

Реализация системы обработки изображений большого динамического диапазона как части комплекса оптического моделирования

Е.Д. Бирюков¹, Б.Х. Барладян¹, Е.Ю. Денисов¹, А.Г. Волобой¹

¹ ИПМ им. М.В. Келдыша РАН, Миусская пл. 4, Москва, 125047, Россия

Аннотация

В данной статье описаны подходы к реализации программного модуля обработки изображений большого динамического диапазона как части комплекса оптического моделирования и фотореалистичной компьютерной графики. Комплексу оптического моделирования и расчета освещенности нужна собственная программа обработки изображений, поскольку возможностей существующих подобных программ, как правило, бывает недостаточно для задач анализа результатов моделирования. Это связано с тем, что при обработке и анализе результатов требуется специальная функциональность, которая чаще всего отсутствует в обычных программах обработки изображений. В статье представлена программа, разработанная нашим коллективом и содержащая функциональность, ориентированную на оптические задачи. В первую очередь, это функция сжатия динамического диапазона изображений для отображения на экране монитора. Кроме того, в число таких функций входят дополнительные визуальные эффекты для улучшения качества отображения, такие, как принудительное обесцвечивание ярких бликов на изображениях, а также фильтрация изображений для снижения шума, образующегося в результате использования стохастических методов. Для анализа результатов требуется передавать из системы моделирования в модуль анализа и обработки дополнительные данные помимо самих изображений. Описана многослойная структура данных в сохраняемых файлах изображений, являющихся результатами оптического моделирования.

Ключевые слова

Изображения большого динамического диапазона, сжатие динамического диапазона, фильтрация изображений, цветовая коррекция изображений, структура данных графических форматов.

Implementation of High Dynamic Range Image Processing Module as a Part of Lighting Simulation Software System

E.D. Birukov¹, B.Kh. Barladian¹, E.Yu. Denisov¹, A.G. Voloboy¹

¹ Keldysh Institute of Applied Mathematics, Miusskay sq. 4, Moscow, 125047, Russia

Annotation

This article describes approaches to implementation of software module for processing of high dynamic range images, as a part of lighting simulation and photorealistic computer graphics software. Lighting simulation software requires its own image processing application because functionality of existing ones usually is not enough for lighting simulation analysis purposes. Processing and analysis of lighting simulation results usually requires the certain functionality that is absent in many image processing programs. The article describes an application developed by us which contains such functionality related to lighting simulation. At first, it is a dynamic range compression function for displaying images on

ГрафиКон 2022: 32-я Международная конференция по компьютерной графике и машинному зрению, 19-22 сентября 2022 г., Рязанский государственный радиотехнический университет им. В.Ф. Уткина, Рязань, Россия

EMAIL: birukov@gin.keldysh.ru (Е.Д. Бирюков); obb@gin.keldysh.ru (Б.Х. Барладян); eed@gin.keldysh.ru (Е.Ю. Денисов); voloboy@gin.keldysh.ru (А.Г. Волобой)

ORCID: 0000-0003-4297-6813 (Е.Д. Бирюков); 0000-0002-2391-2067 (Б.Х. Барладян); 0000-0002-0614-9100 (Е.Ю. Денисов); 0000-0003-1252-8294 (А.Г. Волобой)



© 2022 Copyright for this paper by its authors.
Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

computer monitors. Besides this, our application includes additional functionality for improving image quality, such as forced desaturation of brightest highlights on the images, and also image filtering for decreasing noise caused by stochastic methods used in simulation. For simulation result analysis it is required to transfer additional data (besides image itself) to the processing module. Such layer-based data structure of high dynamic range image files generated by lighting simulations software is described.

Keywords

High dynamic range images, dynamic range compression, image filtering, image color correction, graphical file format data structure.

1. Введение

Программные комплексы фотореалистичной трехмерной графики и оптического моделирования часто включают в себя встроенное средство обработки двумерных изображений, являющихся результатами моделирования. Существует много широко-известных программ обработки изображений (например, Adobe PhotoShop[1], ACDSee[2], Corel PaintShop Pro[3], а также свободные программы, такие, как GIMP). Однако, они, как правило, либо не содержат необходимый функционал, либо настолько избыточны, что пользоваться ими становится неудобно.

