



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 «Радиоэлектроника и лазерная техника»

КАФЕДРА «Лазерные и оптико-электронные системы»

ОТЧЕТ ПО ПРОИЗВОДСТВЕННОЙ ПРАКТИКЕ

Студент РЛ2-81
(Группа)

(Подпись, дата)

Барашев Д.Р.
(И.О.Фамилия)

Руководитель практики

(Подпись, дата)

Литвинов И.С.
(И.О.Фамилия)

2023 г.



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 «Радиоэлектроника и лазерная техника»

КАФЕДРА «Лазерные и оптико-электронные системы»

ОТЧЕТ ПО ПРОИЗВОДСТВЕННОЙ ПРАКТИКЕ

Студент Барашев Д.Р.
фамилия, имя, отчество

Группа РЛ2-81

Тип практики _____

Название предприятия АО ЦНИИ Циклон

Студент	_____	_____
	<i>подпись, дата</i>	<i>фамилия, и.о.</i>

Руководитель практики	_____	_____
	<i>подпись, дата</i>	<i>фамилия, и.о.</i>

Оценка _____

2023 г.

GoPro Hero 11:.....	3
Технические характеристики	3
Алгоритмы:	6
HyperSmooth Стабилизация:.....	7
TimeWarp	8
Алгоритмы автоматической экспозиции:	10
Алгоритмы с высоким динамическим диапазоном (HDR):.....	11
Метод Дебвека	15
Метод Робертсона.....	16
Метод Мертенса.....	18
Метод Рейнхарда	20
Метод Дюранда и Дорси.....	22
Алгоритмы автофокусировки.....	23
Алгоритмы автоматического баланса белого (AWB)	24
Реализация метода Дебвека	25
Выводы:	40

GoPro Hero 11:

Технические характеристики

<p>Socionext GoPro GP2</p> <p>4GB LPDDR4 Package on Package</p>	<p>Imaging "GoPro GP2" processor (M20V) ARMV8 / CEVA DSP / Target ID 0x2751</p> <p>Processor</p> <p>Micron MT53D512M64D4NZ-053 (GoPro Hero 11) Code: D9VQG [end of life] Micron MT53E512M64D2NZ-46 (GoPro Hero 11 Mini) Code: D8CZK</p>
<p><u>Qualcomm</u> <u>QCA9377</u></p>	<p>High performance, low power single-stream 11ac MU-MIMO and Bluetooth 5 in a single chip solution.</p>

<p>NEW</p> <p>Sony IMX677L</p>	<p>New Taller sensor 27.5MPixel 1/1.9inch (8.42mm) 8:7 aspect ratio, SLVS Interface</p> <p>Geometry 5663px x 4949 12-bit Used 5598px x 4926 10-bit (27.13MP) Output Video: 5312px x 3984px</p> <p>Sony IMX677L is not offered yet on their line card</p>
<p><u>Bosch</u> <u>BMI260 IMU</u></p>	<p>Accelerometer / Gyro pair 6-axis sensor.</p>
<p>GoPro Hero 11 Only</p> <p><u>UBlox MAX-M10S</u></p>	<p>GPS Receiver u-blox M10. 1v8 device. Updated for part shortages.</p> <p>[Same as Hardware B on HERO10]</p>
<p>GoPro Hero 11 Mini Only</p> <p>Audio</p>	<p>Mono Only</p>
<p>GoPro Hero 11Audio</p>	<p>Stereo Audio</p>
<p>Qorvo ACT9150</p>	<p>PMIC (Power Management IC) for Socionext. Power supply, charging, and LED control</p>
<p><u>Qorvo</u> <u>ACT88760</u></p>	<p>PMIC #2 0v8 AVS regulator</p>
<p>GoPro Hero 11 Only</p> <p>Controller <u>Sitronix</u> <u>st7701s</u></p>	<p>Rear LCD (0xC3) 16.7M-color (368px by 552px) 320px by 552px used Manufacturers: BOE, NVD, Tianma</p>
<p>GoPro Hero 11 Only</p> <p>FocalTech ft3308</p>	<p>Rear Touch sensor 368px by 552px</p>

GoPro Hero 11 Only Sitronix st7796h	Front 320RGB x 480 dot 262K LCD Manufacturers: Tianma or Truly Color
GoPro Hero 11 Mini Only Controller ssd1309 oLED	oLED on top 72 x 32px
NEW <u>STM32G081RBI6</u> Security Enclave "Micronesia 3.0?"	<ul style="list-style-type: none"> • USB-C PD • Accessory authenticator to ATSHA204 • Battery Authentication • Reset Handler Monitor • RTC - Always "on" • Encrypted and SIGNED boot & app • Authenticates BQ Battery sensor
<u>Samsung</u> <u>KLM8G1GETF-B041</u>	8GB eMMC 5.1 HS400 interface
Pericom (Diodes Inc.) PI3DBS12612	USB Type-C port demultiplexor
Pericom (Diodes Inc.) PI3USB9281	USB BC2.1 Charge Detector
GoPro Enduro Battery	Chem-ID 0x3936 TI Fuel Gauge TI SN27Z562

Процессор GP2, разработанный GoPro, является примером прикладной интегральной схемы (ASIC). ASIC специально разработаны для конкретной цели - в данном случае обработка изображений и видео для линейки экшн-камер GoPro.

ASIC, как и процессор GP2, как правило, высоко оптимизированы для их конкретного применения. Они предназначены для повышения

производительности, энергоэффективности и функциональности по сравнению с процессорами общего назначения.

Процессор GP2, в частности, предназначен для выполнения таких задач, как:

1. Обработка сигнала изображения, которая включает в себя преобразование необработанных данных с датчика изображения камеры в формат, который может отображаться или храниться.
2. Кодирование и декодирование видео, которое представляет собой процесс сжатия видеоданных для хранения или передачи, а затем распаковки их для просмотра.

Алгоритмы:

Камеры GoPro используют набор передовых алгоритмов и технологий обработки видео. Последней моделью GoPro является Hero 11 Black, которая использует следующие ключевые функции обработки видео:

1. **HyperSmooth Стабилизация:** фирменный алгоритм стабилизации видео GoPro, HyperSmooth 5.0 в Go Pro 11, использует комбинацию аппаратного и программного обеспечения для прогнозирования движений и коррекции дрожания камеры, в результате чего получаются исключительно плавные кадры. Он доступен во всех разрешениях и частоте кадров.

2. **TimeWarp:** Эта функция применяет сверхгладкую стабилизацию к замедленным видеороликам, обеспечивая невероятно плавное воспроизведение замедленных видеороликов, даже когда камера находится в движении.

3. **Алгоритмы с высоким динамическим диапазоном (HDR):** Алгоритмы HDR захватывают и отображают больший диапазон яркости изображения или видео, обеспечивая большую детализацию как в светлых, так и в темных областях. Функция GoPro SuperPhoto + HDR настраивает это автоматически.

