

# Отчет по задаче "Вычислительная ниткография"

Склярова Ванесса

June 20, 2021

## Постановка задачи

Цель: Реализовать алгоритм построения схемы расположения нитей на основе метода вычислительной рентгеновской томографии (BPT, computed tomography). В качестве тестового изображения взять свою фотографию.

## Метод

Алгоритм расчета нитей основан на популярном подходе в рентгеновской томографии filtered backprojection. Проецирование исходного изображения на детектор под заданными углами называется преобразованием Радона. Преобразование Радона это интегральное преобразование функции. Оно имеет простой геометрический смысл — это интеграл от функции вдоль заданной прямой. Таким образом, сканирование объекта с разных сторон это получение значения интеграла функции, которое задаёт распределение коэффициента поглощения вдоль заданной линии (рентгеновского луча). Совокупность этих проекций  $p_\theta(t)$  представляет из себя так называемые синограммы, поскольку образом каждой точки является синусоида. Получение объекта из синограммы называется операция backprojection. Цель алгоритма FBP заключается в восстановлении функции объекта  $f(x, y)$  по дискретной выборке из синограммы. Backprojection вычисляется по формуле:

$$f_{bp} = \int p_\theta(x \cos \theta + y \sin \theta) d\theta$$

При применении его без фильтрации получается размытое изображение. Если мы сделаем fft преобразование, то заметим отсутствие high frequency компонент. Поэтому мы применяем светку с high pass Ramp фильтром, чтобы подавить low frequency компоненты и усилить high frequency, поскольку они отвечают за четкость изображения. По сути он компенсирует спарсность высокочастотных значений в Фурье плоскости (график 1). После этого, если мы сделаем ifft, то получим функцию  $q_\theta(t)$  и наш filtered backprojection алгоритм запишется как:

$$f_{bp} = \int q_\theta(x \cos \theta + y \sin \theta) d\theta, \quad q_\theta(t) = \int P_\theta(\omega) |\omega| e^{i2\pi\omega t} d\omega$$

Схему алгоритма фильтрации можно увидеть на графике 2.

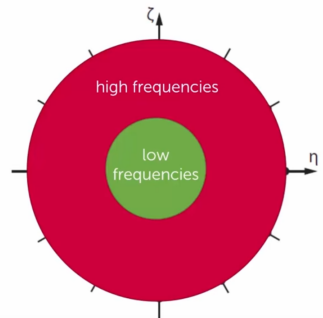


Figure 1: Распределение низких и высоких частот в Фурье плоскости

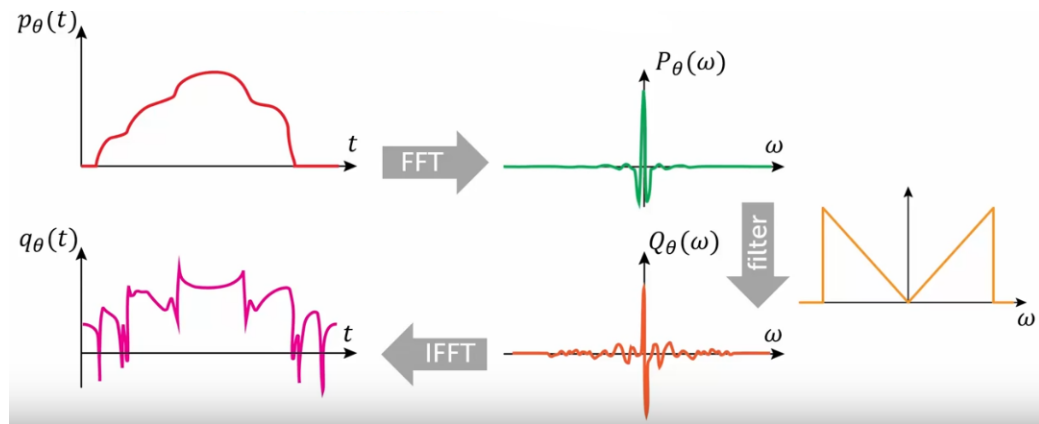


Figure 2: Схема фильтрации

Чтобы получить "эффект нитей" необходимо полученную после этапа фильтрации синогрammu  $q_\theta(t)$  предварительно проредить. Таким образом мы по сути исключаем некоторые пиксели из рассмотрения. После прореживания синограммы мы делаем нормировку с занулением точек синограммы, значения которых меньше 0 и нормируем оставшийся диапазон к отрезку  $[0, 1]$ . Также используем бинарное распыление синограммы, в котором для каждого пикселя выходной синограммы поставим 1 с вероятностью, указанной в этом пикселе, иначе поставим 0. После всех преобразований над синограммой мы делаем back-projection по указанной выше формуле. Также рассматривается возможность контрастирования полученного изображения для усиления эффекта.

