Содержание

[Задание 2](#_Toc150895265)

[Результаты работы 4](#_Toc150895266)

[Код программы 18](#_Toc150895267)

# Задание

1. Выбрать входную функцию и число *m*, исходя из варианта. Построить график 𝑓(𝑟). Здесь и далее для каждого графика следует строить отдельно графики/изображения амплитуды и фазы. Входную область ограничить радиусом 𝑅 = 5.
2. Восстановить изображение в двумерный массив и построить это изображение.
3. Реализовать преобразование Ханкеля методом численного интегрирования (например, методом левых прямоугольников). Размеры входной и выходной областей должны совпадать. Применить преобразование ко входной функции и получить выходную 𝐹(𝜌). Построить её график, а также восстановить двумерную функцию 𝐹(𝜌)exp(𝑖𝑚𝜃) и построить её изображение.
4. Реализовать двумерное преобразование Фурье через БПФ. Применить его ко входной двумерной функции . Построить изображение выходной функции, сравнить его с результатом, полученным для преобразования Ханкеля. Если изображения амплитуд сильно отличаются, попытаться увеличить число точек дискретизации.
5. Исследовать скорость выполнения двумерного БПФ и преобразования Ханкеля, варьируя число точек дискретизации. Сделать выводы.

Вариант 14:



