**Инструкция для упражнений по обработке изображений и данных на языке программирования python**

**I) Обработке изображений (для решения задачи исследования зависимости интенсивности оптического излучения плазмы от расстояния z)**

**1. Подготовка перед обработкой изображения**

1.1) Отрегулируйте (поверните) изображения так, чтобы источник света располагался слева направо (см. рис.1).



Рис.1 Изображение эксперимента

1.2) Исходные изображения имеют разные размеры, поэтому сначала нам нужно привести их к одному размеру (например, 300x200). Мы сделаем это с помощью библиотеки Pillow на языке программирования python (с python обработка изображений довольно проста).

Ознакомиться с работой с этой библиотекой можно по ссылке:

<https://habr.com/ru/post/681248/>

Так же, как вы можете найти другие библиотеки в python, которые поддерживают обработку изображений, чтобы работать максимально эффективно.

1.3) Некоторые важные понятия, которые нужно знать об обработке изображений

* Изображение это прямоугольная таблица имеющие некоторое количество строк и некоторое количество столцов и в каждой ячейке такой прямоугольной таблицы записано число которое коридует цвет в градиациях. Изображение распадается на квадратные области однородного цвета называемые пикселями (пересечения строк и столцов). Для цветных изображений пиксель — это кортеж (R-Red, G-Green, B-Blue). Любой пиксель изображения может иметь значение от 0 до 255, и это будет его интенсивность.

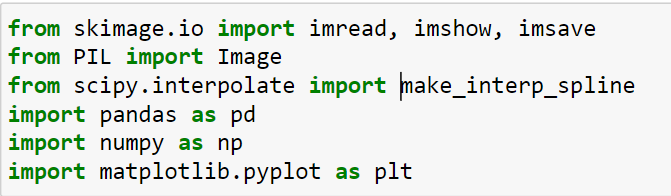


* Яркость пикселя просто: I = (R + G + B) / 3.

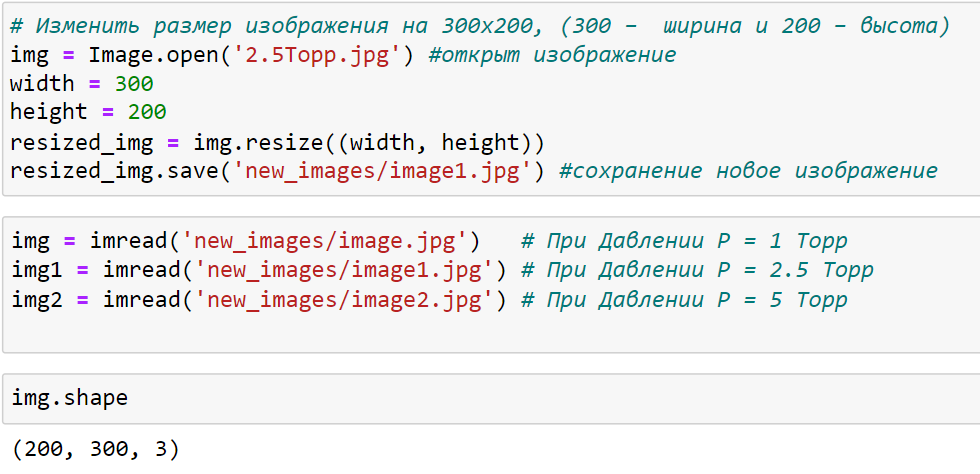
**2. Обработка изображений с помощью языка программирования python**

Во-первых, на вашем компьютере вам нужно приложение, которое может запускать программы на языке программирования python (например, Pycharm, Anaconda, Jupyter Notebook, ...). В этой интрукции я использую Jupyter Notebook для работы.

- Сначала импорт нужный библиотеки:



- Изменить размер изображения на 300х200, (300 – ширина и 200 – высота)



Cоздайте новую папку для сохранения файлов изображений после изменения размера, назовите папку “new\_images”.

Перед расчетом яркости пикселей я предполагал, что яркости пикселей на одном (вертикальном) столбце, в одинаковых освещенных областях будут равны.

Задача изучение продольное распределение яркости I при различных давлениях плазмообразующего газа P (были проведены эксперименты при Р = 1 Торр, 2.5 Торр и 5 Торр).

