Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Саратовский государственный технический университет

имени Гагарина Ю.А.»

Кафедра «Прикладные информационные технологии»

**Яркостные преобразования**

**Отчет по курсу**

**«Методы обработки сигналов и изображений**

Выполнил:

Студент группы м2-ИФСТ-11

Маркелов Александр Сергеевич

Проверил:

ст. преподаватель каф. ПИТ

Пиминов Дмитрий Алексеевич

Саратов 2025

1. **Теоретическая Часть**

### Определение понятия «Яркостные преобразования»

Яркостные преобразования — это класс методов обработки изображений, направленных на изменение значений интенсивности (яркости) пикселей с целью улучшения визуального восприятия, выделения деталей, повышения контрастности или подготовки изображения к дальнейшему анализу (например, в задачах компьютерного зрения). Эти преобразования могут быть применены как к монохромным (чёрно-белым) изображениям, так и к цветным, где чаще всего изменяется яркостная составляющая в определённой цветовой модели (например, Y в YCbCr или V в HSV). Яркостные преобразования являются фундаментальной частью цифровой обработки изображений, поскольку яркость напрямую влияет на восприятие структуры и содержания изображения человеком или алгоритмом.

Основная идея яркостных преобразований заключается в применении функции преобразования *T(I)*, где *I* — исходное значение интенсивности пикселя, а *T(I)* — новое значение, вычисляемое по заданному правилу. Преобразование может быть точечным (зависит только от значения данного пикселя) или контекстным (учитывает соседние пиксели), но в данном контексте мы сосредоточимся на точечных операциях, которые наиболее распространены.

### Виды яркостных преобразований и их описание

Яркостные преобразования включают в себя широкий спектр методов, различающихся по типу функции преобразования и целям применения. Рассмотрим основные виды:

#### 1. Линейное преобразование (растяжка контраста)

* **Описание**: Линейное преобразование изменяет значения яркости пикселей с использованием линейной функции вида ,где *a* — коэффициент усиления (контраст), а  *b* — смещение (яркость). Один из распространённых подвидов — растяжка контраста, при которой диапазон яркости изображения расширяется до максимально возможного (например, от 0 до 255 для 8-битных изображений). Формула для растяжки:

, где *Imin* и *Imax* ​ — минимальное и максимальное значения яркости в исходном изображении.

* **Пример использования:** Улучшение видимости деталей на изображениях с низким контрастом, таких как медицинские рентгеновские снимки или фотографии, сделанные в условиях плохого освещения. Пример можно увидеть на рисунке 1.

Изображение выглядит как млекопитающее, примат, обезьяна, Мартышковые

Контент, сгенерированный ИИ, может содержать ошибки.Изображение выглядит как млекопитающее, примат, Мартышковые, обезьяна

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 1 - Линейное преобразование. Сравнение с оригиналом.

На рисунке 1 благодаря применению линейной коррекции можно увидеть более детально узор на кофте человека. Но видно, что часть волос Справа отсутствует, что более выделено на рисунке 2.

Изображение выглядит как млекопитающее, примат, обезьяна, Мартышковые

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 2 - "Линейное облысение" – недостаток использования линейной преобразования изображения.

* **Особенности:** Метод прост в реализации, но может усиливать шум в изображении, если он присутствует.

#### 2. Изменение яркости

* **Описание**: Это частный случай линейного преобразования, где изменяется только параметр смещения *b*: *T(I)=I+b*. Положительное значение *b* делает изображение светлее, отрицательное — темнее. Значения выходной яркости обычно обрезаются в диапазоне [0, 255], чтобы избежать переполнения.
* **Пример использования**: Коррекция недоэкспонированных (слишком тёмных) или переэкспонированных (слишком светлых) фотографий. На рисунке 3 представлено изображение с разными настройками яркости (от более темного, к более яркому).



Рисунок 3 - вариации изменения яркости для одного изображения.

