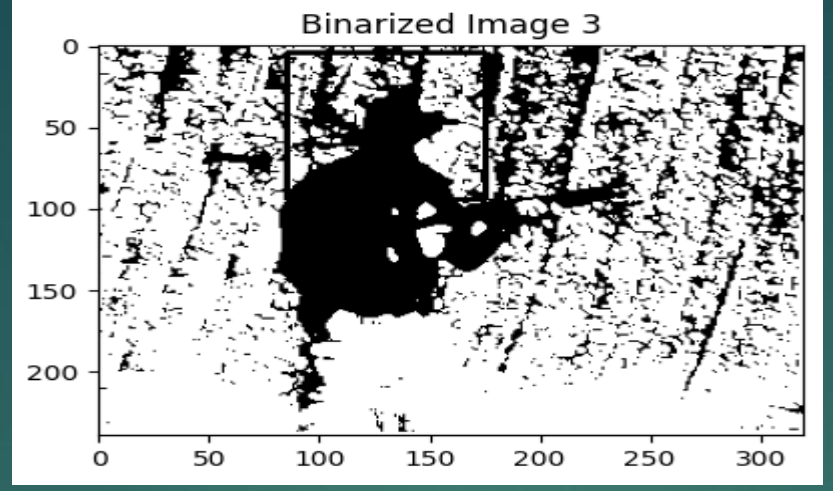
2

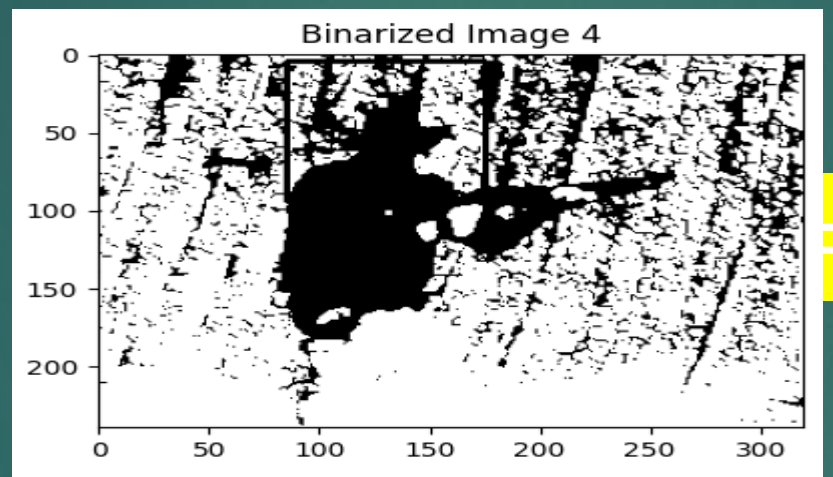
Ready images

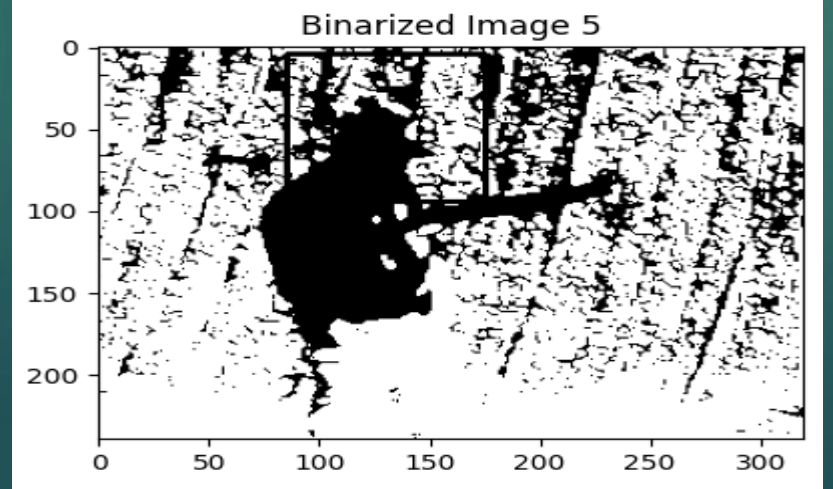
3



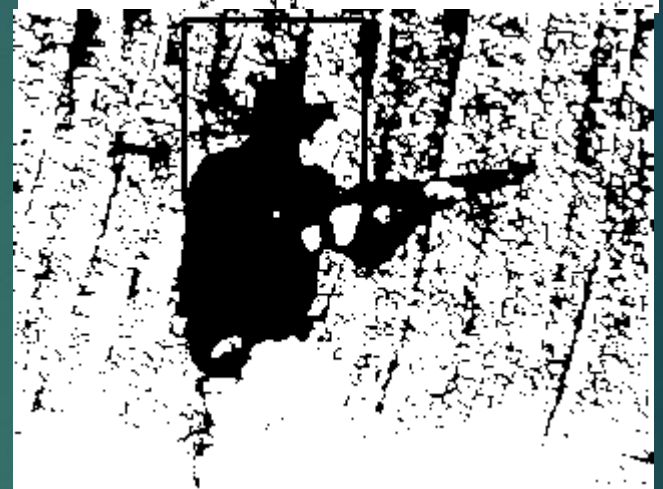
X







=



