

Московский государственный университет имени М.В. Ломоносова
Факультет вычислительной математики и кибернетики
Кафедра математических методов прогнозирования



Отчёт по лабораторной работе №3 курса “Обработка и распознавание изображений”

Зыков Валерий
317 группа

Москва
2024

Содержание

1	Постановка задачи	3
2	Описание данных	3
3	Описание метода решения	4
4	Описание программой реализации	8
5	Эксперименты	9
5.1	Класс I	9
5.2	Класс II	9
5.3	Класс III	9
5.4	Класс IV	10
5.5	Расстояния между векторами степеней вершин	10
6	Выводы	11
	Список литературы	12

1 Постановка задачи

В данной лабораторной работе требовалось разработать и реализовать программу для классификации изображений моделей графов, построенных из магнитной головоломки.

Программа должна обеспечить:

- ввод и отображение на экране изображений в формате JPEG;
- сегментацию изображений на основе точечных и пространственных преобразований;
- генерацию признаков описаний структуры графов на изображениях;
- построение классификатора изображения в соответствии с заданным набором эталонов.

Задача состоит в построении меры сходства изображений на основе выделения и анализа формы графа, составленного из деталей головоломки. Нужно разработать и реализовать алгоритм, входом которого является изображение, а выходом – описание признаков формы изображенной модели.

2 Описание данных

В качестве исходных данных прилагается набор из 16 цветных изображений моделей, построенных из деталей магнитной игры-головоломки в формате 1024×768 с разрешением 72 dpi. Всего задано 4 структуры графа, эталоны которых представлены на изображениях 2.jpg (класс I), 3.jpg (класс II), 4.jpg (класс III), 5.jpg (класс IV). На остальных изображениях представлены графы, изоморфные четырём эталонным образцам (см. рис. 1).

В задание входят задачи двух уровней сложности: Intermediate, Expert.

- Класс **Intermediate**: решение задачи для изображений на белом фоне.
- Класс **Expert**: решение задачи для изображений на пёстром фоне.

Данная работа соответствует уровню сложности **Intermediate**. Все данные изображения, соответствующие этому уровню сложности приведены на рис. 1. Назовем эти изображения обучающими. Программа должна хорошо работать не только на обучающих изображениях, но и на тестовых изображениях аналогичного типа.

