МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ

(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)

КАФЕДРА 305

«ПИЛОТАЖНО-НАВИГАЦИОННЫЕ И ИНФОРМАЦИОННО-ИЗМЕРИТЕЛЬНЫЕ КОМПЛЕКСЫ»



Дисциплина «Специальные технологии 1 (обработка изображений)»

Отчет по лабораторной работе № 1

Вариант №3

Выполнил: студент группы М3О-506С-20

Орлов П.А.

Принял: доцент кафедры 305

к.т.н., Нгуен Н.М.

Цель работы:

Разработка программы для реализации поиска прямых на изображении с помощью преобразования Хафа.

# 1. Теоретическая часть

# 1.1 Основы преобразования Хафа

Пусть задано точек, являющихся значимыми пикселями изображения. Требуется найти среди них подмножество точек, лежащих на некоторых прямых.

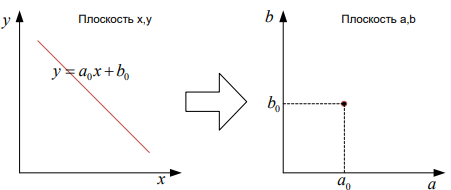
Каждая прямая на плоскости описывается уравнением вида:

(1.1)

Рассмотрим конкретную прямую, например:

(1.2)

и плоскость , в которой переменными (координатами) являются параметры уравнения (1.1), т.е. и (рисунок 1.1). Плоскость ab называют плоскостью Хафа, или плоскостью параметров.

  
Рисунок 1.1 – Образ прямой на плоскости параметров

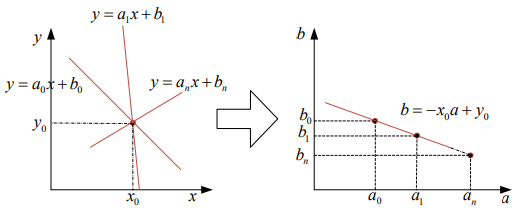
Прямая (1.2) характеризуется на плоскости изображения двумя параметрами: и (т.е. пара , однозначно определяет прямую на плоскости ). Тогда можно сказать, что она характеризуется точкой с координатами (, ) на плоскости .

Теперь берем конкретную точку (,) на плоскости . Через неё проходят бесконечно много прямых, характеризуемых различными значениями коэффициентов и . В этом случае и являются константами (параметрами уравнения), а и - переменными. Преобразуем уравнение (1.1) в следующий вид:

(1.3)

На плоскости уравнение (11.1) соответствует прямой с параметрами и .

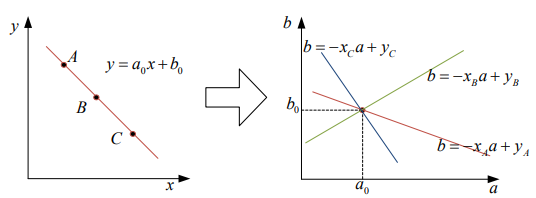
Следовательно, точка (,), или точнее, семейство прямых, проходящих через точку (,) на плоскости , представляет собой прямые на плоскости . Это иллюстрируется на рисунке 1.2.

  
Рисунок 1.2 – Образ точки на плоскости параметров

Действительно, каждая прямая на плоскости изображения (т.е. плоскости ) соответствует одной и только одной точке на плоскости Хафа, поэтому множество прямых, проходящих через точку (,), соответствует множеству точек на плоскости , координаты которых удовлетворяют уравнению прямой с параметрами и , т.е. уравнению (1.2). Известно, что такие точки должны находиться на одной и только одной прямой, заданной уравнением (1.2).

На рисунке 1.3 слева представлены три точки , , , находящиеся на одной прямой, описывающейся уравнением (1.2). Образами этих трех точек на плоскости Хафа являются три прямые:

(1.4)

  
Рисунок 1.3 – Образ точек, лежащих на одной прямой

Поскольку три точки , , находятся на прямой (1.2), их координаты удовлетворяют уравнению (1.2), т.е.:

(1.5)

Равенства (1.5) означают, что три прямые (1.4) пересекаются в точке (,). Это значит, что образы точек, находящихся на одной прямой, на плоскости Хафа пересекаются в одной точке. Таким образом, задача поиска точек прямой на изображении сводится к задаче поиска точки пересечения на плоскости Хафа, что и является более простой.

Преобразование Хафа позволяет найти все прямые, проходящие через некоторые множества пикселей изображения, путем определения координат точек пересечения образов этих пикселей на плоскости Хафа.

