Bharati Vidyapeeth (Deemed to be University)

Department of Engineering and Technology

Kharghar, Navi Mumbai

**Department of Artificial Intelligence and Machine Learning**



**EXPERIMENT 9**

|  |  |
| --- | --- |
| **Name: Shivam Bhosle** | **Roll No: 02** |
| **Subject: COMPUTER VISION** | **Class/Batch: B1** |
| **Date of Performance:** | **Date of Submission:** |

|  |
| --- |
| **AIM** |

To demonstrate Image Segmentation in Python.

|  |
| --- |
| **Theory/Procedure/Algorithm** |

The process of splitting images into multiple layers, represented by a smart, pixel-wise mask is known as Image Segmentation. It involves merging, blocking, and separating an image from its integration level. Splitting a picture into a collection of Image Objects with comparable properties is the first stage in image processing. Scikit-Image is the most popular tool/module for image processing in Python.

Image segmentation is the art of partitioning an image into multiple smaller segments or groups of pixels, such that each pixel in the digital image has a specific label assigned to it. Pixels with the same label have similarity in characteristics.

After segmentation, the output is a region or a structure that collectively covers the entire image. These regions have similar characteristics including colors, texture, or intensity.

**Code:**

import matplotlib.pyplot as plt

import numpy as np

from skimage.io import imread, imshow

from skimage.color import rgb2hsv, rgb2gray

from skimage.exposure import histogram, cumulative\_distribution

from skimage.filters import threshold\_otsu

sail= imread("C:\\Users\\Shivam 007\\Downloads\\balloon.webp")

imshow(sail)

sail\_gray = rgb2gray(sail)

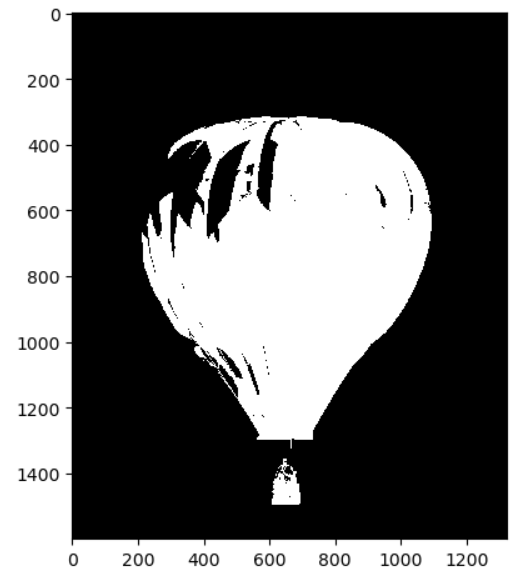
imshow(sail\_gray)

th = 0.4

sail\_gray\_bw = sail\_gray<th

imshow(sail\_gray\_bw)

sail\_hsv = rgb2hsv(sail)



fig, ax = plt.subplots(1, 3, figsize=(12,4))

ax[0].imshow(sail\_hsv[:,:,0], cmap='gray')

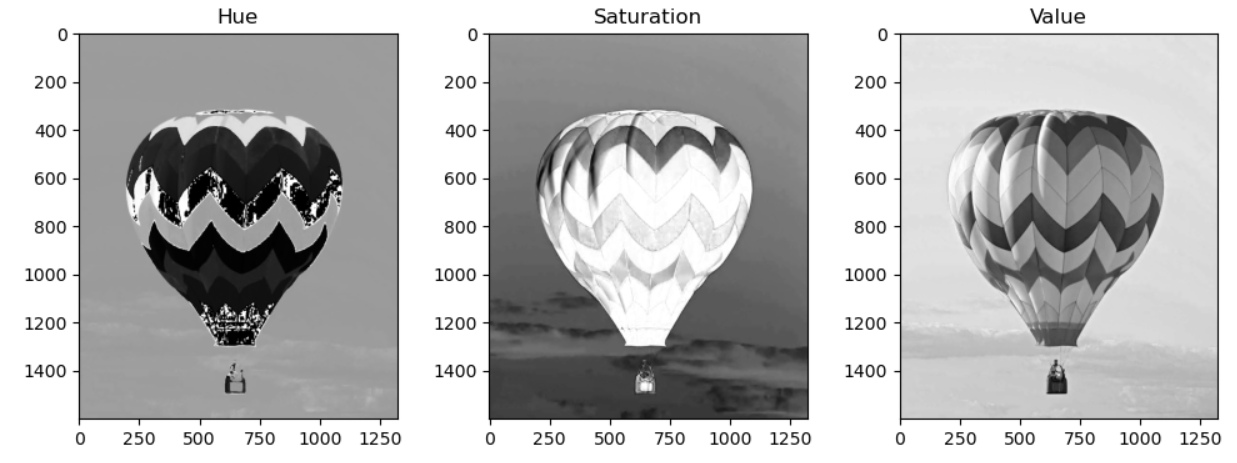
ax[0].set\_title('Hue')

ax[1].imshow(sail\_hsv[:,:,1], cmap='gray')

ax[1].set\_title('Saturation')

