Marker based localization

Thresholding

Team members

Niharika Jayanthi Dheeraj Kamath

Under the guidance of
Sanam Shakya

Goal

In this chapter,
* We will learn how to threshold images.
* We will see: **cv2.threshold()**

Theory

Thresholding is the simplest method of image segmentation. From a grayscale image, thresholding can be used to create binary images. Here, the matter is straight forward. If pixel value is greater than a threshold value, it is assigned one value (may be white), else it is assigned another value (may be black). The function used is cv2.threshold. First argument is the source image, which should be a grayscale image. Second argument is the threshold value which is used to classify the pixel values. Third argument is the maxVal which represents the value to be given if pixel value is more than (sometimes less than) the threshold value. OpenCV provides different styles of thresholding and it is decided by the fourth parameter of the function. Different types are:
- `cv2.THRESH_BINARY`
- `cv2.THRESH_BINARY_INV`
- `cv2.THRESH_TOZERO`

- `cv2.THRESH_TRUNC`
- `cv2.THRESH_TOZERO_INV`

How to use the function?
The function takes three inputs. First is the source image(img), second argument is the threshold value which is used to classify the pixel values. Third argument is the Max_value which represents the value to be given if pixel value is more than (sometimes less than) the threshold value.

```
cv2.threshold(img, 127, 255, cv2.THRESH_BINARY)
```

Additional Resources

1) http://docs.opencv.org/doc/tutorials/imgproc/threshold/threshold.html

Code

Below we will see an example on how to use thresholding to images. Consider the sunflower image below:

Figure 1: Input image

```
'''
*********************************************************************************
*                                                                               *
*                   IMAGE PROCESSING ( THRESHOLDING OPERATION )                 *
*                                                                               *
*                                                                               *
*  TEAM MEMBERS : NIHARIKA JAYANTHI, DHEERAJ KAMATH                             *
*                                                                               *
*  MENTOR : SANAM SHAKYA                                                        *
*                                                                               *
*  FILENAME : THRESHOLDING.PY                                                   *
*                                                                               *
*  THEME : DEVELOP MODULES FOR IMAGE PROCESSING AND ROBOT LOCALISATION          *
*          USING MARKERS                                                        *
*                                                                               *
*  FUNCTIONS : cv3.threshold(), cv2.cvtColor(), cv2.imread()                    *
*                                                                               *
*  GLOBAL VARIABLES : NONE                                                      *
*                                                                               *
*********************************************************************************
'''

########################

import cv2                  #Import OpenCV
import numpy as np          #Import Numpy

########################

'''
Initialisation of the camera is done by the function cv2.imread
'''
img = cv2.imread('sun.jpg')

'''
To convert the image into grayscale we use the command
cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
'''
gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)

#############################################################################################
'''
* FUNCTION NAME : ret,thresh = cv2.threshold(img, Min_value,Max_value,cv2.THRESH_BINARY)
* INPUT         : Input is a source image which should be in grayscale.
* OUTPUT        : A thresholded image will  be displayed.
* LOGIC         : The function takes three inputs. First is the source image(img),
```

```
                  second argument is the threshold value which is used to classify the
                  pixel values.
                  Third argument is the Max_value which represents the value to be given
                  if pixel value is more than (sometimes less than) the threshold value.
* EXAMPLE CALL  : ret,thresh = cv2.threshold( img, 0, 255, cv2.THRESH_BINARY)
* ADITIONAL INFO: Different types are:
                  cv2.THRESH_BINARY
                  cv2.THRESH_BINARY_INV
                  cv2.THRESH_TRUNC
                  cv2.THRESH_TOZERO
                  cv2.THRESH_TOZERO_INV
'''
#####################################################################
ret, thresh  = cv2.threshold(gray,127,255,cv2.THRESH_BINARY)
ret, thresh1 = cv2.threshold(gray,127,255,cv2.THRESH_BINARY_INV)
ret, thresh2 = cv2.threshold(gray,127,255,cv2.THRESH_TRUNC)
ret, thresh3 = cv2.threshold(gray,127,255,cv2.THRESH_TOZERO)
ret, thresh4 = cv2.threshold(gray,127,255,cv2.THRESH_TOZERO_INV)

'''
* To display the image we use the function cv2.imshow()
'''
cv2.imshow("Thresh", thresh)
cv2.imshow("Thresh1", thresh1)
cv2.imshow("Thresh2", thresh2)
cv2.imshow("Thresh3", thresh3)
cv2.imshow("Thresh4", thresh4)

'''
* cv2.waitKey() - Waits for the user to click any key.
* cv2.destroyAllWindows() - Closes all output tabs.
'''
cv2.waitKey(0)
cv2.destroyAllWindows()
```

1. Output Images:

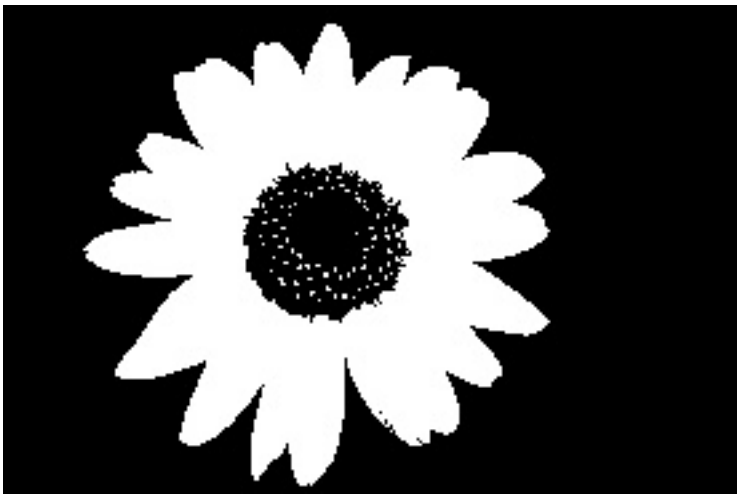

Figure 2: Thresh_Binary

Figure 3: Thresh_Binary_Inv

Figure 4: Thresh_Trunc

Figure 5: Thresh_To_Zero

Figure 6: Thresh_To_Zero