



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Московский государственный технический университет имени
Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

Расчетно-пояснительная записка к курсовому проекту.

Тема Обработка Raw видео.

Студент Нитенко М.Ю.

Группа ИУ7-53Б

Преподаватели (научный руководитель?)

Оглавление

Введение	2
1 Аналитическая часть	4
1.1 Процесс обработки кадра.	4
1.2 Формат файла.	4
1.2.1 DNG	4
1.3 Алгоритмы демозаики.	5
1.3.1 Билинейная интерполяция	5
1.3.2 AND там или какой-нибудь	6
1.4 Цветовая модель.	6
1.4.1 CIE XYZ.	6
1.4.2 ProPhoto RGB.	7
1.5 Преобразования цветовой модели.	7
1.6 Настройка изображения.	8
1.6.1 Яркость.	8
1.6.2 Контрастность.	8
1.6.3 Насыщенность.	8
1.6.4 Баланс цветов.	9
2 Конструкторская часть	11
3 Технологическая часть	12
4 Исследовательская часть	13
Заключение	14
Литература	15

Введение

RAW видео — это видео содержащее необработанную информацию об изображении с сенсора камеры.

Главный элемент цифровых камер — сенсор, при попадании света на сенсор на нем накапливается заряд. Из этих зарядов формируется изображение.

Однако без дополнительных средств любой свет воспринимается сенсором одинаково, и на выходе получается черно-белое изображение. Наиболее распространенными способами записи цветного изображения в одну экспозицию являются: фильтр Байера, над одной матрицей или разделение изображения на три цвета, красный, зеленый и синий, и обработка каждого из них отдельной матрицей. [1]

Несмотря на то что метод разделения на три матрицы дает наиболее качественный результат, в большинстве камер среднего ценового сегмента установлена одна матрица с фильтром Байера.

Фильтр Байера состоит из 25% красных элементов, 25% синих и 50% зеленых элементов, как показано на рисунке 1.

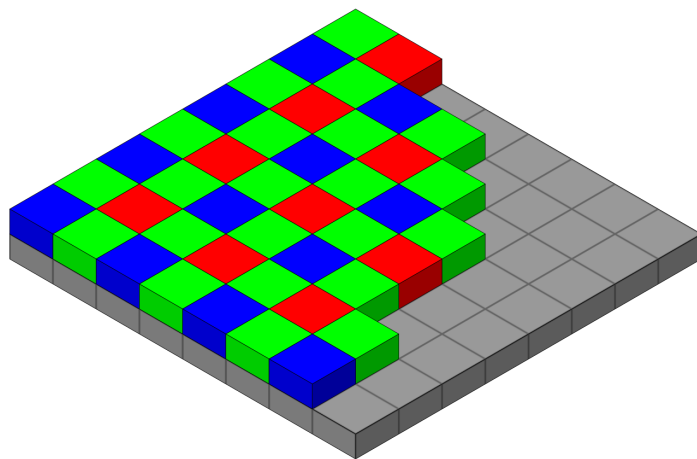


Рисунок 1 – Фильтр Байера.

Изображение с такого фильтра дает возможность создания цветного изображения, однако без обработки оно не будет таковым. Поэтому необходимо произвести процесс демозаики, который приведет изображение к корректному виду.

После этого можно приступать к остальным настройкам изображения, таким как: преобразование цвета, настройка баланса белого, тональных

кривых, контрастности, насыщенности и так далее.

Таким образом, цель данной работы — реализовать ПО позволяющее просматривать, обрабатывать и сохранять RAW видео.

Чтобы достигнуть поставленной цели, требуется решить следующие задачи:

- реализовать открытие и отображение RAW файлов;
- реализовать инструменты для обработки видео;
- реализовать возможность сохранения модифицированного видео;
- реализовать пользовательский интерфейс.

1 Аналитическая часть

В данном разделе описаны необходимые для обработки данных с сенсора алгоритмы.

1.1 Процесс обработки кадра.

Raw-кадр является набором значений с матрицы, поэтому для показа без обработки не пригоден.

Типичная обработка включает в себя:

- декодинг данных, например если каждому пикселю соответствуют 14 бит информации они, скорее всего, лежат последовательно и их придется декодировать;
- демозаика, то есть устранение фильтра Байера;
- преобразование цвета из пространства цвета камеры в общепринятое;
- изменение гаммы и прочих параметров изображения.