Какой же функционал необходим исследователю или дизайнеру в области оптики? Во-первых, для изображений, полученных в результате оптического моделирования, часто требуется использовать собственный специальный формат данных, позволяющий записывать не просто значения цветов пикселей, а значения физических величин (яркости, освещенности и т.д.), которые необходимо преобразовывать в пригодный для отображения на экране монитора формат. При этом необходимо также сжатие динамического диапазона изображений, т.к. динамический диапазон обычного монитора составляет всего два порядка. Во-вторых, в некоторых случаях требуются дополнительные режимы отображения вспомогательных результатов моделирования, например, достигнутой точности или числа протрассированных лучей. Наиболее часто используемый режим в этом случае – это отображение в искусственных цветах (режим «тепловой карты»). Также иногда результат записывается в полярных координатах, или представляет собой сферическую панораму, и такие данные тоже необходимо преобразовать в пригодный для представления на экране вид. Наконец, в некоторых случаях бывает необходимо обеспечить дополнительное взаимодействие между программой просмотра изображений и основной программой оптического моделирования для обратной связи. Это необходимо, когда, например, на изображении необходимо выделить некоторую область, и координаты этой области передаются для последующих вычислений в основную программу.

Все эти потребности были реализованы в системе обработки изображений большого динамического диапазона, входящей в состав разработанного нами комплекса оптического моделирования и фотореалистичной трехмерной графики.

2. Анализ и сжатие динамического диапазона изображений

Первоначально система обработки изображений большого динамического диапазона, разработанная нами, представляла собой небольшую служебную программу. Ее единственной функциональностью являлось отображение изображений большого динамического диапазона в цветовые координаты монитора. Задача сжатия динамического диапазона чрезвычайно важна, поскольку изображения, полученные в результате моделирования распространения света, выражены в реальных физических величинах, и необходимо иметь встроенное средство быстрого их просмотра.

В нашей системе обработки изображений на данный момент реализованы следующие три варианта операторов сжатия динамического диапазона:

1. Нелинейный оператор сжатия динамического диапазона [4-5].
2. Простой линейный оператор сжатия, контролируемый единственным параметром уровня максимальной яркости.
3. Табулированный оператор сжатия, задаваемый таблицей значений, редактируемой пользователем.

Нелинейный оператор сжатия динамического диапазона основан на улучшенном операторе Тумблина-Русмеера [5-6] и дополнительной формуле, предложенной Рейнхардом и др. [5, 7-8],

которая сжимает главным образом большие яркости. Управление этим оператором осуществляется путем задания максимального значения величины яркости или освещенности, которое не будет обрезано, и адапционной яркости сцены.

Первоначально оператор сжатия определялся следующей формулой:

$$L_d = m(L_{wa}) * L_{da} * \left(\frac{L_w}{L_{wa}}\right)^{\left(\frac{\gamma_w}{\gamma_d}\right)} \quad (1)$$

где L_w – исходные значения физических яркостей пикселей, L_{da} – адапционная яркость экрана (типичное значение порядка 10-30 cd/m²), L_{wa} – адапционная яркость сцены, $m(L_{wa})$ – коэффициент, зависящий от адапционной яркости сцены, предотвращающий anomalously серые ночные изображения, а γ_w и γ_d определяются, используя формулу Стивенса для чувствительности контраста. В общем случае, в результате использования улучшенного оператора сжатия Тумблина-Русмеера, яркость некоторых пикселей монитора, соответствующих наиболее яркой части сцены, может превышать максимальную яркость монитора и соответствующая часть изображения будет перенасыщена. Для преодоления этой проблемы наш оператор использует формулу, сжимающую большие яркости, предложенную Рейнхардом и др.:

$$L_{df}(x, y) = \frac{L_d(x, y) \left(1 + \frac{L_d(x, y)}{L_{white}^2}\right)}{1 + L_d(x, y)} \quad (2)$$

где L_d – яркость пикселя на экране, полученная в формуле (1), а L_{white} – минимальная яркость, отображаемая в чисто белый цвет.

