

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Таныгин Максим Олегович

Должность: и.о. декана факультета фундаментальной и прикладной информатики

Дата подписания: 21.09.2023 13:12:44

Уникальный программный ключ:

65ab2aa0d384efe8480e6a4c888edd0c475e411a

МИНОБРАЗОВАНИЯ РОССИИ

Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Юго-Западный государственный университет»
(ЮЗГУ)

Кафедра программной инженерии



УТВЕРЖДАЮ:

проректор по учебной работе

Локтионова О.Г.

2022 г.

МЕТОДЫ СЕГМЕНТАЦИИ ИЗОБРАЖЕНИЙ

Методические указания для выполнения лабораторных работ по
дисциплине «Методы и алгоритмы обработки изображений» для
студентов направления подготовки 09.03.04
«Программная инженерия»

Курск 2022

УДК 004.932

Составитель: Р.А. Томакова

Рецензент

Кандидат технических наук, доцент *А.В. Малышев*

Методы сегментации изображений: методические указания для проведения лабораторных работ и выполнения самостоятельной внеаудиторной работы по дисциплине «Методы и алгоритмы обработки изображений» для студентов всех форм направления подготовки 09.03.04 «Программная инженерия»/ Юго-Зап. гос. ун-т; сост. Р.А. Томакова. Курск, 2022. 19с.

Рассмотрена методика изучения методов сегментации изображений, с использованием встроенных функций системы MATLAB, реализующих операции по сегментации изображений, и приобретение практических навыков применения этих функций, применяемых для работы с файлами изображений.

Методические указания составлены в соответствии с федеральным государственным образовательным стандартом высшего образования направления подготовки 09.03.04 «Программная инженерия» и на основании учебного плана направления подготовки 09.03.04 «Программная инженерия» (профиль «Разработка программно-информационных систем»).

Предназначены для студентов, обучающихся по направлению подготовки 09.03.04 «Программная инженерия» (профиль «Разработка программно-информационных систем») всех форм обучения.

Текст печатается в авторской редакции

Усл. печ. л.	Подписано в печать	Формат 60×84 1/16.	
	. Уч. - изд. л.	. Тираж 25 экз. Заказ.	Бесплатно.
	Юго-Западный государственный университет. 305040, г. Курск, ул. 50 лет Октября, 94.		

МЕТОДЫ СЕГМЕНТАЦИИ ИЗОБРАЖЕНИЙ

Цель и задачи лабораторного занятия (лабораторной работы): методов сегментации изображений, с использованием встроенных функций системы MATLAB, реализующих операции сегментации изображений;

приобретение практических навыков и умений формирования этих функций, применяемых для работы с файлами изображений.

Планируемые результаты обучения (формируемые знания, умения, навыки и компетенции):

Код и наименование индикатора достижения компетенции, закрепленного за дисциплиной : ПК-2.1 -Готовит отчёты, публикации, презентации по результатам выполненной работы

Необходимые материально-техническое оборудование и материалы:

1. Класс ПЭВМ - Athlon 64 X2-2.4; Cel 2.4, Cel 2.6, Cel 800.
2. Мультимедиа центр: ноутбук ASUS X50VL PMD T2330/14"/1024Mb/ 160Gb/ сум-ка/проектор inFocus IN24+ .
3. Экран мобильный Draper Diplomat 60x60
4. Доступ в сеть Интернет.

Шкала оценивания и критерии оценивания выполненной лабораторной работы:

Форма контроля	Минимальный балл		Максимальный балл	
	балл	примечание	балл	примечание
Лабораторная работа № 1 (Работа с файлами различных типов изображений в среде MATLAB)	2	Выполнил, но «не защитил»	4	Выполнил и «защитил»

План проведения лабораторного занятия (лабораторной работы)

1. ЗАДАНИЯ ДЛЯ САМОСТОЯТЕЛЬНОГО РЕШЕНИЯ

1. Выполнить сегментацию изображений файла Clouds.bmp методом разделения.
2. Выполнить сегментацию изображений файла Construc.bmp методом выращивания областей.
3. Выбрать цветовые области из изображения файла bike.bmp, задавая индексы с помощью гистограммы.
4. Выполнить яркостный срез полноцветного изображения файла bike.bmp, задавая диапазон r от 0.2 до 0.8; g от 0.2 до 0.7; b от 0.1 до 0.7.

2. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Сегментация изображения представляет собой разделение изображения на несколько областей (сегментов, т.е. формируется множество пикселей, которые также называют суперпикселями) по сходству свойств (признаков) в их точках [2, 3, 4].

Цель сегментации заключается в упрощении и/или изменении изображения, так чтобы его было проще и легче анализировать. Сегментация изображений обычно используется для того, чтобы выделить объекты и границы (линии, кривые, и т. д.) на изображениях. Более точно, сегментация изображений – это процесс присвоения меток каждому пикселю изображения, так что пиксели с одинаковыми метками имеют общие визуальные характеристики [6, 7, 8].

Если R – область изображения, сегментация – это процесс разбиения этой области на R_1, R_2, \dots, R_n областей, удовлетворяющих условиям:

$$1) \bigcup_{i=1}^n R_i = R;$$

$$2) \forall R_i, i = \overline{1, n} - \text{связные области};$$

$$3) R_i \cap R_j = \emptyset, \forall i, j, i \neq j;$$

Признаки подразделяются на *естественные* и *искусственные*. Естественные признаки устанавливаются простым (визуальным) анализом изображения, а искусственные – в результате специальной обработки различных измерений [9, 10].

Примерами естественных признаков являются *структура, текстура, яркость объекта*. Примеры искусственных признаков: *гистограммы распределения яркости, спектр и др.*

К основным видам сегментации изображений относится *сегментация по яркости, цветовым координатам, контурам, форме*.

Сегментация – весьма трудный и до конца не алгоритмизированный процесс для произвольных изображений. Наиболее распространены методы сегментации, основанные на определении *однородных яркостей* (цветов) или *однородностей типа текстур*.

Методы сегментации можно разделить на два класса: *автоматические* – не требующие взаимодействия с пользователем и *интерактивные* – использующие пользовательский ввод непосредственно в процессе работы. Для анализа биомедицинских материалов более интересны автоматические методы сегментации.

Задачи автоматической сегментации делятся *на два класса*:

1) *выделение областей изображения с известными свойствами;*

2) *разбиение изображения на однородные области.*

В первом случае задача сегментации состоит в поиске определенных областей, о которых имеется априорная информация (например, мы знаем цвет, форму областей, или интересующие нас области представляют собой изображения известного объекта). Методы этой группы узко специализированы для каждой конкретной задачи. Сегментация в такой постановке используется в основном в задачах машинного зрения (например, анализ сцен, поиск объектов на изображении).

Во втором случае априорная информация о свойствах областей не используется, но на само разбиение изображения накладываются некоторые условия (например, все области должны быть однородны по цвету и текстуре). Так как при такой постановке задачи сегментации *не используется априорная информация* об изображенных объектах, то *методы этой группы*

универсальны и применимы к любым изображениям. В основном сегментация в такой постановке применяется на начальном этапе решения задачи, для того чтобы получить представление изображения в более удобном виде для дальнейшей работы.

На результаты сегментации очень сильно влияет как выбор критерия однородности, так и способ построения однородных регионов. Далеко не всегда для изображения есть единственно «правильная» сегментация, и далеко не всегда задача сегментации имеет единственное решение. По той же причине нет и объективного критерия оценки качества разбиения изображения.

Для грубой оценки качества метода в конкретной задаче обычно фиксируют несколько свойств, которыми должна обладать хорошая сегментация. Качество работы метода оценивается в зависимости от того, насколько полученная сегментация обладает этими свойствами [10]. Наиболее часто используются следующие свойства:

- 1. Однородность регионов (однородность цвета или текстуры).*
- 2. Непохожесть соседних регионов.*
- 3. Гладкость границы региона.*
- 4. Маленькое количество мелких «дырок» внутри региона и т.д.*

2.1 Сегментация методом выращивания областей

Для сегментации изображения применяется метод выращивания областей, основная идея которого заключается в группировании пикселей изображения в более крупные области по заранее заданным критериям роста. Определяются «*центры кристаллизации*», а затем на них наращиваются области путем добавления к каждому центру тех соседних пикселей, которые по своим свойствам близки к центру кристаллизации (например, имеют яркость или цвет в определенном диапазоне).

Сегментация изображений в среде MATLAB позволяет осуществить различные методы обработки изображений [1, 5]. Например, применение функции **regiongrow** позволяет выполнять выращивание областей.