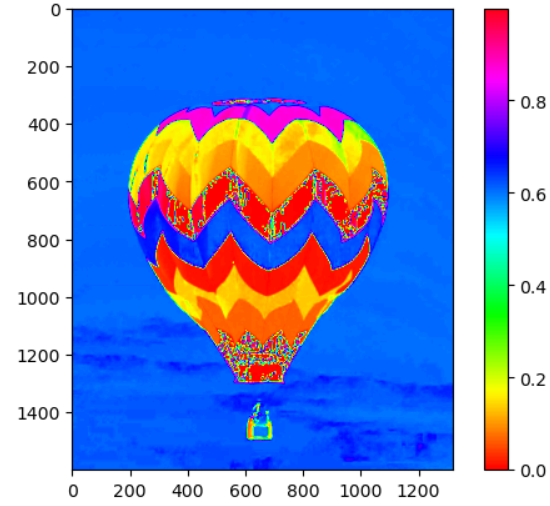
ax[2].imshow(sail\_hsv[:,:,2], cmap='gray')

ax[2].set\_title('Value');



plt.imshow(sail\_hsv[:,:,0],cmap='hsv')

plt.colorbar()



import matplotlib.pyplot as plt

fig, axes = plt.subplots(1, 3, figsize=(12, 4))

# First subplot

lower\_mask = sail\_hsv[:,:,0] > 0.90

upper\_mask = sail\_hsv[:,:,0] < 1

mask = upper\_mask \* lower\_mask

axes[0].imshow(mask)

axes[0].set\_title("Mask 1")

red = sail[:,:,0] \* mask

green = sail[:,:,1] \* mask

blue = sail[:,:,2] \* mask

sail\_masked = np.dstack((red, green, blue))

axes[0].imshow(sail\_masked)

axes[0].set\_title("Red Masked 1")

# Second subplot

lower\_mask = sail\_hsv[:,:,0] > 0.08

upper\_mask = sail\_hsv[:,:,0] < 0.19

mask = upper\_mask \* lower\_mask

axes[1].imshow(mask)

axes[1].set\_title("Mask 2")

red = sail[:,:,0] \* mask

green = sail[:,:,1] \* mask

blue = sail[:,:,2] \* mask

sail\_masked = np.dstack((red, green, blue))

axes[1].imshow(sail\_masked)

axes[1].set\_title("Yellow Masked 2")

# Third subplot

lower\_mask = sail\_hsv[:,:,0] > 0.58

upper\_mask = sail\_hsv[:,:,0] < 0.7

mask = upper\_mask \* lower\_mask

axes[2].imshow(mask)

axes[2].set\_title("Mask 3")

red = sail[:,:,0] \* mask

green = sail[:,:,1] \* mask

blue = sail[:,:,2] \* mask

sail\_masked = np.dstack((red, green, blue))

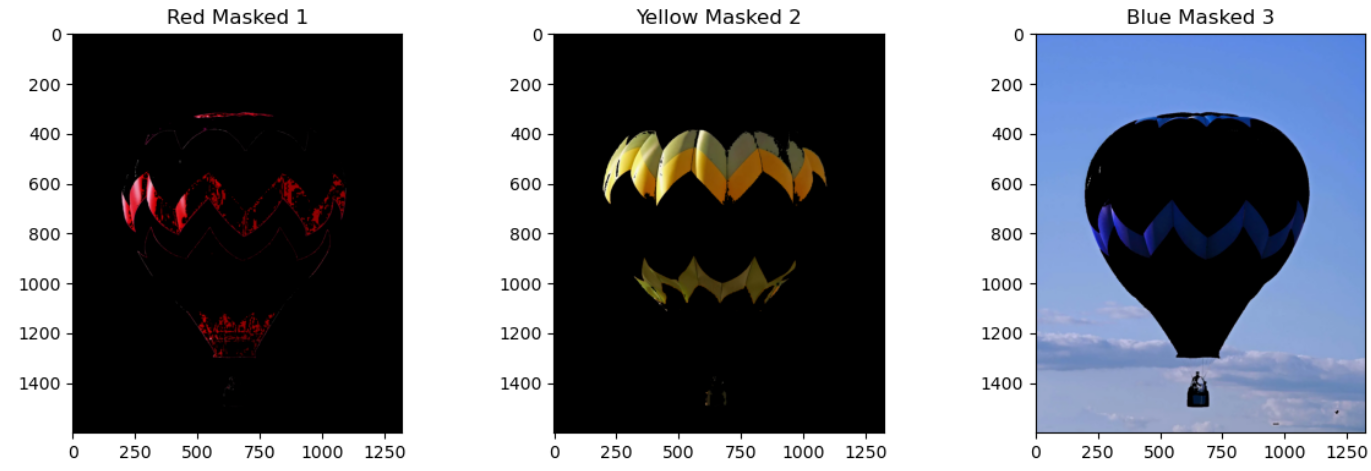
axes[2].imshow(sail\_masked)

axes[2].set\_title("Blue Masked 3")

# Adjust spacing and display the plot

plt.tight\_layout()

plt.show()



|  |
| --- |
| **Conclusion** |

The experiment successfully demonstrated how to Segment Image in Python.

|  |
| --- |
| **Assessment** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Timely Submission**  **(7)** | **Presentation**  **(06)** | **Understanding**  **(12)** | **Total**  **(25)** | **Sign** |
|  |  |  |  |  |