
ORIGINAL IMAGE

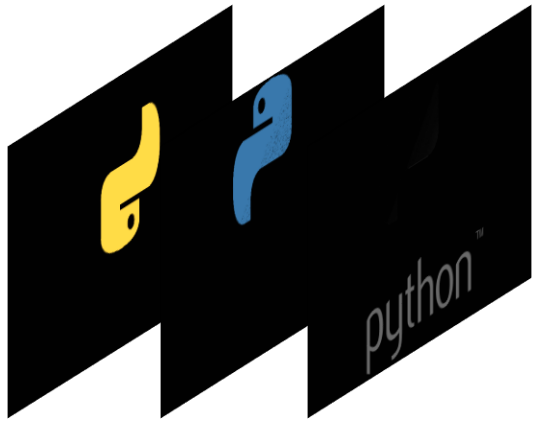
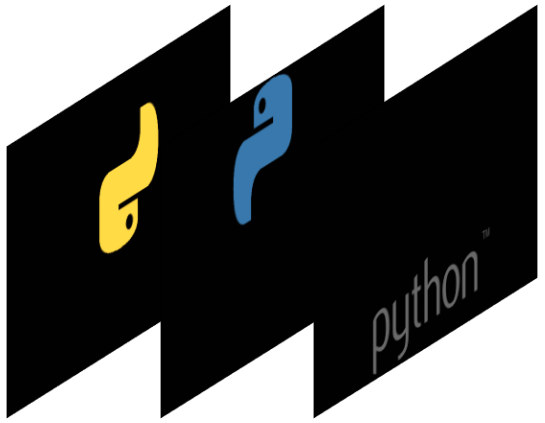



IMAGE SEGMENTATION




MORPHOLOGICAL TRANSFORMATIONS


OUTPUT IMAGE

ORIGINAL IMAGE

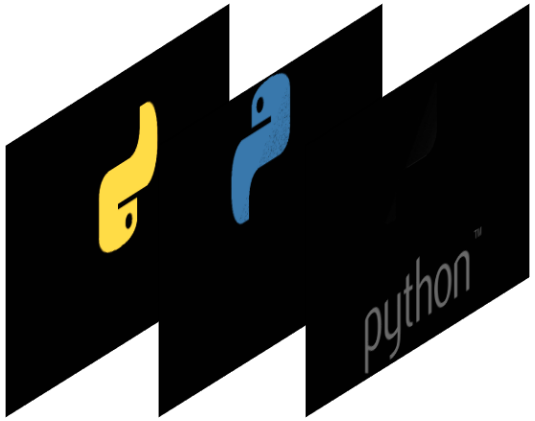
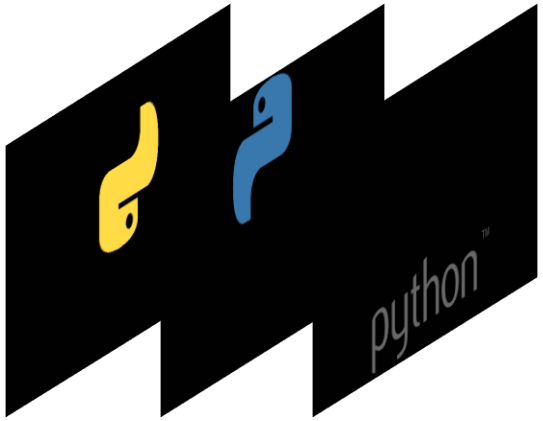


IMAGE SEGMENTATION



MORPHOLOGICAL TRANSFORMATIONS

OUTPUT IMAGE

```
import cv2
import numpy as np

input_image = cv2.imread('Test.png')
hsv_input_image = cv2.cvtColor(input_image,cv2.COLOR_BGR2HSV)
```

