

ГУАП
КАФЕДРА № 44

ОТЧЕТ
ЗАЩИЩЕН С ОЦЕНКОЙ
ПРЕПОДАВАТЕЛЬ

доцент, канд. техн. наук

Н. В. Соловьёв

должность, уч. степень,
звание

подпись, дата

инициалы, фамилия

ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №1

РАЗРАБОТКА WINDOWS-ПРИЛОЖЕНИЯ

по курсу: ЦИФРОВАЯ ОБРАБОТКА ИЗОБРАЖЕНИЯ

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР. № 4842

Д. А. Калмыков

подпись, дата

инициалы, фамилия

Санкт-Петербург 2021

1. Индивидуальное задание

Выделение контура LoG-фильтрами с разными размерами маски

2. Теоретические положения

Информация, позволяющая отличать объекты друг от друга по их изображениям, в значительной степени содержится в контурных линиях.

Следовательно, выделение контурных линий повышает различимость фрагментов изображения. Основным принципом большинства методов выделения контурных линий, отделяющих соседние фрагменты, является вычисление частных производных от функции яркости по координатам.

Так как яркость изображения является функцией двух переменных, градиент функции яркости в каждой точке определяется как двумерный вектор

$$\mathbf{G}[z(x, y)] = \begin{bmatrix} G_x \\ G_y \end{bmatrix}, \text{ где } G_x = \frac{\partial z(x, y)}{\partial x}, G_y = \frac{\partial z(x, y)}{\partial y} - \text{частные производные.}$$

Для дискретных изображений вычисление частных производных сводится к вычислению разности яркостей соседних пикселей различными способами, т.е. фактически к пространственной фильтрации путем свертки с различными по размеру и значению коэффициентов масками разностных фильтров. Тогда,

$$G_x = \sum_{(s,t) \in S_{xy}} h_x(s,t)z(s,t), \quad G_y = \sum_{(s,t) \in S_{xy}} h_y(s,t)z(s,t),$$

В лабораторной работе используются LoG фильтры, для которых частные производные первого порядка будут равны по обеим осям.

$$H_{LoG} = \begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

3. Процесс выполнения работы

Для каждого пикселя изображения необходимо вычислить сумму произведений значений яркости пикселя и коэффициента маски, соответствующего данному пикселю окрестности исследуемого пикселя.

Был использован специальный модуль `opencv` для того, чтобы упростить задачу загрузки изображения в память в виде массива, вывода изображения на экран и сохранения получившегося изображения в виде файла формата `.jpg`

Проблема выхода маски за пределы изображения решена проходом, начиная с 1 элемента и заканчивая элементом `res_image.shape[0] - 2`, где `res_image.shape`—это длина изображения в пикселях.

Код программы и результат выполнения представлены ниже.

4. Код программы

```
import cv2
import math

# загрузка изображения
image = cv2.imread('romashka.jpg', cv2.IMREAD_GRAYSCALE)
(h, w) = image.shape[:2]

# изменение параметров длины и ширины изображения
res_image = cv2.resize(image, (int(w/1.6), int(h/1.6)))
res_image_1 = cv2.resize(image, (int(w/1.6), int(h/1.6)))
res_image_7 = cv2.resize(image, (int(w/1.6), int(h/1.6)))

# фильтр LoG 3x3
transform_matrix_3 = ((0, -1, 0), (-1, 4, -1), (0, -1, 0))

for i in range(1, res_image.shape[0] - 2):
    for j in range(1, res_image.shape[1] - 2):
        g_x = 0
        for di in range(-1, 2):
            for dj in range(-1, 2):
                g_x += res_image_1[i + dj][j + di] *
transform_matrix_3[1 + di][1 + dj]

        res_image[i][j] = math.sqrt(g_x**2 + g_x**2)

cv2.imshow('rom', res_image)
cv2.waitKey(0)
cv2.imwrite('lions-3x3.jpg', res_image)
```

5. Результаты выполнения программы



Рисунок 1 – исходное изображение 1



Рисунок 2 – обработанное изображение 1



Рисунок 3 – исходное изображение 2

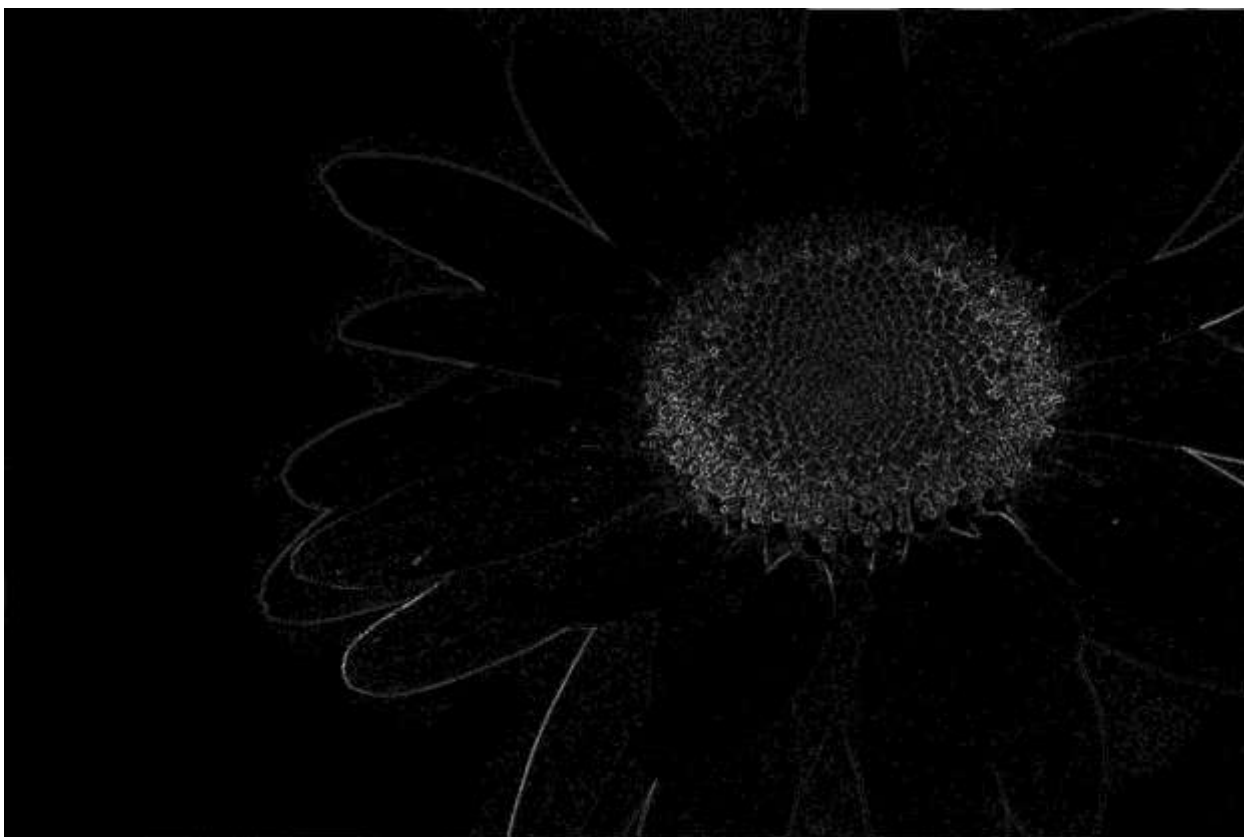


Рисунок 4 – обработанное изображение 2

6. Вывод

Выполнена лабораторная работа “Разработка Windows-приложения”, получены навыки обработки изображения LoG-фильтром размера 3x3.