**Лабораторный журнал проекта «Луч Ч»**

Что было до этого:

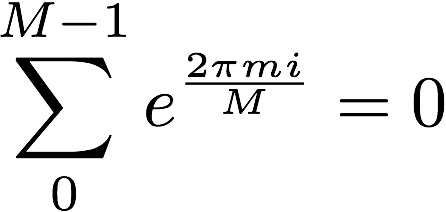
Основная мысль состояла в том, что изображение мы получаем, нас не устраивает только разрешение. Увеличить разрешение можно с помощью механических движений высокой точности. Смысл такой: под нашей матрицей расчерчиваем матрицу с более мелкой клеткой и считаем, что значение пикселя в нашей матрице — это среднее значение мелких клеток, которые лежат внутри нашей. Получается, будто итоговое изображение имеет некоторые элементы оригинального изображения свернутого с PSF(point-spread function). Механическими сдвигами мы находим все элементы свернутого изображения, а потом разворачиваем. Так как от шума никуда не деться, то обратная свертка не обычная, а винеровская.

Написана прошивка для Ардуино, которая управляем шаговыми двигателями и получает команды по ком-порту.

Написана программа для ПК, которая отправляет команды на движение и собирает изображения. С реализацией и тестированием непосредственно алгоритма обработки возникли затруднения.

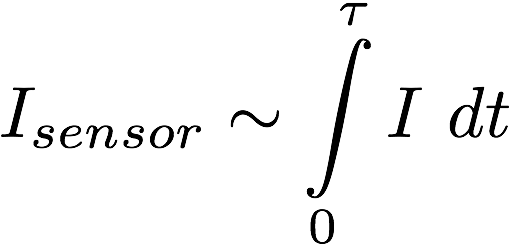
1) Собирая свернутое изображение из сэмплов (полученное изображение при некотором значении механических сдвигов), мы считаем что размеры свернутого изображения = размеры сэмпла \* размеры psf. В случае, когда мы проделываем эту процедуру искусственно, равенство выше может и не выполняться. Поэтому для тестирования надо подбирать специальные размеры изображений. (h % k == 1)

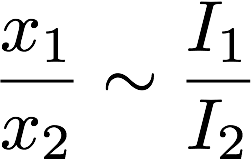
2) Не всегда можно выполнить обратную свертку. Там может быть несколько причин, пока удалось вычленить одну, связанную с формулой DFT



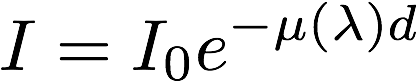
21.10.2016

1. Получил модель сенсора в матрице.



Из модели следует, что числа в матрице пропорциональны интенсивностям.

2. Прикинул уменьшение интенсивности для рентгеновских лучей энергией 60 кэВ при прохождении 0.6 мм меди и 1 мкм алюминия. Алюминий уменьшает интенсивность на 0.01%. Пользовался законом:



Дальнейшие планы:

• Найти распределение рентгеновских лучей по энергиям

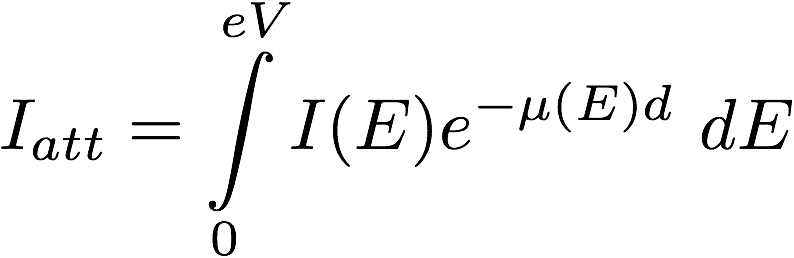
• Поискать материал, пропускающий мягкие лучи лучше, чем жесткие

• Провести еще эксперимент проверяющие формулы выше

24.10.2016

Нашел программу SpekCalc для расчета спектра излучение вольфрамового анода при различных разгоняющих напряжениях и фильтрах. Она выдает результаты в виде зависимости относительных интенсивностей от энергии фотонов.