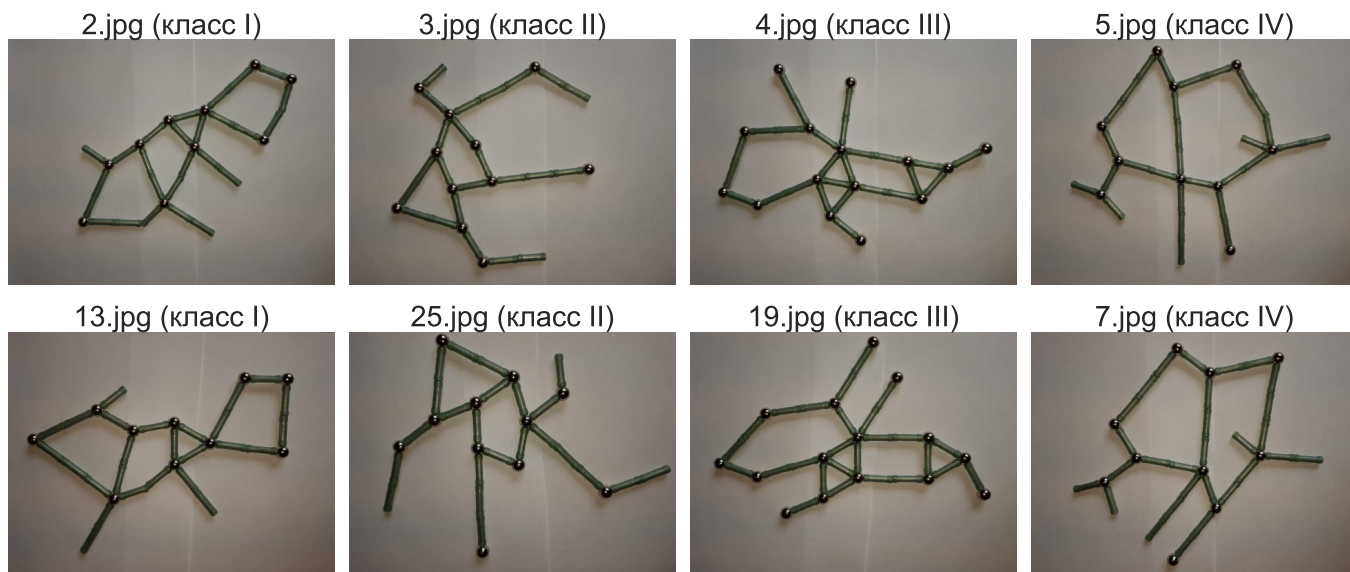


Рис. 1. Примеры входных изображений

3 Описание метода решения

Изначально входное изображение переводится в полутоновое и потом бинаризуется с некоторым порогом. Результат этих операций изображен на рис. 2.

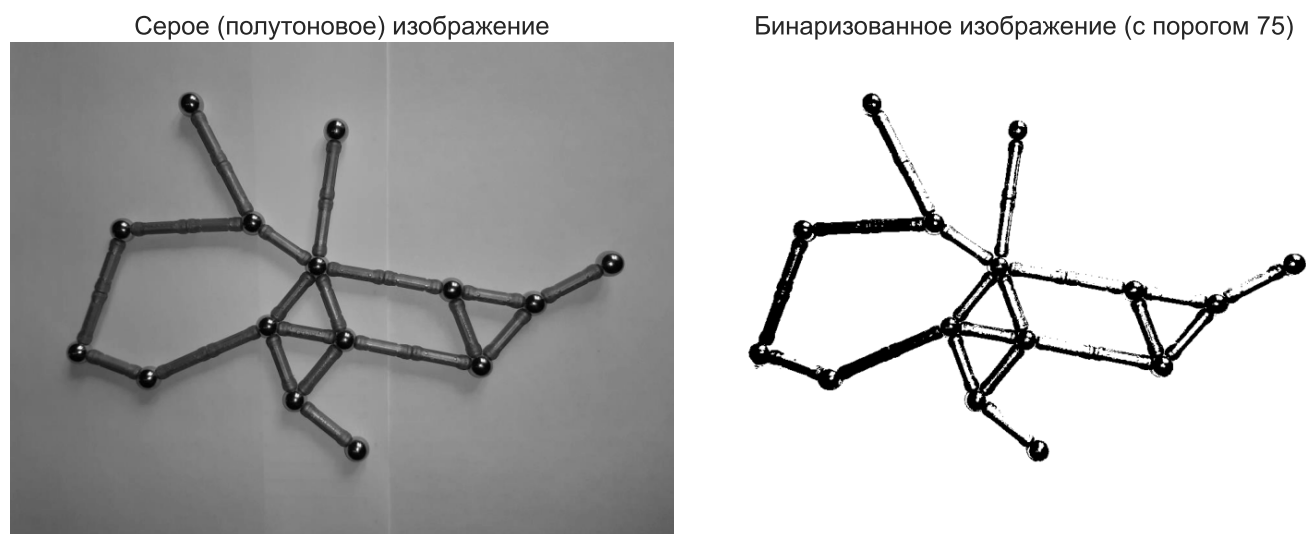


Рис. 2. Полутоновое и бинаризованное (черно-белое) изображение

Далее к полученному бинаризованному (черно-белому) изображению применяется морфологическая операция “Закрытие” с квадратным ядром из единиц (размер ядра был выбран равным 15). Это делается с целью устранения белых засветов на черно-белом изображении. Результат этой операции приведен на рис. 3.

После этого к полученному изображению применяется алгоритм скелетизации Чжана [4]. К построенному скелету применяется алгоритм стрижки DSE [5]. Результат этих действий изображен на рис. 4.



