



Федеральное государственное автономное образовательное учреждение высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

Метод сжатия статических изображений без потерь на основе алгоритма Хаффмана

Студент: Ковалец Кирилл Эдуардович ИУ7-42М
Научный руководитель: Новик Наталья Владимировна

Москва, 2025

Цель и задачи

Цель работы: разработать метод сжатия статических изображений без потерь на основе алгоритма Хаффмана.

Задачи:

- провести аналитический обзор известных методов сжатия статических изображений;
- разработать метод сжатия статических изображений без потерь на основе алгоритма Хаффмана;
- разработать программное обеспечение для демонстрации работы созданного метода;
- провести сравнение разработанного метода с аналогами по степени сжатия изображений.

Сравнение методов сжатия без потерь

- К1 — возможность кодирования данных за один проход;
- К2 — отсутствие необходимости в таблице частот пикселей сжимаемого изображения;
- К3 — наличие в зашифрованном сообщении информации для распаковщика;
- К4 — наличие у каждого сжатого пикселя своего кода.

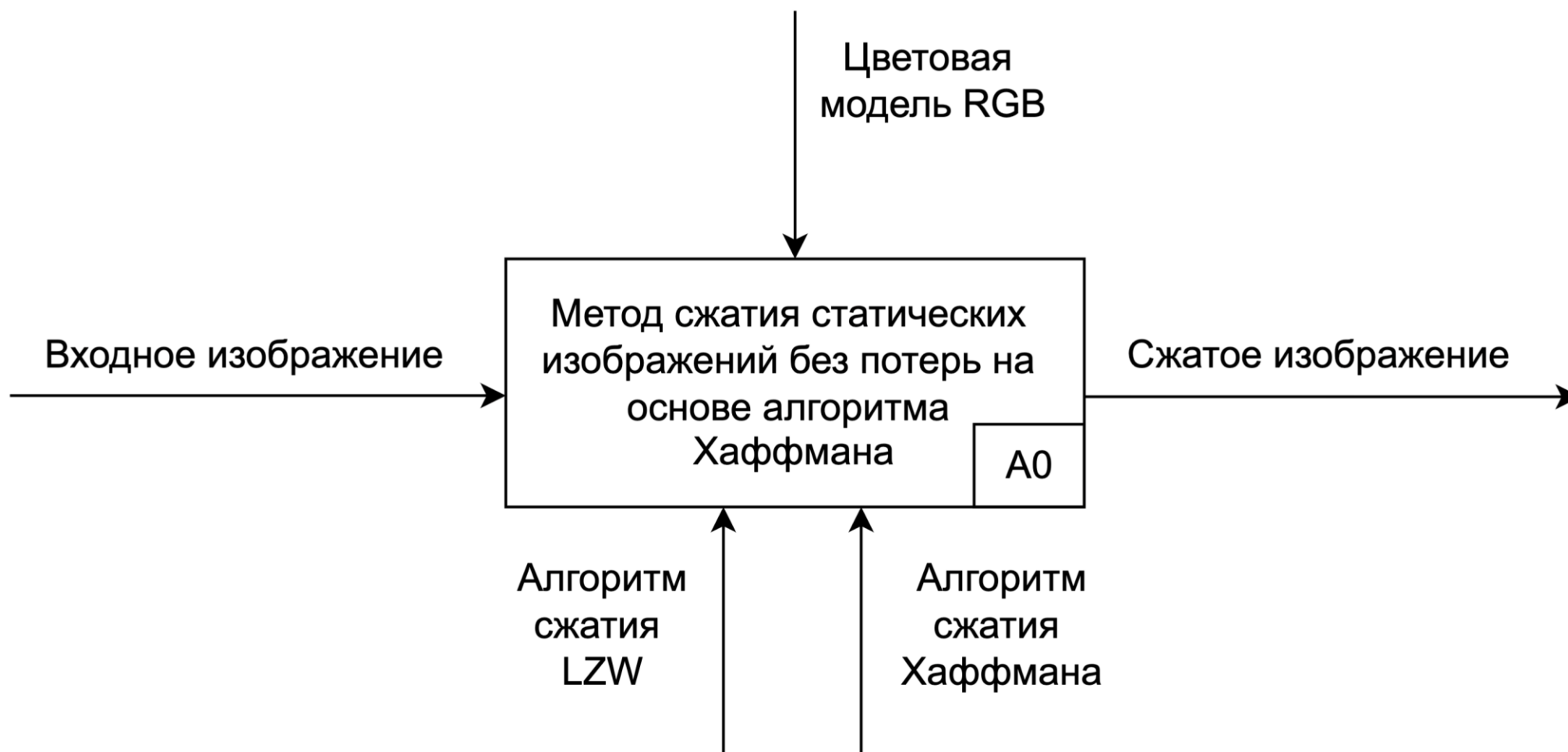
Метод сжатия	К1	К2	К3	К4
RLE	+	+	—	—
LZW (словарный метод)	+	+	+	—
Унарное кодирование	+	—	+	+
Метод Хаффмана	—	—	+	+
Арифметическое кодирование	+	—	+	—