* **Особенности**: Простейший метод, не влияющий на контраст, но может приводить к потере деталей в крайних областях диапазона (например, при "выгорании" белого).

#### 3. Изменение контрастности

* **Описание**: Здесь используется только коэффициент усиления *a*: *T(I)=aI*. Если *a>1*, контраст увеличивается, если *0<a<1*, контраст уменьшается. Для сохранения среднего уровня яркости часто добавляют смещение: *T(I)=a(I−Imean)+Imean* ​, где *Imean* ​ — средняя яркость изображения.
* **Пример использования**: Усиление различий между объектами на изображении, например, для выделения текста на фоне.
* **Особенности**: Может усиливать различия между областями, но при слишком большом *a* теряются детали из-за обрезания значений.

**Пример использования изменения контрастности представлен на рисунке 4, на котором заметно, что в зависимости от настроек контрастности изменяется визуальный фокус на разных областях изображения.**



Рисунок 4 - Пример изменения контрастности для одного изображения.

#### 4. Нелинейное преобразование (гамма-коррекция)

* **Описание**: Нелинейное преобразование использует степенную функцию: , где *c* — константа нормализации (обычно *c=1*), а *γ* — параметр гаммы. Если *γ<1*, изображение становится светлее в тёмных областях, если *γ>1*, тёмные области усиливаются, а светлые сжимаются. Для 8-битных изображений входные и выходные значения нормализуются в диапазоне [0, 1], а затем масштабируются обратно:
* **Пример использования**: Компенсация нелинейного восприятия яркости человеческим глазом или коррекция изображения для устройств с разной гаммой (например, мониторов). Пример использования представлен на рисунке 5.

Изображение выглядит как млекопитающее, примат, обезьяна, Мартышковые

Контент, сгенерированный ИИ, может содержать ошибки.Изображение выглядит как млекопитающее, примат, обезьяна, Земное животное

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 5 - Нелинейное преобразование, сравнение с оригиналом.

При использовании нелинейного яркостного преобразования заметно, что более темные участки изображения выделились. У мужчины на фотографии кофта стала черной, а родинки и прочие недостатки кожи стали более заметны.

* **Особенности**: Позволяет выборочно улучшать видимость в определённых диапазонах яркости, но требует подбора параметра *γ*.

#### 5. Эквализация гистограммы

* **Описание**: Это адаптивное преобразование, основанное на кумулятивной функции распределения (CDF) гистограммы яркости. Каждый уровень яркости *I* преобразуется так, чтобы итоговое распределение стало более равномерным: *T(I)=CDF(I)⋅(L−1)*, где *L* — максимальный уровень яркости (255 для 8-битных изображений), а CDF(*I*) — накопленная вероятность для уровня *I*.
* **Пример использования**: Улучшение контрастности изображений с узким диапазоном яркости, таких как туманные пейзажи или медицинские снимки. Пример использования данного преобразования представлен на рисунке 6.

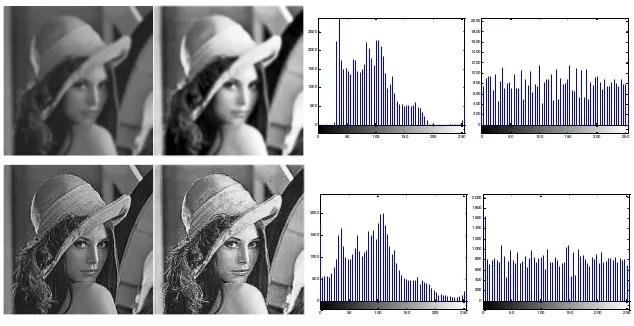


Рисунок 6 - Использование Эквилизации гистограммы для изображения.

* **Особенности**: Автоматически адаптируется к данным изображения, но может усиливать шум или создавать неестественный вид.

#### 6. Пороговая обработка

* **Описание**: Простейшее преобразование, при котором значения яркости сравниваются с порогом *Tthr*​:

Это превращает изображение в бинарное (чёрно-белое).