Для изображений с очень большим динамическим диапазоном яркостей уровень белого L_{white} может быть выбран практически как любая достаточно большая величина, обычно близкая к максимальной яркости сцены. Линейный оператор сжатия управляется единственным параметром – максимальным значением величины яркости или освещенности. Этот параметр может задаваться как непосредственно, так и через набор фотографических параметров, таких как время экспозиции, относительное отверстие (диафрагменное число) объектива и чувствительность камеры.

Органы управления операторами сжатия показаны на рисунке 1.

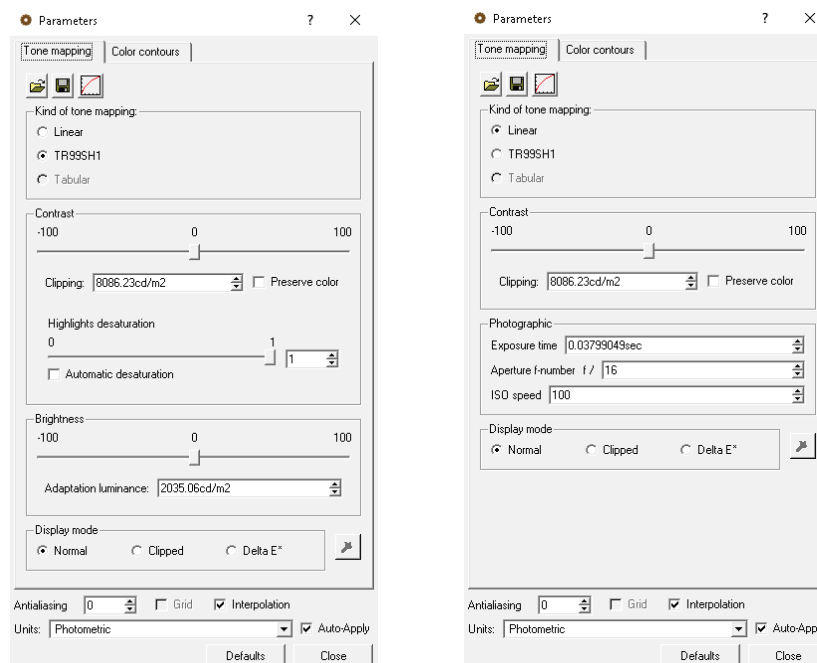


Рисунок 1 – Органы управления операторами сжатия динамического диапазона (слева – нелинейный оператор, справа – линейный)

Для визуального анализа изображения с большим динамическим диапазоном в разработанном модуле предусмотрен режим отображения в искусственных цветах. Этот режим предназначен для представления значений физических величин в виде, удобном для восприятия человеком. Интерфейс управляющих параметров в этом случае позволяет контролировать как цвета визуализации, так и значения яркостей, освещенностей или точности вычислений, соответствующих этим цветам. Пример отображения в режиме искусственных цветов с диалоговым окном управления параметрами этого режима показан на рисунке 2.

