

EX-09--Thresholding of Images

› Aim

To segment the image using global thresholding, adaptive thresholding and Otsu's thresholding using python and OpenCV.

› Software Required

1. Anaconda - Python 3.7
2. OpenCV

› Algorithm

› Step1:

Load the necessary packages.

› Step2:

Read the Image and convert to grayscale.

› Step3:

Use Global thresholding to segment the image.

› Step4:

Use Adaptive thresholding to segment the image.

› Step5:

Use Otsu's method to segment the image.

› Step 6:

Display the results.

' Program

Developed By: Kersoan P
Register No : 212221230050

```
# Load the necessary packages
```

```
import cv2
import numpy as np
import matplotlib.pyplot as plt
```

```
# Read the Image and convert to grayscale
```

```
image=cv2.imread("image.jpg",1)
image=cv2.cvtColor(image,cv2.COLOR_BGR2RGB)
image_gray=cv2.imread("image.jpg",0)
```

```
# Use Global thresholding to segment the image
```

```
ret,thresh_img1=cv2.threshold(image_gray,86,255,cv2.THRESH_BINARY)
ret,thresh_img2=cv2.threshold(image_gray,86,255,cv2.THRESH_BINARY_INV)
ret,thresh_img3=cv2.threshold(image_gray,86,255,cv2.THRESH_TOZERO)
ret,thresh_img4=cv2.threshold(image_gray,86,255,cv2.THRESH_TOZERO_INV)
ret,thresh_img5=cv2.threshold(image_gray,100,255,cv2.THRESH_TRUNC)
```

```
# Use Adaptive thresholding to segment the image
```

```
thresh_img7=cv2.adaptiveThreshold(image_gray,255,cv2.ADAPTIVE_THRESH_MEAN_C,cv2.THRESH_BINARY,1
thresh_img8=cv2.adaptiveThreshold(image_gray,255,cv2.ADAPTIVE_THRESH_GAUSSIAN_C,cv2.THRESH_BINA
```

```
# Use Otsu's method to segment the image
```

```
ret,thresh_img6=cv2.threshold(image_gray,0,255,cv2.THRESH_BINARY+cv2.THRESH_OTSU)
```

```
# Display the results
```

```
titles=["Gray Image","Threshold Image (Binary)","Threshold Image (Binary Inverse)","Threshold I
    ,"Threshold Image (To Zero-Inverse)","Threshold Image (Truncate)","Otsu","Adaptive Thres
images=[image_gray,thresh_img1,thresh_img2,thresh_img3,thresh_img4,thresh_img5,thresh_img6,thre
for i in range(0,9):
    plt.figure(figsize=(10,10))
```

```
plt.subplot(1,2,1)
plt.title("Original Image")
plt.imshow(image)
plt.axis("off")
plt.subplot(1,2,2)
plt.title(titles[i])
plt.imshow(cv2.cvtColor(images[i],cv2.COLOR_BGR2RGB))
plt.axis("off")
plt.show()
```

Output

Original Image



Global Thresholding

Original Image



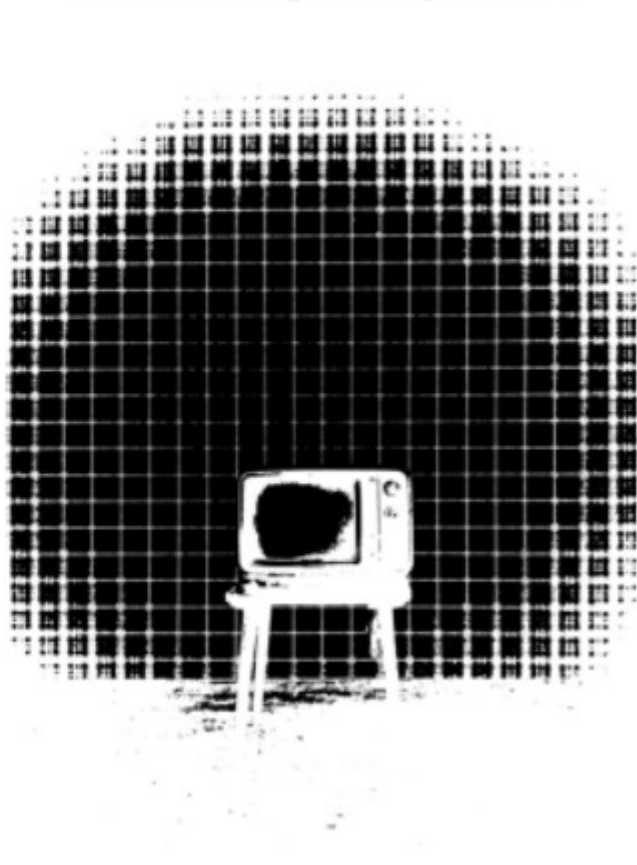
Threshold Image (Binary)



Original Image



Threshold Image (Binary Inverse)