4. Процессор GP2: Этот процессор выполняет задачи обработки сигнала изображения и обеспечивает расширенные функции камеры. Он играет ключевую роль во всем - от стабилизации изображения до кодирования и декодирования видео.

5. Алгоритмы автоматической экспозиции: камеры GoPro автоматически настраивают настройки камеры для поддержания правильной экспозиции в зависимости от условий освещения и объектов в кадре.

6. Цветокоррекция / градуировка: Камеры GoPro поставляются с функцией Protune, которая позволяет регулировать цвет, ограничение ISO, резкость и другие параметры качества видео.

7. Алгоритмы автофокусировки: Хотя это и не традиционная система автофокусировки, как в зеркальных камерах, камеры GoPro используют объектив с фиксированным фокусом, предназначенный для того, чтобы запечатлеть как можно больше в резком фокусе.

8. Автоматический баланс белого: алгоритм AWB GoPro корректирует цветовой баланс изображения, чтобы оно выглядело более естественным при различных условиях освещения.

HyperSmooth Стабилизация:

HyperSmooth - передовая технология цифровой стабилизации изображения, которая помогает обеспечить плавность видеосъемки и отсутствие дрожаний.

Захват дополнительной информации: функция HyperSmooth работает за счет захвата немного большего количества информации по краям кадра, чем то, что вы в конечном итоге увидите в финальном видео. Эта дополнительная информация необходима для процесса стабилизации, поскольку она дает программному обеспечению больше возможностей для настройки отснятого

материала без уменьшения видимой области видео. По этой причине, когда включается стабилизация, поле зрения становится немного уже.

Анализ движения: Далее технология HyperSmooth анализирует движение видео, путем сравнения последовательных кадров и определения того, как объекты перемещались между ними. Это позволяет оценивать движение камеры во время процесса съемки.

Регулировка кадра: Как только движение камеры известно, технология HyperSmooth может настроить видео таким образом, чтобы противодействовать этому движению. Перемещая изображение из одного положения в другое или слегка поворачивая его, чтобы компенсировать перемещение камеры.

Сглаживание видео: Результатом этих настроек является видео, в котором каждый кадр более точно совпадает со следующим, что приводит к более плавному видео, свободному от дрожаний.

Важно отметить, что технология HyperSmooth предназначена для уменьшения незначительных непреднамеренных перемещений камеры, таких как небольшие сотрясения, возникающие при удержании камеры рукой. Более крупные движения, такие как панорамирование камеры для слежения за объектом, обычно не компенсируются технологией стабилизации.

TimeWarp

TimeWarp - это термин, обозначающий функцию стабилизации замедленного видео. Замедленные видеоролики - это видеоролики, в которых время, кажется, течет быстрее. Обычно они создаются путем съемки серии фотографий с заданными интервалами, а затем объединения их в видео. Однако традиционные замедленные видеозаписи могут стать неустойчивыми при движении камеры, и именно здесь на помощь приходит TimeWarp.

Вот упрощенное объяснение того, как работает TimeWarp:

Захват кадров с интервалами: Точно так же, как традиционное замедленное видео, TimeWarp работает, захватывая кадры с заданными интервалами. Точный интервал зависит от скорости выполнения действия. Например, при ходьбе GoPro может снимать кадр каждые полсекунды, в то время как для более быстрых действий, таких как вождение автомобиля, он может снимать кадр каждые две секунды.

Применение SuperSmooth стабилизации: Что отличает TimeWarp от традиционного замедленного видео, так это то, что он применяет к отснятому материалу сверхгладкую стабилизацию GoPro. Это помогает устранить дрожание, которое может возникать при движении камеры, в результате чего получается более плавное и удобное для просмотра замедленное видео.

Создание видео TimeWarp: Как только все кадры будут сняты и стабилизированы, они объединяются в единое видео, причем каждый кадр занимает долю секунды в конечном видео. В результате получается видео, в котором время, кажется, течет намного быстрее, чем обычно, что и является сутью замедленного видео.

Одной из уникальных особенностей TimeWarp является возможность замедления до реального времени нажатием кнопки, что позволяет запечатлеть ключевые моменты во всех деталях. Затем вы можете снова увеличить скорость, чтобы продолжить замедленную съемку.

Важно отметить, что качество видео TimeWarp, как и любого другого видео, будет зависеть от условий, в которых оно снято. Хорошее освещение и устойчивое движение камеры приведут к получению лучшего видео с искажением времени.

Алгоритмы автоматической экспозиции:

Автоэкспозиция (АЭ) - это ключевая функция в цифровой обработке изображений, отвечающая за автоматическую настройку параметров экспозиции для обеспечения надлежащей яркости изображения или видео. Алгоритм АЭ управляет такими параметрами, как ISO, диафрагма и выдержка, в зависимости от текущих условий освещения.

Как правило, существует несколько типов алгоритмов АЭ:

Средний замер (или матричный замер): Этот метод оценивает яркость по всему кадру изображения и устанавливает экспозицию на основе средней яркости. Это хорошо работает для равномерно освещенных сцен, но может не сработать в высококонтрастных сценах, где есть очень яркие и очень темные области.

Измерение с взвешиванием по центру: Этот метод придает большее значение центру кадра при расчете средней яркости. Это полезно в ситуациях, когда объект съемки находится в середине кадра, например, на портретах.

Точечный замер: Этот метод оценивает только небольшую область вокруг центра кадра (или другой выбранной точки фокусировки). Это полезно, когда объект съемки значительно ярче или темнее фона, поскольку обеспечивает правильную экспозицию объекта, даже если остальная часть сцены оказывается недоэкспонированной или переэкспонированной передержкой.

Частичный замер: Этот метод аналогичен точечному измерению, но оценивает большую площадь. Это нечто среднее между точечным измерением и центровзвешенным измерением.

Оценочный замер (или многозонный замер): Этот метод разбивает сцену на несколько зон и оценивает каждую из них отдельно. Затем он использует сложный алгоритм для определения правильной экспозиции на

основе яркости каждой зоны и их важности. Точный алгоритм может значительно варьироваться в зависимости от разных камер и может включать такие факторы, как расстояние фокусировки, информация о цвете и многое другое.

Измерение с учетом выделений: этот метод определяет приоритетность выделений на изображении, гарантируя, что они не будут переэкспонированы и сохраняют свою детализацию. Это особенно полезно в сценах с яркими источниками света или отражениями.

Точная реализация этих методов может значительно различаться у разных камер и производителей, и многие камеры предоставляют опции для выбора используемого метода или для ручного переопределения АЭ.

Большинство продвинутых камер, включая камеры GoPro, обычно предлагают **средний замер (или матричный замер), центровзвешенный замер и точечный замер.**

В дополнение к этому, камеры GoPro имеют автоматический режим низкой освещенности. Когда эта функция активирована, камера автоматически отрегулирует частоту кадров в зависимости от условий освещения, чтобы улучшить характеристики при низкой освещенности и качество изображения.