Синтаксис

text[g, NR, SI, TI] = regiongrow(f, S, T),

где **f** – это сегментируемое изображение; **S** – массив (с размерами как у **f**) или скаляр. Если **S** – массив, то он содержит 1 в тех позициях, где расположены центры кристаллизации и 0 во всех остальных местах. Если **S** является скаляром, то он задает значение яркости пикселей, которые становятся центрами кристаллизации.

T может быть массивом (с размерами, как у **f**) или скаляром. Если **T** – массив, то его элементы являются локальными пороговыми значениями для **f**. *Скаляр T определяет глобальный порог.*

Если **S=a** и **T=b** при сравнении яркости считаем, что пиксель похожий на **a**, если модуль разности их значений не больше **b**.

Выходом **g** является сегментированное изображение, причем элементы каждой области помечены одним и тем же целым числом.

NR – параметр, равен числу выделенных областей.

SI – изображение, на котором отмечены центры кристаллизации.

TI – изображение, содержащее пиксели, связной области.

SI, TI – изображения с размерами как у **f**.

Задание 1. Создать функцию **regiongrow**.

```
function [g, NR, SI, TI] = regiongrow(f, S, T)
f=double (f);
if numel(S) == 1
SI = f == S;
S1 = S
else
SI = bwmorph(S, 'shrink', Inf);
J = find(SI);
S1 = f(J)
end
TI = false(size(f));
for K = 1:length(S1)
seedvalue = S1(K);
S = abs(f - seedvalue) <= T;
TI = TI | S;
```

```
end
[g, NR] = bwlabel(imreconstruct(SI, TI));
```

Задание 2. Выполнить сегментацию методом наращивания областей для изображения, хранящегося в файле Finance.bmp, используя функцию **regiongrow**.

```
[x,map] = imread('c:\image\Finance.bmp');
I = im2double(ind2gray(x,map));
figure,imshow(I)
S = 0.9783; T = 0.0651; % эти значения находятся
экспериментально (взяты из анализа изображения)
[g, NR, SI, TI] = regiongrow(I, S, T);
figure,imshow(TI)
```

Сегментация методом разделения

Метод разделения используется для сегментации изображений различных видов.

В основу метода положена идея разбиения изображения (или выделенного фрагмента) на непересекающиеся блоки, которые с помощью введенного критерия проверяются на однородность.

Функция сегментации полутоновых изображений методом разделения qtdecomp

Синтаксис

A = qtdecomp (I, threshold, mindim)

Функция **qtdecomp** осуществляет сегментацию полутоновых изображений *методом разделения*. В функции **qtdecomp** каждый блок разбивается на 4 неперекрывающихся блока одинакового размера.

На первом шаге алгоритма блоком считается все изображение. Мельчайшим по размерам является блок, который нельзя разделить на 4 блока одинакового размера, т. е. такой блок, у которого число строк или число столбцов нечетное. Таким образом, в функции **qtdecomp** рекомендуется использовать изображения с размерами, равными степеням двух.

Функция **A = qtdecomp(I,threshold,mindim)** осуществляет сегментацию полутонового изображения I методом разделения и помещает результат в разреженный массив A (тип данных sparse MATLAB). Элементам матрицы A(r,c), соответствующим

координатам левых верхних углов блоков на исходном изображении I , присваиваются значения, определяющие размеры каждого блока.

В качестве критерия однородности выбирается критерий, основанный на утверждении: *блок считается однородным, если разница между максимальным и минимальным значением пикселей блока меньше параметра $threshold$.*

Параметр **mindim** определяет минимальный размер блока.

Функция получения блоков из квадродерева результатов сегментации `qtgetblk`

Синтаксис

`[vals, idx] = qtgetblk(I, A, dim)`

Функция возвращает в массив **vals** все блоки размером **dim**, получившиеся в результате сегментации полутонового изображения I с помощью функции **qtdecomp**. В параметре A передается разреженный массив, описывающий квадродерево с результатами сегментации. Координаты левых верхних углов блоков, помещенных в массив **vals**, находятся в векторе **idx**. Если нет ни одного блока размером **dim**, то всем возвращаемым параметрам присваиваются значения пустых **vals** матриц.