**Результаты работы**

**Задание 1:**

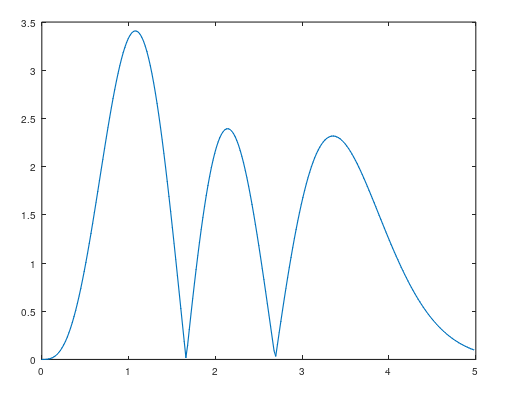


Рисунок 1 – График амплитуды входного радиально-вихревого оптического пучка

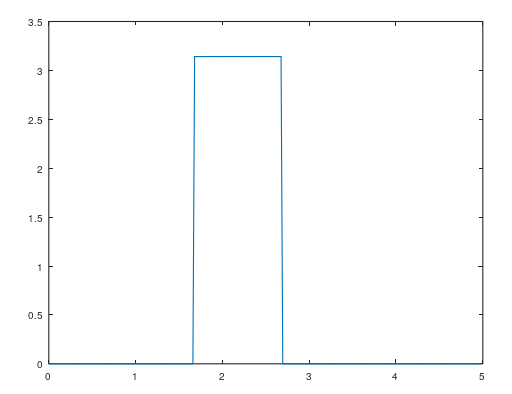


Рисунок 2 – График фазы входного радиально-вихревого оптического пучка

**Задание 2:**

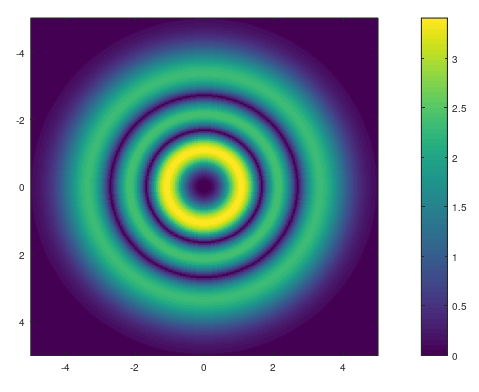


Рисунок 3 – Изображение амплитуды входного радиально-вихревого оптического пучка

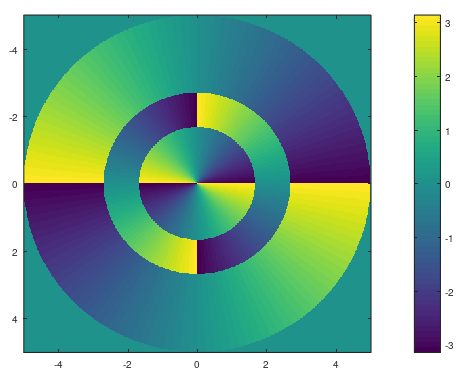


Рисунок 4 – Изображение фазы входного радиально-вихревого оптического пучка

**Задание 3:**

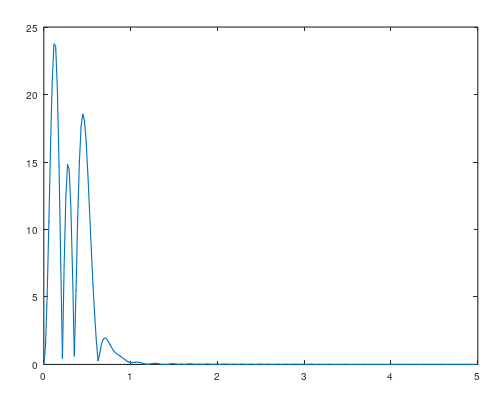


Рисунок 5 – График амплитуды результата преобразования Ханкеля

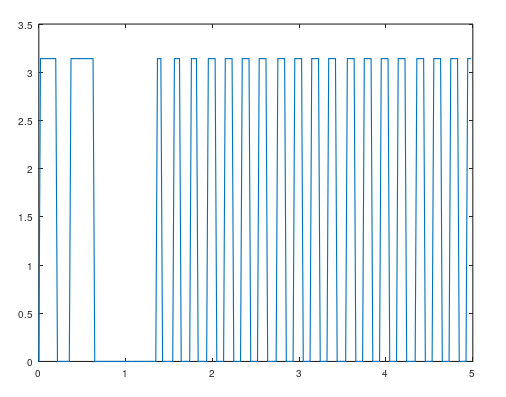


Рисунок 6 – График фазы результата преобразования Ханкеля

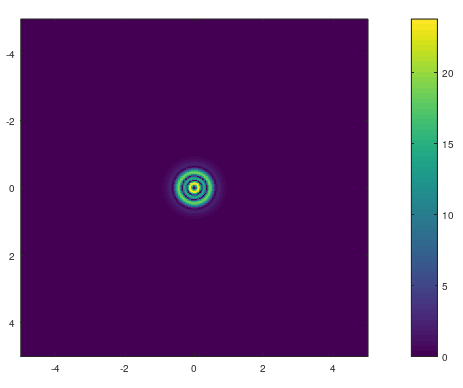


Рисунок 7 – Изображение амплитуды результата преобразования Ханкеля

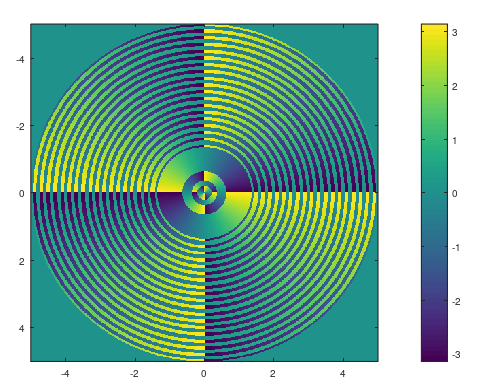


Рисунок 8 – Изображение амплитуды результата преобразования Ханкеля

**Задание 4:**

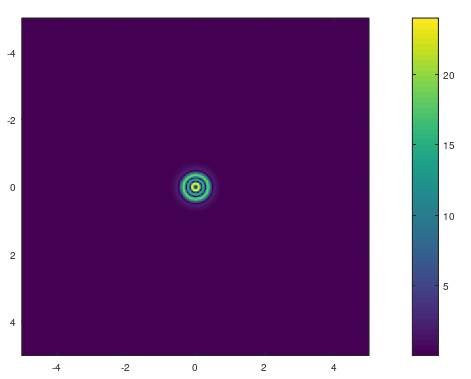


Рисунок 9 – Изображение амплитуды результата преобразования Фурье

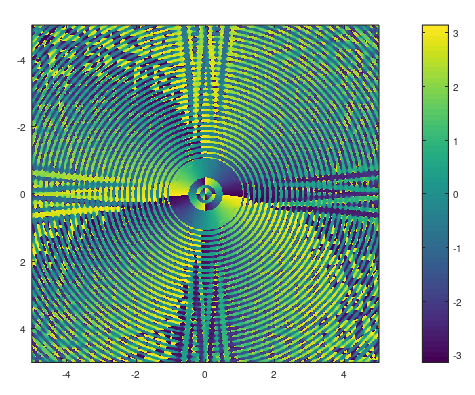


Рисунок 10 – Изображение фазы результата преобразования Фурье

**Задание 5:**

Таблица 1. Скорость выполнения двумерного БПФ и преобразования Ханкеля при M = 2048.