Выбор цветовой модели

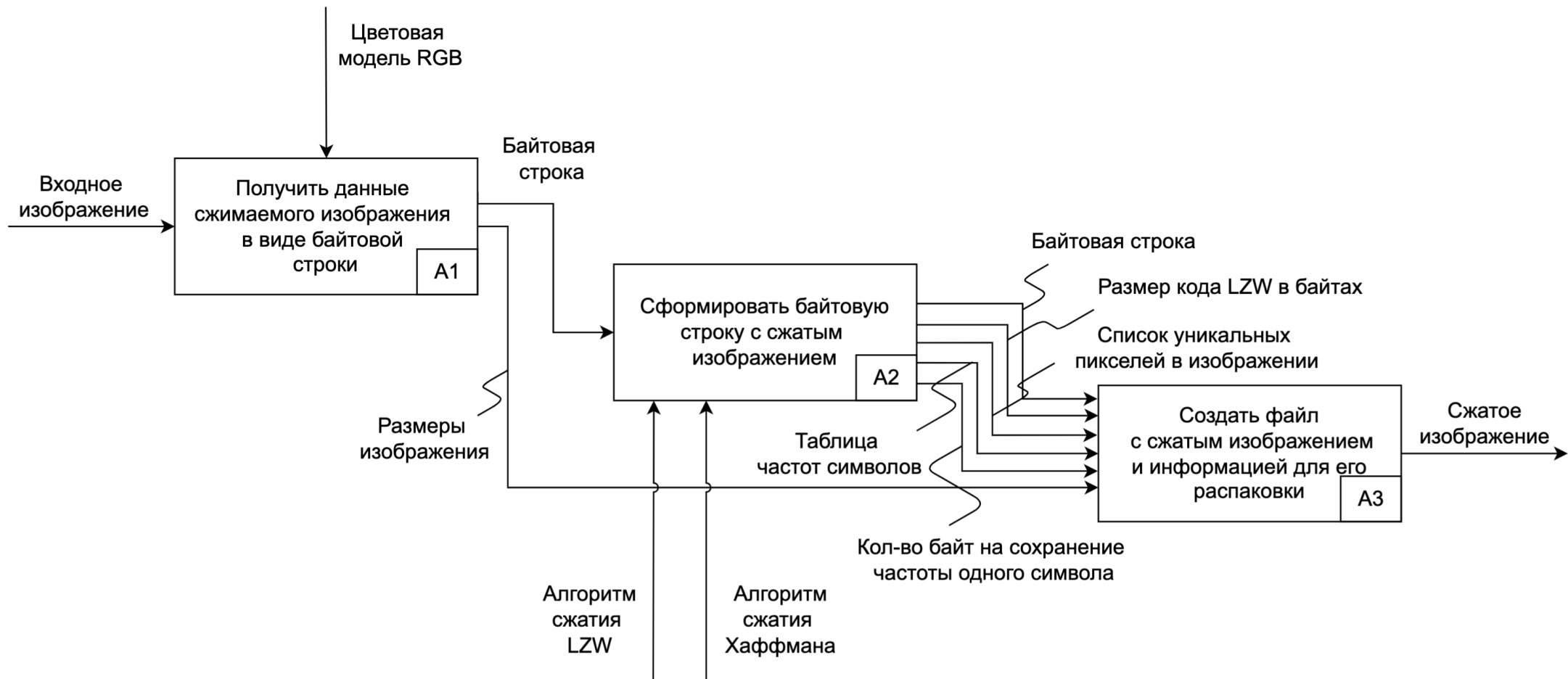
- K1 — класс метода по принципу действия;
- K2 — количество байт для кодирования одного пикселя;
- K3 — наличие поддержки альфа-канала;
- K4 — отсутствие отдельного канала для яркости.

Метод сжатия	K1	K2	K3	K4
RGB	аддитивный	3	—	+
RGBA	аддитивный	4	+	+
CMYK	субтрактивный	4	—	+
LAB	перцепционный	3	—	—
HSB	перцепционный	3	—	—