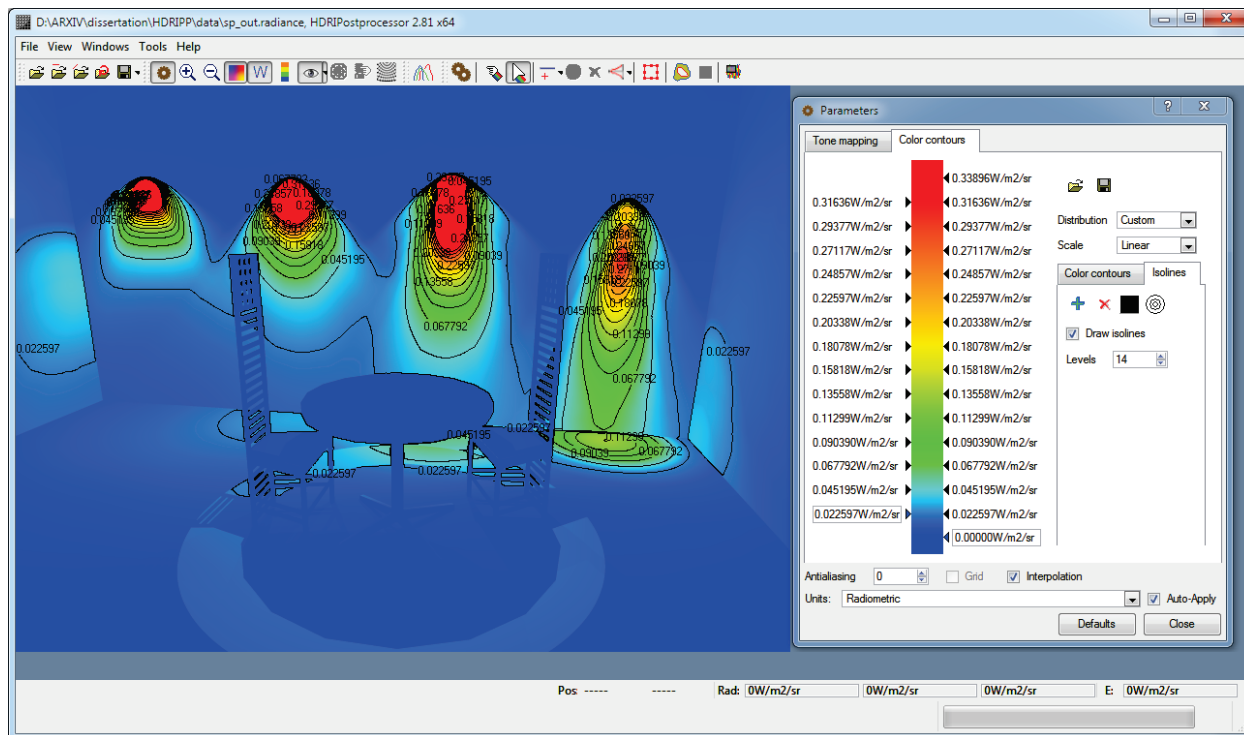


Рисунок 2 – Отображение физических величин в искусственных цветах

3. Послойная структура данных изображения

Первоначально файлы, предназначенные для загрузки в программу обработки изображений, содержали только значения яркости или освещенности для каждого пикселя. Однако, впоследствии возникла необходимость записи дополнительных данных. Для этого была предоставлена возможность добавления в файлы новых таблиц с данными. Эти таблицы также называются слоями. Большинство слоев, аналогично самому изображению, содержит данные, соответствующие пикселям изображения. Но при этом есть возможность создавать слои, не привязанные к координатной сетке изображения. Например, для изображений, сохраненных в цветовой модели с распределением по длинам волн, в качестве отдельного слоя записывается массив длин волн, для которых было сосчитано это изображение. Слои обозначаются именами, которые фиксируются для определенных данных, и при загрузке в программу эти данные ищутся и считываются по таким именам. Сейчас основными слоями, которые записываются в файл помимо самого изображения, являются следующие: изображение сетки; матрица величин ошибки вычислений; матрица значений количества попаданий лучей в каждую ячейку; координаты точки в трехмерном пространстве, соответствующие каждому пикселю.

Исторически сложилось, что некоторые слои были включены непосредственно в «основной» файл с изображением, в то время как другие слои записываются в отдельные файлы, которые имеют имя, совпадающее с именем основного файла, и отличающиеся только расширением. В частности, в отдельные файлы вынесены слой со значением координат точек в трехмерном пространстве, соответствующих пикселям, и слой с индексами объектов в исходной сцене, которые соответствуют данным точкам. Кроме того, если в рамках одного моделирования производится вычисление нескольких типов величин (яркости, освещенности и т.д.) для одной и той же камеры или измерительного прибора, то эти данные также записываются в отдельные файлы.

3.1. Слой значений ошибок вычислений

Слой со значениями оценки ошибки вычислений, соответствующих пикселям, и слой со значениями количества попаданий лучей в каждый пиксель были добавлены для оптического моделирования методом стохастической трассировки лучей с использованием виртуальных измерительных приборов.

Ошибки вычислений отображаются в искусственных цветах. Для их отображения предназначен отдельный режим, в который можно переключиться в основном диалоге параметров. При этом доступны те же элементы управления, что и для отображения самого изображения в режиме искусственных цветов, как это видно на рисунке 3.