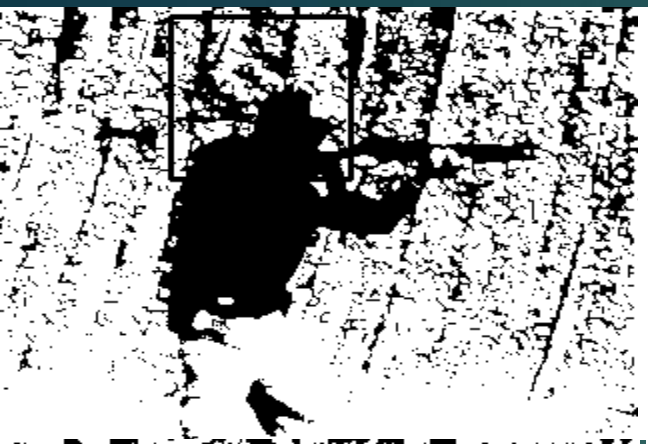
Not Ready images

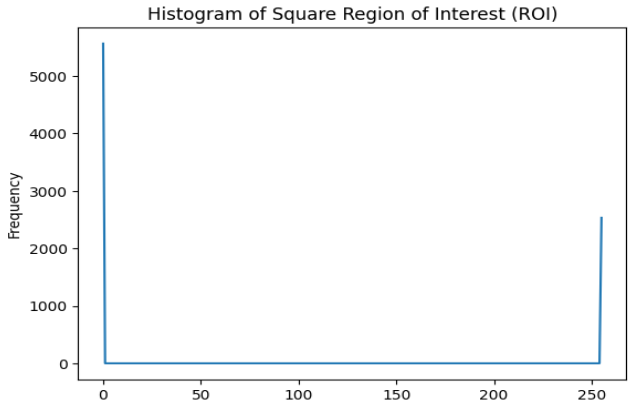
4

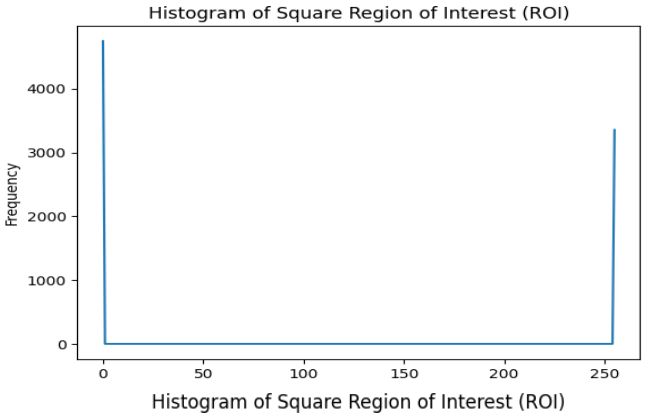
5

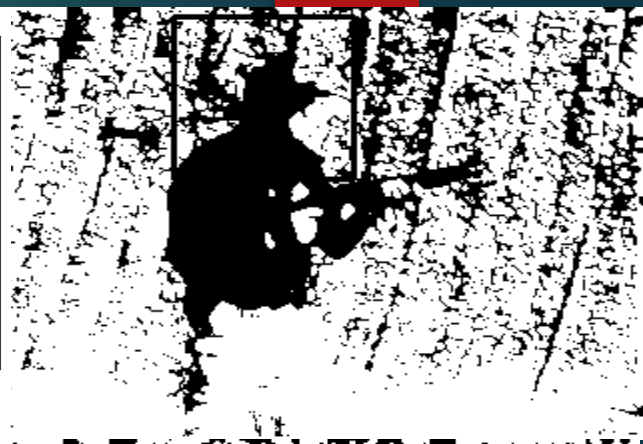
6

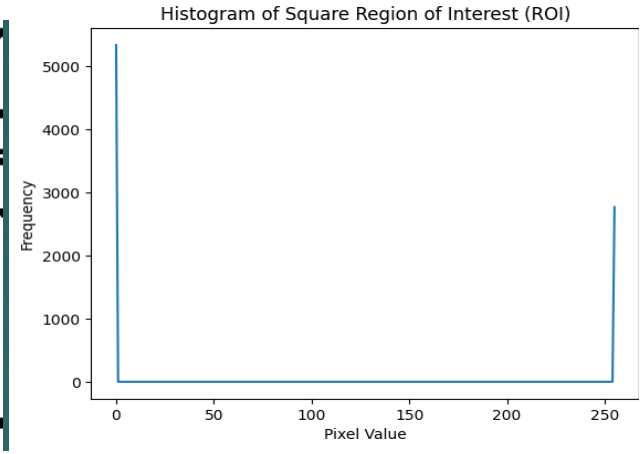
5. Generate histograms of the segmented gray level image. Or calculate the histogram or histogram of Oriented Gradients feature vector of the segmented gray level image