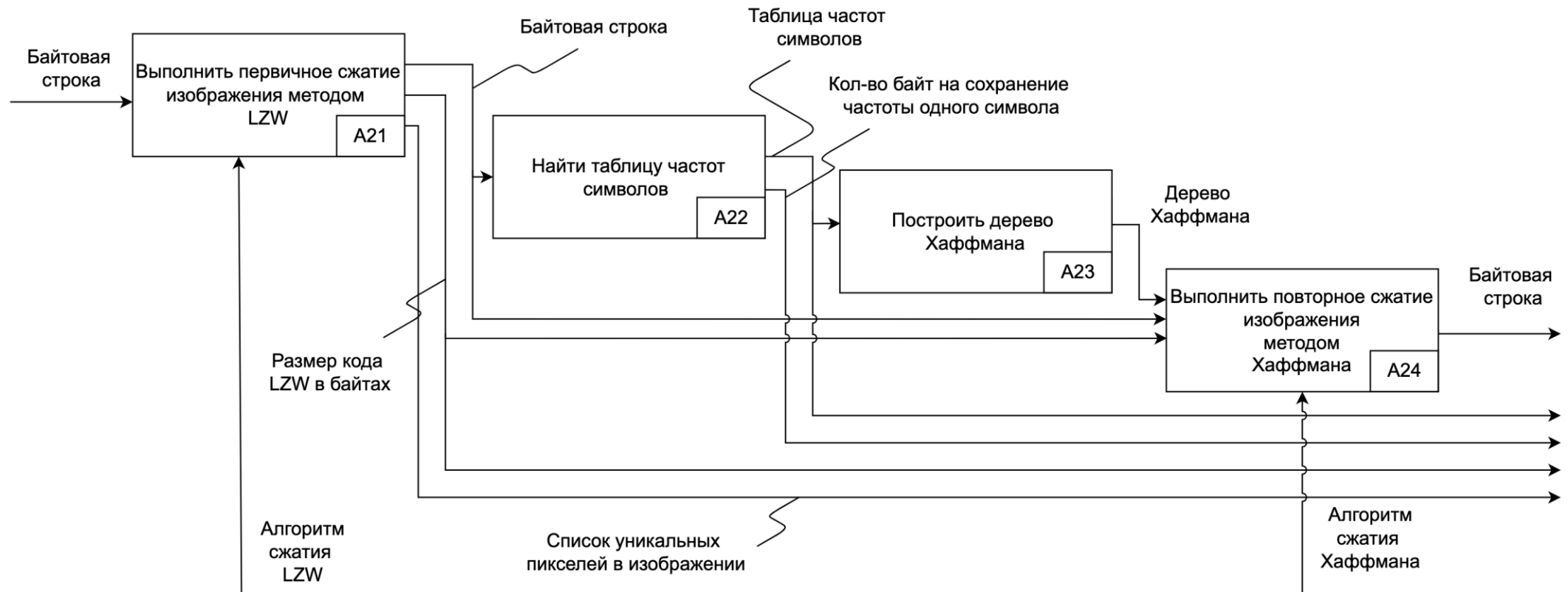
Формализованная постановка задачи в нотации IDEF0



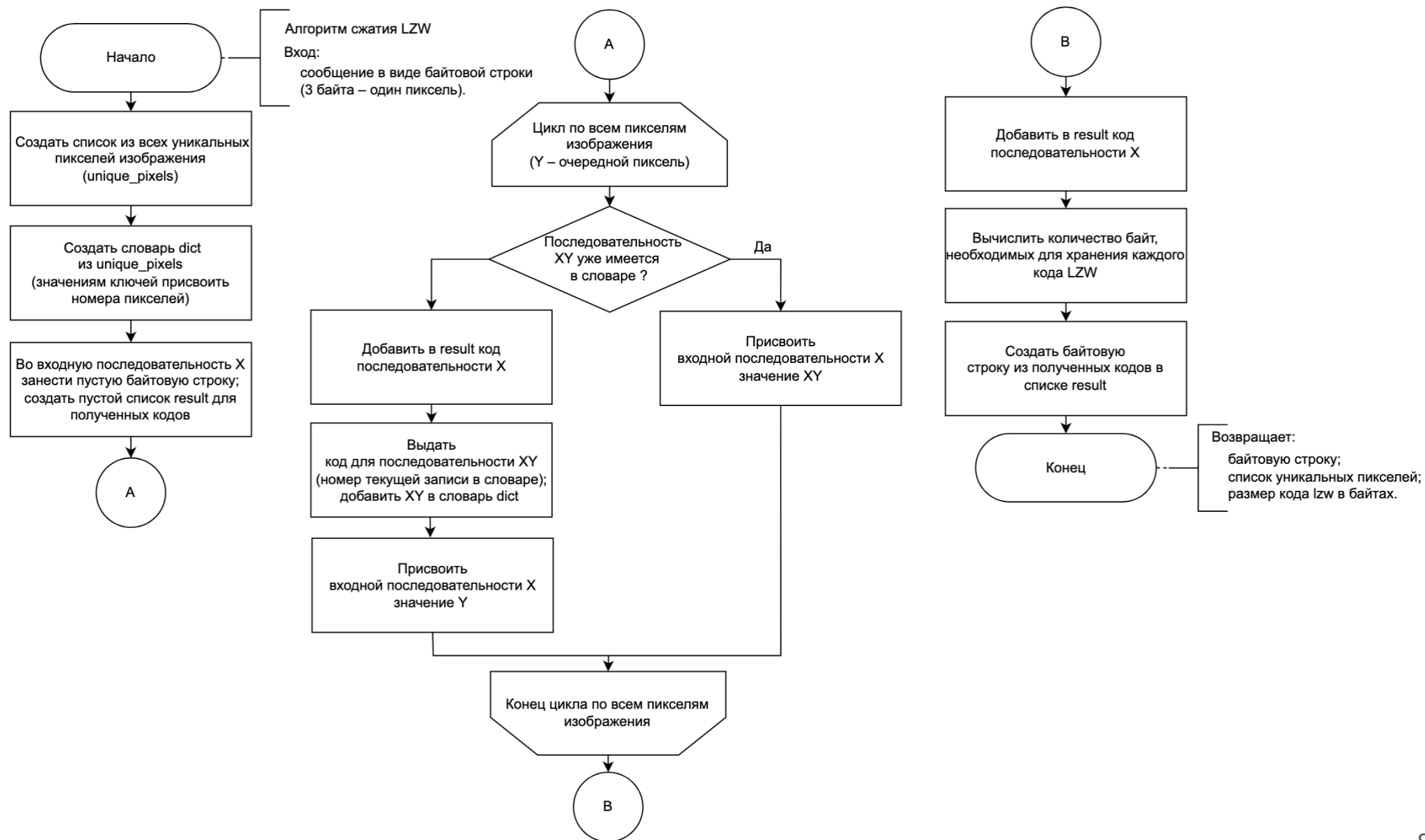
Детализированная IDEF0-диаграмма первого уровня разработанного метода