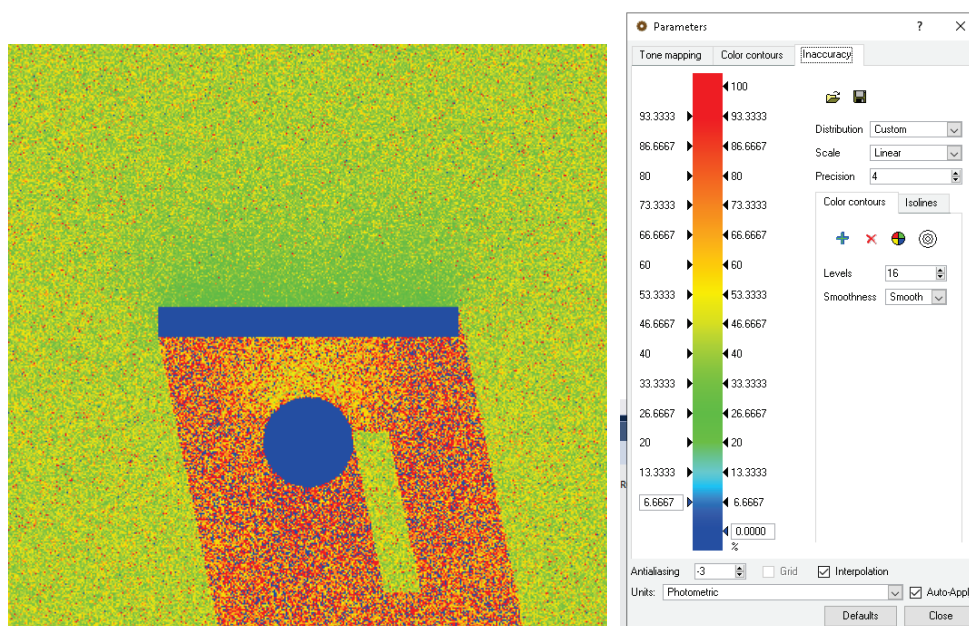


Рисунок 3 – Изображение и параметры просмотра ошибок вычислений

Этот режим позволяет обнаружить участки сцены, где особенности геометрии не позволяют добиться требуемой точности вычислений, и исправить их. Также есть возможность отобразить слой со значениями количества лучей, попавших в каждый пиксель. Этот слой также отображается в искусственных цветах аналогично предыдущему. Пример отображения данного слоя показан на рисунке 4. В результате анализа такого изображения можно сделать выводы об эффективности используемых алгоритмов трассировки лучей, что позволит при необходимости либо выбрать другой алгоритм, либо изменить параметры используемого.

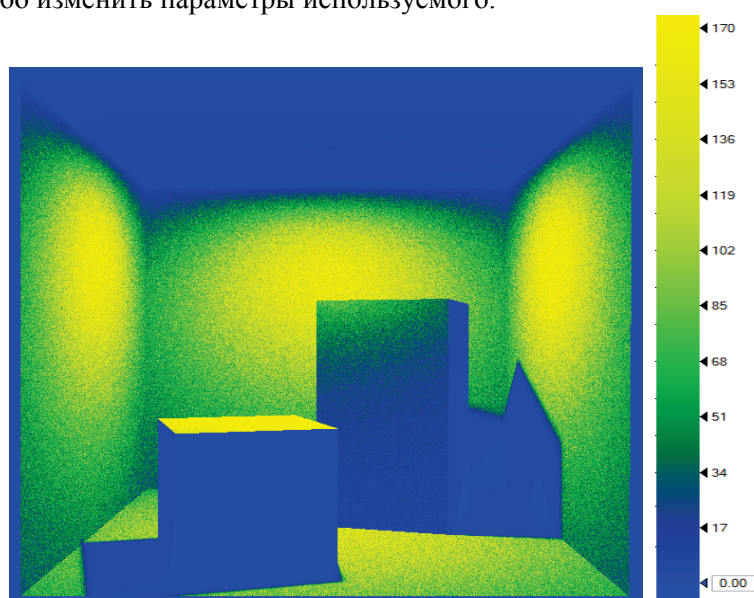


Рисунок 4 – Пример отображения в режиме просмотра количества лучей на пиксель

3.2. Слой 3D координат точек виртуальной сцены

Вместе с изображением можно записать также слой, содержащий трехмерные координаты ближайшего объекта виртуальной сцены, которая соответствовала данному пикселю. Одним из случаев, когда эти данные оказываются необходимыми, является измерение расстояния между точками. Данная функциональность позволяет выбрать два пикселя на изображении и вычислить расстояние между точками объектов сцены в трехмерном пространстве, соответствующими этим пикселям, используя записанные значения координат. Диалоговое окно для измерения расстояний показано на рисунке 5.