1.2 Формат файла.

1.2.1 DNG

Digital Negative (DNG) — формат хранения raw-изображений, основан на формате изображений TIFF. CinemaDNG является форматом хранения raw-видео и представляет собой набор DNG файлов.

DNG хранит в себе изображение и набор метаданных, для обработки кадра используются следующие поля:

- BitsPerSample — количество битов для описания каждого сэмпла (пикселя);
- BlackLevel — уровень черного, все значения меньшие или равные ему считаются минимальными;

- WhiteLevel — уровень белого, все значения больше или равные ему считаются максимальными;
- ColorMatrix — матрица для преобразования из цветового пространства XYZ в цветовое пространство камеры;
- CameraCalibration — матрица для преобразования из идеального цветового пространства камеры в цветовое пространство конкретной камеры. [2] (<— этот референс относится ко всему списку, его тут оставить или переместить куда ?)

Для обработки файлов можно использовать библиотеку «Tiny DNG Loader», эта библиотека небольшая по размеру и поддерживает необходимые поля. (<— сравнить с libraw? она вроде тоже может грузить кадры, но при этом там еще куча всего ненужного, вот так можно написать)

1.3 Алгоритмы демозаики.

Одной из главных задач обработки RAW видео является устранение эффектов фильтра Байера.

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25

Рисунок 1.1 – Пронумерованный фильтр Байера.

1.3.1 Билинейная интерполяция

Билинейная интерполяция использует среднее значение двух или четырех соседних пикселей соответствующего цвета, например: значения синего

и красного цвета для пикселя 8 находится по формулам 1.1:

$$B8 = \frac{B7 + B9}{2}, \quad R8 = \frac{R3 + R13}{2} \quad (1.1)$$

Данный алгоритм считается одним самых быстрых и часто используется для интерполяции видео в реальном времени.

1.3.2 АНД там или какой-нитб

еще один, например медленнее? а если на фпс то надо писать про артефакты? если я на фпс работаю, или лучше по качеству если на качество? обязательно самописный?

1.4 Цветовая модель.

Цветовая модель — это математическая модель описания представления цветов в виде кортежей чисел, называемых цветовыми компонентами или цветовыми координатами. Изображение с матрицы находится в цветовой модели камеры и для правильного представления картинки необходимо преобразование цветов. Например, формат DNG хранит в себе матрицу для преобразования изображения из цветовой модели камеры в CIE XYZ D50.

1.4.1 CIE XYZ.

В цветовой модели CIE XYZ каждый элемент кортежа примерно соответствует одной из колбочек человеческого глаза: X — длинноволновым, Y — средневолновым и Z — коротковолновым.

1.4.2 ProPhoto RGB.

Цветовая модель в которой значения кортежа означают значения основных цветов: красного, зеленого и синего. Остальные цвета получаются сочетанием базовых. Цвета такого типа называются аддитивными.

Цветовая модель ProPhoto RGB покрывает 90% возможных цветов модели CIE LAB и является рекомендованной в спецификации DNG цветовой моделью. [2]

1.5 Преобразования цветовой модели.

Для преобразования часто используются матрицы. Пусть CM — матрица преобразующая XYZ D50 в цветовое пространство камеры, тогда CM^{-1} будет матрицей переводящей цветовое пространство матрицы в XYZ D50. Пусть XTP :

$$XTP = \begin{bmatrix} 1.3460 & -0.2556 & -0.0511 \\ -0.5446 & 1.5082 & 0.0205 \\ 0.0 & 0.0 & 1.2123 \end{bmatrix} \quad (1.2)$$

матрица преобразующая XYZ D50 в ProPhoto RGB. Тогда для преобразования изображения из цветового пространства камеры в ProPhoto RGB необходимо произвести умножение:

$$T_{ProPhotoRGB} = XTP * CM * T_{CC} \quad (1.3)$$

где T_{CC} — кортеж с значениями цвета в пространстве камеры.

