

Министерство образования Республики Беларусь  
Учреждение образования «Белорусский государственный университет  
информатики и радиоэлектроники»

Факультет информационных технологий и управления  
Кафедра интеллектуальных информационных технологий  
Дисциплина «Обработка изображений в интеллектуальных системах»

**ОТЧЁТ**  
к лабораторной работе №4  
на тему  
**«СЕГМЕНТАЦИЯ ИЗОБРАЖЕНИЙ»**

БГУИР 6-05-0611-03 130

Выполнил студент группы 321701  
СЕМЕНЯКО Владимир Дмитриевич

---

(дата, подпись студента)

Проверил  
САЛЬНИКОВ Даниил Андреевич

---

(дата, подпись преподавателя)

Минск 2025

# 1 ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ

Реализовать программно сегментацию посредством выделения границ областей или посредством разметки точек области.

## 2 ВЫПОЛНЕНИЕ РАБОТЫ

Программа была реализована на языке Python. Было реализовано два способа сегментации: сегментация по выделению границ областей (метод Кэнни — Canny edge detection), сегментация по разметке точек области (метод Watershed — водораздел).

### Листинг 1 – Код программы

```
import cv2
import numpy as np
import matplotlib.pyplot as plt

image = cv2.imread('image.jpg')
gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
plt.figure(figsize=(12,6))
plt.subplot(1,3,1)
plt.title('Оригинал')
plt.imshow(cv2.cvtColor(image, cv2.COLOR_BGR2RGB))
edges = cv2.Canny(gray, 30, 100)

blur = cv2.medianBlur(gray, 5)

ret, thresh = cv2.threshold(blur, 0, 255, cv2.THRESH_BINARY + cv2.THRESH_OTSU)

kernel = np.ones((3,3), np.uint8)
opening = cv2.morphologyEx(thresh, cv2.MORPH_OPEN, kernel, iterations=2)

sure_bg = cv2.dilate(opening, kernel, iterations=3)
dist_transform = cv2.distanceTransform(opening, cv2.DIST_L2, 5)
ret, sure_fg = cv2.threshold(dist_transform, 0.7*dist_transform.max(), 255, 0)

sure_fg = np.uint8(sure_fg)
unknown = cv2.subtract(sure_bg, sure_fg)

ret, markers = cv2.connectedComponents(sure_fg)

markers = markers + 1
markers[unknown == 255] = 0

markers = cv2.watershed(image, markers)
image[markers == -1] = [255, 0, 0]

plt.subplot(1,3,2)
plt.title('Canny границы')
```

```
plt.imshow(edges, cmap='gray')

plt.subplot(1,3,3)
plt.title('Watershed сегментация()')
plt.imshow(cv2.cvtColor(image, cv2.COLOR_BGR2RGB))
plt.show()
```

В ходе выполнения работы на вход программы было подано изображение. На Рисунке 1 представлено исходное изображение. Было подобрано контрастное изображение, чтобы было явно видно результат сегментации.

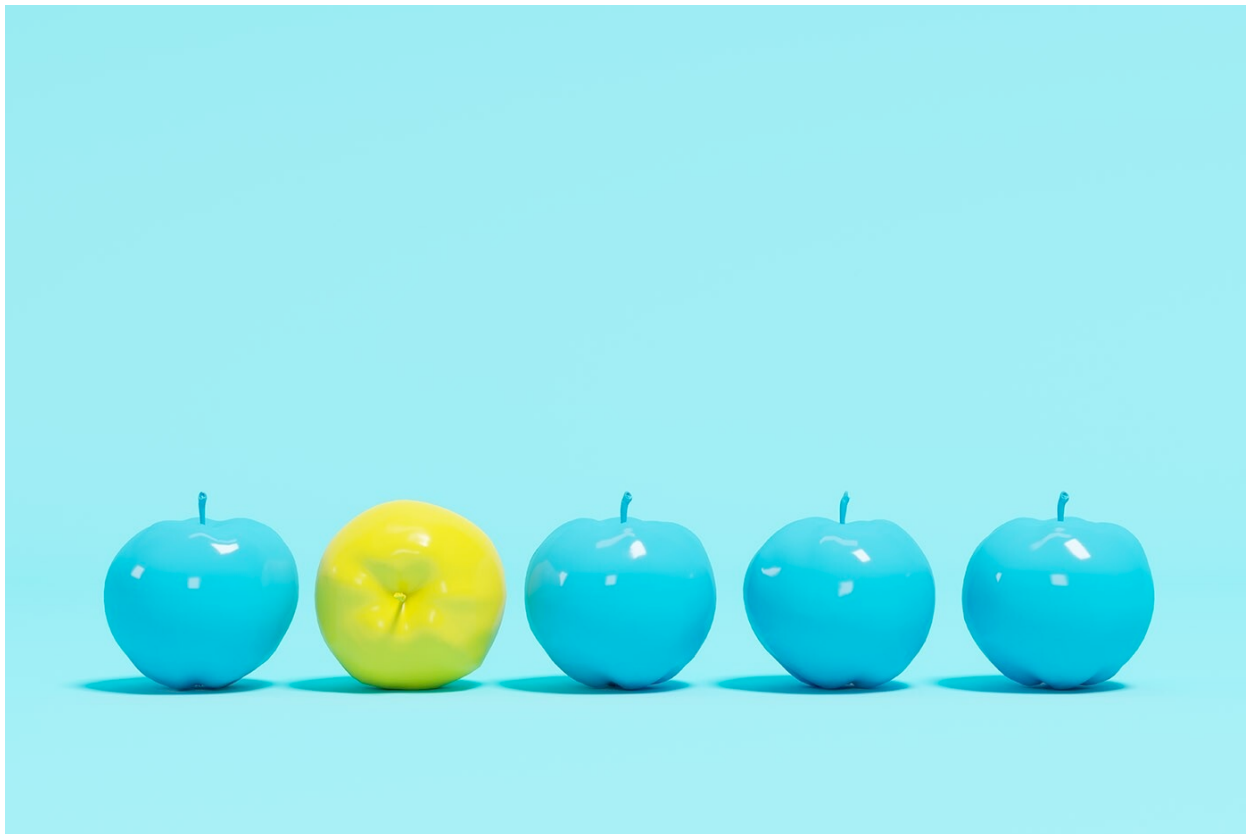


Рисунок 1 – Исходное изображение

### **3 АЛГОРИТМЫ СЕГМЕНТАЦИИ ИЗОБРАЖЕНИЙ**

В данной работе реализованы два подхода к сегментации изображений:

- а) Сегментация на основе выделения границ областей (метод Кэнни).
- б) Сегментация на основе разметки точек областей (метод водораздела, Watershed).

### 3.1 Алгоритм 1. Сегментация посредством выделения границ (метод Кэнни)

Метод Кэнни (*Canny Edge Detection*) предназначен для выделения контуров (границ) объектов на изображении. Он состоит из нескольких этапов:

а) Преобразование исходного изображения в оттенки серого:

$$I_{gray} = f_{gray}(I_{rgb})$$

б) Сглаживание (фильтрация Гаусса) для удаления шума:

$$I_{smooth} = G_{\sigma} * I_{gray}$$

где  $G_{\sigma}$  — Гауссов фильтр с дисперсией  $\sigma^2$ .

в) Вычисление градиента яркости:

$$G_x = \frac{\partial I_{smooth}}{\partial x}, \quad G_y = \frac{\partial I_{smooth}}{\partial y}$$

$$M = \sqrt{G_x^2 + G_y^2}, \quad \theta = \arctan\left(\frac{G_y}{G_x}\right)$$

где  $M$  — модуль градиента (интенсивность границы),  $\theta$  — направление границы.

г) Подавление немаксимумов (Non-Maximum Suppression) — удаляются пиксели, не соответствующие локальным максимумам вдоль направления градиента.

д) Двойная пороговая фильтрация:

$$E(x, y) = \begin{cases} 1, & \text{если } M(x, y) > T_{high} \\ 0, & \text{если } M(x, y) < T_{low} \\ \text{неопределён,} & \text{иначе} \end{cases}$$

где  $T_{high}$  и  $T_{low}$  — верхний и нижний пороги.

е) Гистерезис соединяет слабые края, если они связаны с сильными.

Результатом алгоритма является бинарная карта границ:

$$I_{edges} = f_{Canny}(I_{gray}, T_{low}, T_{high})$$

### 3.2 Алгоритм 2. Сегментация посредством разметки областей (метод водораздела)

Метод водораздела (*Watershed Segmentation*) относится к семейству морфологических алгоритмов сегментации. Он рассматривает изображение как топографическую поверхность, где яркость пикселя соответствует высоте. Алгоритм можно описать следующими шагами:

а) Преобразование изображения в градации серого:

$$I_{gray} = f_{gray}(I_{rgb})$$

б) Удаление шума фильтром:

$$I_{blur} = f_{median}(I_{gray})$$

в) Пороговая бинаризация:

$$I_{bin}(x, y) = \begin{cases} 1, & I_{blur}(x, y) > T \\ 0, & I_{blur}(x, y) \leq T \end{cases}$$

где  $T$  определяется автоматически по методу Отсу.

г) Морфологические операции:

$$I_{open} = (I_{bin} \ominus B) \oplus B$$

где  $\ominus$  — эрозия,  $\oplus$  — дилатация,  $B$  — структурный элемент.

д) Определение фона и переднего плана:

$$I_{bg} = f_{dilate}(I_{open}), \quad I_{fg} = f_{dist}(I_{open}) > \alpha \cdot \max(f_{dist}(I_{open}))$$

где  $f_{dist}$  — преобразование расстояния (distance transform),  $\alpha \in [0, 1]$ .

е) Нахождение неизвестных областей:

$$I_{unknown} = I_{bg} - I_{fg}$$

ж) Маркировка областей:

$$L = f_{label}(I_{fg})$$

Все метки увеличиваются на 1:

$$L = L + 1, \quad L[I_{unknown} = 1] = 0$$

з) Применение алгоритма водораздела:

$$L' = f_{watershed}(I_{rgb}, L)$$

Контуры, найденные водоразделом, помечаются как границы ( $L' = -1$ ).

Результатом работы алгоритма является изображение, разделённое на области, каждая из которых соответствует отдельному объекту.

После применения алгоритмов получим следующие результаты:

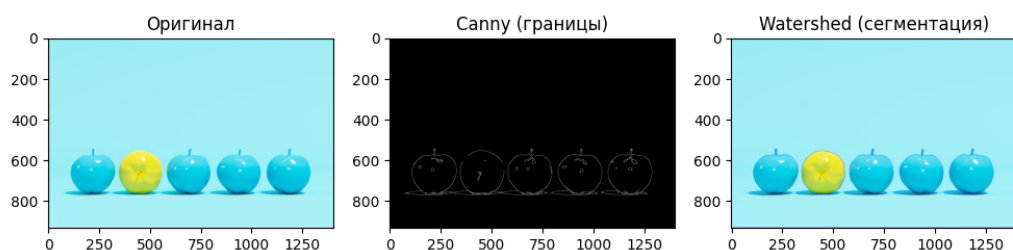


Рисунок 1 – Результаты сегментации в сравнении с исходным изображением

## ВЫВОД

В ходе выполнения лабораторной работы были изучены и реализованы основные методы сегментации изображений — метод выделения границ (Canny) и метод разметки областей (Watershed).