Рис. 3. Результат закрытия

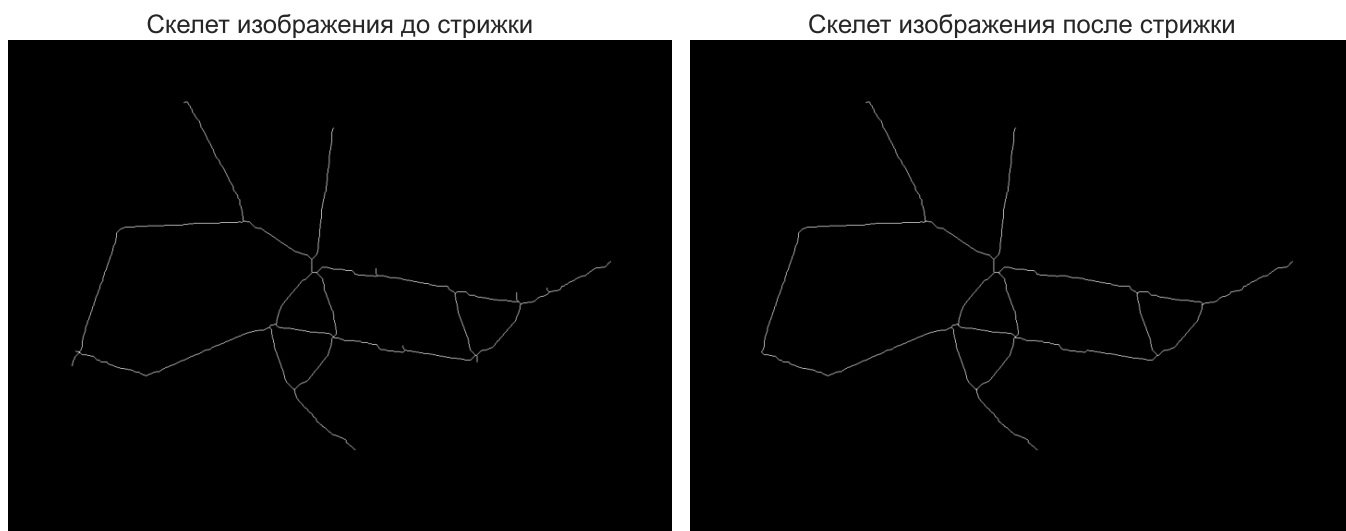


Рис. 4. Построение скелета изображения

Скелет изображения является графом. Для классификации изображения предлагается посчитать количество вершин его скелета разных степеней. Заметим, что в местах железных шариков часто присутствуют несколько вершин скелета (см. рис. 5).

Нам бы хотелось, чтобы одному железному шарiku соответствовало не более одной вершины скелета. Чтобы добиться этого, будем считать одной обобщенной