$T_{ProPhotoRGB}$ находится в цветовом пространстве ProPhoto RGB, но яркость все еще закодирована линейно, для правильного отображения необходимо применить гамма-коррекцию:

$$\gamma(u) = \begin{cases} 16u, & u \leq 0.001953 \\ u^{1/1.8} & \end{cases} \quad (1.4)$$

где u — одна из компонент цвета. [3]

1.6 Настройка изображения.

1.6.1 Яркость.

Преобразование яркости определяется как:

$$br(a) = a + N \quad (1.5)$$

где a — значение яркости пикселя, а N — желаемое увеличение в яркости. [4]

1.6.2 Контрастность.

Преобразование контрастности определяется как:

$$con(a) = a * N \quad (1.6)$$

где a — значение яркости пикселя, а N — желаемое увеличение в яркости (чтобы, например, увеличить контрастность на 50% необходимо умножить на 1.5). [4]

1.6.3 Насыщенность.

Для изменения насыщенности изображения в цветовой модели RGB можно воспользоваться умножением матриц.

Пусть $F(x, y)$ — вектор:

$$F(x, y) = [f_R, f_G, f_B, 1]^T \quad (1.7)$$

где f_R, f_G, f_B — значения цвета в точке x, y .

Тогда $G(x, y)$ — вектор содержащий значения цвета с иной насыщенностью:

$$G(x, y) = [g_R, g_G, g_B, g_w]^T \quad (1.8)$$

где g_R, g_G, g_B — новые значения цвета в точке x, y , а g_w не используется.

Высчитать $G(x, y)$ можно по формуле 1.9:

$$G(x, y) = T * F(x, y) \quad (1.9)$$

где T — матрица преобразования:

$$T_{sat}(s) = \begin{bmatrix} \alpha + s & \beta & \gamma & 0 \\ \alpha & \beta + s & \gamma & 0 \\ \alpha & \beta & \gamma + s & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1.10)$$

где $\alpha = 0.3086(1 - s)$, $\beta = 0.6094(1 - s)$ и $\gamma = 0.0820(1 - s)$.

Значения $s < 1$ приводят к уменьшению насыщенности, значения > 1 — к увеличению. [5]

1.6.4 Баланс цветов.

Для изменения баланса цвета в цветовой модели RGB можно воспользоваться умножением матриц.

Пусть $F(x, y)$ — вектор:

$$F(x, y) = [f_R, f_G, f_B, 1]^T \quad (1.11)$$

где f_R, f_G, f_B — значения цвета в точке x, y .

Тогда $G(x, y)$ — вектор содержащий значения цвета с иным балансом цвета:

$$G(x, y) = [g_R, g_G, g_B]^T \quad (1.12)$$

где g_R, g_G, g_B — новые значения цвета в точке x, y .

Высчитать $G(x, y)$ можно по формуле 1.13:

$$G(x, y) = T * F(x, y) \quad (1.13)$$

где T — матрица преобразования:

$$T_{sat}(s) = \begin{bmatrix} R_{wb} & 0 & 0 \\ 0 & G_{wb} & 0 \\ 0 & 0 & B_{wb} \end{bmatrix} \quad (1.14)$$

где R_{wb} , G_{wb} и B_{wb} — коэффициенты для каждого из основных цветов.

Вывод

В данном разделе был проведен обзор необходимых для реализации алгоритмов.

2 Конструкторская часть

Вывод

3 Технологическая часть

Вывод

4 Исследовательская часть

Вывод

Заключение

В ходе выполнения лабораторной работы была проделана следующая работа:

- замерено время выполнения алгоритмов;

Литература

- [1] Сердце цифровой фотокамеры: ПЗС-матрица (часть четвёртая) [электронный ресурс]. Режим доступа: <https://www.ferra.ru/review/multimedia/71885.htm> (дата обращения: 05.11.2020).
- [2] ADOBE SYSTEMS INCORPORATED. Digital Negative (DNG) Specification.
- [3] ANSI, USA. Specification of ROMM RGB.
- [4] Wilhelm Burger, Mark James Burge. Principles of Digital Image Processing: Fundamental Techniques. Springer International Publishing, 2009. Vol. 1. P. 55–59.
- [5] Janglin Chen, Wayne Cranton, Mark Fihn. Handbook of Visual Display Technology. Springer International Publishing, 2016. Vol. 1. P. 461–462.