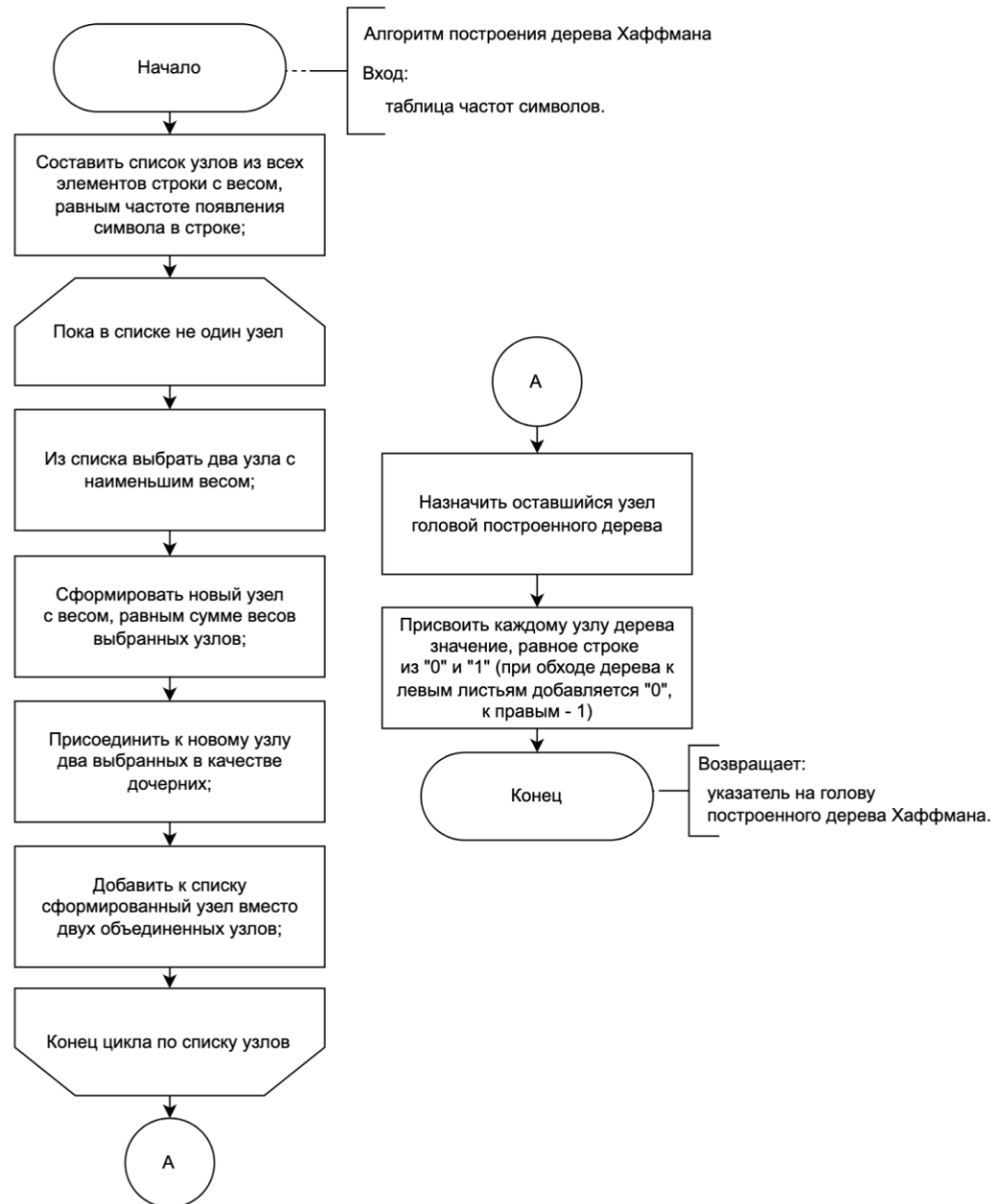
Детализированная IDEF0-диаграмма уровня А2 разработанного метода