Функция замены блоков – результатов сегментирования `qtsetblk`

Синтаксис

`ID = qtsetblk(Is, A, dim, vals)`

Функция создает новое полутоновое изображение **ID**, заменяя в исходном полутоновом изображении все блоки размера **dim**, полученные в результате сегментации с помощью функции **qtdecomp**, на блоки из массива **vals**. В параметре A передается разреженный массив, описывающий квадродерево с результатами сегментации. Данная функция используется для преобразования изображения в соответствии с результатами сегментации методом деления.

Рассмотрим работу функции **qtdecomp** совместно с функциями **qtgetblk** и **qtsetblk** для полутонового изображения размера 8×8 пикселей. Формат представления данных – `uint8`. Будем считать, что блок изображения является однородным, если величина разброса яркостей пикселей в блоке не превышает 10 градаций яркости. Установим минимально возможный размер блока. В нашем случае он равен двум.

Будем считать, что к объекту относятся блоки, средняя яркость которых не превышает 50. Требуется изменить исходное изображение так, чтобы пикселям блоков, относящихся к объекту, было присвоено значение 1, а пикселям блоков, не относящихся к объекту, 0.

Задание 3. Выполнить сегментацию небольшого текстового изображения методом разделения.

Исходное изображение:

```
>> I = [ 10 11 10 15 20 25 47 51
11 14 17 13 27 29 52 55
12 13 11 10 24 47 56 60
13 14 11 13 49 54 74 77
15 16 43 48 79 82 87 86
17 18 45 50 85 80 80 84
29 51 50 59 80 83 83 85
59 61 58 61 81 85 86 88 ];
```

Сегментация методом разделения: размер минимального блока 2×2 ; блок считается однородным, если в его пределах яркость изменяется меньше, чем на 10 градаций.

```
>> A = qtdecomp(I,10,2);
```

Для удобства визуального анализа предварительно преобразуем разреженную матрицу **A** в обычную матрицу **M** с помощью функции **full**.

```
>> M = full(A)
```

```
M =
4 0 0 0 2 0 2 0
0 0 0 0 0 0 0 0
0 0 0 0 2 0 2 0
0 0 0 0 0 0 0 0
2 0 2 0 4 0 0 0
```

```

0 0 0 0 0 0 0 0
2 0 2 0 0 0 0 0
0 0 0 0 0 0 0 0

```

В результате сегментации получили 2 блока размером 4×4 (левая верхняя и правая нижняя части изображения и 8 блоков размером 2×2).

Переберем в цикле все возможные размеры блоков: 8, 4, 2.

```

>> dim = 8;
>> while dim >= 2
% получить в переменной blocks все блоки размера dim.
[blocks,idx] = qtgetblk(I,A,dim);
[x y n] = size(blocks);
% если блоки такого размера есть в quadro-дереве,
if n > 0
% то перебираем все блоки размера dim
for j = 1:n
% если среднее значение яркости пикселей в пределах блока
меньше 50
if (mean2(blocks(:,:,j)) < 50)
% то заменяем значения всех пикселей блока на 1,
blocks(:,:,j) = ones(dim,dim);
else
% иначе заменяем значения всех пикселей на 0.
blocks(:,:,j) = zeros(dim,dim);
end;
end % end for
% устанавливаем новые значения всех пикселей размера dim
I = qtsetblk(I,A,dim,blocks);
end; % end if
dim = dim/2;
end % end while % ссылка на 1 из списка литературы
% получившееся изображение I
>> I
I =
1 1 1 1 1 1 0 0
1 1 1 1 1 1 0 0
1 1 1 1 1 1 0 0

```

```

1 1 1 1 1 1 0 0
1 1 1 1 0 0 0 0
1 1 1 1 0 0 0 0
0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0

```

Задание 4. Выполнить сегментацию реального изображения из файла cotton3.bmp

```

rgb = imread('c:\Image\cotton3.bmp');
I = im2double(rgb2gray(rgb));
figure,imshow(I)
T=graythresh(I);
A = qtdecomp(I,0.1,2);
dim = 8;
while dim >= 2
[blocks,idx] = qtgetblk(I,A,dim);
[x y n] = size(blocks);
if n>0
for j = 1:n
if (mean2(blocks(:,j))<T)
blocks(:,j) = ones(dim,dim);
else
blocks(:,j) = zeros(dim,dim);
end
end
I1 = qtsetblk(I,A,dim,blocks);
end
dim = dim/2;
end
figure,imshow(I1)

```

Функция выбора интересующей области по цвету roicolor

Синтаксис

BW = roicolor(S,low,high)

BW = roicolor(S,v)

Для любого варианта вызова функции **roicolor** бинарное изображение формируется по следующему *алгоритму*:

пикселю бинарного изображения $BW(r, c)$ присваивается значение 1, если яркость пикселя $S(r, c)$ исходного полутонового изображения или индекс $S(r, c)$ палитрового изображения принадлежит диапазону $[low, high]$ или любому из значений вектора v . В противном случае $BW(r, c)$ присваивается значение 0.