Благодаря этой программе обобщил подсчет уменьшение интенсивности. Если I(E) — распределение интенсивности (мощности) по энергиям на входе, то на выходе получаем суммарную интенсивность:



Например, для алюминиевой фольги толщиной 18 мкм при прохождении рентгеновских лучей с максимальной энергией 80 кэВ теряется 5% мощности.

Дальнейшие планы:

• Конечно же, экспериментальная проверка этих выкладок.

• Уменьшить разгоняющее напряжение трубки

25.10.2016

Почему-то сегодня работал совсем не по плану, а по наитию. Пока, на самом деле, я не очень хорошо себе представляю как матрица превращает интенсивности в числа.

Собрал примерную версию установки, в которой все не рушится при съемке. Соответственно, подсчитал чувствительности пикселей и шум. Получил какие-то кривые результаты с очень маленькими чувствительностями для некоторых пикселей.

Дальнейшие планы:

• Интерпретация чувствительности разных пикселей

• Опыты с алюминиевой фольгой

26.10.2016

Поставил цель в работе на ближайшие несколько дней: различить на картинке 18-микронный слой фольги. Мешает чувствительность пикселей. Потому что рябь то там, то тут появляется. На фоне этой ряби теряется вся информация.

27.10.2016

Дополнил установку неподвижной подставкой для матрицы. Настроил с Антоном датчик таким образом, что он выдает более-менее одинаковые картинки. Появилась повторяемость. На волне измерил чувствительность разных пикселей для этой конфигурации и использовал ее для просмотра. В некоторых случаях, таких как тонкая фольга, чувствительность здорово помогает и фольгу становится видно лучше. В случае большого количества деталей или большого количества поглощенной энергии, чувствительность только мешает и размазывает картинку.

Получил первую часть условия успешной обратной свертки на размеры свернутого сигнала и размеры шаблона. Если обозначить за N размер свернутого сигнала, а за K — размер шаблона, то при K \* НОД(N,n) = N, n-й элемент DFT шаблона будет равен нулю. Чтобы защититься от этого, N не должно делится на K.

Дальнейшие планы:

• Учесть возможную цикличность точек на комплексной окружности

• Учесть возможное влияние двумерности сигнала

• Получить численное значение различимости 18-микронной фольги

28.10.2016

Обсудил с Андреем Владимировичем возможность уменьшения разгоняющего напряжения в рентгеновской трубке. Нужно будет обратиться к Юрию Ивановичу, чтобы он нарисовал схему работы трубки и указал на те вещи, которые можно поменять извне.

Получил первую часть условия успешной обратной свертки на размеры свернутого сигнала и размеры шаблона. Она появляется из-за возможной цикличности точек на комплексной плоскости. Во вчерашних обозначениях, при НОД(N,K) \* НОД(N,n) = N, n-й элемент DFT шаблона будет равен нулю. Чтобы защититься от этого, N не должно иметь общих делителей с K.

Эта формула интуитивная, слегка проверенная на простых наборах. Было бы здорово ее доказать.

Дальнейшие планы:

• Написать функцию обратной свертки, учитывающую мою находку. Применить эту функцию в работе.

• Обратиться к Юрию Ивановичу за помощью по схеме.

• Проверить функцию обратной свертки на картинке с енотом.

31.10.2016

Использовал выведенное условие для улучшения функции обратной свертки. Чудесным образом она теперь работает для любых размеров шаблона.

Подытожил все свои алгоритмические изыскания пятью экспериментами в питоновском блокноте. Удается разбить изображение на сэмплы, сохранить их на диск, прочитать и восстановить изображение даже при наличии шума.

Написал функции чтения сэмплов с диска и свертки из сэмплов на С++. Это первая часть работы по тестированию.

Попросил Юрия Ивановича нарисовать схему электрического управления рентгеновской установкой.

Дальнейшие планы:

* Реализовать обратную свертку на С++
* Протестировать алгоритм вручную запихивая картинки в папку

01.11.2016

Целый день потратил на функцию обратной свертки на С++, но все-таки написал. А все из-за OpenCV. До чего уж неочевидная библиотека.

Пришлось перейти из QtCreator’a в Visual Studio, потому что в первом отладчик работал катастрофически медленно. К счастью переход произошел быстро.

Дальнейшие планы:

* Протестировать алгоритм вручную запихивая картинки в папку