Первичное сжатие с использованием метода LZW



Построение дерева Хаффмана



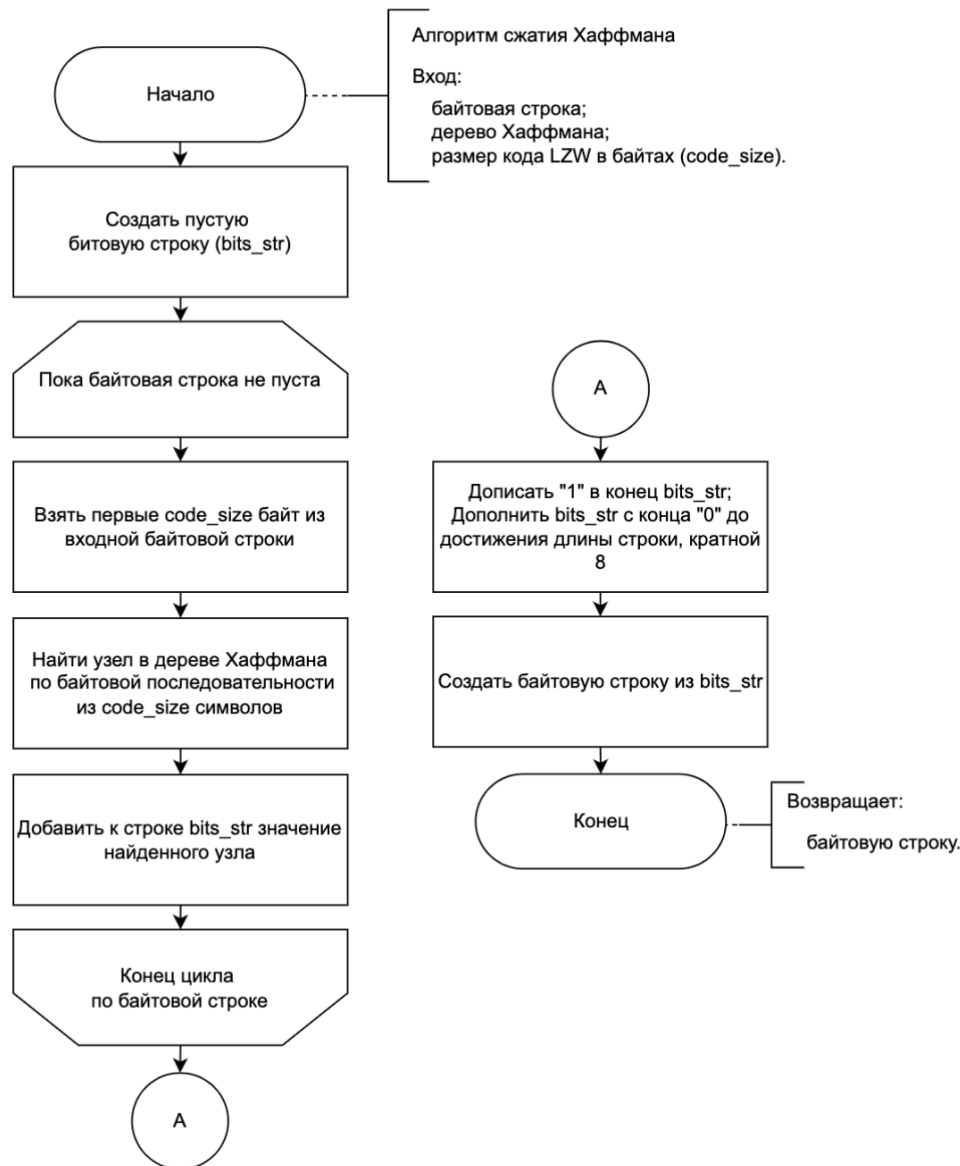
Формулы для построения дерева Хаффмана

1. Для узла v , соответствующему символу s_i : $w(v_i) = f(s_i)$.
2. Вес узла, полученного в результате объединения двух узлов с минимальными весами (v_{left} и v_{right}): $w(v) = w(v_{\text{left}}) + w(v_{\text{right}})$.
3. Средняя длина кода L вычисляется по формуле: $L = \sum_{i=1}^n f(s_i) \cdot |c(s_i)|$.
4. Количество узлов N в дереве Хаффмана для n символов: $N = 2n - 1$.

Используемые обозначения:

- $S = \{s_1, s_2, \dots, s_n\}$ – алфавит из n символов;
- s_i – символ алфавита S , где $i \in \overline{1, n}$;
- $f(s_i)$ – частота появления символа s_i в изображении;
- $c(s_i)$ – последовательность битов от корня дерева до листка символа s_i ;
- $w(v_i)$ – вес узла v_i .