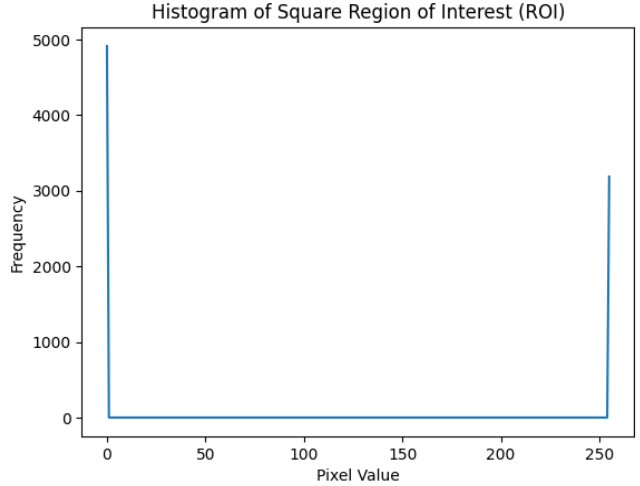


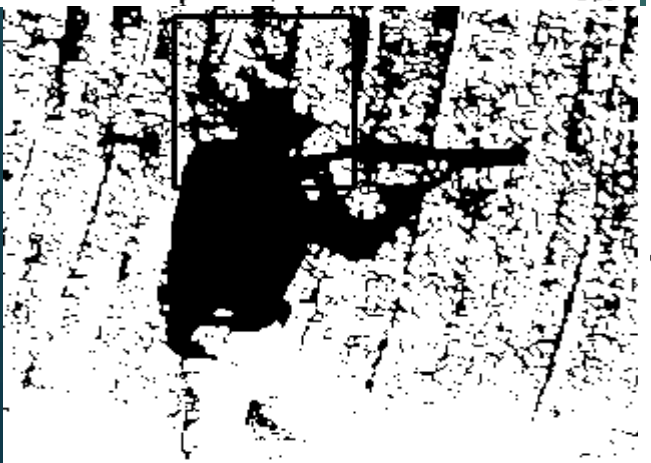


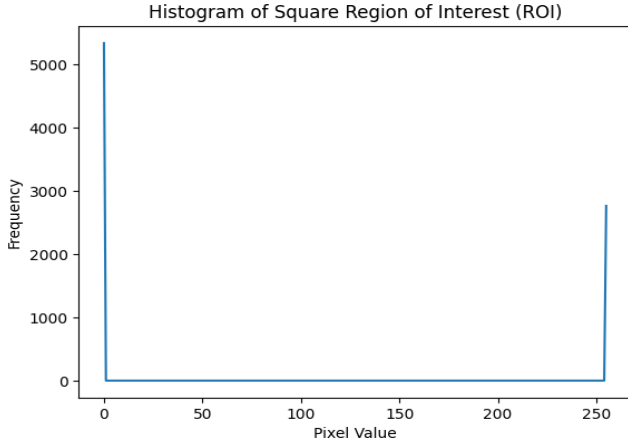


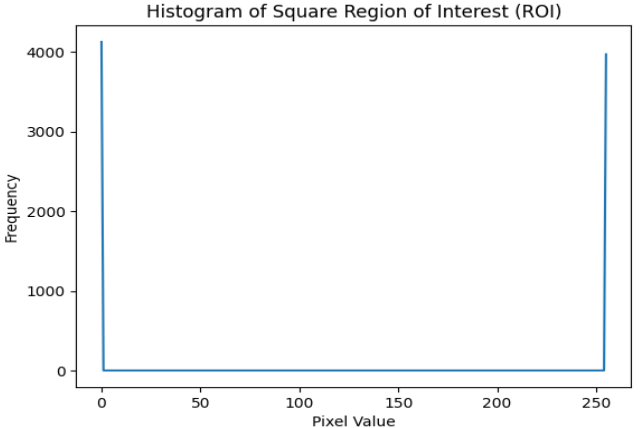












6.Applying two methods of histogram similarity
i. Chi-squared
ii. Correlation coefficient

Chi-squared

```
# Compute the chi-squared distance(sample)
chi_squared_distance4 = cv2.compareHist(hist4, hist5, cv2.HISTCMP_CHISQR)
```

Results for Chi-Squared

Threshold set for this calculation is 1 lowest as possible
Chi-squared distance between first and second ready to shoot images is 31, so they are not similar
Chi-squared distance between first and third ready to shoot images is 30.3, so they are not similar

Chi-squared distance between first and second not ready to shoot images is 14.19, so they are not similar
Chi-squared distance between first and third not ready to shoot images is 194.3, so they are not similar

Using correlation coefficient method

Correlation threshold = 0.95
Correlation coefficient for image 1 and image 2 for ready is 1.0 which is a perfect value, so they are similar
Correlation coefficient for image 1 and image 3 for ready is 0.99 which is a perfect value, so they are similar

Correlation coefficient for image 1 and image 2 for not ready is 0.999 which is a perfect value, so they are similar
Correlation coefficient for image 1 and image 3 for not ready is 0.988 which is a perfect value, so they are similar

CONCLUSION

The correlation coefficient methods is the best method among two which fulfill the task for this project. The proposal was that 3 ready to shoot images are similar and 3 not ready to shoot images are also similar. However, Chi-squared doesn't seems to be working well for this model but the correlation is the excellent model with accuracy 100%.

total_prediction= 2
total_positive=2
accuracy= (total_positive/ total_prediction*100)
Accuracy=100%

##########################THE END##########################