

Федеральное государственное автономное образовательное учреждение  
высшего образования  
**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ  
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**  
Факультет систем управления и робототехники

**Лабораторная работа №4**  
**Сегментация изображений**

Студенты: Бахтаиров Р.А.,  
Сайфуллин Д.Р.  
Поток: Тех.Зр R23 1.1  
Преподаватель: Шаветов С.В.

Санкт-Петербург, 2025 г.

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>2</b>
<b>2</b>	<b>Изображение, которое будем использовать для проверки функций</b>	<b>2</b>
<b>3</b>	<b>Изначальный код работы</b>	<b>5</b>
<b>4</b>	<b>Бинаризация изображений</b>	<b>5</b>
4.1	Среднее арифметическое . . . . .	5
4.2	На основе модуля градиента . . . . .	6
4.3	Статистический метод Отсу . . . . .	7
4.4	Адаптивный метод . . . . .	8
<b>5</b>	<b>Сегментация 1: по принципу Вебера</b>	<b>8</b>
<b>6</b>	<b>Сегментация 2: с помощью пространства CIA Lab</b>	<b>9</b>
<b>7</b>	<b>Сегментация 3: текстуры</b>	<b>11</b>
<b>8</b>	<b>Выводы</b>	<b>14</b>
<b>9</b>	<b>Ответы на вопросы</b>	<b>14</b>

## 1 Цель работы

Целью этой работы будет изучение методов сегментации изображений по разным признакам

## 2 Изображение, которое будем использовать для проверки функций



Рис. 1: Изначальное изображение



Рис. 2: Изначальное изображение



Рис. 3: Изначальное изображение



Рис. 4: Изначальное изображение

### 3 Изначальный код работы

Весь код работы будет находиться в прикрепленном файле. Далее в работе будут разбираться методы, которые мы применили в этом коде, но не он сам. В коде по ссылке приведены все разъяснения

<https://github.com/RomanBakhtairov/CV-lab-4>

### 4 Бинаризация изображений

Бинаризацию можно назвать довольно простым и прямолинейным методом преобразования изображения. В её основе лежит замена пикселей по следующему правилу.

$$I_{\text{new}}(x, y) = \begin{cases} 0, & I(x, y) \leq t, \\ 1, & I(x, y) > t \end{cases}$$

Как можно понять, если пиксель по интенсивности меньше нашего порога  $t$ , то он равен 0, иначе 1. Конечно, стоит понимать, что под 1 подразумевается максимальное значение интенсивности. В итоге мы получим чёрно-белое изображение без градации цвета.

Стоит также отметить, что "зону выделения" можно ограничить не только снизу, но и сверху, причём сделать это одновременно. Достаточно лишь добавить ещё один порог:

$$I_{\text{new}}(x, y) = \begin{cases} 0, & I(x, y) \leq t_1, \\ 1, & t_1 < I(x, y) \leq t_2, \\ 0, & I(x, y) > t_2 \end{cases}$$

Вот пример применения данного метода

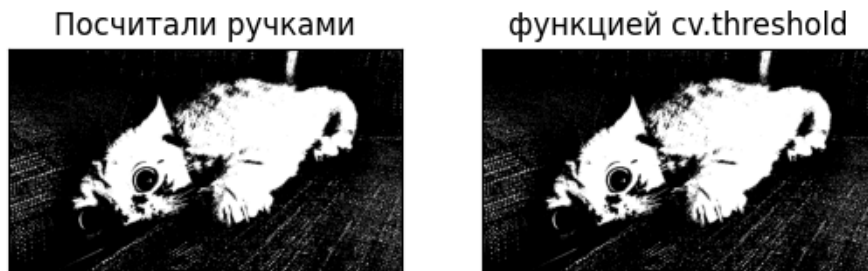


Рис. 5: Сравнение разных реализаций определённого нами порога  $t = 150$

Как можно видеть, метод работает.