В GoPro также есть функция ProTune, которая обеспечивает ручное управление цветом, балансом белого, выдержкой и пределом ISO. Это полезная функция для пользователей, которые хотят иметь больше контроля над настройками экспозиции и внешним видом своего видео.

Алгоритмы с высоким динамическим диапазоном (HDR):

Высокий динамический диапазон (HDR) в контексте обработки изображений и видео - это технология, которая обеспечивает больший динамический диапазон яркости, чем могут предложить стандартные методы цифровой обработки изображений или фотографические методы.

Он работает путем съемки нескольких фотографий одной и той же сцены с разными уровнями экспозиции (захват теней, полутонов и бликов по отдельности), а затем объединения их в одно изображение. Каждое из изображений в идеале имеет разное время экспозиции - одно короткое (для светлых областей), одно среднее (для полутонов) и одно продолжительное (для темных областей). Процесс объединения этих экспозиций в единое изображение называется тональным отображением.

Камеры GoPro, включая GoPro HERO 11, поддерживают режим HDR, который автоматизирует этот процесс. Когда вы делаете снимок в режиме HDR, камера захватывает несколько изображений с разной экспозицией и обрабатывает их внутри системы для создания единого, насыщенного по тону изображения. Этот режим полезен для сцен с высокой контрастностью, например, с очень яркими и очень темными участками.

В видео применение HDR более сложное из-за движения в кадрах, но некоторые принципы остаются прежними - речь идет о захвате и отображении большего диапазона яркости.

Серия GoPro Hero, включая Hero 11, также включает в себя расширенную функцию под названием SuperPhoto. SuperPhoto автоматически настраивает HDR, локальное отображение тонов и шумоподавление в камере для достижения наилучшего качества фотографий.

Комбинирование кадров при съемке в высоком динамическом диапазоне (HDR) используется для получения большего диапазона яркости, чем это возможно при однократной экспозиции. Это особенно полезно в сценах с высокой контрастностью, где есть как очень яркие, так и очень темные области.

Вот пошаговый обзор этого процесса:

Съемка с несколькими экспозициями: Сначала камера делает несколько снимков одной и той же сцены с разными уровнями экспозиции. Как правило, будет одно недоэкспонированное изображение (для захвата деталей в светлых областях или бликов), одно переэкспонированное изображение (для захвата деталей в темных областях или тенях) и одно правильно экспонированное изображение.

Объедините изображения: Далее эти изображения объединяются в одно. Самый простой способ сделать это - взять самые темные участки с переэкспонированного изображения, самые яркие участки с недоэкспонированного изображения и средние тона с правильно экспонированного изображения. Это гарантирует сохранение деталей во всем диапазоне яркости.

Отображение тонов: Заключительным шагом является отображение тонов, которое масштабирует диапазон яркости объединенного изображения обратно до диапазона, который может быть отображен или распечатан. Цель здесь состоит в том, чтобы максимально сохранить яркость, контрастность и цветовые соотношения оригинальной сцены.

Комбинируя таким образом несколько экспозиций, HDR-фотография может отображать более широкий диапазон уровней яркости, чем стандартная цифровая фотография или человеческое зрение. Это может привести к получению более ярких и реалистичных изображений, особенно при просмотре HDR-контента на HDR-совместимом дисплее.

Когда дело доходит до HDR-видео, концепция в чем-то схожа, но реализация отличается из-за движения и природы видео в реальном времени. Вместо объединения нескольких видеок кадров HDR-видео обычно включает в себя захват и обработку большего диапазона информации о яркости и цвете в пределах отдельных кадров. Однако конкретные методы могут варьироваться в зависимости от используемой камеры и программного обеспечения.

Существует несколько алгоритмов или методов, обычно используемых в процессе получения изображений в формате HDR:

Метод Дебвека: Также известный как **метод карты сияния**, он использует изображения сцены с разной экспозицией для вычисления функции отклика камеры и карты сияния сцены.

Метод Робертсона: Этот метод направлен на восстановление функции отклика камеры и карты освещенности сцены по последовательности изображений, сделанных с разным временем экспозиции. Это итеративный метод, который поочередно решает для карты сияния и функции отклика до тех пор, пока не произойдет сходимость.

Мертенса-Каутца-Ван Рита (MKVR) или метод Мертенса: Этот метод, как упоминалось ранее, объединяет изображения с разной экспозицией в единую карту яркости, которая охватывает гораздо больший диапазон интенсивностей, но для этого не требуется знание значений экспозиции или функции отклика камеры. Веса для объединения изображений вычисляются на основе контрастности, насыщенности и хорошей экспозиции.

Метод Рейнхарда: Названный в честь Эрика Рейнхарда, этот метод не требует расчета кривой отклика камеры. Он масштабирует интенсивность пикселей таким образом, чтобы результирующее изображение максимально приближалось к человеческому восприятию цвета и яркости.

Метод Дюранда и Дорси: Этот метод основан на двухмасштабной модели визуальной адаптации для рендеринга изображений с высоким динамическим диапазоном. Он предназначен для приближения зрения в самых разнообразных условиях просмотра, от скотопического (почти темного) до фотопического (очень яркого).

Эти методы могут быть использованы для объединения изображений и отображения тонов, и часто требуют дальнейшей обработки и ручной настройки для получения конечного HDR-изображения. Выбор метода будет зависеть от конкретных потребностей обрабатываемого изображения или последовательности изображений.

Метод Дебвека

Метод создания изображений с высоким динамическим диапазоном (HDR), разработанный Дебвеком, является одним из наиболее часто используемых методов в области компьютерной графики и фотографии. Вот упрощенный обзор того, как это работает:

Захват изображений в брекетинг: Сначала вам нужно сделать серию снимков с разными уровнями экспозиции. Это часто делается с помощью функции камеры, называемой "автоматический брекетинг", где вы можете настроить камеру на автоматическую съемку серии снимков с разной экспозицией (обычно с разной выдержкой при сохранении постоянной диафрагмы и ISO).

Вычисление функции отклика камеры (CRF): Цель здесь состоит в том, чтобы оценить реакцию камеры на различные уровни освещенности, т.е. определить, как камера соотносит определенную интенсивность света со значениями цвета. Это достигается путем решения системы уравнений, которые связывают значения пикселей на изображениях, заключенных в

квадратные скобки, с соответствующей им интенсивностью света с учетом времени экспозиции. Этот шаг имеет решающее значение, поскольку цифровые камеры применяют нелинейные преобразования к полученным значениям интенсивности света.

Создание карты яркости HDR: как только у нас будет CRF, мы сможем использовать ее для объединения наших изображений, заключенных в квадратные скобки, в единое изображение, которое имеет гораздо более широкий диапазон значений цвета и яркости, чем любое из отдельных изображений. Другими словами, для каждого местоположения пикселя мы определяем соответствующую яркость сцены, комбинируя значения пикселей из каждого изображения, принимая во внимание CRF и время экспозиции.

