

Курсовой проект по дисциплине компьютерная графика

Метод сохранения качества изображения при понижении его разрядности

Студент

Капустин А.И.гр. ИУ7-52

Руководитель курсового проекта Оленев А.А.



Москва 2016

Цель:

- Получить изображение схожего визуального качества при уменьшении разрядности изображения до 8 бит

Методы реализации:

Для достижения нужного эффекта были выбраны алгоритмы: Белого, Синего, Розового, Фиолетового, Коричневого, Флойда-Стейнберга, Ложного Флойда-Стейнберга, Джарвиса, Джунка и Нинка, а так же были рассмотрены иные алгоритмы для сравнения.

Для оценки качества изображения были выбраны метрики PSNR и SSIM.

Зачем нужно уменьшать разрядность изображений?

- Значительное уменьшение размера
- Поддержка разными типами устройств, не поддерживающими большое количество цветов
- Использование в веб-разработке



Выбор палитры

Если веб-браузер не в состоянии правильно отобразить тот или иной цвет, то он пытается подобрать похожий или смешивает несколько соседних цветов. Может так случиться, что исходный цвет будет замещён чем-то совершенно неподходящим.

Используя безопасную палитру(web-safe), вы будете уверены, что цвета будут выглядеть неизменно хорошо и без искажений даже на очень старых компьютерах, вплоть до 8-битного режима отображения с поддержкой всего 256 цветов.

Описание алгоритма перевода в новую палитру

1. Раскладываем каждый пиксель изображения на три составляющих(RGB)
2. Для каждой составляющей по формуле $51 * (((\text{pixel.color}()) + 25) / 51 + \text{move})$ – выполнить переход в новую палитру, где `move` – случайное значение, полученное на основе алгоритмов распределения

Описание алгоритма, основанном на диффузии ошибок

1. Раскалываем каждый пиксель изображения на три составляющих(RGB).
2. Сохраняем значение пикселя, переводим пиксель в новую палитру
3. Находим разницу между старым и новым значением пикселя(по каждой составляющей RGB)
4. Распределяем полученное значение по соседним пикселям

Метрики оценки качества изображений: PSNR

$$PSNR = 20 \log_{10} \frac{MAX_i}{\sqrt{MSE}}$$

Где MAX_i - это максимальное значение, принимаемое пикселем изображения, MSE - среднеквадратичное отклонение. Для двух монохромных изображений I и K размера $m \times n$, одно из которых считается зашумленным приближением другого, вычисляется так

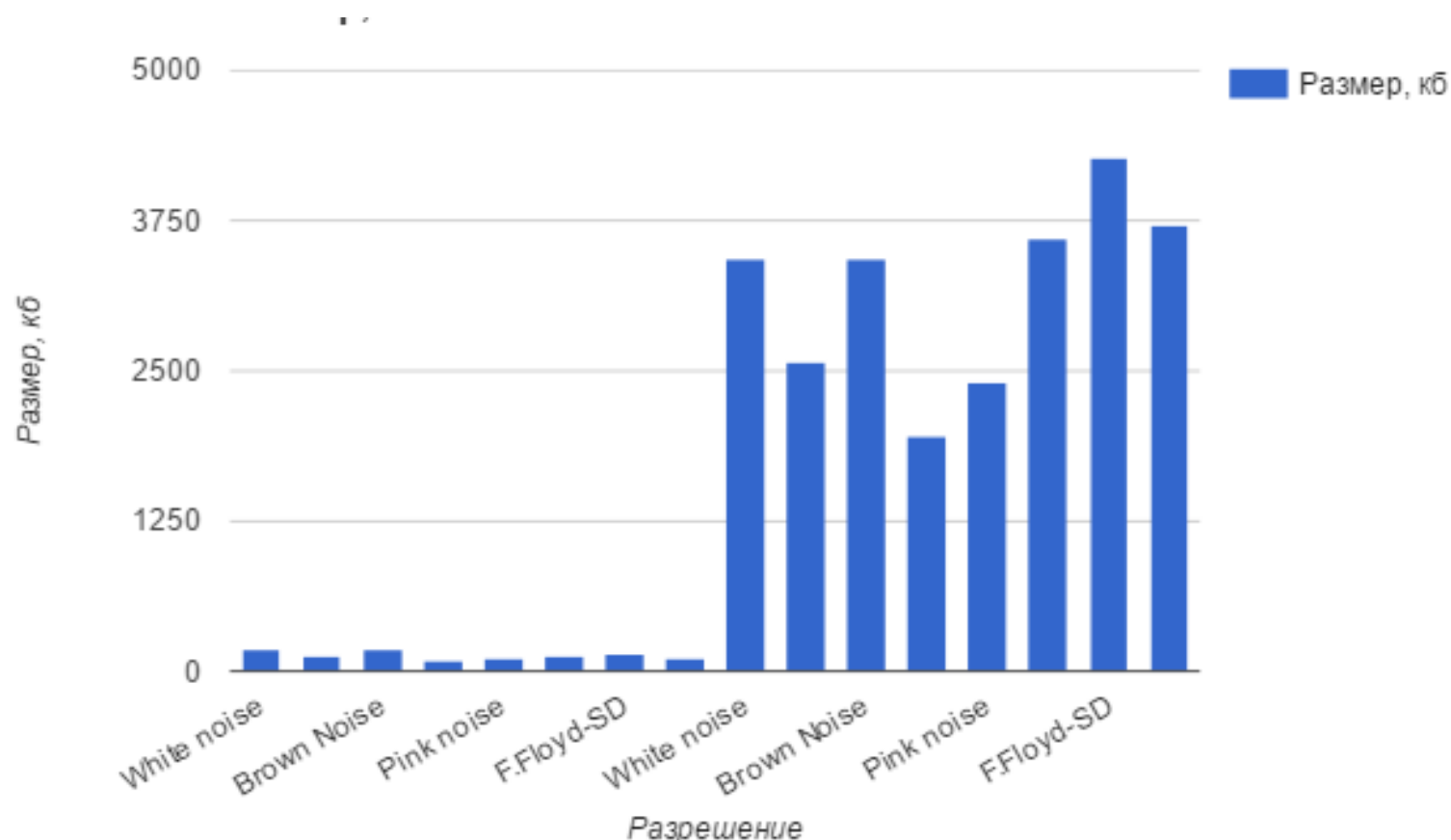
$$MSE = \frac{1}{m * n} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} |I(i,j) - K(i,j)|^2$$