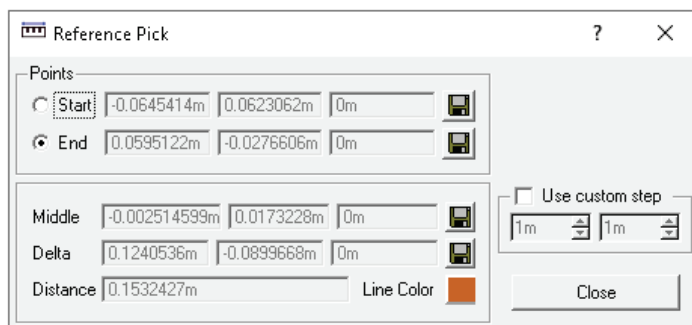


Рисунок 5 – Диалог измерения расстояния между точками исходной трехмерной сцены

Возможность измерения расстояния между двумя точками была предусмотрена в основной программе комплекса оптического моделирования, которая предоставляет возможность перемещения точки положения наблюдателя в пространстве. Но добавление такой возможности также в программу обработки изображений заметно упрощает работу, поскольку пользователю при анализе результатов нет необходимости каждый раз переключаться между двумя программами и самостоятельно вручную искать в сцене точку, соответствующую данному пикселю на изображении.

В некоторых случаях такой ручной поиск даже не представляется возможным, если при моделировании используется сверхширокоугольная камера (с углом обзора, близким к или даже превышающим 180 градусов), поскольку отображение сцены в рабочей области системы моделирования не позволяет использовать такие широкие углы обзора.

Также данная функциональность позволяет вычислять расстояние и на плоскости. Как правило, это необходимо при работе с изображениями, представляющими собой результаты вычислений на виртуальных измерительных приборах. Поскольку размеры сенсоров задаются реальными физическими размерами, а не просто количеством пикселей, то вычисляемое расстояние между точками в этом случае также выражается реальными физическими размерами. Предусмотрена возможность задавать пользовательский масштаб координатных осей на изображении, выраженный в процентах относительно пикселей исходного изображения, и вычислять расстояние между пикселями в этом масштабе. Такая возможность предусмотрена как для изначально двухмерных изображений, не содержащих сведений о трехмерных координатах, так и для изображений с такими данными. В этом случае происходит принудительное переключение режима отображения координат с трехмерного на двухмерный. Внешний вид окна отображения расстояний в двумерном режиме показан на рисунке 6.

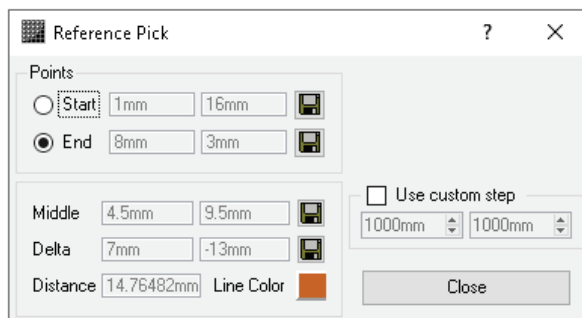


Рисунок 6 – Диалог измерения расстояния между точками в двумерном режиме

4. Дополнительные функции отображения

4.1. Применение фильтров

Если моделирование распространения света использует стохастический процесс, то итоговое изображение неизбежно будет зашумленным. Для удаления зашумленности предназначены алгоритмы фильтрации изображений. В нашей системе обработки изображений в настоящее время реализовано два простейших алгоритма фильтрации – фильтр по среднему значению и медианный фильтр. Фильтр по среднему значению вычисляет для каждого пикселя среднее арифметическое соседних пикселей и присваивает данному пикселю это значение. Использование такого фильтра приводит к размытию изображения. Медианный фильтр для данного пикселя собирает массив значений соседних пикселей, сортирует и в качестве цвета пикселя выбирает значение срединного элемента в массиве.