## Реализация

Код алгоритма написан на python.

На вход программе подаются параметры строки такие как:

image\_path – путь до картинки,

rate – доля нитей прореживания (в отрезке  $[0, 1]$ ),

num\_angles – число используемых углов,

is\_contrast – использовать контрастирование или нет,

visualize – визуализировать, полученные на каждом этапе картинки или нет.

Пример запуска: `python threads_sklyarova.py -- image_path 'photo.jpg' -- number_angles 60 -- rate 0.3`.

Указанные параметры являются default.

Результат сохраняется в текущую директорию result.png и если `is_contrast == True`, то также сохраняется контрастированное изображение contrasted\_result.png

Основные функции:

**preprocess\_image** – функция для препроцессинга картинки. Делает crop центральной части, resize, convert to grayscale и вписывает изображение в окружность.

**projection** – делает преобразование Радона для исходной картинки.

**get\_ramp\_fourier\_filter** – возвращает рампы фильтр в фурье области.

**filtered\_projection** – возвращает синогрammu после свертки с рампы фильтром.

**cuttering** – прореживает синогрammu в rate раз.

**normalize** – нормализация синограммы.

**binarize** – бинаризация синограммы.

**back\_projection** – функция для получения объекта из его проекций.

**contrast** – зашкаливает пиксели высокой яркости.

## Результаты

Примеры работы алгоритма можно видеть на графике 3. Пунктом а обозначено исходное изображение. Пункт б соответствует результату backprojection алгоритма для заданного числа углов без применения фильтрации. Можно заметить, что high frequency компоненты отсутствуют и изображение получается сильно размытым, однако эта проблема решается после применения фильтрации (пункт в). Пункт г отвечает за результат применения backprojection алгоритма после прореживания синограммы. Д и е получаются после нормализации и бинарного распыления синограммы без и с контрастированием после backprojection.

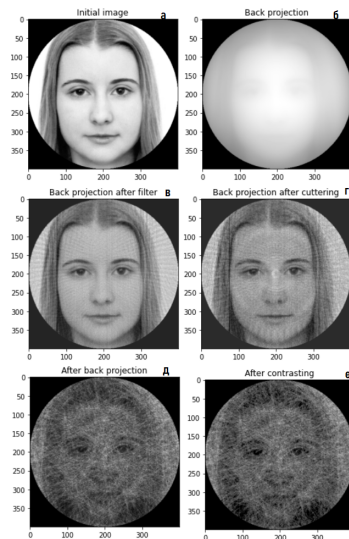


Figure 3: Пример построения алгоритмом схемы расположения нитей

Синограммы представлены на графике 4. Преобразование Радона исходной картинке дает пункт а. Пример фильтрации синограммы и прореживания представлен в пунктах б и в соответственно. Г и д соответствуют синограммам, полученным после нормализации и бинаризации.

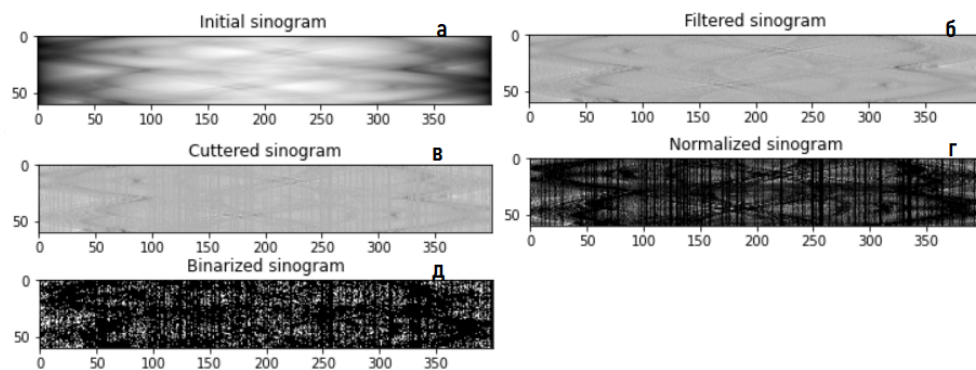


Figure 4: Синограммы, полученные в ходе работы алгоритма.