Выполнение сжатия подготовленных данных методом Хаффмана



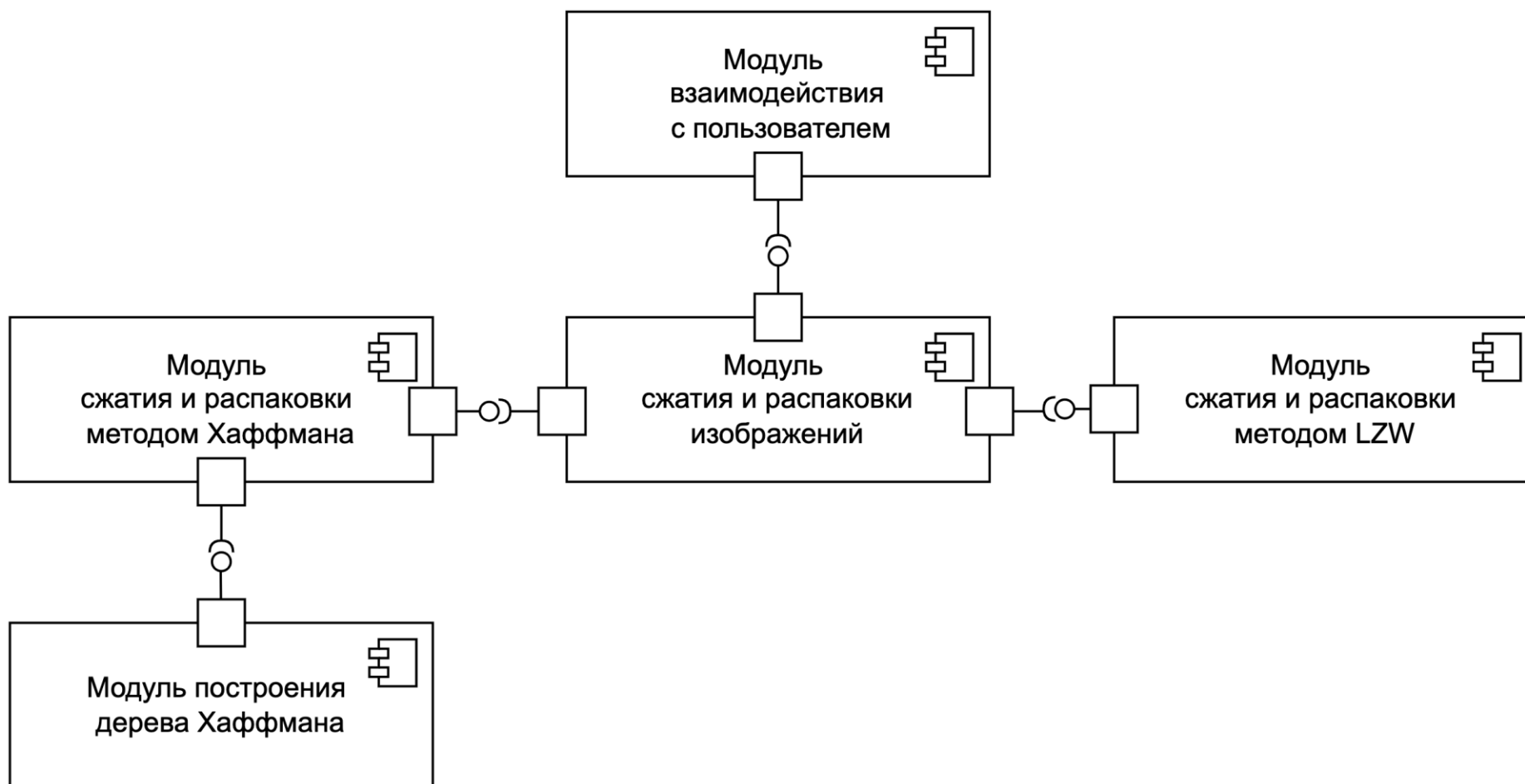
Особенности разработанного гибридного метода

- Применение первичной обработки изображения методом LZW для удаления избыточности и уменьшения количества обрабатываемых СИМВОЛОВ.
- Восстановление словаря LZW по мере распаковки изображения с использованием списка уникальных пикселей и величины кода одного пикселя.
- При обработке подготовленных данных методом Хаффмана за один символ принимается код последовательности пикселей, сгенерированный на первом этапе метода.
- Выравнивание битовой строки, полученной на последнем этапе сжатия, до длины, кратной 8.

Используемые программные средства для реализации метода

- **Python** — язык программирования;
- **tkinter** — библиотека для создания графического интерфейса;
- **matplotlib.pyplot** — модуль, предоставляющий функции для создания графиков;
- **matplotlib.offsetbox** — модуль, предоставляющий возможность размещения текстовых и графических элементов на построенных графиках;
- **bitarray** — библиотека для работы с массивами битов (использовалась при сжатии данных методом Хаффмана);
- **progress** — библиотека, используемая для отображения прогресса этапов сжатия и распаковки изображений.

UML-диаграмма компонентов разработанного ПО

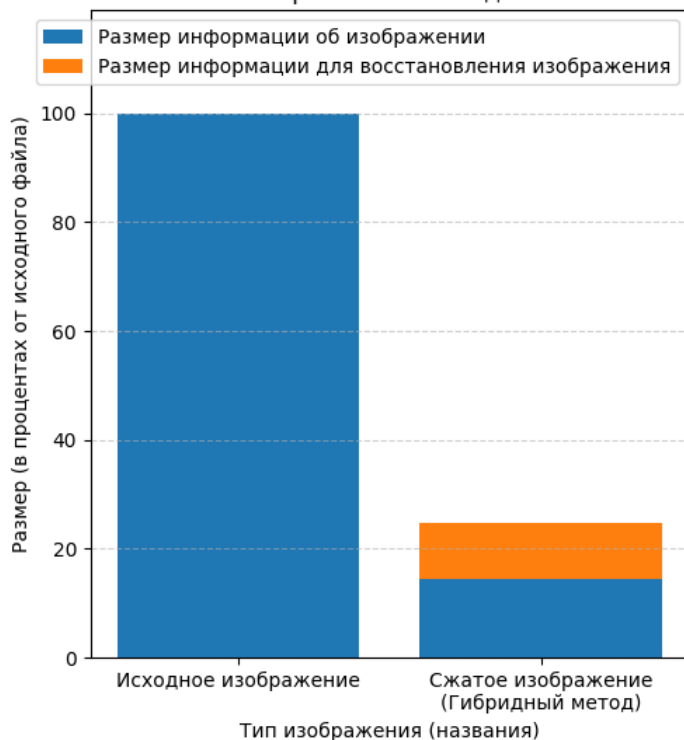


Результаты сжатия изображения



Сжимаемое изображение

Диаграмма сравнения размеров сжатого изображения с исходным



Выпускная квалификационная работа (Ковалец Кирилл ИУ7-42М)