Отображение тонов (необязательно): HDR-изображения содержат широкий диапазон значений яркости, часто более широкий, чем то, что может быть отображено на стандартных экранах или напечатано. Итак, для визуализации HDR-изображений на стандартных устройствах может использоваться процесс, называемый тональным отображением. Этот процесс преобразует цвета из широкого диапазона, доступного в HDR-изображении, в более узкий диапазон, который может быть отображен.

Важно отметить, что, хотя метод Дебевека является мощным, он не идеален для сцен со значительным движением, поскольку требует нескольких снимков, сделанных за определенный промежуток времени. Кроме того, метод предполагает, что сцена статична и что единственным изменением между различными изображениями является уровень экспозиции.

Метод Робертсона

Метод Робертсона, предложенный Эриком Рейнхардом, является еще одним подходом к созданию изображений с высоким динамическим диапазоном (HDR). Это включает в себя следующие шаги:

Захват изображений в брекетинг: Аналогично методу Дебевека, вам нужно будет сделать серию снимков с разными уровнями экспозиции. Это часто связано с использованием функции камеры, называемой "автоматический брекетинг", когда камера автоматически делает серию снимков с разной экспозицией. Различные уровни экспозиции часто достигаются за счет изменения выдержки при сохранении постоянной диафрагмы и ISO.

Оценивание функции отклика камеры (CRF) и карты яркости одновременно: Основное различие между методом Робертсона и методом Дебевека заключается в том, что подход Робертсона одновременно оценивает функцию отклика камеры (CRF) и карту яркости. Этот итеративный метод обеспечивает первоначальную оценку CRF и карты сияния, а затем уточняет их на последующих итерациях. В некоторых случаях этот метод может быть более точным, чем метод Дебевека, особенно когда первоначальное предположение для CRF близко к истинной функции.

Отображение тонов (необязательно): Как и в случае с методом Дебевека, результирующее HDR-изображение по методу Робертсона содержит широкий диапазон значений яркости, часто более широкий, чем то, что может быть отображено на стандартных экранах или напечатано. Для визуализации HDR-изображений на стандартных устройствах может использоваться процесс, называемый тональным отображением. Этот процесс преобразует цвета из широкого диапазона, доступного в HDR-изображении, в более узкий диапазон, который может быть отображен.

Стоит отметить, что, хотя метод Робертсона может давать более точные результаты в определенных сценариях, он также может быть более интенсивным с точки зрения вычислений из-за его итеративного характера. Как и в случае с методом Дебевека, метод Робертсона также предполагает, что сцена статична и что единственным изменением между различными изображениями является уровень экспозиции.

Метод Мертенса

Функция слияния Мертенса - это тип функций, используемых в изображениях с высоким динамическим диапазоном (HDR). Он назван в честь своего создателя Фредерика Драго и иногда также известен как метод Драго.

Метод создания изображений с высоким динамическим диапазоном (HDR), разработанный Мертенсом, также известен как *exposure fusion*. В отличие от других методов, таких как методы Дебевека или Робертсона, он явно не вычисляет карту яркости или функцию отклика камеры. Вместо этого он напрямую объединяет изображения с разной экспозицией в единое изображение высокого качества без создания промежуточного HDR-изображения.

Метод, по сути, работает путем присвоения различных весовых коэффициентов различным значениям пикселей, основанным на их контрастности, насыщенности и хорошей экспозиции.

Контрастность: Пикселям с более высокой контрастностью (разница в интенсивности с окружающими их пикселями) присваиваются более высокие веса.

Насыщенность: пикселям, которые являются более насыщенными (более интенсивный цвет), присваиваются более высокие веса.

Хорошая экспозиция: пикселям, которые не близки ни к черному, ни к белому, присваиваются более высокие веса.

После вычисления этих весовых коэффициентов метод Мертенса объединяет изображения таким образом, чтобы приоритизировать пиксели с более высокими весовыми коэффициентами, генерируя таким образом выходное изображение, которое объединяет наиболее экспонированные части каждого входного изображения. Результатом является объединение входных изображений, которое позволяет запечатлеть сцену с высоким динамическим

диапазоном в одном выходном изображении. Это слитное изображение затем может быть преобразовано в тональное отображение для создания отображаемого HDR-изображения.

Эта функция уникальна тем, что, в отличие от других методов HDR, для нее не требуется знание значений экспозиции или функции отклика камеры. Он работает путем объединения изображений с разной экспозицией в единую карту сияния, которая охватывает гораздо больший диапазон интенсивностей.

Метод Мертенса включает в себя следующие этапы:

Захват изображений в брекетинг: Как и в случае с другими методами, вам необходимо захватить серию изображений с разными уровнями экспозиции.

Вычисление показателей качества: Для каждого изображения в стеке экспозиции вычислите показатели качества для каждого пикселя. Эти показатели качества обычно основаны на таких факторах, как насыщенность, контрастность и хорошая экспозиция. Затем эти показатели объединяются, чтобы сформировать карту качества для каждого изображения.

Объединение изображений: метод Мертенса использует многомасштабный алгоритм для объединения изображений вместе на основе их карт качества. Идея состоит в том, чтобы извлечь больше информации из изображений, которые имеют более высокие оценки качества в соответствии с картой качества.

Приведение результата в пределы отображаемого диапазона: Поскольку процесс слияния может привести к значениям пикселей, выходящим за пределы отображаемого диапазона, результирующее изображение нормализуется, чтобы привести его в пределы отображаемого диапазона.

Одно из преимуществ метода Мертенса перед другими заключается в том, что он, как правило, лучше справляется с движением в сцене. Поскольку он не полагается на создание промежуточного HDR-изображения, он может более изящно обрабатывать несоответствия в стеке изображений. Однако полученное изображение технически не является HDR-изображением; скорее, это изображение с низким динамическим диапазоном (LDR), которое было объединено с нескольких экспозиций для сохранения широкого спектра деталей.

Метод Рейнхарда

Метод Рейнхарда, разработанный Эриком Рейнхардом и его коллегами, представляет собой разновидность техники отображения тонов, используемой для преобразования изображений с высоким динамическим диапазоном (HDR) в формат, который может отображаться на устройствах, поддерживающих только стандартный динамический диапазон (SDR).

Целью тонального отображения является воспроизведение внешнего вида изображений с высоким динамическим диапазоном при отображении на устройствах, которые не могут полностью воспроизвести исходный динамический диапазон. Он пытается сохранить визуальную взаимосвязь между различными цветами и интенсивностями, даже несмотря на то, что абсолютные значения, возможно, придется уменьшить.

Метод Рейнхарда позволяет достичь этого с помощью нескольких шагов:

Захват HDR-изображения: Как и при использовании других методов, для начала вам понадобится HDR-изображение. Это часто достигается путем

объединения набора изображений, сделанных при разных уровнях экспозиции, с использованием таких методов, как методы Дебевека или Робертсона.