Переходим к расчету яркости при давлении P = 1 Торр (рис.2)

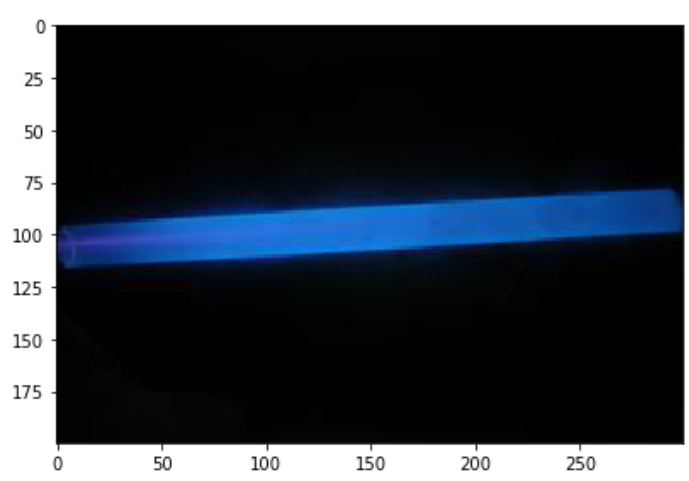
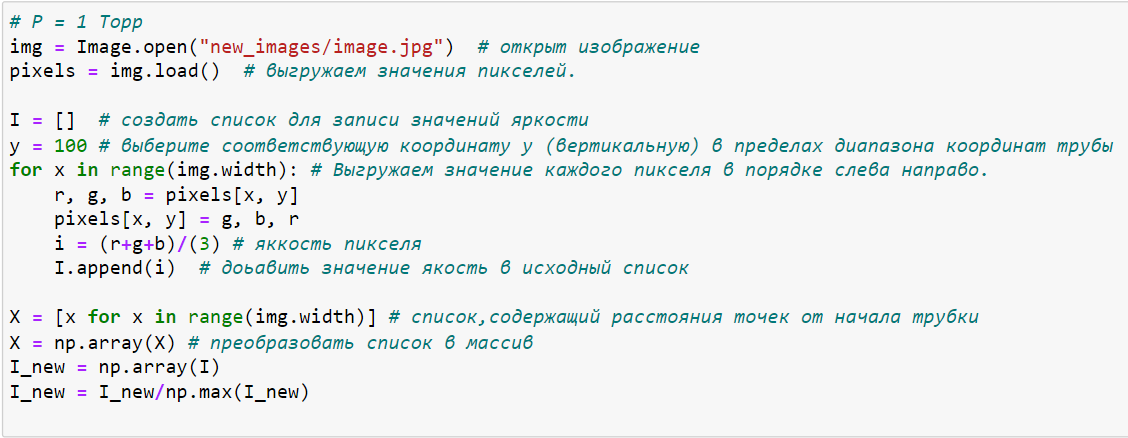


Рис.2 при Р = 1 Торр

Здесь обратите внимание, что начало координат будет верхней левой точкой, ось y направлена ​​вниз, а ось x слева направо.

В координатном распределении трубки будем вычислять яркости пикселей с координатами y = 100 в порядке слева направо. Для следующих изображений (при изменении давления) мы соответственно выбираем другое значение y (поскольку положение распределения трубки на изображениях не совпадает).

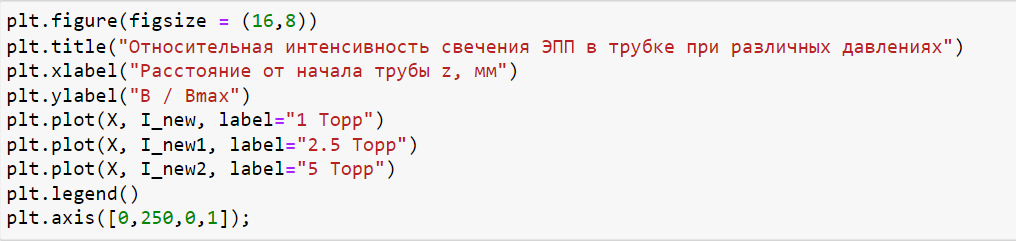
Давайте посмотрим код ниже, посчитаем яркость с помощью языка программирования python.