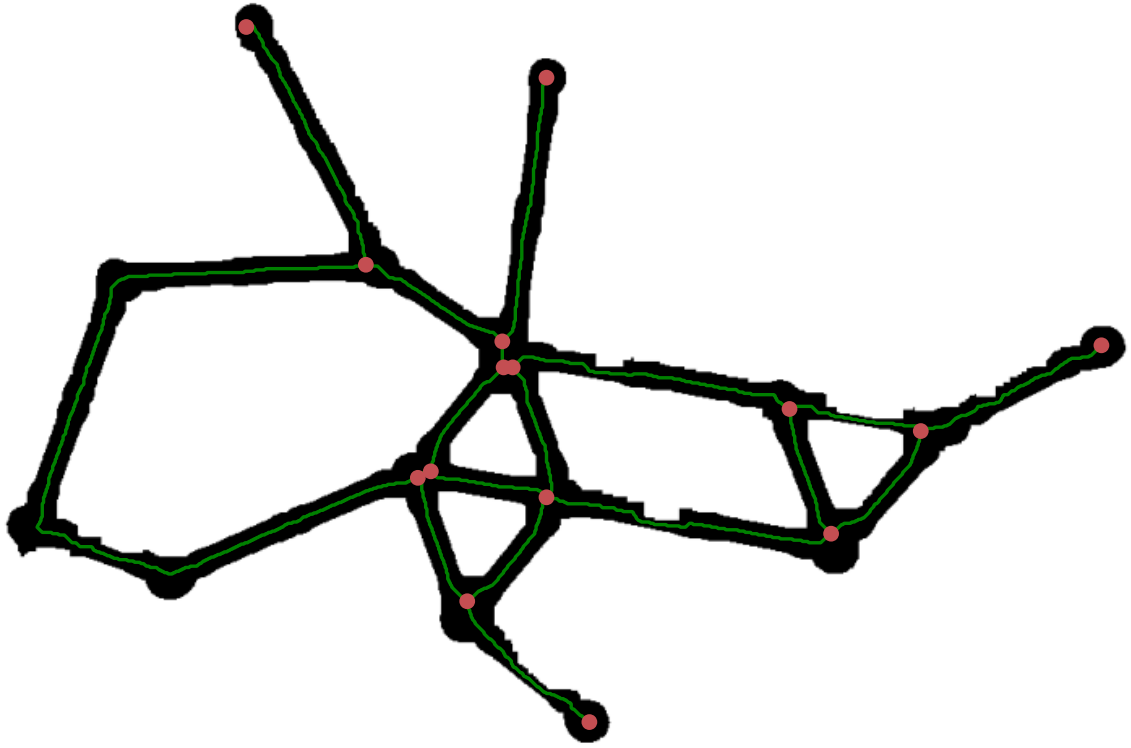


Рис. 5. Вершины скелета (изображены красными точками)

вершиной те несколько вершин, которые отстоят недалеко друг от друга, т.е. те, что находятся внутри некоторого круга небольшого радиуса. Предположим, что в таком круге находятся вершины v_1, \dots, v_n . Тогда степень полученной обобщенной (склеенной) вершины v будет вычисляться по формуле

$$\deg(v) = \sum_{i=1}^n \deg(v_i) - 2(n - 1).$$

Результат склейки вершин изображен на рис. 6.

После этого находится вектор (n_1, \dots, n_k) , где n_i - количество обобщенных вершин степени i у данного скелета. Так для скелета на рис. 6 соответствующий вектор будет равен $(4, 0, 5, 2, 1)$. Назовем полученный вектор вектором степеней вершин.