Наибольший интерес представляют методы подбора того самого порога, чтобы выделить то, что нам необходимо. Вот некоторые из таких методов

#### 4.1 Среднее арифметическое

Для данного метода определим порог бинаризации как среднее арифметическое интенсивности изображения:

$$t = \frac{I_{\text{max}} - I_{\text{min}}}{2}$$

Вот пример применения данного метода:



## Бинаризация по среднему значению



Рис. 6: Получилось  $t = 128$

У этого метода есть кое-какая проблема. Если изображение в целом тёмное, при этом имеет небольшой контрастный участок с очень высокой яркостью, то существует немаленькая возможность полностью закрыть изображение.

### 4.2 На основе модуля градиента

Идея данного метода заключается в том, чтобы "выделить" интенсивности пикселей, рядом с которыми происходит резкий перепад значений градиента. Таким образом можно провести бинаризации по контуру двух сильно разных по интенсивности объектов. Таким образом мы определяем градиент:

$$G(x, y) = \max \{|I(x + 1, y) - I(x - 1, y)|, |I(x, y + 1) - I(x, y - 1)|\}$$

Таким образом определяем значение порога

$$t = \frac{\sum_{x=0}^{X-1} \sum_{y=0}^{Y-1} I(x, y) G(x, y)}{\sum_{x=0}^{X-1} \sum_{y=0}^{Y-1} G(x, y)}.$$

Пример работы данного алгоритма:

### Бинаризация по градиенту



Рис. 7: Получилось  $t = 101$

## 4.3 Статистический метод Отсу

Данный метод обычно и применяют, когда необходимо выполнить бинаризацию изображения. Идея его заключается в создании двух классов, дисперсию внутри которых минимизируют, а между которыми максимизируют. Ниже представлен алгоритм работы:

1. Вычисление гистограммы интенсивностей изображения и вероятности  $p_i = \frac{n_i}{N}$  для каждого уровня интенсивности, где  $n_i$  — число пикселей с уровнем интенсивности  $i$ ,  $N$  — число пикселей в изображении.
2. Задание начального порога  $t = 0$  и порога  $k \in (0, L)$ , разделяющего все пиксели на два класса, где  $L$  — максимальное значение интенсивности изображения. В цикле для каждого значения порога от  $k = 1$  до  $k = L - 1$ :
  - (а) Вычисление вероятностей двух классов  $\omega_j(k)$  и средних арифметических  $\mu_j(k)$ , где  $j = \overline{1, 2}$ :

$$\omega_1(k) = \sum_{s=0}^k p_s, \quad (1)$$

$$\omega_2(k) = \sum_{s=k+1}^L p_s = 1 - \omega_1(k), \quad (2)$$

$$\mu_1(k) = \sum_{s=0}^k \frac{s \cdot p_s}{\omega_1}, \quad (3)$$

$$\mu_2(k) = \sum_{s=k+1}^L \frac{s \cdot p_s}{\omega_2}. \quad (4)$$

- (б) Вычисление межклассовой дисперсии  $\sigma_b^2(k)$ :

$$\sigma_b^2(k) = \omega_1(k)\omega_2(k) (\mu_1(k) - \mu_2(k))^2.$$



(с) Если вычисленное значение  $\sigma_b^2(k)$  больше текущего значения  $t$ , то присвоить порогу значение межклассовой дисперсии  $t = \sigma_b^2(k)$ .

(d) Наилучший порог будет соответствовать максимуму межклассовой дисперсии

Условно можно воспринимать данный метод, как "кластеризацию" интенсивностей, порогом же будет граница между этими "кластерами".

Ниже представлен пример работы этого алгоритма:



Рис. 8: Получилось  $t = 113$

#### 4.4 Адаптивный метод

Идея работы адаптивного метода заключается в работе не со всем изображением, а лишь с его фрагментами.



Рис. 9: Локально по среднему

Как видим, получилось довольно интересно

## 5 Сегментация 1: по принципу Вебера

Данный способ сегментации идейно прост. Мы разделяем изображение на слои, используя функцию Вебера. Каждому слою мы присваиваем значение минимального элемента внутри него. Сам "шаг" разделения - функция Вебера - основана на идеи,

что человеческий глаз плохо воспринимает значения серого между такими слоями, так что можно применять такую группировку без сильной потери в информации на изображении.