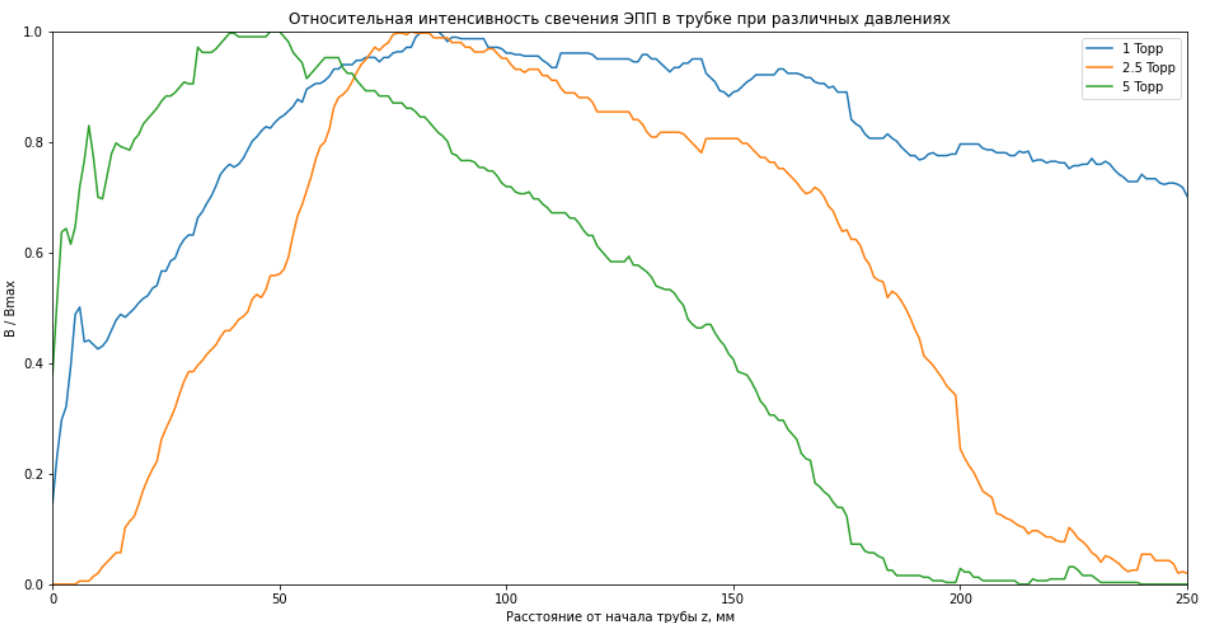
Запустив приведенный выше код, мы получим 2 массива. Массив значений расстояния от начала трубки и массив значений яркостей пикселей.

- Выполните те же шаги, что и выше, для случаев давления P = 2.5; и 5 торр.

Мы можем построить график зависимости интенсивности B(z) от расстояния z (от начала трубки).



Запустив приведенный выше код, мы получим график

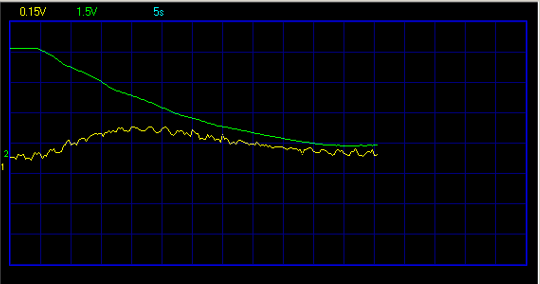


(Осталось сгладить график)

**II) Исследование зависимости между интенсивностей оптического излучения и давлением в камере.**

***Через программное обеспечение (Осциллограф)***

При откачке, давление в камере постепенно снижалось с 5 Торр до 0.5 Торр, через программное обеспечение получили график изменения давления и график изменения интенсивности свечения ЭПП. Зеленая линия показывает изменение давления в камере, желтая линия показывает изменение интенсивности свечения ЭПП.



**Note:** The file will be saved as **.txt**, to convert data to excel file is very simple, just open the saved **.txt** file and copy all data to a new ecxel file (**Ctr+A → Ctr+V**).

