Санкт-Петербургский национальный исследовательский университет информационных технологий механики и оптики

Цифровая обработка сигналов

**Лабораторная работа №4**

**Студент:**

Черезов Игорь Юрьевич

**Группа:**

P3400

**Преподаватель:**

Тропченко Андрей Александрович

Санкт-Петербург

2020

# Цель работы

Определение возможностей применения медианного фильтра для подавления импульсных помех

# Задание

**Задание № 1**

1. Используя полученное изображение лица человека, при помощи приведенной в разделе 2.1.1 функции осуществите его сегментацию методом центроидного связывания. Необходимо выделить лицо человека, добившись выделения максимально возможной области;
2. По результатам выполнения п.1, выведите на экран соответствующий фрагмент полутонового изображения, используя результат сегментации как маску. Полученные изображения и программный код (за исключением программного кода реализации функции *regiongrow*) поместите в отчет).

**Задание № 2**

Используя представленный выше программный код, осуществите квадратичное разложения полученного для данного упражнения изображения. Постарайтесь выбрать порог, обеспечивающий наиболее точное выделение основной части изображения.

Программный код и полученные результаты поместите в отчет.

**Задание № 3**

Используя функцию *splitmerge* примените метод разделения-слияния на полученном для данного задания изображении. Постарайтесь выбрать параметры *matOj, minMean, maxMean* таким образом, чтобы добиться максимально возможного выделения основного объекта изображения.

Программный код и полученные изображения поместите в отчет.

**Задание № 4**

Не используя функцию *edge* создайте три функции *Roberts.m, Sobel.m, Previtt.m*, реализующих работу каждого из рассмотренных выше фильтров с помощью масок по алгоритму, представленному на рисунке 8.

Выполнение функций проверьте на заранее полученных для этого задания изображениях разного типа.

Программный код и полученные изображения поместите в отчет. В отчете сделайте выводы о фильтрах Собеля и Превитта, сравнив их эффективность работы по полученным результатам.

# Выполнение

## Задание 1

Код

%Read image from file

img = imread("wonder-woman-bw.jpg");

%Create seed array and fill in the seeds

seeds = zeros(size(img));

seeds(300, 300) = 1;

seeds(240, 352) = 1;

seeds(240, 360) = 1;

%Call regiongrow function to seperate image by regions

[g NR SI] = regiongrow(img, seeds, 30);

%Show created images and save them to files

figure, imshow(g);

imwrite(g, "ww-region.jpg");

figure, imshow(img);

imwrite(SI, "ww-seeds.jpg");

figure, imshow(SI);

img(~g) = 0;

figure, imshow(img);

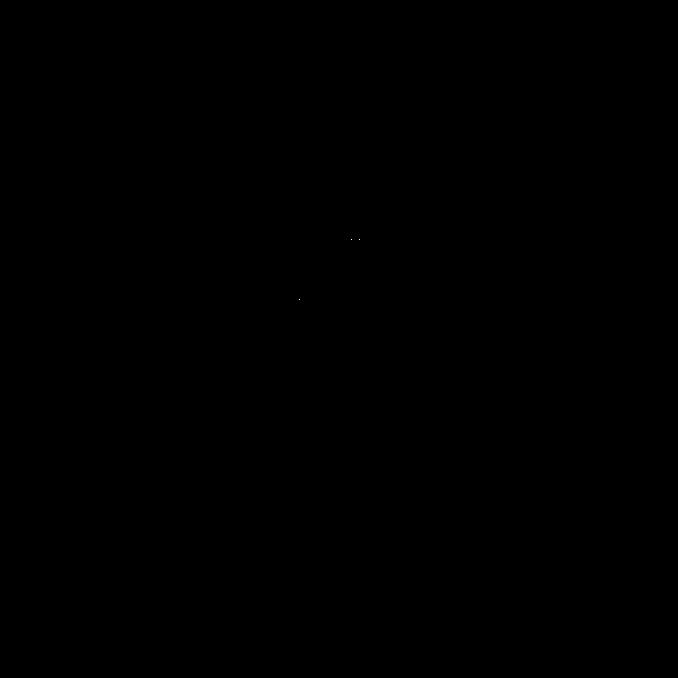
imwrite(img, "ww-mask.jpg");