Метрики оценки качества изображений: SSIM

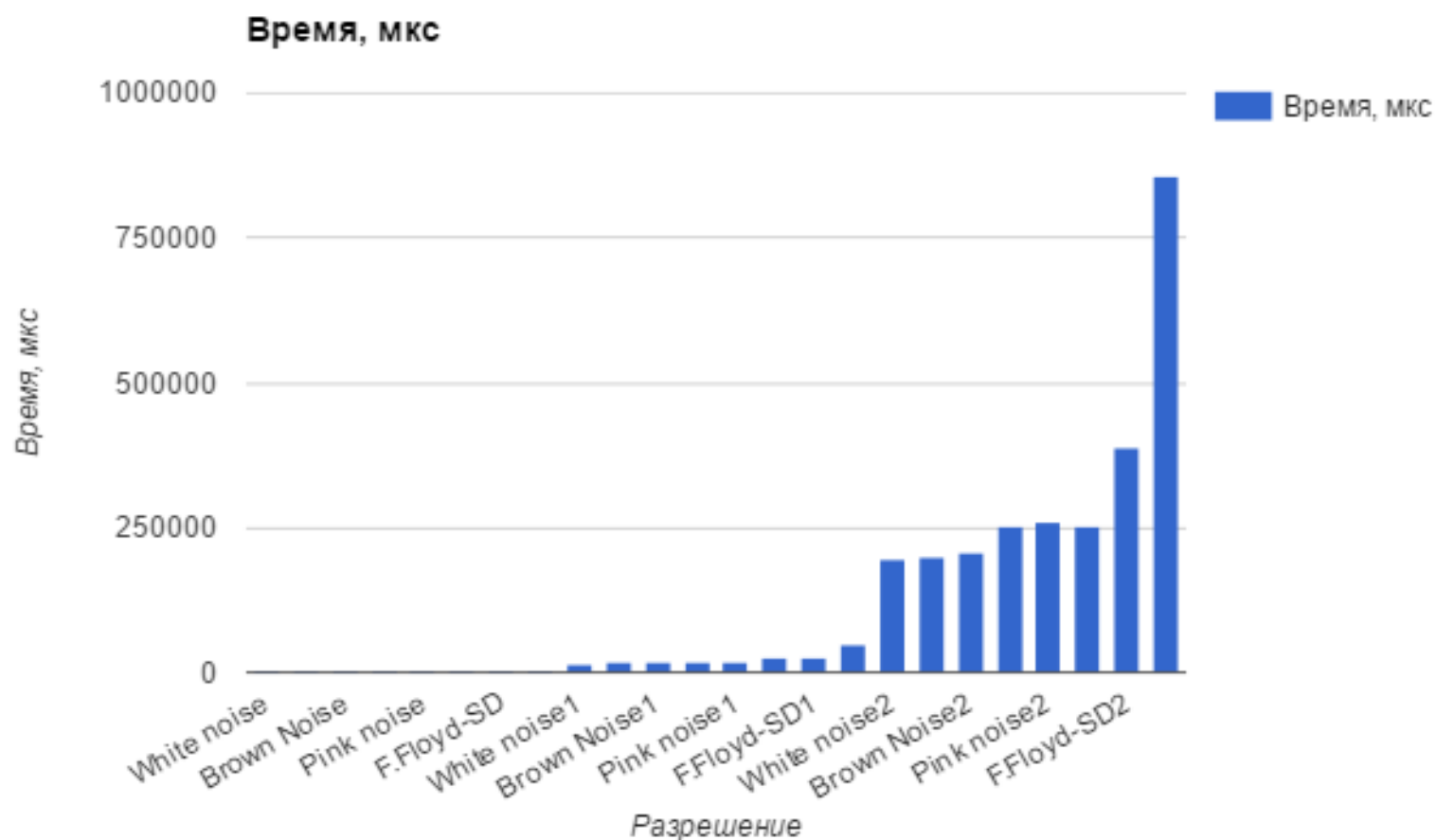
$$SSIM(x,y) = \frac{(2\mu_x\mu_y + c_1)(\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}$$