Вычисление яркости: Следующим шагом в методе Рейнхарда является вычисление яркости изображения. Яркость - это мера интенсивности света, который люди воспринимают по изображению. Другими словами, это яркость изображения, воспринимаемая человеческим глазом. Это часто делается с использованием формулы, которая взвешивает вклад каждого цветового канала в зависимости от того, насколько чувствительны к нему человеческие глаза.

Масштабирование яркости: затем значения яркости масштабируются таким образом, чтобы они соответствовали диапазону, который может отображаться стандартными устройствами. Рейнхард предложил два подхода для достижения этой цели: глобальный оператор, который масштабирует все изображение на основе средней яркости, и локальный оператор, который масштабирует разные части изображения по-разному, сохраняя больше деталей.

Отображение тонов: затем масштабированные значения яркости объединяются с исходной информацией о цвете для получения конечного изображения, которое может отображаться на стандартных устройствах. Полученное изображение должно произвести на зрителя впечатление, похожее на то, которое возникло бы у него при просмотре реальной сцены с высоким динамическим диапазоном.

Метод тонального отображения Рейнхарда широко используется благодаря своей простоте и визуально приятным результатам, которые он дает. Однако, как и все операторы отображения тонов, это приблизительное решение, и оно не может полностью воспроизвести впечатления от просмотра реальной сцены с высоким динамическим диапазоном.

Метод Дюранда и Дорси

Метод Дюранда и Дорси, также известный как двусторонняя фильтрация, - это метод, который в основном используется для отображения тонов в изображениях с высоким динамическим диапазоном (HDR). Этот метод помогает сохранить края и детали при одновременном уменьшении мелкомасштабных контрастов, что позволяет лучше управлять преобразованием изображений из HDR в изображения со стандартным динамическим диапазоном (SDR).

Этот метод включает в себя несколько этапов:

Захват HDR-изображения: Это похоже на другие методы HDR, при которых вы снимаете серию изображений с разными уровнями экспозиции и объединяете их в одно HDR-изображение.

Декомпозиция: Метод Дюранда и Дорси использует двусторонний фильтр для декомпозиции изображения на два слоя: базовый слой, кодирующий крупномасштабные вариации (или низкочастотные компоненты), и детализирующий слой для мелкомасштабных вариаций (или высокочастотных компонентов). Двусторонний фильтр может сохранять края, поскольку он сглаживает изображение, сохраняя при этом четкие границы, что делает его сглаживающим фильтром, сохраняющим края.

Сопоставление тонов: Сопоставление тонов применяется только к базовому слою. Здесь часто используется логарифмическое сжатие, уменьшающее динамический диапазон базового слоя, не затрагивая слой деталей. Цель состоит в том, чтобы сжать крупномасштабные вариации освещения, сохранив при этом детали меньшего масштаба.

Перекомпоновка: слой деталей и базовый слой с отображением тонов затем повторно объединяются для создания конечного изображения с

отображением тонов, пригодного для отображения на стандартных устройствах.

Метод Дюранда и Дорси популярен благодаря своей способности создавать изображения, которые сохраняют много деталей и контрастности, но при этом уменьшают общий динамический диапазон. Также стоит упомянуть, что, как и все методы отображения тонов, это приблизительное изображение, и результирующее изображение все равно может отличаться от того, как человеческий глаз воспринимал бы реальную сцену с высоким динамическим диапазоном.

Алгоритмы автофокусировки

Алгоритмы автофокусировки являются неотъемлемой частью многих современных систем обработки изображений, включая цифровые фотоаппараты и даже камеры в смартфонах. Они работают путем автоматической настройки объектива камеры для получения фокуса на объекте, устраняя необходимость в ручной настройке фокуса. В отрасли используется несколько методов автофокусировки:

Автофокусировка с определением контраста (CDAF): Это метод пассивной автофокусировки, который работает путем измерения контраста внутри сцены. Камера настраивает свой объектив и выбирает настройку фокусировки, при которой изображение получается максимально контрастным, что соответствует наиболее точной фокусировке. Этот метод часто используется в беззеркальных камерах и компактных камерах наведения и съемки. Однако в условиях низкой освещенности это может быть медленным процессом и вызывать трудности.

Автофокусировка с определением фазы (PDAF): Это более быстрый и эффективный метод автофокусировки, часто используемый в цифровых

зеркальных камерах. Он работает путем разделения входящего света на два изображения и последующего их сравнения. Объектив настраивается до тех пор, пока два изображения не выровняются идеально, что указывает на то, что изображение находится в фокусе. Этот метод особенно хорошо работает для отслеживания движущихся объектов.

Лазерная автофокусировка: Этот метод работает путем направления лазерного луча на объект съемки и измерения времени, необходимого для отражения света. Затем камера может рассчитать расстояние до объекта съемки и соответствующим образом настроить объектив. Этот метод быстр и надежен, но его эффективность может снижаться при увеличении расстояния.

Отображение глубины: Некоторые камеры, особенно в смартфонах, используют два объектива для создания карты глубины сцены, которую можно использовать для определения наилучшей фокусировки. Концепция аналогична тому, как люди используют два глаза для восприятия глубины.

Гибридный автофокус: Это комбинация методов автофокусировки с определением фазы и контрастности. Он использует определение фазы, чтобы быстро приблизиться к идеальной фокусировке, а затем использует определение контраста для точной настройки фокуса, получая лучшее из обоих миров.

Двухпиксельная автофокусировка: Это вариант автофокусировки с определением фазы, при котором каждый пиксель на датчике изображения используется для определения фокуса, что приводит к более быстрой и точной автофокусировке.

Алгоритмы автоматического баланса белого (AWB)

Предположение о сером мире: Этот алгоритм предполагает, что среднее значение всех цветов в сцене является нейтрально-серым, и соответствующим образом настраивает баланс белого.

Идеальный отражатель: предполагается, что где-то на изображении есть белый объект, и настраивается баланс таким образом, чтобы самый яркий объект отображался белым.

Постоянство цвета: Оно включает в себя такие алгоритмы, как отображение гаммы и байесовские методы, которые направлены на имитацию способности зрительной системы человека распознавать цвет объекта независимо от источника света.

Нейронные сети: В последние годы нейронные сети стали использоваться для процесса автоматического баланса белого. Это включает в себя обучение алгоритма на наборе данных изображений с правильным балансом белого, чтобы он мог вносить коррективы в новые изображения.

Оценка цветовой температуры: Этот алгоритм оценивает температуру источника света и соответствующим образом корректирует цвета на изображении. Это может быть особенно эффективно при наличии единственного доминирующего источника света.

Реализация метода Дебвека

Рассмотрим подробнее HDR, а в частности метод Дебвека с применением отображения тонов с помощью методов Драго, Рейнхарда и Мантиюка,

Методы Драго (Drago), Рейнхарда (Reinhard) и Мантиюка (Mantiuk) являются алгоритмами отображения тонов (Tonemapping) и используются для преобразования линейного изображения высокой динамической области (HDR) в изображение с низкой динамической областью (LDR), которое может быть правильно отображено на обычном экране или устройстве вывода. Однако каждый из этих методов имеет свои особенности и производит различные эффекты.