Изображение 1 Оригинальное изображение



Изображение 2 Результат центроидного связывания



Изображение 3 Точки кристаллизации



Изображение 4 Наложение маски на исходное изображение

## Задание 2

i = imread("batman\_jl.jpg");

s = qtdecomp(i, 0.5);

blocks = repmat(uint8(0), size(s));

for dim = [512 256 128 64 32 16 8 4 2 1];

numblocks = length(find(s==dim));

if (numblocks > 0)

values = repmat(uint8(1),[dim dim numblocks]);

values(2:dim,2:dim,:) = 0;

blocks = qtsetblk(blocks,s,dim,values);

end

end

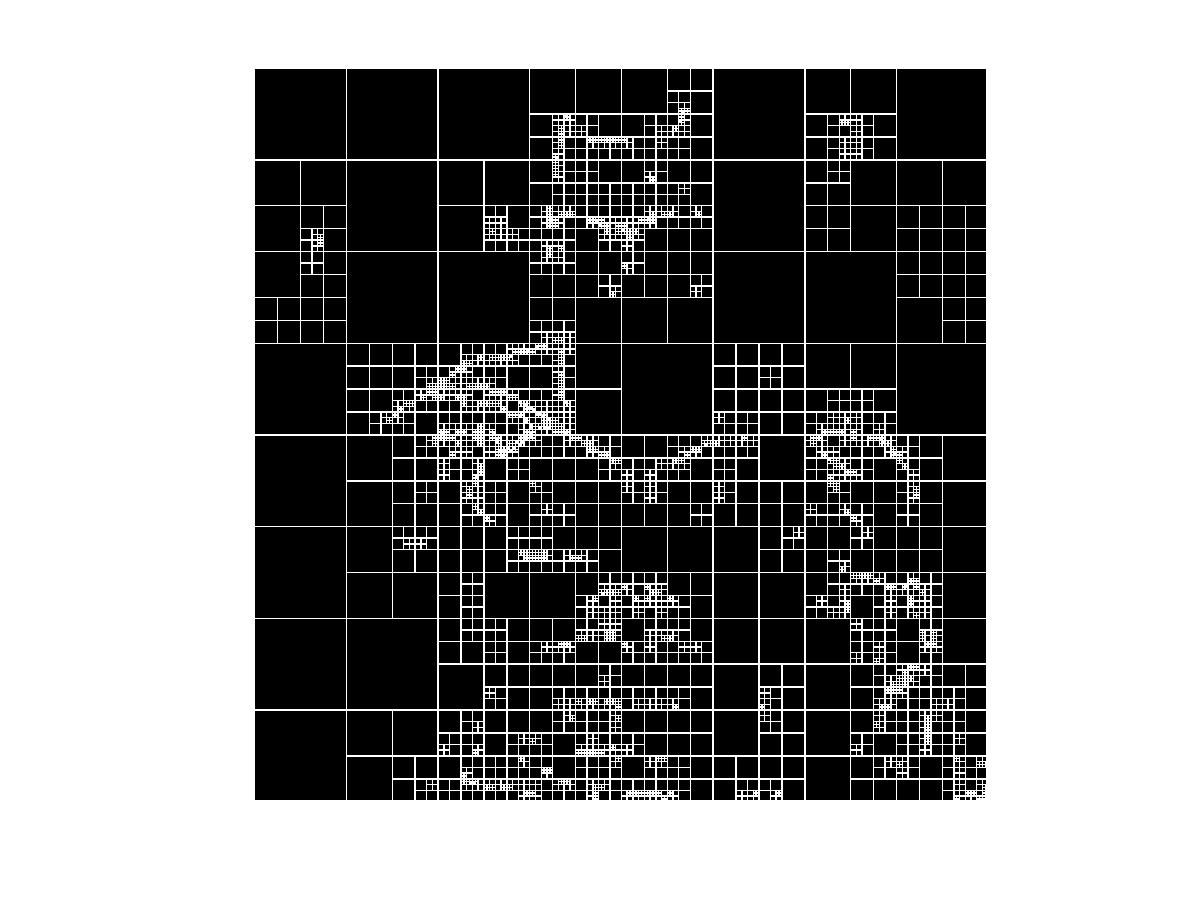
blocks(end,1:end) = 1;

blocks(1:end,end) = 1;

imshow(i), figure, imshow(blocks,[]);