Задание 5. Выбрать цветовые области из изображения файла chip.bmp, задавая индексы с помощью гистограммы.

```
>> [x,map]=imread('C:\Image\chip.bmp');
>> figure,imhist(x,map),title('histogramma')
>> figure,imshow(x,map),
>> bw=roicolor(x,9,12);
>> figure,imshow(bw),title('9 - 12')
>> x1=immultiply(bw,x);
>> figure,imshow(x1,map),title('9 - 12')
>> bw=roicolor(x,3,8);
>> figure,imshow(bw),title('3 - 8')
>> x1=immultiply(bw,x);
>> figure,imshow(x1,map),title('3 - 8')
```

Яркостный срез

Этот метод позволяет выделить определенный диапазон яркости:

$$\forall A_{i,j} \quad B_{i,j} = \begin{cases} 0, & A_B < A_{i,j} < A_H \\ A_{i,j}, & A_H \leq A_{i,j} \leq A_B \end{cases}$$

или

$$B_{i,j} = \begin{cases} 0, & A_B < A_{i,j} < A_H \\ K, & A_H \leq A_{i,j} \leq A_H \end{cases}$$

где A_H – нижнее граничное значение определяемого диапазона яркости; A_B – верхнее граничное значение определяемого диапазона яркости.

Для выполнения яркостного среза можно использовать функцию **imrindex**.

Синтаксис

$P = \text{impixel}(S,c,r)$

Функция **impixel** возвращает значения красной, зеленой и синей составляющих цвета для заданных координат – c и r – векторов значений столбцов и строк.

Задание 6. Выполнить яркостный срез полноцветного изображения файла cotton3.bmp.

```
[img] = imread('c:\Image\cotton3.bmp');
[m,n,k] = size(img)
img = im2double(img);
R = zeros(m,n,3);
z = [0.1,0.8;0.1,0.8; 0.1,0.9];
for y = 1:m
for x = 1:n
b = impixel(img,x,y);
if ((b(1)>= z(1,1))&(b(1)<= z(1,2)))& ((b(2)>= z(2,1))&(b(2)<=
z(2,2))) ... & ((b(3)>= z(3,1))&(b(3)<= z(3,2)))
R(x,y,1) = b(1); R(x,y,2) = b(2); R(x,y,3) = b(3);
else
R(x,y,1) = 0; R(x,y,2) = 0; R(x,y,3) = 0;
end
end
end
figure, imshow(img)
figure, imshow(R)
```