Тут μ_x среднее значение для первой картинки, μ_y для второй, σ_x среднеквадратичное отклонение для первой картинки, и соответственно σ_y для второй, σ_{xy} это уже ковариация, c_1 и c_2 - поправочные коэффициенты, которые нужны вследствие малости знаменателя

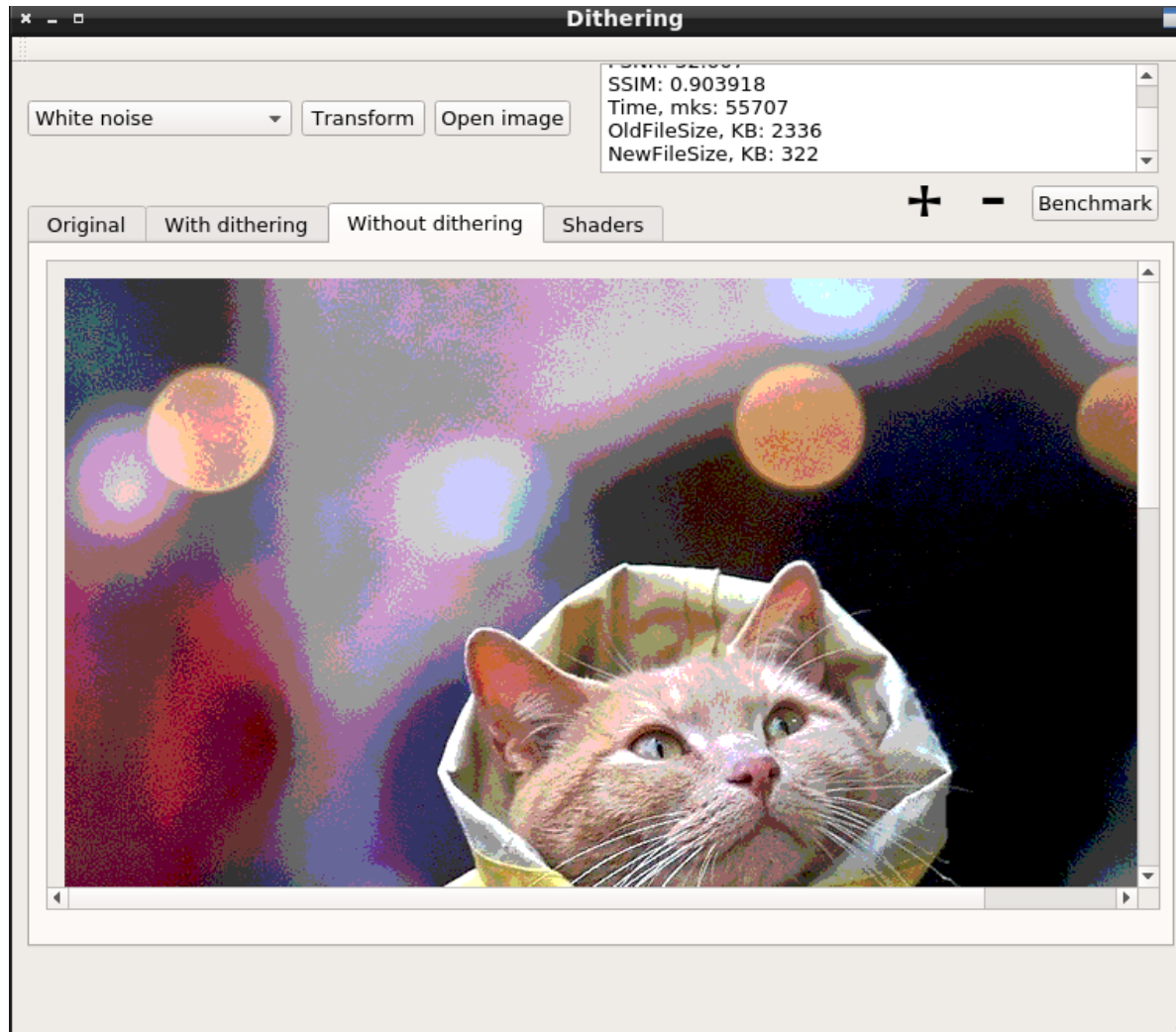
Размер изображения после дизеринга



Скорость работы алгоритмов



Интерфейс программы.



Заключение

В данной работе были реализованы различные алгоритмы дизеринга, было произведено сравнение и анализ этих алгоритмов. Программа позволяет получить изображение схожего визуального качества при значительном уменьшении размера. Был получен вывод о том, что для различных целей следует использовать различные алгоритмы дизеринга, универсального алгоритма дизеринга не существует.