Сама функция Вебера

$$W(I) = \begin{cases} 20 - \frac{12I}{88}, & 0 \leq I \leq 88, \\ 0.002(I - 88)^2, & 88 < I \leq 138, \\ \frac{7(I - 138)}{117} + 13, & 138 < I \leq 255. \end{cases}$$

Пример применения сегментации с помощью функции Вебера



Рис. 10: Гоголь

Видно разделение на слои, но также есть некоторая зернистость

## 6 Сегментация 2: с помощью пространства CIE Lab

В отличие от цветового профиля RGB в цветовом пространстве Lab цвет пикселей отделён от яркости. Это позволяет работать с объектам одного цвета, даже если они освещены по-разному. L - яркость от 0 до 100, а - от зелёного(-128) до красного(127), b - от синего до жёлтого(в том же диапазоне). Для каждого пикселя мы вычисляем значения евклидовой нормы от разности необходимых параметров a и b к тем, которые принадлежат пикселю. В итоге получается двухмерная карта значений расстояния между пикселем и цветом. Затем проводим тоже самое с другими наборами пробных цветов кластеризации (a,b)

В итоге мы сравниваем значения на каждой из этих карт расстояний до каждого цвета и присуждаем пиксель к тому классу, к которому он окажется ближе всех.

Данный метод работает шире, чем, к примеру, подбор только красных пикселей. Он позволяет выделить пиксели, которые, к примеру, "скорее красные, чем синие". Что позволяет выполнять менее ограниченную кластеризацию.

Для сегмента 1



Рис. 11: Выделили небо

Для сегмента 2

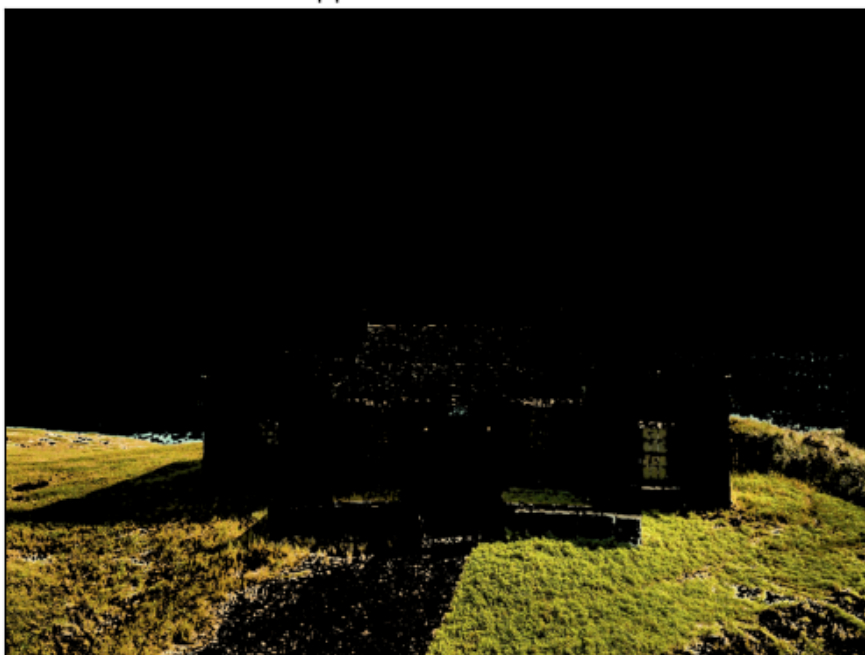


Рис. 12: Выделили траву

Для сегмента 3



Рис. 13: Целились в дом

Данный метод справился хорошо, хотя метки пришлось выбирать вручную

## 7 Сегментация 3: текстуры

В данном разделе попытаемся сегментировать изображение, используя разделение по характеристикам текстур.