При реализации фильтров существует возможность выбора значения, которое необходимо использовать в качестве критерия. Это могут быть либо отдельные массивы значений каждого цветового канала, что приводит к очень низкой скорости работы, особенно для медианного фильтра [9-10], либо (для медианного фильтра) использование векторной медианы, что также достаточно затратно по времени [12], либо использование некоторого единственного физического значения, соответствующего пикселю (яркости или освещенности). В нашей системе применяется последний вариант.

Для управления фильтрацией изображений можно задавать следующие параметры: тип фильтра (по среднему значению либо медианный), размер окна в пикселях и количество проходов алгоритма (рисунок 7).

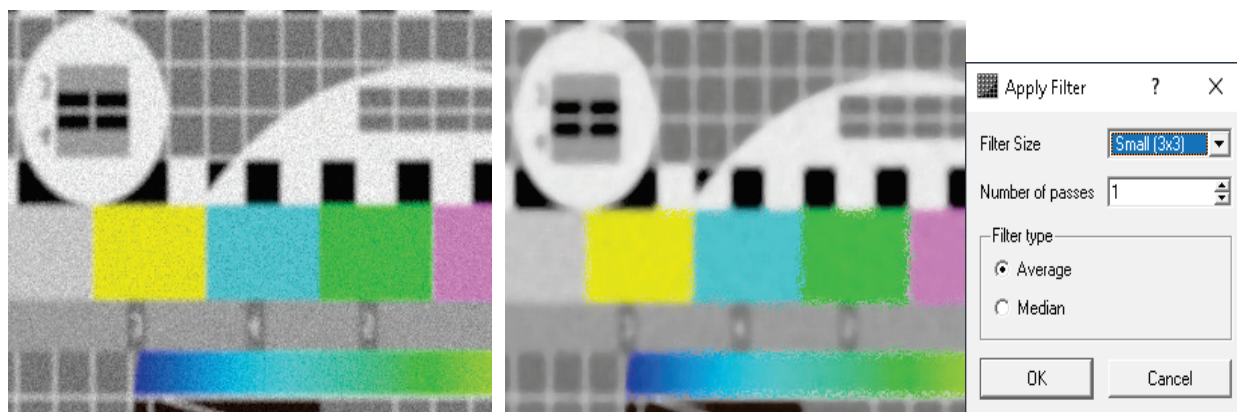


Рисунок 7 – Результат работы медианного фильтра (слева – исходное изображение, справа – после применения фильтра) и диалог с его параметрами

На компьютере с процессором AMD Ryzen 9 3900X 12-Core Processor и 32 ГБ оперативной памяти время работы медианного фильтра с окном 5x5 занимает примерно 1 секунду для изображения формата 1366x768 пикселей и около 2.5 минут для изображения размером 15000x15000 пикселей.

4.2. Система принудительного обесцвечивания бликов

В некоторых случаях сжатия динамического диапазона оказывается недостаточно для корректного отображения изображений с большим динамическим диапазоном, и оказывается целесообразным дополнительно обрабатывать наиболее яркие участки изображения (блики). Это может быть необходимо, чтобы симулировать особенности человеческого зрения или фотографической аппаратуры, когда наиболее яркие участки кажутся просто белыми, т.е. пересвеченные области «теряют» цвет в нашем восприятии [12-13]. Поэтому наша программа обработки изображений содержит метод обесцвечивания бликов.

Метод работает по следующему алгоритму:

1. Рассчитывается среднее значение цветовых каналов данного пикселя;
2. Находится разность между каждым конкретным цветовым каналом и средним значением;

3. Яркость данного пикселя делится на яркость самого яркого пикселя в изображении. Это значение будет индивидуальным коэффициентом снижения насыщенности данного пикселя;
4. Из каждого цветового канала вычитается его разница со средним значением канала, умноженная на индивидуальный коэффициент снижения насыщенности.

Таким образом, значения цветовых каналов станут ближе друг к другу, и цветовая насыщенность снизится.

Для более качественной автоматической настройки алгоритма была произведена его привязка к статистическим параметрам изображения. Яркие блики, которым требуется засветка ради нормального отображения основной части изображения, по определению не могут занимать слишком большую площадь относительно площади всего этого изображения. Поэтому окончательный вариант алгоритма предусматривал получения такого значения яркости, чтобы только 10% пикселей изображения имели бы большую яркость, дополнительное удвоение этой величины и использование полученного значения в качестве порогового для снижения насыщенности. Коэффициент снижения насыщенности считается равным нулю для этого порогового значения и всех пикселей меньшей яркости, и равным единице для максимального значения яркости на изображении. Для всех промежуточных значений применяется линейная интерполяция. Результат применения этого метода приведен на рисунке 8.