* **Пример использования**: Выделение объектов на фоне, например, в задачах сегментации или распознавания текста.
* **Особенности**: Полезно для простых задач, но теряет градации серого и детали, что особенно видно на примере (рисунок 7).

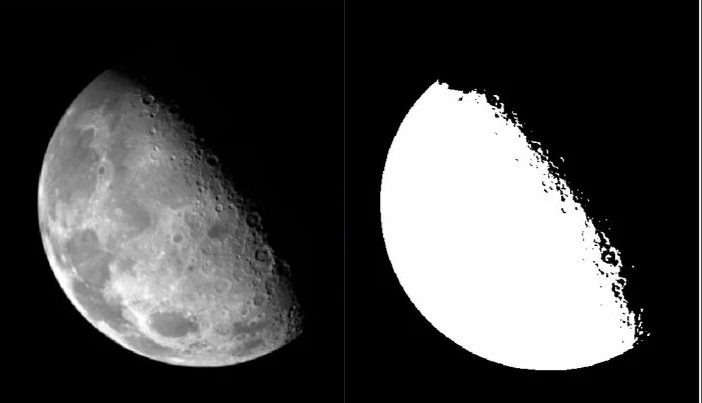


Рисунок 7 - Пример пороговой обработки яркостного преобразования изображения.

#### 7. Инверсия яркости

* **Описание**: Преобразование, при котором значения яркости заменяются на противоположные: *T(I)=255−I* для 8-битных изображений. Это создаёт эффект "негатива", и делает темные участки – светлыми, а светлые участки – темными.
* **Пример использования**: Обработка фотоплёнки или выделение деталей в специфических задачах (например, астрономические снимки).
* **Особенности**: Простое и обратимоe преобразование, не изменяющее контраст.

**Пример выполнения инверсии яркости продемонстрирован на рисунке 8.**



Рисунок 8 - Инверсия яркости изображения.

На данном примере видно, что листья, которые находятся в тени от предмета за кадром или же за тенью других листьев, стали ярче, а те, что находились под прямыми лучами источника света, стали более темными. Эффект яркости в примере достигнут.

### Примеры применения в реальных задачах

1. **Фотография**: Гамма-коррекция используется для адаптации изображения к восприятию человеческим глазом, а эквализация — для улучшения снимков в условиях плохой видимости.
2. **Медицина**: Линейная растяжка контраста помогает выделить структуры на рентгеновских или МРТ-снимках.
3. **Компьютерное зрение**: Пороговая обработка применяется для сегментации объектов, а инверсия — для анализа астрономических данных.
4. **Графика**: Изменение яркости и контраста используется в редакторах для создания художественных эффектов.

### Заключение

Яркостные преобразования и анализ гистограмм составляют основу обработки изображений, обеспечивая инструменты для решения широкого круга задач — от улучшения эстетических качеств фотографий до подготовки данных для сложных систем компьютерного зрения. Эти методы позволяют манипулировать яркостью и контрастом изображения с высокой точностью, адаптируя его под конкретные нужды: будь то коррекция экспозиции, выделение объектов или устранение аппаратных искажений. Разнообразие подходов — от простых линейных операций до сложных нелинейных преобразований, таких как гамма-коррекция или эквализация гистограммы, — делает их универсальными и применимыми в самых разных областях: фотографии, медицине, астрономии, графике, промышленности и науке.

Гистограммы, в свою очередь, выступают незаменимым инструментом анализа, предоставляя наглядную картину распределения яркости и позволяя принимать обоснованные решения о выборе метода обработки. Они не только помогают диагностировать проблемы изображения, такие как недостаточная контрастность или неправильная экспозиция, но и дают возможность оценить эффективность применённых преобразований, сравнивая состояние "до" и "после". В сочетании с яркостными преобразованиями гистограммы создают мощный аналитический и практический арсенал, который лежит в основе как ручной, так и автоматизированной обработки изображений.