За эти самые характеристики текстур отвечают такие параметры, как дисперсия, среднее значение, энтропия, мера гладкости и т.д. Вот методы вычисления некоторых из них:

Центральный момент порядка  $n$

$$\mu_n(z) = \sum_{i=0}^{L-1} (z_i - m)^n p(z_i),$$

где  $L$  — число уровней интенсивностей,  $m$  — среднее значение случайной величины  $z$ :

Среднее значение:

$$m = \sum_{i=0}^{L-1} z_i p(z_i).$$

Мера гладкости:

$$R = 1 - \frac{1}{1 + \sigma^2(z)},$$

Стандартное отклонение:

$$s = \sigma(z).$$

Энтропия:

$$E = - \sum_{i=0}^{L-1} p(z_i) \log_2 p(z_i). \quad (6)$$

Мера однородности:

$$U = \sum_{i=0}^{L-1} p^2(z_i).$$

Для изначальной сегментации по текстурам посчитаем любой из данных параметров в ограниченной окрестности для каждого пикселя, после чего применим бинаризацию. В итоге получим маску, которая выражает место нахождения текстуры.

После такой сегментации обычно появляется довольно много шума, поэтому с помощью специальных функций мы удаляем все объекты с небольшой площадью и заполняем пикселями пустоты (где это необходимо)

Вот, что получилось у нас, используя сегментацию по мере относительной гладкости:

Сегментировали



Рис. 14: Найденная нами маска

Просто бинаризовали Отсу



Рис. 15: Просто применили Отсу

Данный метод сработал лучше, чем бинаризация по Отсу  
Как можем видеть данный метод довольно хорошо подцепил необходимую текстуру.  
Давайте теперь попробуем оценить некоторые параметры для каждой текстуры

- (a) По среднему(начальному моменту первого порядка)  
 $m_{clouds} \approx 110$  - грубо-периодическая  
 $m_{sky} \approx 14$  - очень гладкая
- (b) По квадратичному отклонению s  
 $s_{clouds} \approx 76.4$  - грубая  
 $s_{sky} \approx 23.8$  - периодически -гладкая
- (c) И то, что называли 3-им моментом, но по факту должно быть коэффициентом асимметрии(так как иначе получаются слишком большие значения)  
 $\mu_{3_{clouds}} \approx -0.32$  - очень грубая  
 $\mu_{3_{sky}} \approx 1.16$  - сильно периодическая

Из чего можно сделать вывод, что облака - грубая текстура, а небо гладкая. Думаю, "неуверенность" некоторых коэффициентов связана с выбросами и тем, что тяжело разделить данное изображение только на 2 текстуры.

## 8 Выводы

В данной работе мы научились сегментировать изображения по нескольким принципам: от отделения всех ненужных цветов по нижней границе до работы с цветами и их кластеризацией. Научились определять текстуры на изображениях, а также познакомились с эффективными методами бинаризации.

## 9 Ответы на вопросы

- (a) В каких случаях целесообразно использовать сегментацию по принципу Вебера?  
-Когда в изображении нету перемешивание цветов с сильными контрастами, что может привести к появлению зернистости. Данный метод также подходит, когда не страшны потери небольших деталей градации интенсивности цветов
- (b) Какие значения имеют цветовые координаты a и b цветового пространства CIE Lab в полутонном изображении?  
-Значения этих параметров не должно влиять на полутонное изображение. Идея формата Lab в том, что за полутонное изображение полностью отвечает канал L
- (c) Зачем производить сегментацию в цветовом пространстве CIE Lab, а не в исходном RGB?  
-Иначе бы на нашу сегментацию очень сильно влияла бы освещённость объектов



(d) Что такое цветовое пространство и цветовой охват?

-цветовое пространство - пространство, с помощью которых мы описываем цвета реального мира в цифровых технологиях. К таким, к примеру, относятся RGB или Lab

цветовой охват - набор цветов, который может представить наше цветовое пространство. Например для RGB таких цветов  $255^3$