Рисунок 8 – Результат работы алгоритма обесцвечивания бликов (слева – исходное изображение, справа – после применения обесцвечивания)

Управление системой обесцвечивания бликов заключается в возможности переключения ручного и автоматического режимов. В ручном режиме пользователь должен сам задавать пороговое значение яркости, а в автоматическом режиме оно задается по вышеописанному алгоритму.

5. Заключение

Описанный модуль анализа и обработки изображений большого динамического диапазона в настоящее время используется в составе комплекса моделирования освещенности и генерации реалистичных изображений Lumiscept [14], разработанного нами. Он позволяет проводить сжатие динамического диапазона для просмотра изображений, построенных с помощью оптического моделирования, анализировать эти изображения с использованием дополнительных слоев данных, применять дополнительные эффекты для улучшения внешнего вида изображений и экспортировать полученные изображения со сжатым диапазоном в обычные растровые графические форматы.

Совершенствование модуля обработки изображений непрерывно продолжается. В настоящее время основными направлениями его развития являются усовершенствование операторов сжатия динамического диапазона вместе с дополнительными алгоритмами уменьшения насыщенности ярких бликов для улучшения отображения, а также поддержка новых файловых форматов большого динамического диапазона и обычной растровой графики для импорта и экспорта изображений.

6. СПИСОК ИСТОЧНИКОВ

- [1] Adobe Photoshop User Manual, URL: <https://helpx.adobe.com/photoshop/user-guide.html>.
- [2] ACDSee Photo Studio Software, URL: <https://www.acdsee.com/en/products/photo-studio-ultimate>.
- [3] Photo Editing Software – PaintShop Pro, URL: <https://www.paintshoppro.com/en/products/paintshop-pro/ultimate/#features>.
- [4] Barladian, B., "Robust Parameter Estimation for Tone Mapping Operator," Proc. 13th International Conference on Computer Graphics and Vision GraphiCon-2003, pp. 106-108, 5 -10 September 2003.
- [5] Б.Х. Барладян, А.Г. Волобой, В.А. Галактионов, Э.А. Копылов., "Эффективный оператор сжатия динамического диапазона яркостей," Программирование, № 5, с. 35-42, 2004.
- [6] Tumblin, J. 1999 Three Methods For Detail-Preserving Contrast Reduction For Displayed Images. PhD thesis, Georgia Institute of Technology, 1999.
- [7] Reinhard, E., Stark, M., Shirley, P., and Ferwerda, J., 2002. 'Photographic Tone Reproduction for Digital Images', ACM Transactions on Graphics, 21(3), July 2002 (Proceedings of SIGGRAPH 2002).
- [8] Reinhard, E., 2002, Parameter estimation for photographic tone reproduction. Journal of graphics tools, 7(1), pp. 45-52, 2002.
- [9] Rafael C. Gonzalez, Richard E. Woods, Steven L. Eddins. Digital Image Processing Using Matlab. Pearson Education, 2004.
- [10] Optimized Median Filter Implementation on FPGA Including Soft Processor. S.S.Tavse, P.M.Jadhav, M.R.Ingle. International Journal of Emerging Technology and Advanced Engineering, Volume 2, Issue 8, August 2012.
- [11] Yike Liu. Noise reduction by vector median filtering. Geophysics, Vol. 78, no. 3, May-June 2013.
- [12] Alessandro Rizzi, Carlo Gatta, Daniele Marini, 2003. A new algorithm for unsupervised global and local color correction. Pattern Recognition Letters, Volume 24, Issue 11, pages 1663-1677, July 2003.
- [13] D. Guo, Y. Cheng, S. Zhuo, T. Sim, 2010. Correcting over-exposure in photographs. Correcting over-exposure in photographs. IEEE Computer Society, 2010.
- [14] Lumicept – Hybrid Light Simulation Softwareю. URL: <http://www.integra.jp/en>.