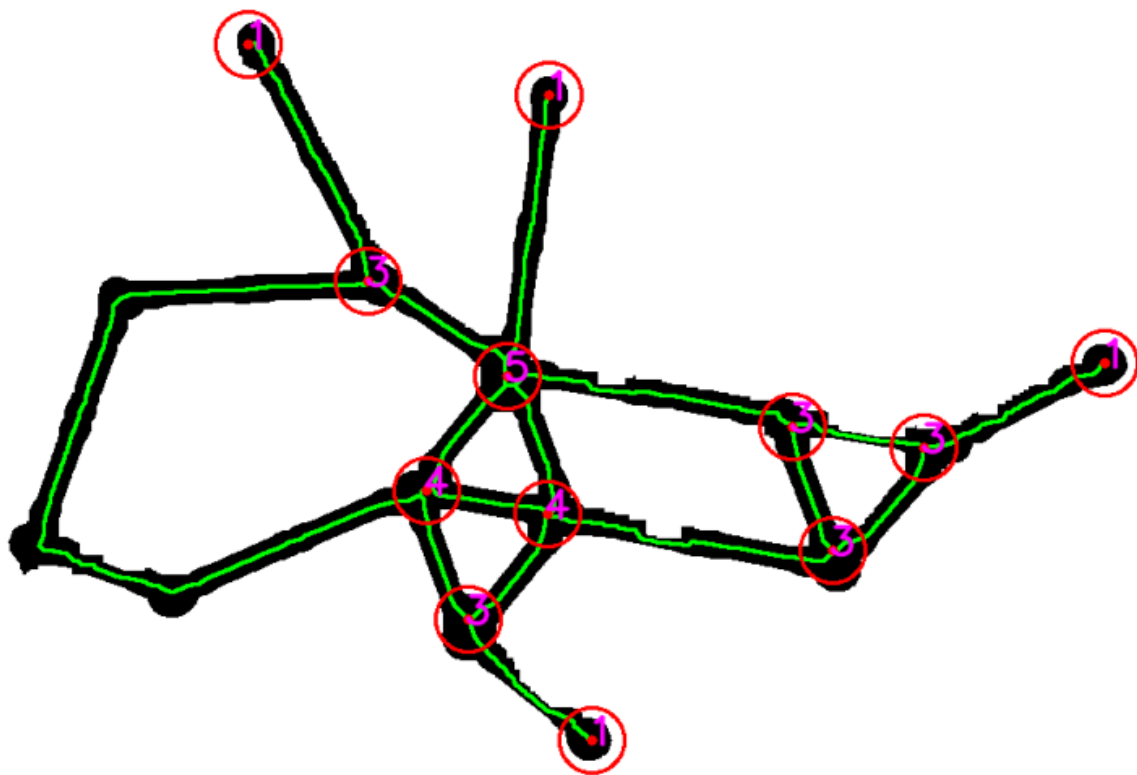


Рис. 6. Вершины скелета после склейки (изображены красными окружностями, внутри которых подписаны их степени)

Таким образом, для обучающих изображений одного класса получился один и тот же вектор степеней вершин:

- для класса I: $(3, 1, 3, 3)$;
- для класса II: $(4, 0, 4, 1)$;
- для класса III: $(4, 0, 5, 2, 1)$;
- для класса IV: $(6, 0, 4, 2)$.

Итоговый классификатор считает евклидово расстояние от вектора степеней вершин скелета поданного на вход изображения до эталонных векторов степеней вершин и относит изображение к классу, соответствующему наименьшему расстоянию. При этом если векторы, между которыми нужно посчитать расстояние, имеют разную длину, то меньшему вектору дописывается справа нужное количество нулей.

4 Описание программой реализации

Программа реализована на языке `python`. Для создания графического интерфейса использовалась библиотека `PyQt5`, для работы с изображениями - библиотеки `OpenCV` и `scikit-image`. Скелеты строились с помощью функции `skimage.morphology.skeletonize`, а стрижка скелета производилась с помощью библиотеки `DSE-skeleton-pruning` [6]. Для работы с графами использовалась библиотека `NetworkX`.

В файле `Readme.txt` описано, как запустить программу. После её запуска нужно нажать кнопку “Загрузить изображение” и в выпадающем окне выбрать изображение с деталями магнитной игры-головоломки, после чего оно отобразится в программе.

В программе можно последовательно выполнить шаги, описанные в пункте “Описание метода решения”, причем промежуточные результаты будут появляться на экране. Параметры по умолчанию подобраны достаточно хорошо для работы с набором обучающих изображений (*Intermediate*), но при желании можно отрегулировать параметры алгоритма. Пример работы программы изображен на рис. 7. В левом нижнем углу программа отображает полученный вектор степеней вершин скелета и результат классификации.