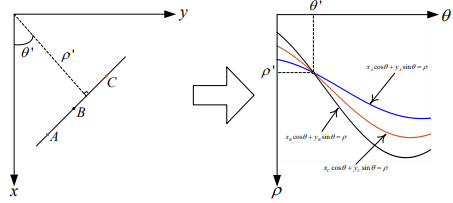
Кроме прямых, преобразование Хафа позволяет найти также любые фигуры определенной формы на изображении

## 1.2 Модификация преобразования Хафа

Уравнение прямой вида (1.1) имеет недостаток, связанный с бесконечным значением параметра a при стремлении прямой к вертикали. Для устранения данного недостатка используем другую формулу описания прямой, а именно:

(1.6)

где - угол наклона прямой, - расстояние от начала системы координат до прямой (рисунок 1.4). В такой форме представления можем описывать любые прямые на плоскости

  
Рисунок 1.4 – Модификация преобразования Хафа

Введем плоскость параметров и построим на ней образы точек изображения, находящихся на одной прямой. В результате получается такой же вывод, за исключением того, что образ точки уже не является прямой, а синусоидой.

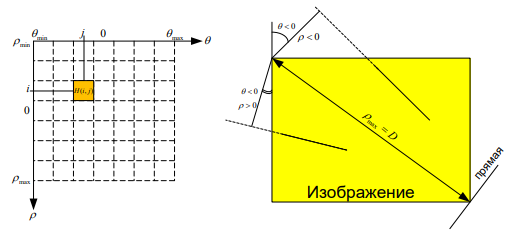
## 1.3 Алгоритмы реализации преобразования Хафа на компьютере

Привлекательность преобразования Хафа с точки зрения вычислений определяется возможностью разбиения пространства параметров (в данном случае плоскости Хафа) на так называемые ячейки накопления, как показано на рисунке 1.5, где [, ] и [,] - предлагаемые диапазоны значений параметров.

Обычно принимают:

(1.7)

где - расстояние между угловыми точками изображения (максимальное расстояние между началом системы координат и прямыми на изображении).

  
Рисунок 1.5 – Дискретизация плоскости Хафа

В ячейке с индексами () накапливается значение для квадрата в пространстве параметров, соответствующего точке (при этом , ). Сначала значения всех ячеек накопления равны , затем для каждой точки выбранного множества значимых пикселей на изображении полагаем параметр равным каждому разрешенному дискретному значению и находим соответствующее значение ρ путем решения уравнения . Затем найденные величины округляются до ближайшего разрешенного дискретного значения . После этого значение соответствующей ячейки накопления увеличивается на единицу: . В конце процедуры значение ячейки равно , которое означает, что точек плоскости изображения лежат на прямой.

## 2. Практическая часть

Листинг программы в MATLAB:

function init\_var\_3()

%функция инициализации начальных параметров согласно варианту №3

global M N x\_1 y\_1 x\_2 y\_2 x\_min x\_max;

M = 101;

N = 101;

x\_1 = 100;

y\_1 = 0;

x\_2 = 50;

y\_2 = 100;

x\_min = 50;

x\_max = 100;

end

function image\_with\_line = image\_with\_line\_based\_on\_two\_points\_create(M, N, x\_1, x\_2, y\_1, y\_2, x\_min, x\_max)

%функция создания изображения с прямой линией, построенной по 2 точкам

a = (y\_2 - y\_1)/(x\_2 - x\_1);

b = y\_1 - a\*x\_1;

image\_with\_line = image\_with\_line\_based\_on\_two\_koef\_create(M, N, a, b, x\_min, x\_max);

end

function image\_with\_line = image\_with\_line\_based\_on\_two\_koef\_create(M, N, a, b, x\_min, x\_max)

%функция создания изображения с прямой линией, построенной по 2

%коэффициентам

image\_with\_line = ones(M, N);

for x = x\_min:1:x\_max

y = a\*x+b;

image\_with\_line(x +1, floor(y+1)) = 0;

end

end

function n = real\_n\_count(x\_min, x\_max, dx)

%функция расчета реального числа разбиений отрезка от x\_min до x\_max с

%шагом dx

n = ceil(x\_max-x\_min/dx);

end

function [x,y,q] = null\_pixel\_count(image)

%функция подсчёта и определения координат нулевых пикселей на изображении

[M, N] = size(image);

x = [];

y = [];

q = 0;

for i = 1:1:M

for j=1:1:N

if image(i,j) == 0

x(end + 1) = i - 1;

y(end + 1) = j - 1;

q = q + 1;

end

end

end

end

function [H, theta, rho] = PRHAF(I, dtheta, drho)

%PRHAF функция преобразования Хафа

%I,H - входное и выходное изображения

%dtheta, drho - шаги дискретизации плоскости Хафа

%theta, rho - вектор значений параметров плоскости Хафа

[M, N] = size(I);

D = sqrt((M-1).^2 + (N-1).^2);

n\_theta = real\_n\_count(-90, 90, dtheta);

n\_rho = real\_n\_count(-D, D, drho);

theta = linspace(-90, 90, n\_theta);

rho = linspace(-D, D, n\_rho);

