

шем интеллектуальном развитии, но и на процессе обучения в дальнейшем. Например, для таких обучающихся не будет проблемой в будущем, во время обучения в УВО качественно выполнять курсовые или дипломные работы и проекты, а тем более и научные статьи в рамках научно-исследовательской работы студентов. Такие студенты с большей вероятностью решат продолжить обучение в углубленном высшем образовании, а в перспективе и научно-ориентированном образовании. Но даже, если студенты решат закончить только бакалавриат или специалитет и работать по профессии, то владение проектно-исследовательскими навыками сделает их более конкурентноспособными на трудовом рынке.

Список использованных источников

1. Румбешта, Е. А. Формирование проектно-исследовательской компетенции учащихся при обучении физике и оценка ее сформированности / Е. А. Румбешта, В. З. Мидуков // Вестник ТГПУ. – 2007. – № 10. – С. 103–108.

УДК 004.932

Эффективные алгоритмы сжатия и обработки изображений в компьютерной графике

Гала Е. В., студент

Кицун М. В., студент

Белорусский национальный технический университет

Минск, Республика Беларусь

Научный руководитель: ст. преподаватель Ражднова А. В.

Аннотация:

В современном мире изображения широко используются в различных областях, включая рекламу, медицину, игровую индустрию, научные исследования и др. Однако, изображения могут занимать большое пространство в памяти и передаваться по сети, что требует эффективных алгоритмов сжатия и обработки изображений.

Изображение является важным компонентом компьютерной графики, которая широко используется в различных областях, таких как медицина, дизайн, анимация и др. Изображения могут быть очень большими и требовательными к ресурсам, поэтому они часто сжимаются и обрабатываются с использованием различных алгоритмов. Целью этой работы является изучение и сравнение эффективных алгоритмов сжатия и обработки изображений в компьютерной графике.

Интересно понять, что несжатое полноцветное изображение размером 2000×1000 пикселей притянет на себя громадные 6 мегабайт. А вот если мы говорим о создании изображений на профессиональных камерах или сканерах с высоким разрешением, их размер может быть еще более огромен. В то время как мы наблюдаем постоянный рост емкости хранилищ, алгоритмы сжатия изображений все еще остаются высоко ценными и актуальными.

Алгоритмы делятся на два основных типа: сжатие изображений без потерь и сжатие изображений с потерями.

Сжатие изображений без потерь – это процесс уменьшения размера файла изображения без потери качества. При сжатии без потерь все пиксели остаются неизменными, исключая ненужную информацию, что позволяет уменьшить размер файла без изменения визуального восприятия изображения. Этот метод сжатия особенно полезен для веб-страниц, электронных документов и других мест, где важно сохранить качество изображения. Некоторые популярные алгоритмы сжатия без потерь включают в себя ZIP, PNG и GIF.

В ходе работы с алгоритмами, мы выяснили, что самыми эффективными алгоритмами сжатия без потерь являются: алгоритм статического кодирования и алгоритм Хаффмана.

Алгоритм статического кодирования.

Алгоритм статического кодирования является методом сжатия данных, основанным на исследовании статистических свойств входного потока. Он работает на этапе компиляции или перед исполнением программы и позволяет уменьшить размер кода без потери информации.

Алгоритм имеет два основных этапа: анализ и использование кодовой таблицы. На этапе анализа происходит изучение частоты встречаемости байтов или символов в исходном коде. На основе этих

результатов создается таблица кодов, в которой самым часто встречающимся символам присваиваются более короткие коды, а реже встречающимся символам – более длинные коды.

На этапе использования кодовой таблицы происходит замена оригинальных символов или байтов на соответствующие коды из таблицы. Таким образом, кодирование позволяет представить оригинальный код более компактно и сократить объем данных. Однако для декодирования необходимо иметь доступ к кодовой таблице.

Алгоритм Хаффмана.

Сначала алгоритм анализирует входные данные, чтобы определить вероятность появления каждого символа. Затем строится бинарное дерево Хаффмана, где каждый символ представлен как лист дерева, а вероятности появления символов определяют их положение в дереве.

Для составления кодового слова для каждого символа, начиная от корня дерева, присваивается '0' или '1' в зависимости от того, находится символ слева или справа от родительского узла. Путь от корневого узла до каждого символа образует его код Хаффмана.