ВХОДНЫЕ ДАННЫЕ

Путь до исходного файла

Путь до директории с результатами

☒ Гибридный метод ☐ Метод Хаффмана ☐ Метод LZW

ЭТАПЫ СЖАТИЯ И РАСПАКОВКИ ИЗОБРАЖЕНИЙ

Сжатие файла sunrise.bmp (Гибридный метод)

Размер изначального файла: 720.1 kB
Кол-во различных пикселей в изображении: 212
Сжатие методом LZW
Кол-во цепочек пикселей в словаре: 61041
Размер кода для метода LZW в байтах: 2
Среднее число пикселей в цепочках: 3.93
Вычисление таблицы частот символов
Размер таблицы частот символов: 24210
Максимальная частота символа: 91 (1 байт(а) на сохранение частоты символа)
Среднее кол-во повторных использований цепочек байт: 2.51
Построение дерева Хаффмана
Сжатие методом Хаффмана
Размер сжатого изображения: 104.6 kB
Размер информации для распаковки файла: 73.3 kB
Размер сжатого файла (вместе с информацией для распаковки): 177.9 kB
Степень сжатия файла: 75.29%

Файл успешно сжат (Гибридный метод)

Построение дерева Хаффмана
Распаковка методом Хаффмана
Распаковка методом LZW
Размер распакованного файла: 720.1 kB

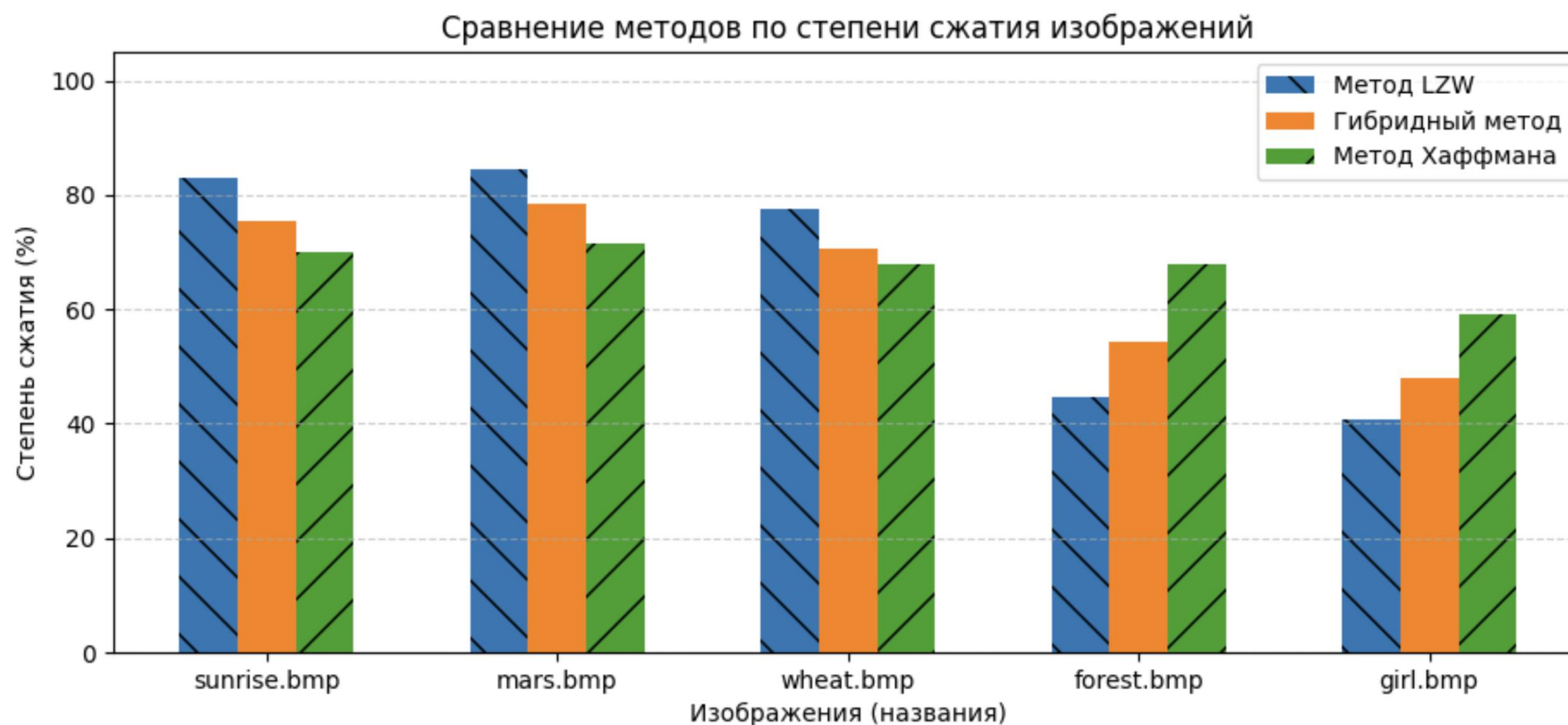
Файл успешно распакован (Гибридный метод)