3. ПРИМЕРЫ ВЫПОЛНЕНИЯ ЗАДАНИЙ

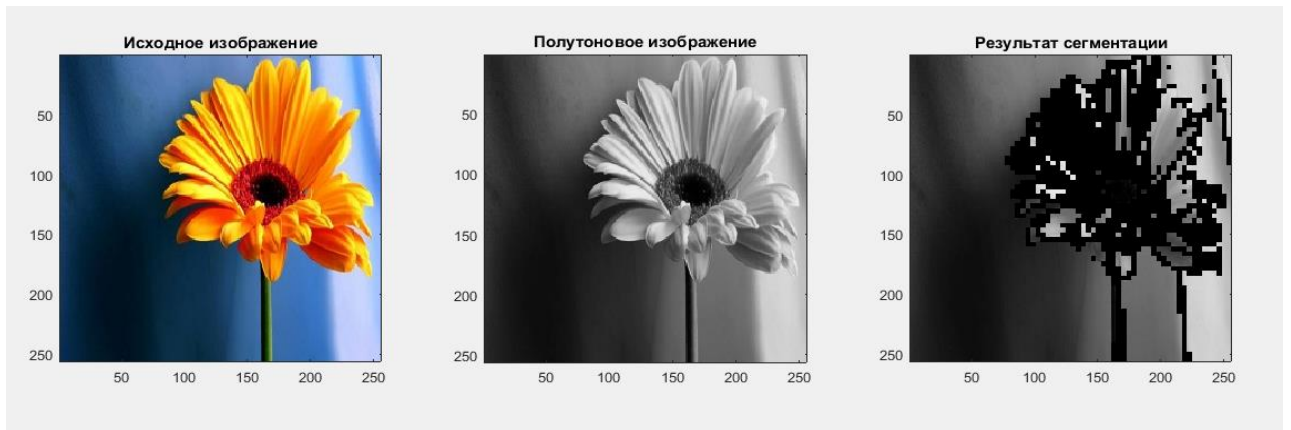


Рисунок 1 – Результат сегментации изображения методом разделения

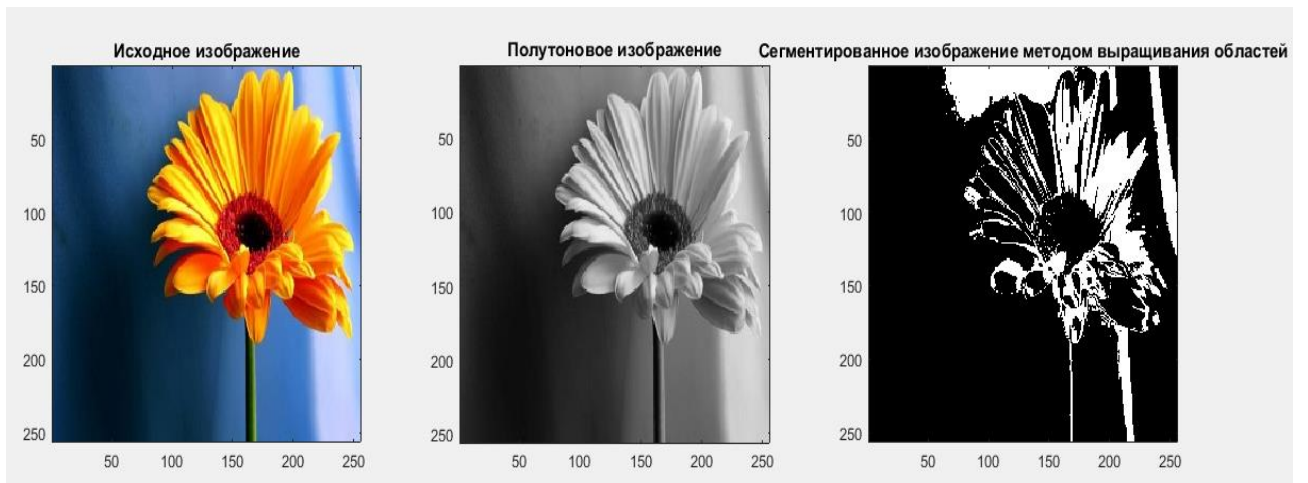


Рисунок 2 – Результат сегментации изображения методом выращивания областей

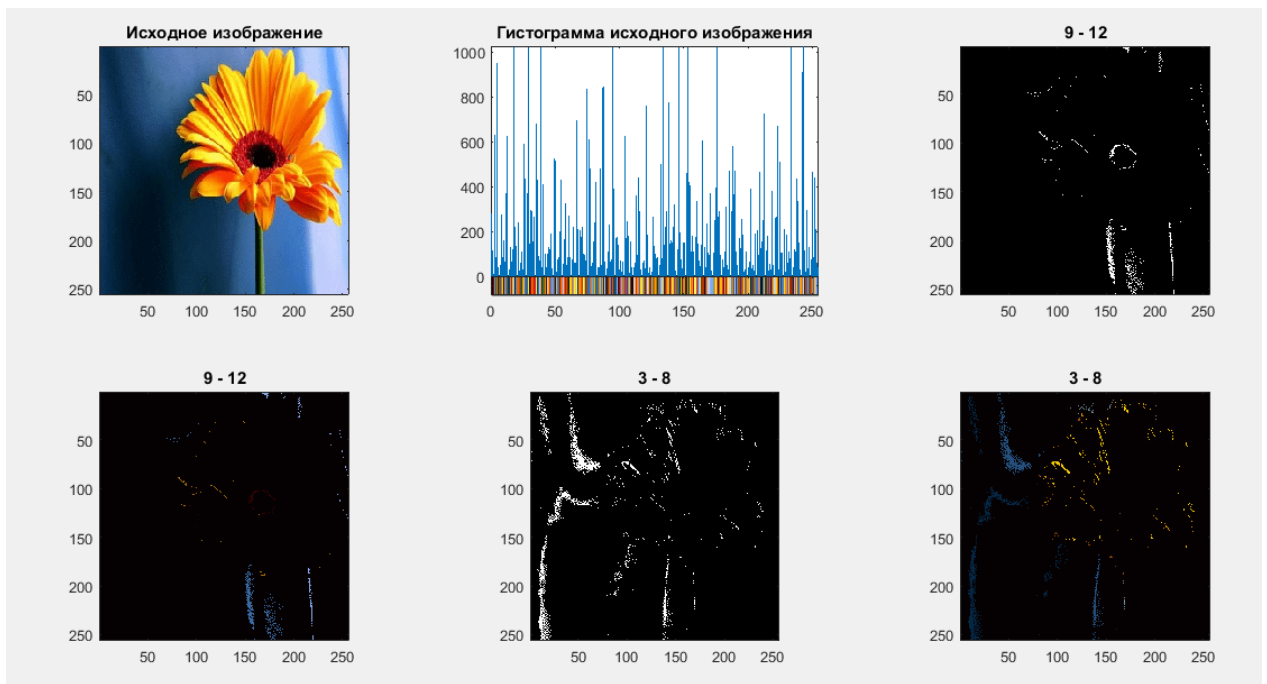


Рисунок 3 – Результат выбора цветных областей из изображения с заданными индексами с помощью гистограммы

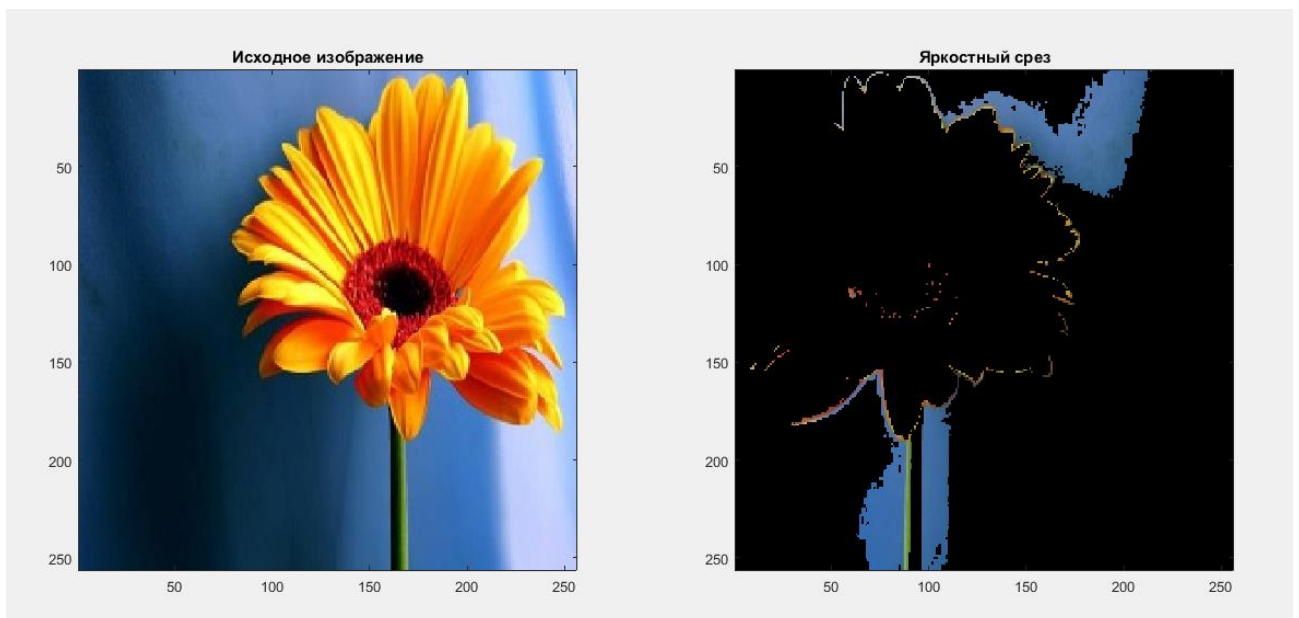


Рисунок 4 – Результат выполнения яркого среза полноцветного изображения

4. КОНТРОЛЬНЫЕ ВОПРОСЫ

1. В чем заключается сегментация изображения?
2. Какие признаки используются для сегментации?
3. В чем заключается алгоритм метода выращивания областей, использующийся для сегментации изображения?
4. В чем заключается алгоритм метода разделения, использующийся для сегментации изображения?
5. Какие параметры являются входными параметрами функции сегментации методом разделения?
6. В чем заключается преобразование яркостного среза?
7. Какие параметры возвращает функция **impixel**?