Изображение 5 Исходное изображение



Изображение 6 Разложение по квадродереву

## Задание 3

Splitmerge

function g = splitmerge(f, mindim, mathOj, minMean, maxMean)

fun = @(region) predicate(region, mathOj, minMean, maxMean);

Q = 2^nextpow2(max(size(f)));

[M, N] = size(f);

f = padarray(f, [Q - M, Q - N], 'post');

S = qtdecomp(f, @split\_test, mindim, fun);

Lmax = full(max(S(:)));

g = zeros(size(f));

MARKER = zeros(size(f));

for K = 1:Lmax

[vals, r, c] = qtgetblk(f, S, K);

if ~isempty(vals)

for I = 1:length(r)

xlow = r(I); ylow = c(I);

xhigh = xlow + K - 1; yhigh = ylow + K - 1;

region = f(xlow:xhigh, ylow:yhigh);

flag = feval(fun, region);

if flag

g(xlow:xhigh, ylow:yhigh) = 1;

MARKER(xlow, ylow) = 1;

end

end

end

end

g = bwlabel(imreconstruct(MARKER, g));

g = g(1:M, 1:N);

function v = split\_test(B, mindim, fun)

k = size(B, 3);

v(1:k) = false;

for I = 1:k

quadregion = B(:, :, I);

if size(quadregion, 1) <= mindim

v(I) = false;

continue

end

flag = feval(fun, quadregion);

if flag

v(I) = true;

end

end

function flag = predicate(region, mathOj, minMean, maxMean)

sd = std2(region);

m = mean2(region);

flag = (sd > mathOj) & (m > minMean) & (m < maxMean);

img = imread("batman\_jl.jpg");

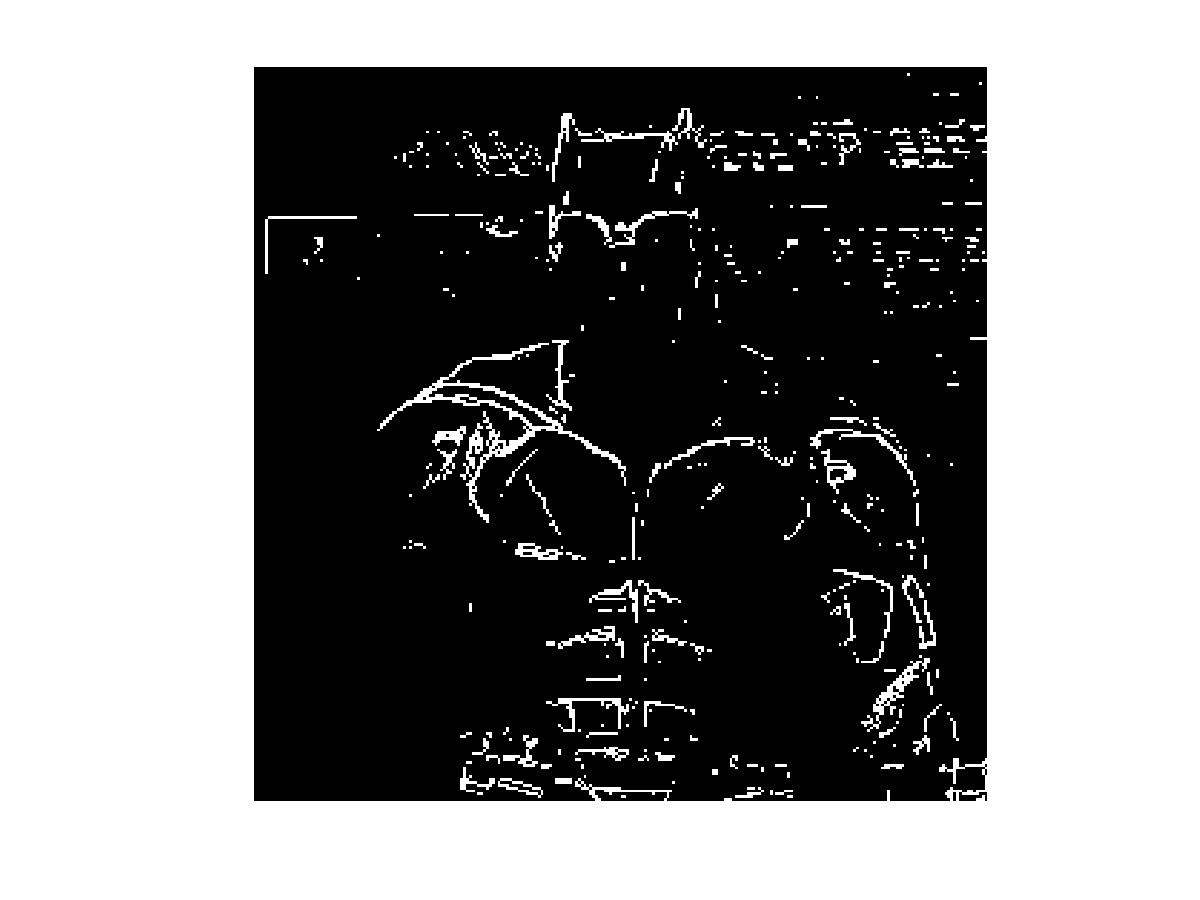
res = splitmerge(img, 4, 10, 0, 500);