Installing Python Modules

pip install opencv-python

ORIGINAL IMAGE

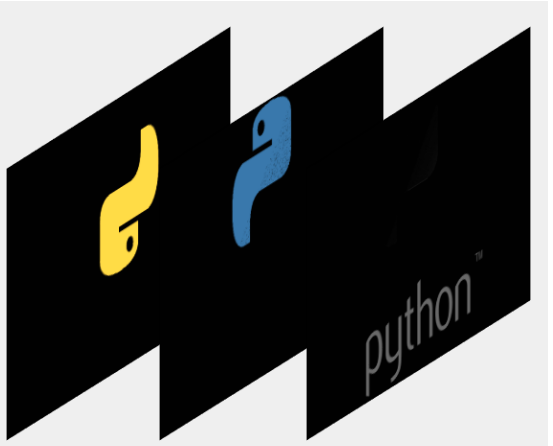
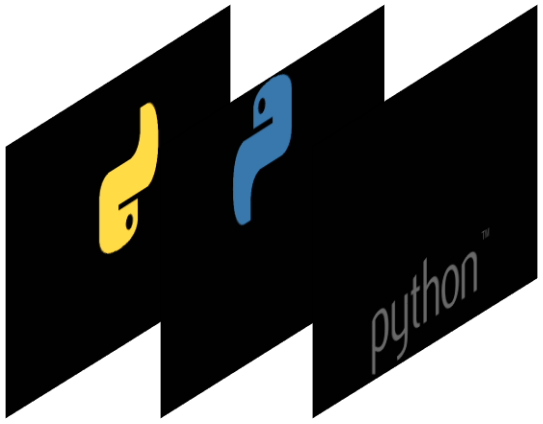


IMAGE SEGMENTATION



MORPHOLOGICAL TRANSFORMATIONS

OUTPUT IMAGE

```
###################Color Boundaries#########################
# Yellow Color Boundaries in HSV
lower_Yellow = np.array([20, 100, 100])
upper_Yellow = np.array([30, 255, 255])
```

Different color range

Similarly to other colors (blue and grey).

```
###############Apply Mask to the image#####################
mask_Yellow = cv2.inRange(hsv_input_image, lower_Yellow, upper_Yellow)
res_Yellow = cv2.bitwise_and(input_image,input_image, mask= mask_Yellow)
```

inRange

Checks if array elements lie between the elements of two other arrays.

$$mask(I) = lower_b(I) \leq src(I) \leq upper_b$$

bitwise_and

Computes bitwise conjunction of the two arrays.

$$res(I) = src(I) \wedge src(I) \quad if \quad mask(I) \neq 0$$

ORIGINAL IMAGE

IMAGE SEGMENTATION

MORPHOLOGICAL TRANSFORMATIONS

OUTPUT IMAGE

```
############Apply Morph to clean the image#################
morph_Yellow = cv2.morphologyEx(res_Yellow, cv2.MORPH_CLOSE, np.ones((5,5),np.uint8))
morph_Blue = cv2.morphologyEx(res_Blue, cv2.MORPH_CLOSE, np.ones((5,5),np.uint8))
morph_Gray = cv2.morphologyEx(res_Gray, cv2.MORPH_OPEN, np.ones((5,5),np.uint8))
```

morphologyEx

The function morphologyEx can perform advanced morphological transformations using an erosion and dilation as basic operations. It is useful for noise removal.
cv.MORPH_OPEN : it applies erosion followed by dilation. For removing external noise.

cv.MORPH_CLOSE : it applies dilation followed by erosion. For removing internal object noise.

ORIGINAL IMAGE

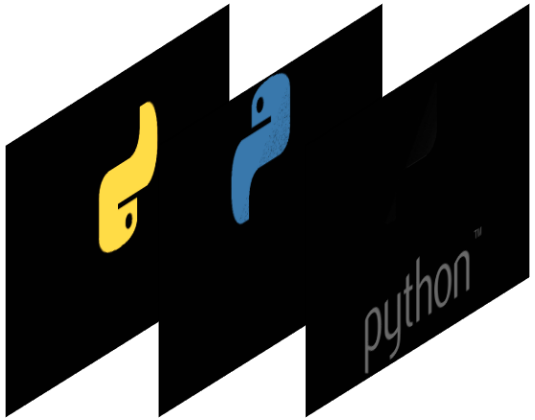
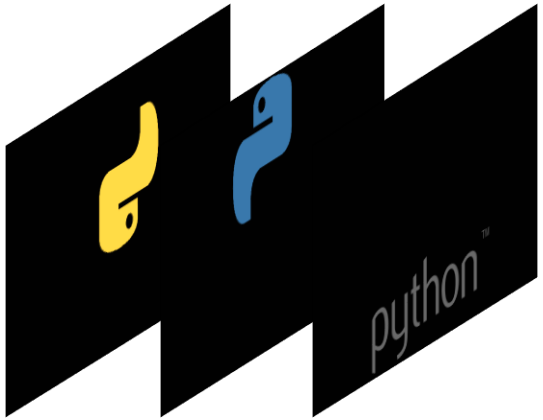


IMAGE SEGMENTATION



MORPHOLOGICAL TRANSFORMATIONS

OUTPUT IMAGE

```
##################Create the full image#####################
image_output = morph_Yellow + morph_Blue + morph_Gray
```

Creating the full image