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Томакова, Р. А. Методы и алгоритмы цифровой обработки изображений : учебное пособие для студентов всех форм обучения по направлениям подготовки 09.03.03 Прикладная информатика; 09.03.04 «Программная инженерия», 09.04.04 «Программная инженерия» (профиль «Разработка информационно-вычислительных систем») / Р. А. Томакова, Е. А. Петрик ; Юго-Зап. гос. ун-т. - Курск : Университетская книга, 2020. - 310 с. - Загл. с титул. экрана. – Текст : электронный.

3. Яне, Бернд. Цифровая обработка изображений : [учебное пособие] / пер. с англ. А. М. Измайловой. - М. : Техносфера, 2007. - 584 с. : ил. + 1 эл. опт. диск (CD-ROM). - (Мир цифровой обработки. XI. 06). - Библиогр.: с. 575-583 (221 назв.). - ISBN 978-5-94836-1 22-2 : 285.00 р. - Текст : непосредственный.

4. Красильников, Н. Н. Цифровая обработка 2D- и 3D-изображений : учебное пособие / Н. Н. Красильников. - СПб. : БХВ-Петербург, 2011. - 608 с. : ил. - ISBN 978-5-9775-07 00-4 : 372.50 р. - Текст : непосредственный.

5. Томакова, Римма Александровна. Интеллектуальные технологии сегментации и классификации биомедицинских изображений : монография / Р. А. Томакова, С. Г. Емельянов, С. А. Филист ; Юго-Западный государственный университет. - Курск : ЮЗГУ, 2012. - 222 с. - Текст : электронный.

6. Томакова, Римма Александровна . Теоретические основы и методы обработки и анализа микроскопических изображений биоматериалов : монография / Р. А. Томакова, С. А. Филист, С. Г. Емельянов ; МИНОБРНАУКИ РФ, Юго-Западный государственный университет. - Курск : ЮЗГУ, 2011. - 202 с. - Текст : электронный.

7. Методы цифровой обработки изображений : учебное пособие / А. Е. Архипов, С. В. Дегтярев, С. С. Садыков, С. Н. Серeda, В. С. Титов. - Курск : КГТУ, 2002 - Ч. 2. - 115 с. - Текст : непосредственный.

9. Технологии сетей связи. Особенности кодирования цифровых факсимильных сообщений : учебное пособие / А. И. Атакищев [и др.] ; Министерство образования Российской

Федерации, Курский государственный технический университет. - Курск :КурскГТУ, 2002. - 159 с. :ил.табл. - Имеется электрон. аналог. - ISBN 5-7681-0111-X : 62.00 р. - Текст : непосредственный.

10. Гонсалес, Р. Цифровая обработка изображений: практические советы : монография / Р. Гонсалес, Р. Вудс ; пер. П. А. Чочиа, Л. И. Рубанова. – 3-е изд., испр. и доп. – Москва :Техносфера, 2012. – 1104 с. :- (Мир цифровой обработки).– URL: <https://biblioclub.ru/index.php?page=book&id=233465> (дата обращения: 23.03.2022).

11. Гонсалес, Р. Цифровая обработка изображений / Р. Гонсалес, Р. Вудс. - М. :Техносфера, 2006. - 1072 с. - (Мир цифровой обработки). - ISBN 5-94836-028-8 : 394.66 р. - Текст : непосредственный.

12. Фисенко, В.Т. Компьютерная обработка и распознавание изображений / В.Т. Фисенко, Т.Ю. Фисенко. – СПб.: СПбГУ ИТМО, 2008. – 192 с.

13. Гонсалес, Р. Цифровая обработка изображений в среде Matlab / Р. Гонсалес, Р. Вудс, С. Эддинс. – М.: Техносфера, 2006. – 616 с.