Выходной сигнал от вакуумного датчика имеет логарифмическую зависимость от давления во всем диапазоне измерени, он зависит от состава и типа измеряемого газа. Преобразование величины напряжения в давление и наоборот производится по следующим формулам:

Vout = a\*log(P) + b (1)

Где:

Vout – напряжение (V)

P – давлениие (мбар), 1 mbar = 0.75 Toрр.

а – усиление в V на одну декаду давления.

b – смещение напряжения в V.

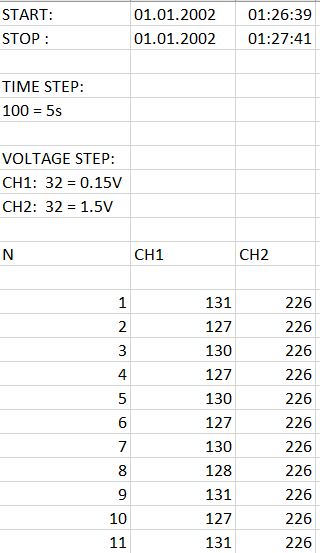
Выходной сигнал зависит от состава и типа измеряемого газа.

- **Фотодиод закреплен в расстоянии z = 7.5cm, 13cm и 20сm от выводного окна**

***+) При нагнетании воздуха в камеру***

Я буду обрабатывать данные в excel и python с случаем z = 7.5cm

Во-первых, вам нужно иметь в виду некоторые параметры при сохранении данных в файле excel.

CH1 – для измерения интенсивность

СН2 – для изиерения давление

VOLTAGE STEP

32 = 1.5 V, это означает, что на момент, когда сигнал достиг 32-го деления шкалы, напряжение на входе канала осциллографа составляло 1.5 В. Для преобразования значения в СН2 в V, нам нужно делить на 32 и умножить на 1.5. Например:

Vout\_max = 226/32\*1.5 = 10.59375

из (1), имеем: Vout = a\*log(P) + b

С значениями Рmax = 5 Topp = 20/3 мбар, Pmin = 0.5 Topp = 2/3 мбар соответствуют значениям Vout\_max = 10.59375 V и Vout\_min = 5.671875

Решив систему уравнений:

10.59375 = a\*log(20/3) + b

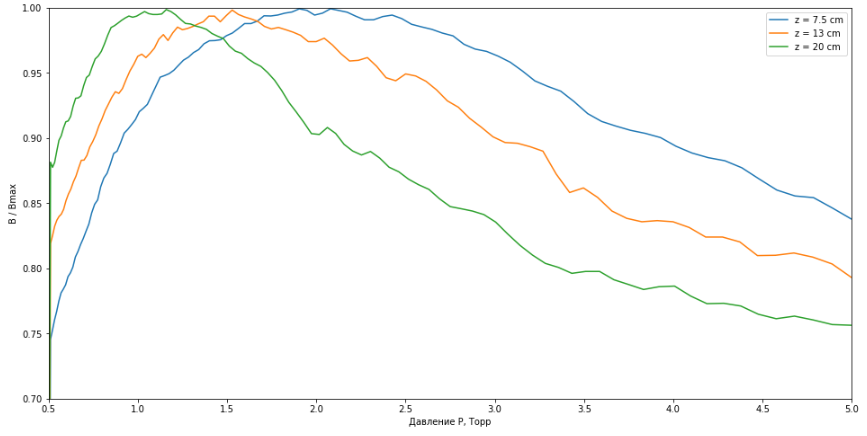
5.671875 = a\*log(2/3) + b

Получаем а = 4.921875, b = 6.5385742 и потом рассчитываем остальные значения давления.

По значению СН1 для вычисления интенсивности, мы хотим взять только отношение В/Bmax поэтому преобразовать значение СН1 не нужно.

После вычисления данных в еcxel, переходим на обработку данных и построение график в python.

После обработки данных используя Эксель и python , построен график зависимости между интенсивностью оптического излучения ЭПП и давлением в камере:



(The graphs above are still unsatisfactory, the desired result should be graphs from smooth curves, I haven't figured out how to do it yet.)

Все ресурсы выше я загрузил на github

Ссылка доступа

<https://github.com/TrinhVanDuc/image_processing.git>