figure, imshow(img);

figure, imshow(res);



Изображение 7 Исходное изображение



Изображение 8 splitmerge



Изображение 9 spltmerge

## Задание 4

function g = grad\_edge(f, thresh, x\_mask, y\_mask)

f = im2double(f);

g = zeros(size(f));

for i = 2:1:size(f,1)-1

for j = 2:1:size(f,2)-1

mask = f(i-1:i+1,j-1:j+1);

gx = sum(sum(mask .\* x\_mask)).^2;

gy = sum(sum(mask .\* y\_mask)).^2;

t = sqrt(gx + gy);

if (t > thresh)

g(i,j) = 1;

end

end

end

end

function edge\_r = Roberts(img, thresh)

rob\_x = [0, 0, 0;

0, -1, 0;

0, 0, 1];

rob\_y = [0, 0, 0;

0, 0, -1;

0, 1, 0];

edge\_r = grad\_edge(img, thresh, rob\_x, rob\_y);

endfunction

function edge\_s = Sobel(img, thresh);

mask\_x = [-1, -2, -1;

0, 0, 0;

1, 2, 1];

mask\_y = [-1, 0, 1;

-2, 0, 2;

-1, 0, 1];

edge\_s = grad\_edge(img, thresh, mask\_x, mask\_y);

endfunction

function edge\_p = Previtt(img, thresh);

mask\_x = [-1, -1, -1;

0, 0, 0;

1, 1, 1];

mask\_y = [-1, 0, 1;

-1, 0, 1;

-1, 0, 1];

edge\_p = grad\_edge(img, thresh, mask\_x, mask\_y);

endfunction

img = imread("batman\_jl.jpg");

edge\_r = Roberts(img, 0.08);

figure, imshow(img);

figure, imshow(edge\_r);

edge\_s = Sobel(img, 0.48);

figure, imshow(edge\_s);

edge\_p = Previtt(img, 0.3);

figure, imshow(edge\_p);



Изображение 10 Исходное изображение



Изображение 11 Границы методом Робертса



Изображение 12 Границы методом Собеля



Изображение 13 Границы методом Превитта

Все использованные методы могут выдать схожий результат, однако для метода Собеля необходимо устанавливать довольно высокое пороговое значение.

# Выводы

В ходе выполнения данной лабораторной работы были изучены следующие методы сегментирования изображений: центроидного связывания, квадратичного разложения, разделения слияния. Также были использованы градиентные фильтры для выделения границ объектов. Сравнение данных фильтров показало, что все они могут выдать схожий результат, однако предполагают различные пороги бинаризации исходного изображения.