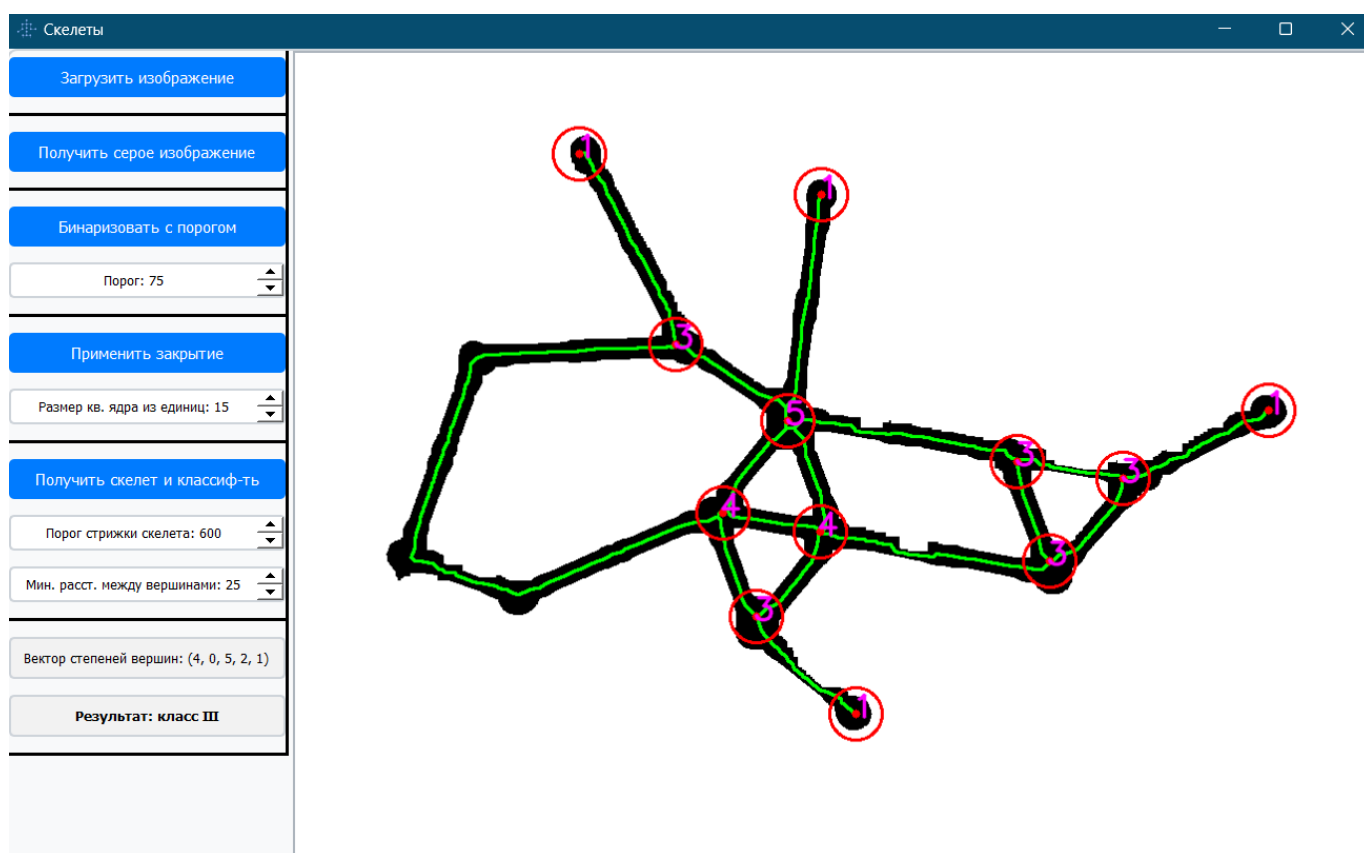


Рис. 7. Пример работы программы

5 Эксперименты

5.1 Класс I

Результат работы программы на обучающих изображениях класса I можно посмотреть на рисунках 8 и 9. Программа дает один и тот же вектор $(3, 1, 3, 3)$ степеней вершин и правильно классифицирует эти изображения.

