# Техническое задание

На разработку графического интерфейса пользователя для скрипта определения коэффициентов виньетирования

## Общие сведения

Скрипт *vignette\_finder.py* предназначен для поиска коэффициентов виньетирования объектива по изображениям равномерно освещенной плоскости. Скрипт не имеет GUI и требует много ручных операций. Требуется на основе алгоритмов данного скрипта создать приложение с GUI, в котором были бы автоматизированы операции по загрузке входных изображений и генерации выходных файлов для конфигурации камеры.

В расчетах используется предположение, что виньетирование (затенение по краям кадра) может быть с приемлемой точностью описано радиально-симметричной моделью. Получаемые параметры необходимы для обработки изображений в фотограмметрическом ПО Agisoft Metashape. Параметры модели: положение центра виньетирования (cx, cy) и шесть полиномиальных коэффициентов. Формулы для расчета коэффициента коррекции:

где   
r – расстояние от корректируемого пикселя с координатами (х,y) до центра виньетирования,   
k – коэффициент коррекции, на который необходимо разделить исходное значение яркости пикселя.

## Алгоритм определения коэффициентов виньетирования

В качестве входных данных используется одноканальное изображение равномерно освещенной плоской мишени («белая стена»). Для уменьшения влияния шумов матрицы несколько исходных снимков с камеры усредняется.

У входного изображения определяются координаты центра виньетирования. Хорошие результаты показало использование центра масс изображения (*ndimage.measurements.center\_of\_mass*), хотя аналитического доказательства того, что центр виньетирования совпадает с центром масс - нет.

Из определения k очевидно, что в центре виньетирования k = 1. Этот факт используется для нормировки яркости изображения. За максимальный уровень изображения принимается среднее значение яркости пикселей в квадрате 11х11 с центром в центре виньетирования.

Следующим шагом строится распределение нормированной яркости пикселя от расстояния до центра виньетирования. Полученное распределение аппроксимируется полиномом 6 степени. Параметры аппроксимации и есть искомые коэффициенты виньетирования.

## Требования к разработке

1. В текущей версии скрипта в качестве входного изображения используется уже усредненное из нескольких исходных снимков. Необходимо перенести усреднение внутрь скрипта, на вход при этом нужно получать набор из нескольких исходных снимков.
2. Необходимо добавить возможность расчета к-тов виньетирования сразу для 5 каналов мультиспектральной камеры. Определить соответствие изображения определенному сенсору можно по наименованию и по метаданным изображения
3. При загрузке изображений требуется проверять (по значениям, записанным в метаданные) что весь набор выполнен при одинаковых параметрах экспозиции, иначе простое усреднение изображений будет некорректно.
4. После аппроксимации необходимо провести анализ качества аппроксимации, путем проверки, что рассчитанные значения к-тов лежат в определенных диапазонах.
5. Вывод результатов необходимо осуществлять в файлы конфигурации в текстовом формате и в формате JSON (описание см. в приложении)
6. Для удобства пользователя необходимо создать GUI, с возможностью выбора входных изображений и пути сохранения результатов
7. *\* Переписать код скрипта, чтобы для перехода от изображения к аппроксимации не требовался промежуточный файл*

## Приложение 1. Формат файлов конфигурации

### Описание Tags.ini

Файл tags.ini – текстовый файл, состоящий из 5 разделов, [Cam0] … [Cam4], в каждом содержатся одинаковое количество полей.

В поле vignetting\_center записываются координаты центра виньетирования (cx, cy)

В поле vignetting\_polynomial записываются значения к-тов k0 – k5

Пример заполнения:

[Cam0]

central\_wavelength=470

band\_name=Blue

wavelength\_fwhm=28

fnumber=1.8

band\_sensitivity=0.83

vignetting\_center=722;539

vignetting\_polynomial=1.45246135e-05;-1.50875788e-07;0;1.65036422e-13;0;-7.14037560e-19

radiometric\_calibration=0.000119266;0

[Cam1]

…

[Cam2]

…

[Cam3]

…

[Cam4]

…

### Описание tags.JSON

Файл tags.JSON содержит аналогичные tags.ini поля и данные, но в формате JSON:

{

"Cams": {

"0": {

"central\_wavelength": "470",

"band\_name": "Blue",

"wavelength\_fwhm": "28",

"fnumber": "1.8",

"band\_sensitivity": "0.83",

"vignetting\_center": "0;0",

"vignetting\_polynomial": "0;0;0;0;0;0",

"radiometric\_calibration": "0.000119266;0"

},

"1": {

"central\_wavelength": "560",

"band\_name": "Green",

"wavelength\_fwhm": "20",

"fnumber": "1.8",

"band\_sensitivity": "0.8",

"vignetting\_center": "0;0",

"vignetting\_polynomial": "0;0;0;0;0;0",

"radiometric\_calibration": "0.000123596;0"

},

"2": {

"central\_wavelength": "665",

"band\_name": "Red",

"wavelength\_fwhm": "14",

"fnumber": "1.8",

"band\_sensitivity": "0.4",

"vignetting\_center": "0;0",

"vignetting\_polynomial": "0;0;0;0;0;0",

"radiometric\_calibration": "0.000246559;0"

},

"3": {

"central\_wavelength": 720,

"band\_name": "Rededge",

"wavelength\_fwhm": 12,

"fnumber": 1.8,

"band\_sensitivity": 0.307,

"vignetting\_center": "0;0",

"vignetting\_polynomial": "0;0;0;0;0;0",

"radiometric\_calibration": "0.000322352;0"

},

"4": {

"central\_wavelength": "840",

"band\_name": "NIR",

"wavelength\_fwhm": "40",

"fnumber": "1.8",

"band\_sensitivity": "0.73",

"vignetting\_center": "0;0",

"vignetting\_polynomial": "0;0;0;0;0;0",

"radiometric\_calibration": "0.000135683;0"

}

}

}