Код Хаффмана для каждого символа – это минимальная последовательность битов, которая позволяет однозначно идентифицировать символ. После построения кодового дерева Хаффмана, данные сжимаются, заменяя каждый символ его кодом Хаффмана. Это позволяет сократить количество битов, необходимых для представления данных. При распаковке сжатых данных используется тот же код Хаффмана, чтобы преобразовать битовую последовательность обратно в исходные символы.

Используя алгоритм Хаффмана, мы можем создать коды для каждого символа в заданном алфавите, учитывая их частоту появления.

Сжатие изображений с потерями – это метод сжатия данных, который используется для уменьшения размера изображений путем удаления некоторой информации, незаметной для человеческого глаза. При сжатии изображения с потерями, исходный файл проходит через алгоритм, который определяет и удаляет детали, содержимое которых не существенно для восприятия изображения визуально. Например, это могут быть высокочастотные детали, шумы или незначительные изменения цвета.

Таким образом, сжатие с потерями приводит к некоторой потере качества изображения, однако оно остается приемлемым для боль-

шинства приложений, поскольку исходное качество все равно сохраняется на достаточно высоком уровне. Однако при повторном сохранении изображения с потерями может привести к дополнительной потере качества, поэтому важно быть осторожным при многократном редактировании и сохранении таких изображений.

Кодирование методом Хаффмана.

Кодирование с использованием алгоритма Хаффмана, описанного выше, основывается на использовании заранее составленной таблицы. При декодировании производится обратное преобразование. Этот алгоритм отличается высокой степенью сжатия (до 5 раз и более), низкой сложностью и отсутствием патентных проблем. Однако он имеет недостаток – возникновение артефактов, которые могут быть заметны для человеческого глаза.

Фрактальное сжатие.

Процесс фрактального сжатия основан на использовании математических моделей, известных как фракталы.

Вначале, исходное изображение разбивается на маленькие блоки данных, называемые фрагментами. Затем выбирается фрагмент, который наиболее соответствует заданному качеству изображения.

Далее, выбранный фрагмент используется для создания математической формулы, называемой кодеком, которая описывает поведение фрактала. Эта формула представляет собой инструкции для повторения структур и деталей изображения.

Затем, с помощью кодека, выбранный фрагмент повторяется и масштабируется для воссоздания исходного изображения. Каждый повтор фрагмента также может быть дальше разделен на более мелкие фрагменты, чтобы добавить больше деталей.

Процесс повторяется до тех пор, пока не будет достигнуто требуемое качество изображения или требуемый уровень сжатия.

Фрактальное сжатие может обеспечить более высокую степень сжатия без потери качества изображения по сравнению с другими методами сжатия, такими как JPEG. Однако, процесс сжатия и восстановления фракталов требует большого количества вычислительных ресурсов, поэтому он может быть более медленным и сложным.

Точность полученного изображения зависит от точности выполнения аффинных преобразований.

Алгоритмы фрактального сжатия сложны в реализации из-за использования целочисленной арифметики и специальных методов, направленных на уменьшение ошибок округления.

Одной из особенностей фрактального сжатия является ярко выраженная асимметрия между процессом сжатия и восстановления изображения, где сжатие требует значительно больше вычислений.

Использование эффективных алгоритмов сжатия и обработки изображений в компьютерной графике имеет большое значение для улучшения производительности систем, снижения затрат на хранение и передачу изображений, а также для обеспечения высокого качества визуализации. В результате проведенного исследования были выявлены преимущества и недостатки часто используемых алгоритмов сжатия и обработки изображений, что может быть основой для дальнейшего развития и совершенствования алгоритмов.

Список использованных источников

1. Д-р Навин Гарг, ПТ – D (Лекция – 19 Сжатие данных) [Электронный ресурс]. – Режим доступа: Lecture - 19 Data Compression – YouTube. – Дата доступа: 29.10.2023.

2. Хабр [Электронный ресурс]. – Режим доступа: habr.com. – Дата доступа: 30.10.2023.

УДК 004.921

Влияние цвета и формы в графическом дизайне на эмоциональное восприятие потребителей

Ганусевич А. А., студент

Гала Е. В., студент

Белорусский национальный технический университет

Минск, Республика Беларусь

Научный руководитель: ст. преподаватель Ражнова А. В.