**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра математического обеспечения и применения ЭВМ**

отчет

**по лабораторной работе №2**

**по дисциплине «Цифровая обработка изображений»**

**Тема: Поиск особенностей изображений**

|  |  |  |
| --- | --- | --- |
| Студентка гр. 7381 |  | Алясова А.Н. |
| Преподаватель |  | Черниченко Д.А. |

Санкт-Петербург

2021

**Постановка задачи.**

Реализовать на языке Python с использованием библиотеки OpenCV программу, выполняющую поиск окружностей в изображении путем преобразования Xo (Hough).

**Входные данные**.

1. Цветное изображение в формате bmp, jpg.
2. Минимальное расстояние между центрами окружностей.
3. Минимальный радиус окружности.
4. Максимальный радиус окружности.

**Выходные данные.**

Цветное изображение в формате bmp (result.bmp).

**Выполнение работы.**

Загрузим цветное изображение в формате .jpg (рис.1).



Рисунок 1

Зададим:

* минимальное расстояние между центрами окружностей
* минимальный радиус окружности .
* максимальный радиус окружности .



Рисунок 2

**Выводы.**

В ходе выполнения лабораторной работы была реализована программа на языке Python с использованием библиотеки OpenCV, выполняющая поиск окружностей в изображении путем преобразования Xo (Hough).

**ПРИЛОЖЕНИЕ А**

**ИСХОДНЫЙ КОД**

import cv2 as cv

import numpy as np

import sys

filename = sys.argv[1]

min\_dis = int(sys.argv[2])

min\_rad = int(sys.argv[3])

max\_rad = int(sys.argv[4])

image = cv.imread(filename)

cv.imshow("Original image", image)

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

cv.imshow("Gray image", gray)

gray = cv.medianBlur(gray, 5)

cv.imshow("Blurred gray image", gray)

circles = cv.HoughCircles(gray, cv.HOUGH\_GRADIENT, 1.7,

param1=80, param2=57, minDist=min\_dis,

minRadius=min\_rad, maxRadius=max\_rad)

if circles is not None:

circles = np.round(circles[0, :]).astype("int")

for (x, y, r) in circles:

cv.circle(image, (x, y), r, (0, 255, 0), 4)

cv.rectangle(image, (x - 5, y - 5), (x + 5, y + 5), (0, 128, 255), -1)

cv.imshow("Circles", image)

cv.imwrite('result.bmp', image)

cv.waitKey(0)