Метод Драго (Drago):

Метод Драго позволяет получить натуральное и фотореалистичное отображение тонов. Он основан на глобальной операции масштабирования и адаптации контрастности. Метод обладает высокой способностью сохранять детали в областях высокой яркости и теней. Параметры метода Драго, такие как `gamma`, `saturation` и `bias`, могут быть настроены для контроля эффекта отображения тонов.

Метод Рейнхарда (Reinhard):

Метод Рейнхарда также обеспечивает натуральное и высококачественное отображение тонов. Он основан на локальной операции масштабирования и адаптации контрастности. Метод позволяет контролировать глобальную интенсивность, освещение, а также адаптацию цвета. Параметры метода Рейнхарда, такие как `gamma`, `intensity`, `light_adapt` и `color_adapt`, могут быть настроены для получения желаемого визуального эффекта.

Метод Мантиюка (Mantiuk):

Метод Мантиюка также предоставляет высокое качество отображения тонов. Он основан на комбинации глобальной и локальной операций масштабирования контрастности. Метод обладает способностью сохранять детали в областях высокой яркости и обеспечивать широкий динамический диапазон. Параметры метода Мантиюка, такие как `gamma`, `scale` и `saturation`, могут быть настроены для контроля эффекта отображения тонов.

Реализация метода Дебвека для HDR-обработки в Python может быть выполнена с помощью библиотеки OpenCV. Этот метод объединяет несколько изображений с разной экспозицией в одно HDR-изображение, которое затем может быть отображено или сохранено. Вот основные шаги:

Считывание изображений: В начале мы читаем несколько изображений с разной экспозицией. Это можно сделать с помощью метода `cv2.imread()`. Необходимо убедиться, что все изображения имеют одинаковый размер.

```
def read_images_and_times():
    # Путь к папке с изображениями
    images_dir = 'Bali-Tukad-Cepung-Waterfall'

    # Получаем список всех .jpg файлов в папке
    filenames = [os.path.join(images_dir, f) for f in os.listdir(images_dir)
    if
        os.path.isfile(os.path.join(images_dir, f)) and
        f.endswith('.jpg')]

    # Упорядочиваем файлы так, чтобы _under был первым, _over был последним
    filenames.sort(key=lambda x: ('_over' in x, '_under' not in x, x))

    # Соответствующее время экспозиции в секундах от наименьшего к
    # наибольшему
    times = np.array([1/6.0, 1.3, 5.0], dtype=np.float32)

    images = []
    for filename in filenames:
        im = cv2.cvtColor(cv2.imread(filename), cv2.COLOR_BGR2RGB)
        if im is None:
            print(f"Error: {filename} is not a valid image file.")
            continue
        images.append(im)

    if not images:
        print("No valid images to process. Exiting.")
        exit()

    return images, times
```

В данном случае используется функция `read_images_and_times` в которой происходит считывание файлов из каталога `images_dir`, для получения нужных изображений в правильном порядке, а также организована проверка на существование изображений. В результате выполнения функции возвращается список изображений и времени экспозиции.

Создание объекта MergeDebevec: Затем мы создаем объект `MergeDebevec` с помощью функции `cv2.createMergeDebevec()`. Этот объект будет использоваться для слияния изображений с разной экспозицией.

Объединение изображений: Объект MergeDebevec затем используется для объединения изображений в одно HDR-изображение с помощью метода process(). В этом методе передаются два аргумента: список изображений и список времен экспозиции.

Тоновое отображение: После создания HDR-изображения его необходимо перевести в LDR (Low Dynamic Range) формат для отображения или сохранения. Для этого мы создаем объект Tonemap с помощью функции cv2.createTonemap(), а затем применяем его к HDR-изображению с помощью метода process(). Объект Tonemap создается с использованием различных методов: Драго, Рейнхарда и Мانتюка

```
def process_images(images, times):
    # Алигнирование
    alignMTB = cv2.createAlignMTB()
    alignMTB.process(images, images)

    # Получаем функцию отклика камеры (CRF)
    calibrateDebevec = cv2.createCalibrateDebevec()
    responseDebevec = calibrateDebevec.process(images, times)

    # Объединение изображения в линейное изображение HDR
    mergeDebevec = cv2.createMergeDebevec()
    hdrDebevec = mergeDebevec.process(images, times, responseDebevec)

    # Отображение тонов с использованием метода Драго для получения 24-
    битного цветного изображения

    gamma = 0.6
    saturation = 0.4
    bias = 0.9

    tonemapDrago = cv2.createTonemapDrago(gamma, saturation, bias)
    ldrDrago = tonemapDrago.process(hdrDebevec)
    ldrDrago = 3 * ldrDrago

    # Отображение тонов с использованием метода Рейнхарда для получения 24-
    битного цветного изображения

    gamma = 1.5
    intensity = 0.0
    light_adapt = 0.0
    color_adapt = 0.0

    tonemapReinhard = cv2.createTonemapReinhard(gamma, intensity,
    light_adapt, color_adapt)
    ldrReinhard = tonemapReinhard.process(hdrDebevec)
```

Отображение тонов с использованием метода Мантиюка для получения 24-битного цветного изображения

```
gamma = 1.3
scale = 0.85
saturation = 0.9

tonemapMantiuk = cv2.createTonemapMantiuk(gamma, scale, saturation)
ldrMantiuk = tonemapMantiuk.process(hdrDebevec)
ldrMantiuk = 3 * ldrMantiuk

return hdrDebevec, ldrDrago, ldrReinhard, ldrMantiuk
```

Сохранение изображения: Наконец, полученное изображение можно сохранить с помощью метода `cv2.imwrite()`. Если это HDR-изображение, оно должно быть сохранено в формате `.hdr`.

```
def save_images(hdrDebevec, ldrDrago, ldrReinhard, ldrMantiuk):
    # Указываем путь к папке, в которую хотим сохранить результаты
    results_dir = 'results'

    # Проверяем, существует ли папка, если нет, то создаем
    if not os.path.exists(results_dir):
        os.makedirs(results_dir)

    # Сохраняем результаты
    cv2.imwrite(os.path.join(results_dir, "hdrDebevec.hdr"), hdrDebevec)
    cv2.imwrite(os.path.join(results_dir, "ldr-Drago.jpg"),
cv2.cvtColor(ldrDrago * 255, cv2.COLOR_RGB2BGR))
    cv2.imwrite(os.path.join(results_dir, "ldr-Reinhard.jpg"),
cv2.cvtColor(ldrReinhard * 255, cv2.COLOR_RGB2BGR))
    cv2.imwrite(os.path.join(results_dir, "ldr-Mantiuk.jpg"),
cv2.cvtColor(ldrMantiuk * 255, cv2.COLOR_RGB2BGR))
```

Важно отметить, что метод Дебвека является одним из нескольких доступных методов для HDR-обработки в OpenCV. Другие включают методы Робертсона и Мерца. Каждый из этих методов имеет свои преимущества и недостатки, и выбор метода зависит от конкретных требований задачи.