H = zeros(n\_rho, n\_theta);

[x,y,q] = null\_pixel\_count(I);

for k=1:q

for j=1:n\_theta

rho\_count = x(k)\*cos(theta(j)\*pi/180) + y(k)\*sin(theta(j)\*pi/180);

i = floor(((rho\_count-rho(1))/drho)+1);

H(i, j) = H(i,j) + 1;

end

end

end

clear all;

close all;

%% Инициализация начальных переменных

global M N x\_1 y\_1 x\_2 y\_2 x\_min x\_max;

init\_var\_3();

dtheta = 0.5;

drho = 1;

%% Построение исходного изображения

image = image\_with\_line\_based\_on\_two\_points\_create(M, N, x\_1, x\_2, y\_1, y\_2, x\_min, x\_max);

figure();

imshow(image);

xlabel('y'), ylabel('x');

axis on;

title('Исходное изображение');

%% Построение проекции исходного изображения на плоскость Хафа

[image\_haf, theta, rho] = PRHAF(image, dtheta, drho);

figure();

imshow(image\_haf,[],'XData',theta,'YData',rho,'InitialMagnification','fit');

xlabel('\theta'), ylabel('\rho');

axis on, axis normal;

title('Проекция исходного изображения на плоскость Хафа')

%% Определение коэффициентов уравнения прямой

[rho\_max\_index, theta\_max\_index] = find(image\_haf == max(image\_haf,[],"all"));

rho\_max = rho(rho\_max\_index(1));

theta\_max = theta(theta\_max\_index(1));

a\_max = -cos(theta\_max\*pi/180)/sin(theta\_max\*pi/180);

b\_max = rho\_max/sin(theta\_max\*pi/180);

%% Построение прямой на исходном изображении

for x = x\_min:1:x\_max

y = a\_max\*x+b\_max;

image(x +1, floor(y+1)) = 0;

end

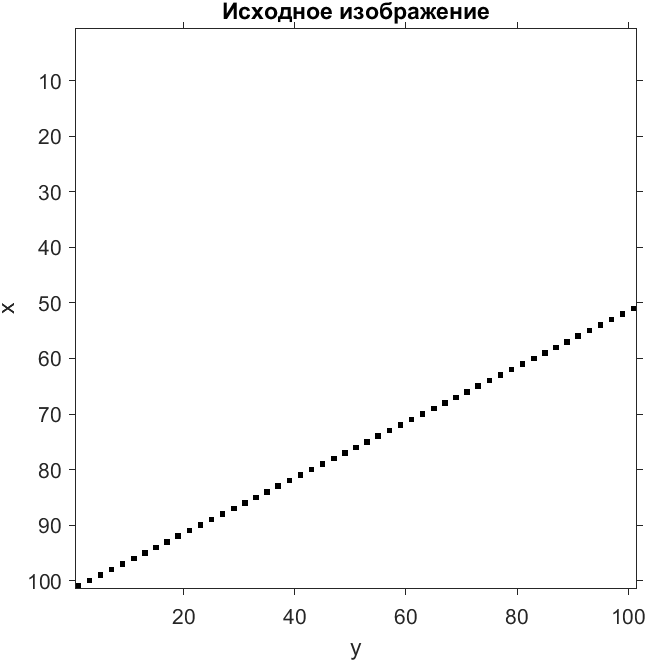
figure();

imshow(image);