ВОЗМОЖНЫЕ ДЕЙСТВИЯ

Пример работы программы

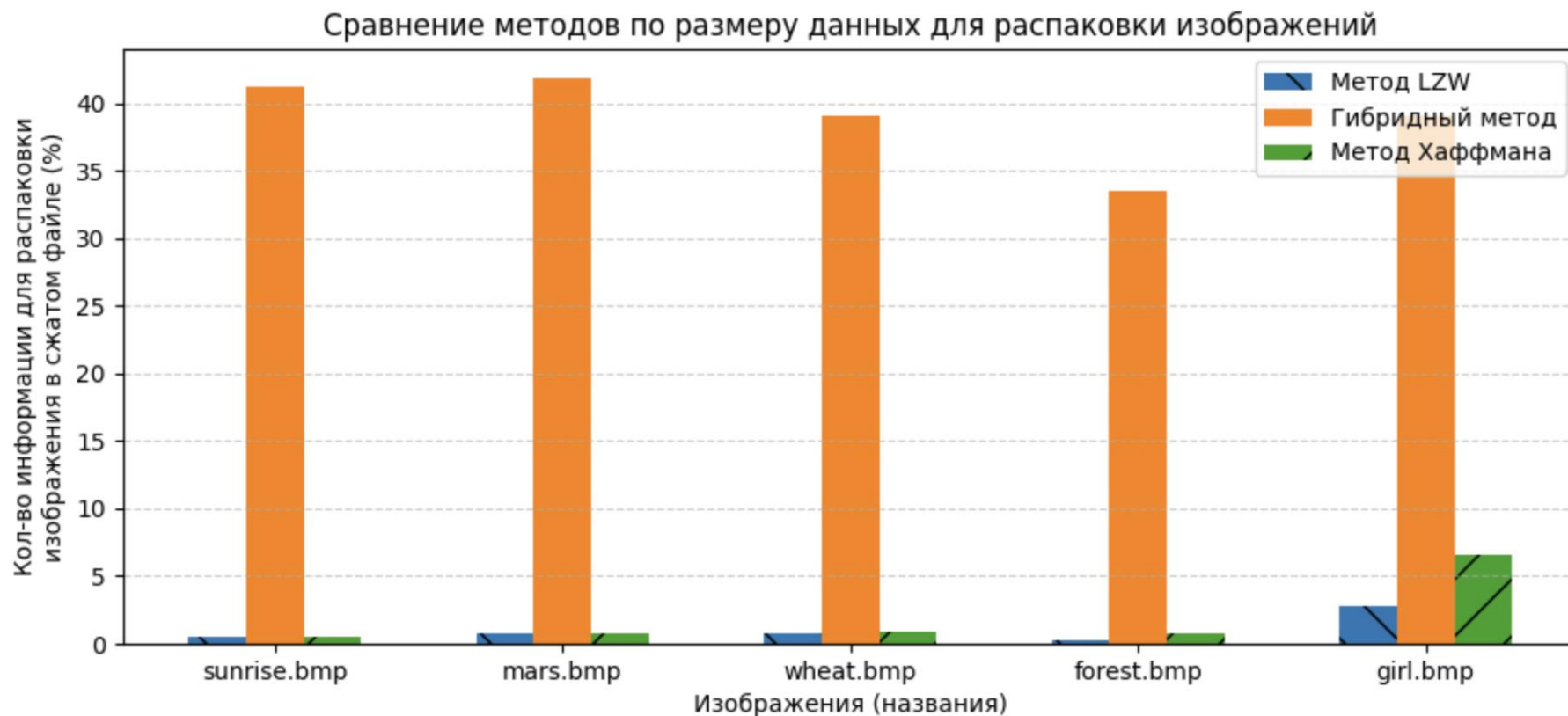
Сравнение методов сжатия изображений (по степени сжатия)

- График показывает, на сколько процентов от изначального размера файла удалось сжать изображение.



Сравнение методов сжатия изображений (по размеру данных для распаковки)

- График показывает, сколько процентов от размера сжатого файла занимает информация для его распаковки.



Заключение

В ходе выполнения работы цель была достигнута, а все поставленные задачи выполнены:

- проведен аналитический обзор известных методов сжатия статических изображений;
- разработан метод сжатия статических изображений без потерь на основе алгоритма Хаффмана;
- разработано программное обеспечение для демонстрации работы созданного метода;
- проведено сравнение разработанного метода с аналогами по степени сжатия изображений.

Направление дальнейшего развития

- Добавить поддержку сжатия файлов, отличных от изображений.
- Уменьшить размер сжатого файла путем оптимизации данных, требуемых для распаковки изображения.
- Разработать алгоритмы управления сжатием файлов в зависимости от особенностей исходных изображений.

Научная публикация

- *Ковалец К. Э., Новик Н. В.* Метод сжатия статических изображений на основе алгоритма Хаффмана // Вестник Российского нового университета. Серия: Сложные системы: модели, анализ и управление. – 2025. – № 2. (Рецензируемое издание ВАК, научные специальности: информатика, вычислительная техника и управление)