Значение этих технологий трудно переоценить. В современном мире, где визуальная информация играет ключевую роль, яркостные преобразования обеспечивают не только улучшение качества изображений, но и их функциональность в задачах, требующих высокой точности — от распознавания лиц до анализа спутниковых снимков. Понимание их теоретических основ, возможностей и ограничений открывает путь к созданию более совершенных систем обработки изображений, способных отвечать вызовам XXI века. Таким образом, яркостные преобразования и гистограммы остаются краеугольным камнем в развитии технологий визуализации, обеспечивая баланс между технической точностью и человеческим восприятием. Их изучение и применение — это шаг к глубокому освоению цифровой обработки изображений, открывающий новые горизонты для творчества и научных открытий.

1. **Практическая часть**
   1. **. Разработка программного обеспечения**

**Общая цель**

Разработанное Java-приложение представляет собой графический редактор для обработки изображений с использованием библиотеки Swing. Приложение реализует основные методы яркостных преобразований и анализа гистограмм.

**Основные возможности**

1. **Загрузка изображений**:
   * Поддерживает популярные форматы изображений (например, PNG, JPEG, BMP).
2. **Преобразования:** 
   * Яркость/контраст/насыщенность (регулируемые слайдерами)
   * Гамма-коррекция
   * Линейное выравнивание гистограммы
   * Преобразование в градации серого
3. **Визуализация**:
   * Отображение оригинального и обработанного изображений
   * Построение гистограмм RGB Ручные настройки:
4. **Управление**:
   * Сброс изменений
   * Сохранение результата

**Технические особенности**

* **Интерфейс**: Разделён на две части:
  + Слева — рабочая область с изображением и панель управления с кнопками "Open", " Grayscale", "Linear Correction", "Nonlinear Correction" и " Save".
  + Справа — гистограмма и ползунки регулировок.

**2.2. Архитектура**

Модель: `ImageModel` - хранение и обработка изображений

Представление: `MainFrame`, `ImagePanel`, `HistogramPanel` - GUI

Контроллер: `ImageController`, `AdjustmentsController` - логика обработки

**2.3. Руководство пользователя**

**1. Загрузка изображения**

* **Что сделать**: Нажмите кнопку "Open".
* **Что произойдёт**: Откроется окно выбора файла. Найдите изображение (например, .jpg или .png) на вашем компьютере, выберите его и нажмите "Открыть".
* **Результат**: В окне появится выбранное изображение с которым можно будет начинать работать.

Изображение выглядит как млекопитающее, примат, снимок экрана, обезьяна

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 9 - Загрузка изображение в ПО.

**2. Применение преобразований**

* **Что сделать**: Используйте панель слайдеров для регулировки:

- Яркость (-100..100)

- Контраст (-100..100)

- Насыщенность (-100..100)

- Гамма (0.1..3.0)

Кнопки специальных преобразований:

- "Grayscale" - преобразование в ч/б

- "Linear Correction" - автоматическое выравнивание гистограммы

* **Результат**: Вы можете редактировать изображение.

Изображение выглядит как текст, снимок экрана, линия, Параллельный

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 10 – Применение преобразований.

**3. Анализ гистограмм**

Гистограмма автоматически обновляется при изменениях

- Отображаются каналы R, G, B и общая яркость(рисунок 10).

Изображение выглядит как силуэт, Графика, снимок экрана, графический дизайн

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 11 - Гистограмма.

**4. Сохранение результатов**

Нажмите "Save"

- Выберите формат (JPG/PNG/BMP) и имя файла

- Изображение сохранится с выбранным расширением

Изображение выглядит как млекопитающее, примат, снимок экрана, обезьяна

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 12 – Сохранение измененного изображения.

2.2**. Код ПО**

https://github.com/Aleksandr1209/ImageConverter.git