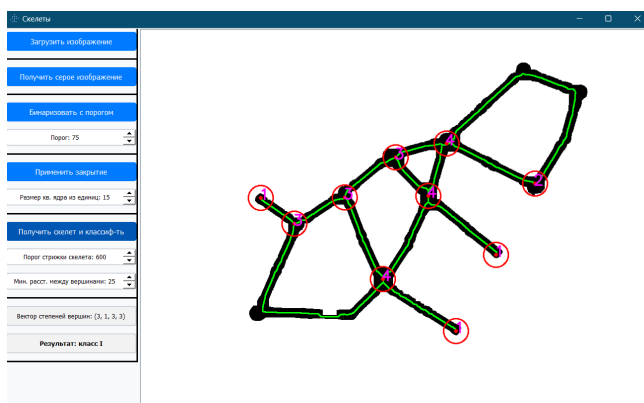


Рис. 8. Результат работы на 2.jpg

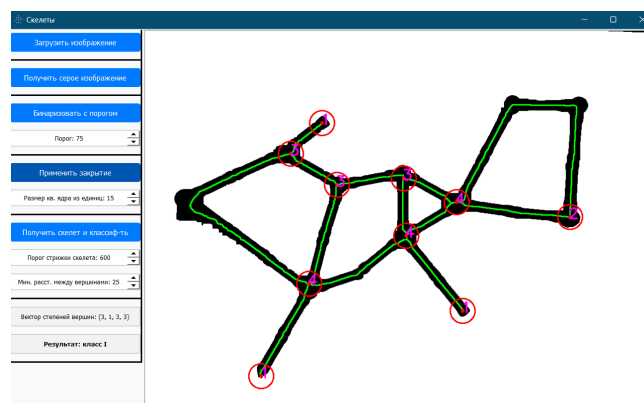


Рис. 9. Результат работы на 13.jpg

5.2 Класс II

Результат работы программы на обучающих изображениях класса II можно посмотреть на рисунках 10 и 11. Программа дает один и тот же вектор $(4, 0, 4, 1)$ степеней вершин и правильно классифицирует эти изображения.

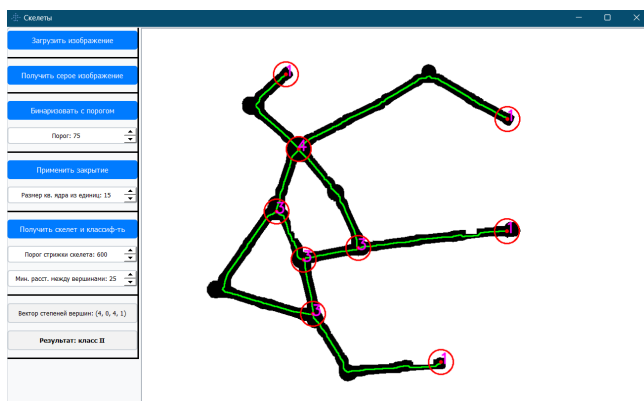


Рис. 10. Результат работы на 3.jpg

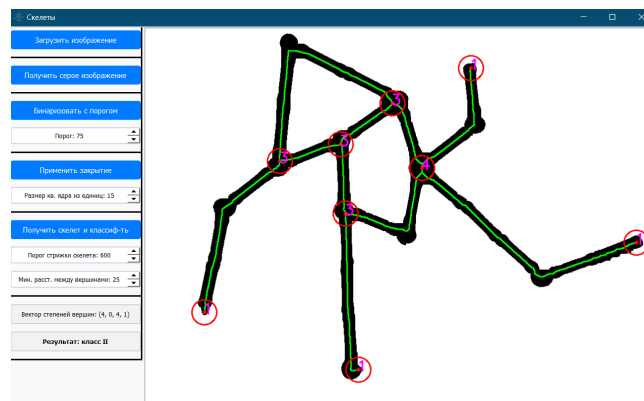


Рис. 11. Результат работы на 25.jpg

5.3 Класс III

Результат работы программы на обучающих изображениях класса III можно посмотреть на рисунках 12 и 13. Программа дает один и тот же вектор $(4, 0, 5, 2, 1)$ степеней вершин и правильно классифицирует эти изображения.