Original Image



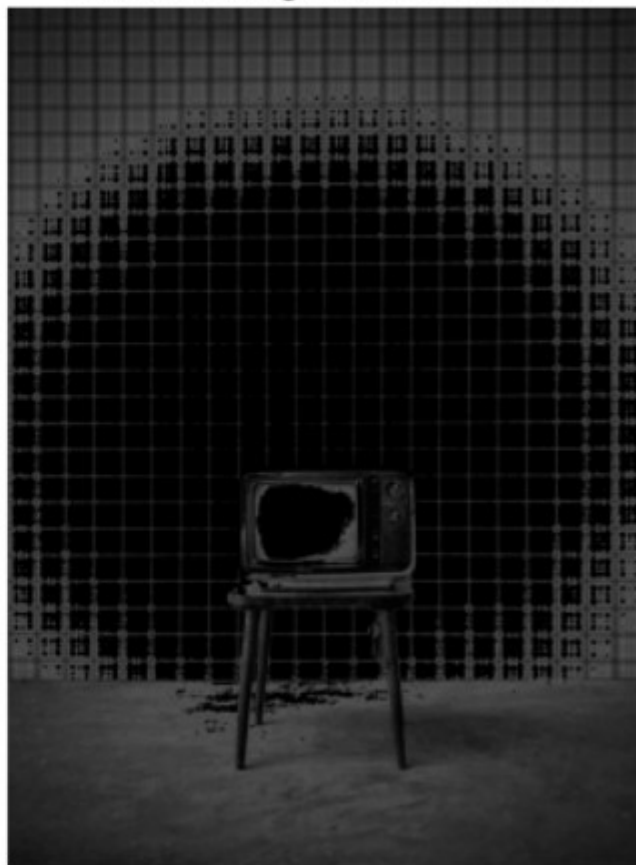
Threshold Image (To Zero)



Original Image



Threshold Image (To Zero-Inverse)



Original Image



Threshold Image (Truncate)



’ Adaptive Thresholding

Original Image



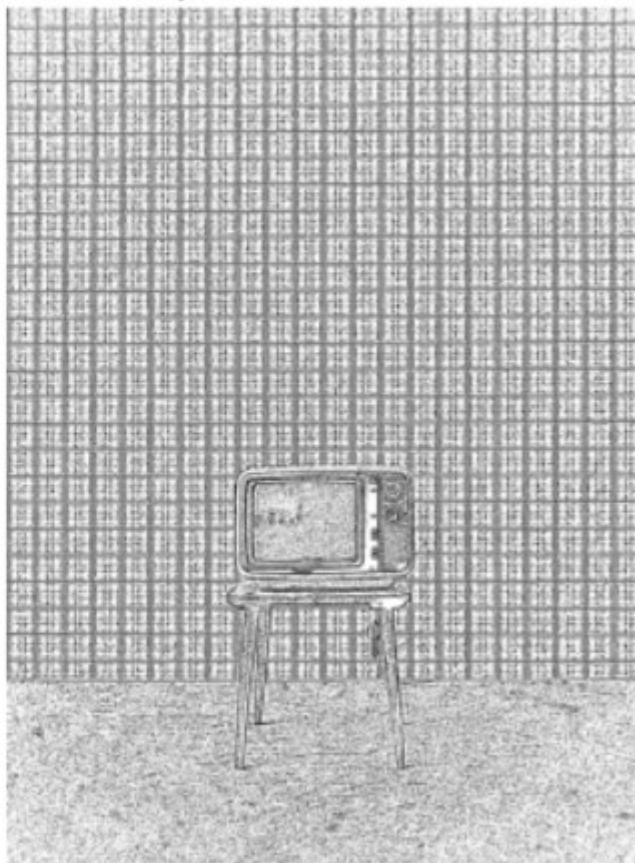
Otsu



Original Image



Adaptive Threshold (Mean)

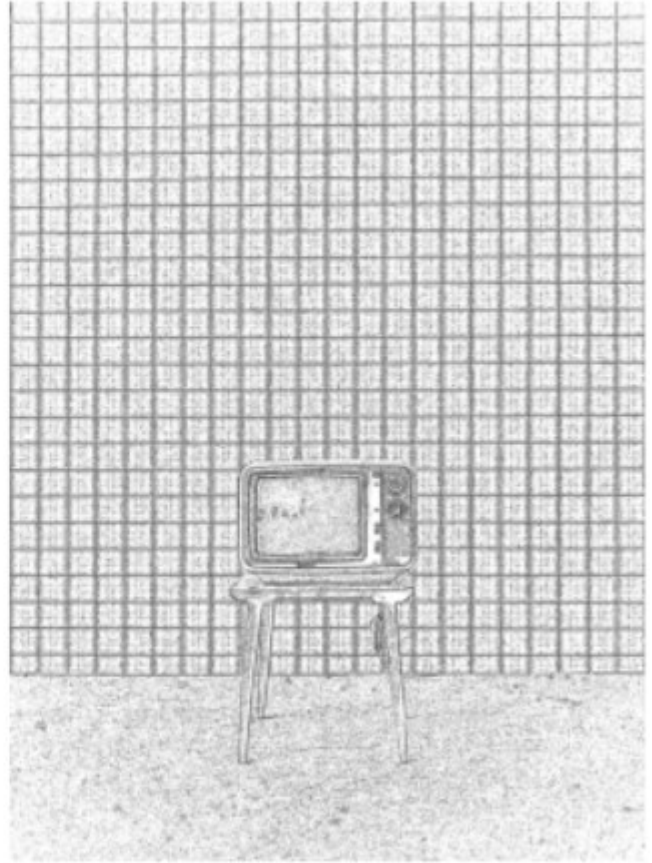


' Optimum Global Thesholding using Otsu's Method

Original Image



Adaptive Threshold (Gaussian)



' Result

Thus the images are segmented using global thresholding, adaptive thresholding and optimum global thresholding using python and OpenCV.