|  |  |  |
| --- | --- | --- |
| N | Преобразование Фурье c. | Преобразование Ханкеля c. |
| 32 | 0,024905 | 0,001860 |
| 64 | 0,048960 | 0, 003245 |
| 128 | 0,084164 | 0,013652 |
| 256 | 0,191259 | 0,047086 |
| 512 | 0,378421 | 0,224038 |

По таблице можем сделать вывод, что преобразование Ханкеля на порядок быстрее, чем БПФ. Поскольку преобразование Ханкеля является частным случаем Фурье-преобразования, получается, что требуется лишь вычислить интеграл на отрезке, а Фурье обрабатывает двумерный массив. Однако при дальнейшем уменьшении шагов дискретизации, разница времени выполнения преобразований уменьшается. Скорость выполнения зависит только от параметра N.

# Код программы

clear;

# БПФ

function F = DFT(func, M, hx)

s = (M - length(func)) / 2;

func = [zeros(s, 1).', func, zeros(s, 1).'];

func = [func(1 + M/2:end), func(1:M / 2)];

F = fft(func) \* hx;

F = [F(1 + M/2:end), F(1:M / 2)];

F = F(s + 1:M - s);

end

# Обобщенный полином Лагерра

function poly = L(n, p, r)

poly = zeros(size(r));

for j = 0:n

poly += power(-1, j) \* nchoosek(n + p, n - j) \* r.^ j / factorial(j);

end

end

# Мода Гаусса-Лагерра

function moda = GL(n, p, r)

p = abs(p);

moda = exp(r.^ 2 / -2) .\* r.^ p .\* L(n, p, r.^ 2);

end

m = 2;

n = 2;

p = -3;

N = 256;

R = 5;

interval = [-R, R];

hx = R / N;

x = 0:hx:(R - hx / 2);

F = GL(n, p, x);

# Графики входной функции

figure(1);

plot(x, abs(F)); # Амплитуда

figure(2);

plot(x, arg(F)); # Фаза

# Восстанавление изображения в двумерный массив

rec\_img = zeros(2 \* N, 2 \* N);

for i = 1:rows(rec\_img)

for j = 1:columns(rec\_img)

alpha = round(sqrt((i - N)^2 + (j - N)^2)) + 1;

if (alpha <= N)

rec\_img(i, j) = F(alpha).\* exp(1i \* m \* atan2(j - N, i - N));

endif

endfor

endfor

figure(3);

imagesc(interval, interval, abs(rec\_img)); # Амплитуда

colorbar;

figure(4);

imagesc(interval, interval, arg(rec\_img)); # Фаза

colorbar;

# Преобразования Ханкеля

tic();

[X, XI] = meshgrid(x, x);

A = (2 \* pi / 1i^ m) \* besselj(m, 2 \* pi \* X.\* XI) .\* X;

H = A \* F.' \* hx;

printf('Time H: %f sec\n', toc());

# График преобразования Ханкеля

figure(5);

plot(x, abs(H)); # Амплитуда

figure(6);

plot(x, arg(H)); # Фаза

# Восстанавливаем изображение для Ханкеля

rec\_img\_H = zeros(2 \* N - 1, 2 \* N - 1);

for i = 1:rows(rec\_img\_H)

for j = 1:columns(rec\_img\_H)

alpha = round(sqrt((i - N)^2 + (j - N)^2)) + 1;

if (alpha <= N)

rec\_img\_H(i, j) = H(alpha).\* exp(1i \* m \* atan2(j - N, i - N));

endif

endfor

endfor

figure(7);

imagesc(interval, interval, abs(rec\_img\_H)); # Амплитуда

colorbar;

figure(8);

imagesc(interval, interval, arg(rec\_img\_H)); # Фаза

colorbar;

# Двумерное преобразованье Фурье через БПФ

M = 2048;

b = N ^ 2 / (4 \* R \* M);

tic();

for i = 1:rows(rec\_img)

rec\_img(i, :) = DFT(rec\_img(i, :), M, hx);

endfor

for j = 1:columns(rec\_img)

rec\_img(:, j) = DFT(rec\_img(:, j).', M, hx).';

endfor

printf('Time F: %f sec\n', toc());

figure(9);

imagesc(interval, interval, abs(rec\_img)); # Амплитуда

colorbar;

figure(10);

imagesc(interval, interval, arg(rec\_img)); # Фаза

colorbar;