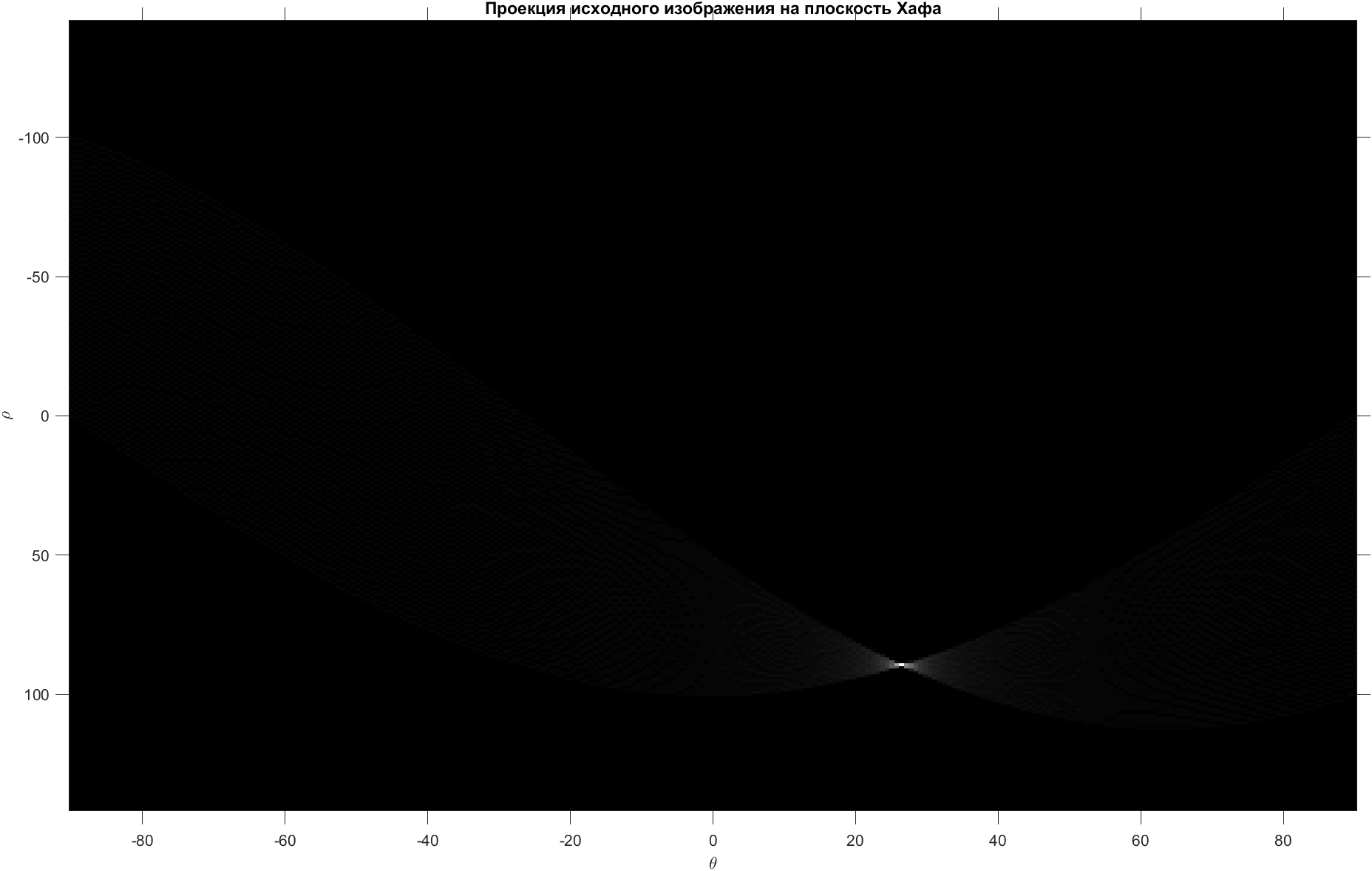
xlabel('y'), ylabel('x');

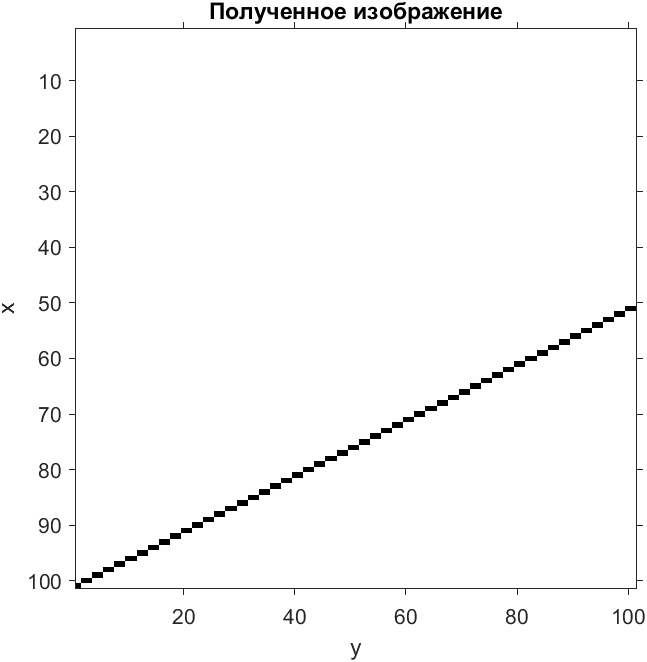
axis on;

title('Полученное изображение');

  
Рисунок 2.1 – Исходное изображение

Значения коэффициентов прямой на исходном изображении: .

  
Рисунок 2.2 – Проекция исходного изображения на плоскость Хафа

  
Рисунок 2.3 – Полученное изображение

Значения коэффициентов прямой, определённые с помощью преобразования Хафа: .

На рисунке 2.3 представлено наложение прямой, найденной с помощью преобразования Хафа, на исходное изображение. Ошибка определения коэффициентов уравнения прямой: .

Вывод:

В ходе выполнения лабораторной работы была разработана программа для реализации поиска прямых на изображении с помощью преобразования Хафа