

INDIVIDUAL REPORT 1

AI & VISION SYSTEM LABORATORY



January 20, 2024

MOHAMED BASIM Z

21R220

**Question No. 14**

Find the number of coins in the given image.

**Software Packages Used**

1. Pycharm IDE

2. Libraries used:

* opencv-python
* numpy

**Program**

import cv2

import numpy as np

# Load the image

image = cv2.imread('path\_to\_your\_image.jpg')

# Convert the image to grayscale

gray = cv2.cvtColor(image, cv2.COLOR\_BGR2GRAY)

# Apply Gaussian blur to reduce noise and improve circle detection

blurred = cv2.GaussianBlur(gray, (15, 15), 0)

# Use HoughCircles to detect circles

circles = cv2.HoughCircles(

blurred,

cv2.HOUGH\_GRADIENT,

dp=1,

minDist=50,

param1=50,

param2=30,

minRadius=10,

maxRadius=50

)

# If circles are found, draw them on the image and count them

if circles is not None:

circles = np.uint16(np.around(circles))

total\_coins = len(circles[0])

print(f'Total number of coins: {total\_coins}')

for i in circles[0, :]:

# Draw the outer circle

cv2.circle(image, (i[0], i[1]), i[2], (0, 255, 0), 2)

# Draw the center of the circle

cv2.circle(image, (i[0], i[1]), 2, (0, 0, 255), 3)

# Display the result

cv2.imshow('Coin Detection', image)

cv2.waitKey(0)

cv2.destroyAllWindows()

else:

print('No coins found in the image.')

**Output :**

**INPUT IMAGE:**

****

**OUTPUT IMAGE:**

****

**C:\Users\BASIM\PycharmProjects\pythonProject2\venv\Scripts\python.exe C:\Users\BASIM\PycharmProjects\pythonProject2\JAGX.py**

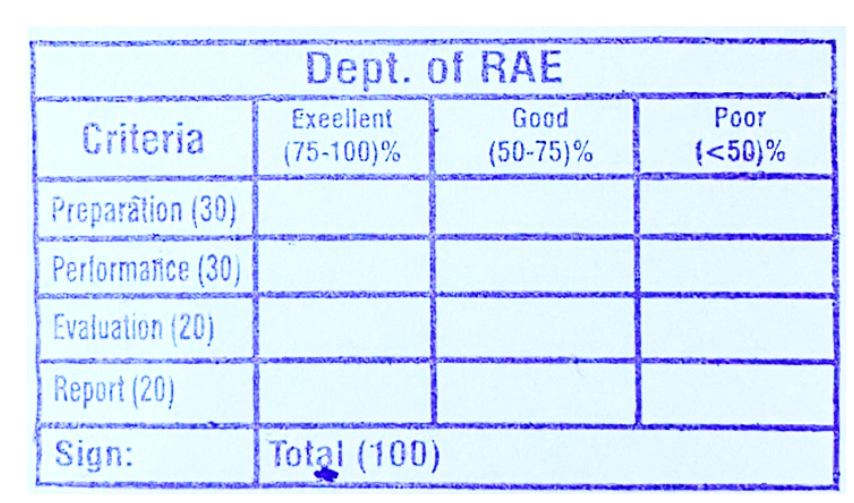
**Total number of coins: 22**

**Process finished with exit code -1**

**Explanation**

Certainly! Here's an explanation of detecting and counting number of coins in points without code:

1. **Library Imports:**
   * The code imports the necessary libraries, OpenCV (cv2) and NumPy (np).
2. **Image Loading:**
   * It loads an image from the file specified by the path ('path\_to\_your\_image.jpg') using cv2.imread().
3. **Image Preprocessing:**
   * The image is converted to grayscale using cv2.cvtColor() to simplify subsequent processing.
   * Gaussian blur is applied to the grayscale image using cv2.GaussianBlur() to reduce noise and facilitate circle detection.
4. **Circle Detection using HoughCircles:**
   * cv2.HoughCircles is employed to detect circles in the preprocessed image.
   * Parameters for the HoughCircles function are set:
   * dp: Inverse ratio of the accumulator resolution to the image resolution.
   * minDist: Minimum distance between the centers of detected circles.
   * param1 and param2: Parameters for the edge detection.
   * minRadius and maxRadius: Minimum and maximum radii of the circles to be detected.
5. **Check for Detected Circles:**
   * If circles are found (circles is not None), the code proceeds to count and draw them.
6. **Count and Draw Circles:**
   * The number of detected circles is counted and stored in total\_coins.
   * For each detected circle, both the outer circle and the center are drawn on the original image using cv2.circle().
7. **Display Result:**
   * The resulting image with drawn circles is displayed using cv2.imshow().
   * The program waits for a key press (cv2.waitKey(0)) and then closes the OpenCV windows using cv2.destroyAllWindows().
8. **Print Result or Handle No Coins:**
   * If no circles are detected, a message is printed: 'No coins found in the image.'
   * If circles are detected, the total number of coins is printed.



**RESULT**

Thus the detection and counting number of coin in the given image was done using OpenCV in python.