Ниже представлен полный код для реализации метода:

```
import cv2
import numpy as np
import os
import time

def read_images_and_times():
    # Путь к папке с изображениями
    images_dir = 'Bali-Tukad-Cepung-Waterfall'

    # Получаем список всех .jpg файлов в папке
    filenames = [os.path.join(images_dir, f) for f in os.listdir(images_dir)
    if
        os.path.isfile(os.path.join(images_dir, f)) and
        f.endswith('.jpg')]

    # Упорядочиваем файлы так, чтобы _under был первым, _over был последним
    filenames.sort(key=lambda x: ('_over' in x, '_under' not in x, x))

    # Соответствующее время экспозиции в секундах от наименьшего к
    # наибольшему
    times = np.array([1/6.0, 1.3, 5.0], dtype=np.float32)

    images = []
    for filename in filenames:
        im = cv2.cvtColor(cv2.imread(filename), cv2.COLOR_BGR2RGB)
        if im is None:
            print(f"Error: {filename} is not a valid image file.")
            continue
        images.append(im)

    if not images:
        print("No valid images to process. Exiting.")
        exit()

    return images, times

def process_images(images, times):
    # АЛИГНИРОВАНИЕ
    alignMTB = cv2.createAlignMTB()
    alignMTB.process(images, images)

    # Получаем функцию отклика камеры (CRF)
    calibrateDebevec = cv2.createCalibrateDebevec()
    responseDebevec = calibrateDebevec.process(images, times)

    # Объединение изображения в линейное изображение HDR
    mergeDebevec = cv2.createMergeDebevec()
    hdrDebevec = mergeDebevec.process(images, times, responseDebevec)

    # Отображение тонов с использованием метода Драго для получения 24-
    # битного цветного изображения

    gamma = 0.6
    saturation = 0.4
    bias = 0.9

    tonemapDrago = cv2.createTonemapDrago(gamma, saturation, bias)
```

```

ldrDrago = tonemapDrago.process(hdrDebevec)
ldrDrago = 3 * ldrDrago

# Отображение тонов с использованием метода Рейнхарда для получения 24-битного цветного изображения

gamma = 1.5
intensity = 0.0
light_adapt = 0.0
color_adapt = 0.0

tonemapReinhard = cv2.createTonemapReinhard(gamma, intensity,
light_adapt, color_adapt)
ldrReinhard = tonemapReinhard.process(hdrDebevec)

# Отображение тонов с использованием метода Мантиюка для получения 24-битного цветного изображения

gamma = 1.3
scale = 0.85
saturation = 0.9

tonemapMantiuk = cv2.createTonemapMantiuk(gamma, scale, saturation)
ldrMantiuk = tonemapMantiuk.process(hdrDebevec)
ldrMantiuk = 3 * ldrMantiuk

return hdrDebevec, ldrDrago, ldrReinhard, ldrMantiuk

def save_images(hdrDebevec, ldrDrago, ldrReinhard, ldrMantiuk):
    # Указываем путь к папке, в которую хотим сохранить результаты
    results_dir = 'results'

    # Проверяем, существует ли папка, если нет, то создаем
    if not os.path.exists(results_dir):
        os.makedirs(results_dir)

    # Сохраняем результаты
    cv2.imwrite(os.path.join(results_dir, "hdrDebevec.hdr"), hdrDebevec)
    cv2.imwrite(os.path.join(results_dir, "ldr-Drago.jpg"),
cv2.cvtColor(ldrDrago * 255, cv2.COLOR_RGB2BGR))
    cv2.imwrite(os.path.join(results_dir, "ldr-Reinhard.jpg"),
cv2.cvtColor(ldrReinhard * 255, cv2.COLOR_RGB2BGR))
    cv2.imwrite(os.path.join(results_dir, "ldr-Mantiuk.jpg"),
cv2.cvtColor(ldrMantiuk * 255, cv2.COLOR_RGB2BGR))

def main():
    images, times = read_images_and_times()
    hdrDebevec, ldrDrago, ldrReinhard, ldrMantiuk = process_images(images,
times)
    save_images(hdrDebevec, ldrDrago, ldrReinhard, ldrMantiuk)

if __name__ == "__main__":
    start_time = time.perf_counter()
    main()
    end_time = time.perf_counter()
    execution_time = end_time - start_time
    print(f"Программа выполнялась {execution_time} секунд")

```

Посмотрим на примере работу нашей реализации:

Расположим в подкаталоге **Bali-Tukad-Cerung-Waterfall** внутри нашего проекта три изображения:

Waterfall_under.jpg с временем выдержки 1/6



Waterfall.jpg с временем выдержки 1.3

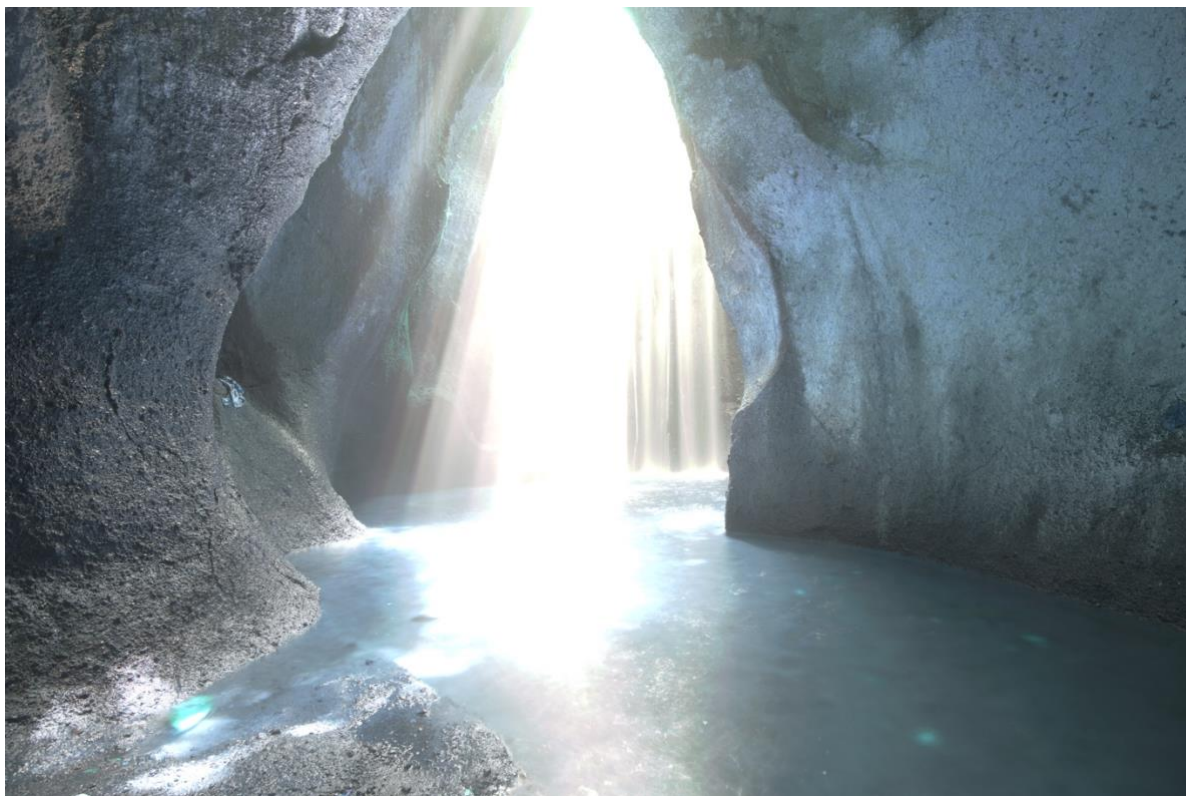


Waterfall_over.jpg с временем выдержки 5.0



В результате выполнения программы получаем четыре изображения: HDR изображение без Tone mapping'a и три изображения, соответствующие LDR изображениям для каждого из методов:

HDR изображение без Tone mapping'a: **hdrDebvec.hdr**



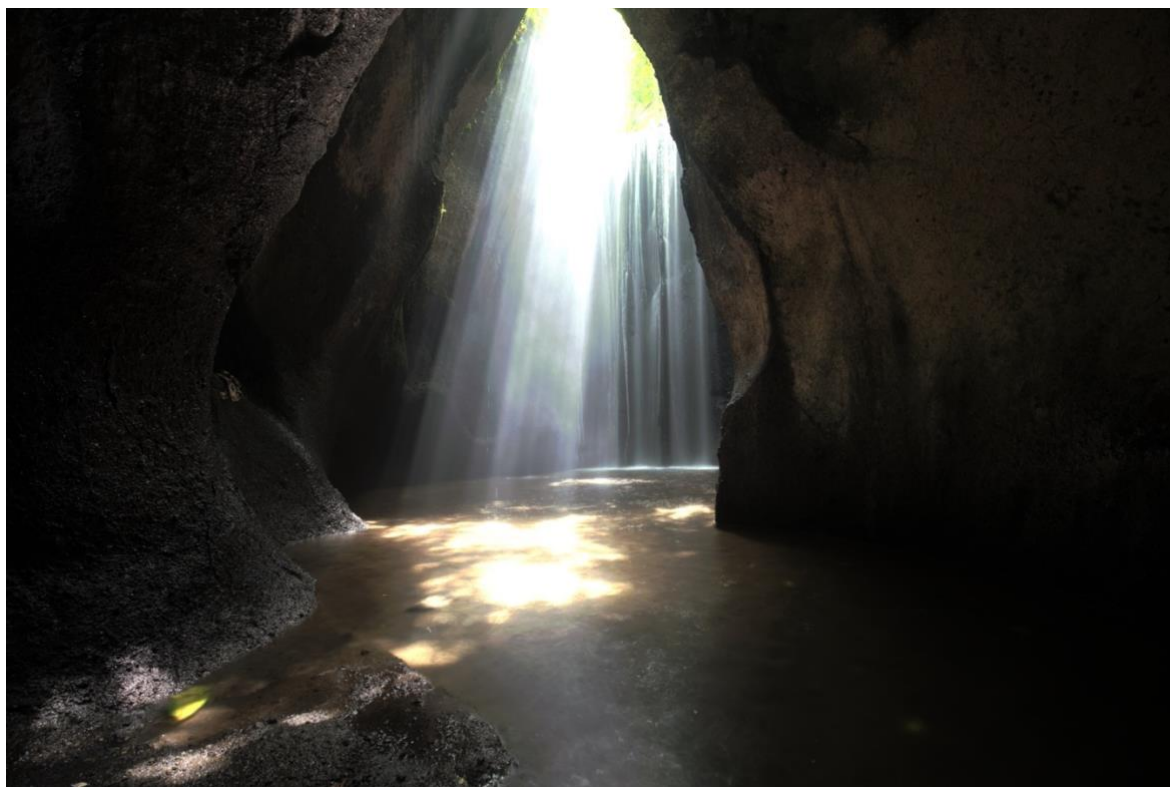
LDR изображение с Tone mapping'ом по методу Мантиюка: **ldr-Mantiuk.jpg**



LDR изображение с Tone mapping'ом по методу Рейнхарда: **ldr-Reinhard.jpg**:



LDR изображение с Tone mapping'ом по методу Драго: **ldr-Draco.jpg**:



Размер изначальных изображений составлял **6720 x 4480** и для выполнения программы потребовалось 30.50065977700433 секунд.

Параметры для каждого из методов получены методом проб и ошибок:

*# Отображение тонов с использованием метода Драго для получения 24-битного
цветного изображения*

gamma = 0.6

saturation = 0.4

bias = 0.9

*# Отображение тонов с использованием метода Рейнхарда для получения 24-битного
цветного изображения*

gamma = 1.5

intensity = 0.0

light_adapt = 0.0

color_adapt = 0.0

*# Отображение тонов с использованием метода Мантюка для получения 24-битного
цветного изображения*

gamma = 1.3

scale = 0.85

saturation = 0.9

Рассмотрим другой пример работы нашей реализации:

Время экспозиции: 1/350



Время экспозиции: 1/180



Время экспозиции: 1/90



После выполнения программы получаем:

LDR изображение с Tone mapping'ом по методу Мантиюка: **ldr-Mantiuk.jpg**



LDR изображение с Tone mapping'ом по методу Рейнхарда: **ldr-Reinhard.jpg**:



LDR изображение с Tone mapping'ом по методу Драго: **ldr-Drago.jpg**:



Параметры для каждого из методов получены методом проб и ошибок:

```
# Отображение тонов с использованием метода Драго для получения 24-битного
цветного изображения
gamma = 0.38
saturation = 0.5
bias = 0.6

# Отображение тонов с использованием метода Рейнхарда для получения 24-битного
цветного изображения
gamma = 0.7
intensity = 1.5
light_adapt = -0.2
color_adapt = 1.0

# Отображение тонов с использованием метода Мантюка для получения 24-битного
цветного изображения
gamma = 0.7
scale = 0.95
saturation = 0.9
```

Размер изначальных изображений составлял **1200 x 740** и для выполнения программы потребовалось 0.7207992710027611 секунд.

Отсюда можно сделать вывод, что **при повышении размера изображения, увеличивается время обработки изображения программой.**

Выводы

Время обработки изображения зависит от размеров конкретных изображений.

Долгое время обработки кадра связано с тем, что функции внутри методов построения карты тонов не оптимизированы под конкретную систему. Чтобы оптимизировать код, необходимо использовать C++.