

# Computer Vision Assignment 9

```
import cv2
import numpy as np
from matplotlib import pyplot as plt
from google.colab.patches import cv2_imshow
import imutils

img = cv2.imread('blackdot.jpg')
cv2_imshow(img)
image = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)

#Set our filtering parameters
#Initialize parameter setting using cv2.simpleblobdetector
params = cv2.SimpleBlobDetector_Params()

#Set area filtering parameters
params.filterByArea = True
params.minArea = 100

#Set circularity filtering parameter
params.filterByCircularity = True
params.minConvexity = 0.2

#Set inertia filtering parameters
params.filterByInertia = True
params.minInertiaRatio = 0.01

#Create a detector with the parameters
detector = cv2.SimpleBlobDetector_create(params)

#Detect blobs
keypoints = detector.detect(image)

#Draw blobs on our image as red circles
blank = np.zeros((1,1))
blobs = cv2.drawKeypoints(image, keypoints, blank, (0,0,255),
                           cv2.DRAW_MATCHES_FLAGS_DRAW_RICH_KEYPOINTS)

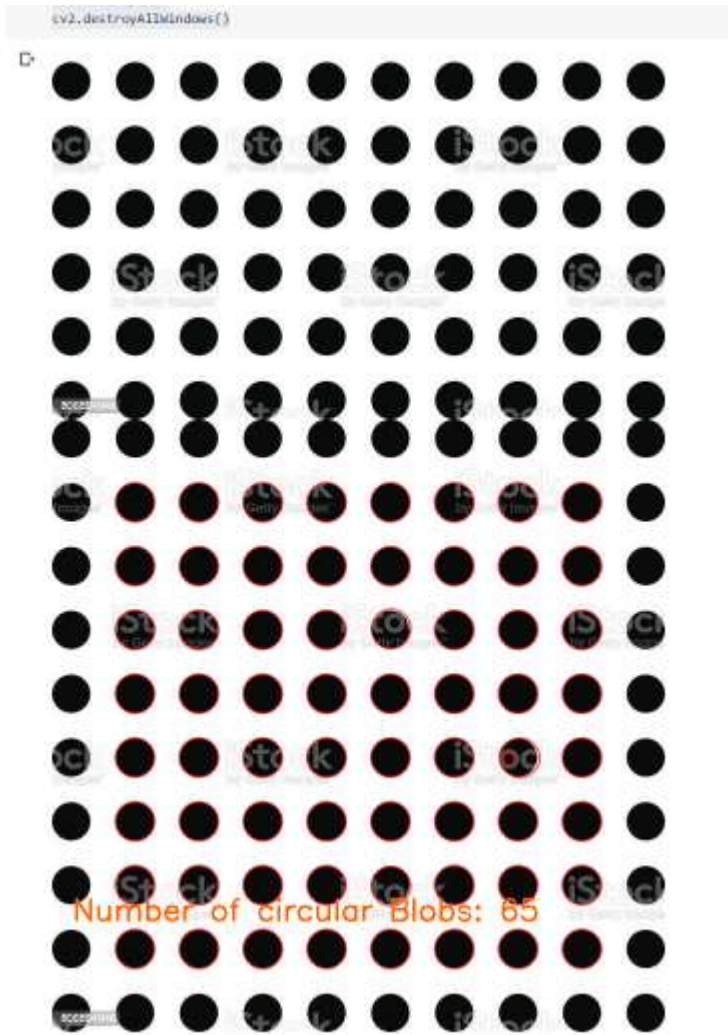
number_of_blobs = len(keypoints)
text = "Number of circular Blobs: " + str(len(keypoints))
cv2.putText(blobs, text, (20, 500),
            cv2.FONT_HERSHEY_SIMPLEX, 1, (0,100,255),2)

#show blobs "Filtering circular blobs only"
```

```

cv2.imshow(blobs)
cv2.waitKey(0)
cv2.destroyAllWindows()

```



```

#histogram gradient
from skimage import feature

image = cv2.imread('dandelions.jfif')
gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
(hog, hog_image) = feature.hog(gray, orientations = 9,
                                pixels_per_cell=(8,8), cells_per_block=(
2,2),
                                visualize = True)

cv2.imshow(image)
cv2.imshow(hog_image)

```



```
#reading image
img1 = cv2.imread('effifile.jfif')
gray1 = cv2.cvtColor(img1, cv2.COLOR_BGR2GRAY)

#keypoints
sift = cv2.xfeatures2d.SIFT_create()
keypoints_1, descriptors_1 = sift.detectAndCompute(img1, None)

img_1 = cv2.drawKeypoints(gray1, keypoints_1, img1)
plt.imshow(img_1)
```

<matplotlib.image.AxesImage at 0x7f9e8ed7b310>