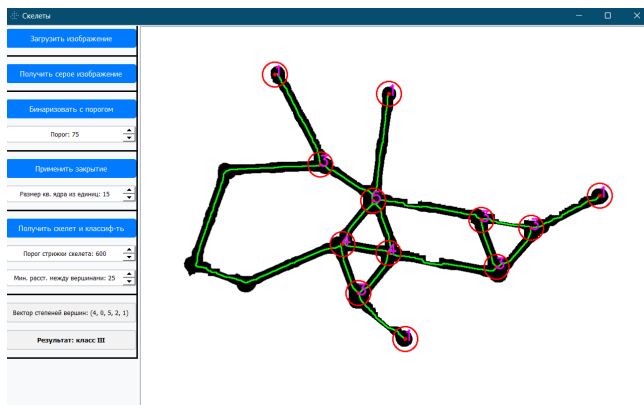


Рис. 12. Результат работы на 4.jpg

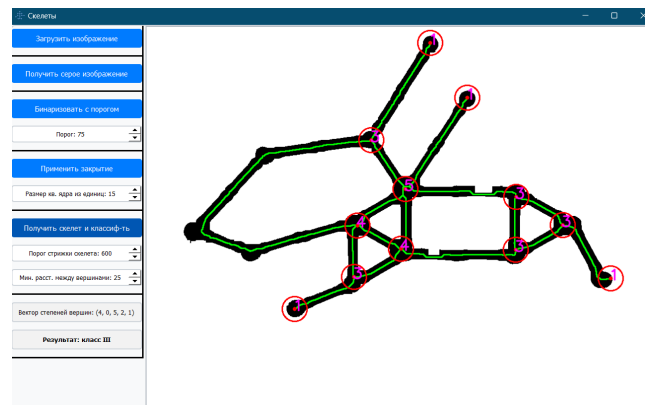


Рис. 13. Результат работы на 19.jpg

5.4 Класс IV

Результат работы программы на обучающих изображениях класса IV можно посмотреть на рисунках 14 и 15. Программа дает один и тот же вектор $(6, 0, 4, 2)$ степеней вершин и правильно классифицирует эти изображения.

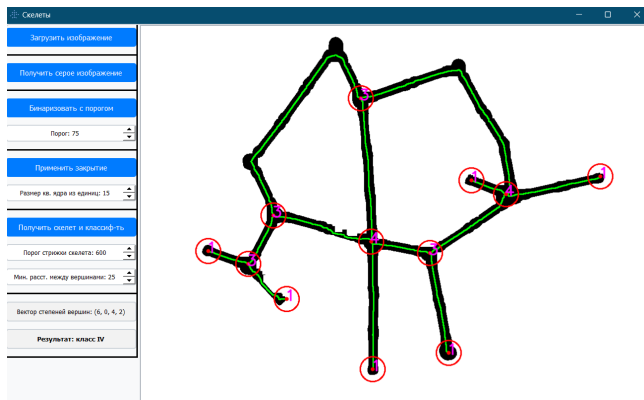


Рис. 14. Результат работы на 5.jpg

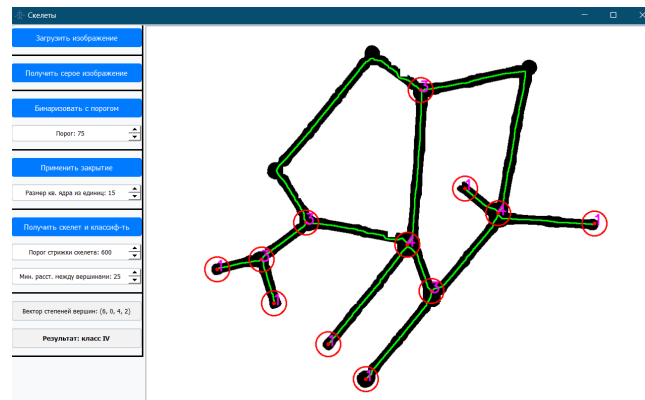


Рис. 15. Результат работы на 7.jpg

5.5 Расстояния между векторами степеней вершин

Квадраты расстояний между векторами степеней вершин скелетов обучающих изображений приведены на рис. 16. Видно, что расстояния внутри одного класса равны нулю, а расстояния между векторами, соответствующими разным классам, больше нуля.

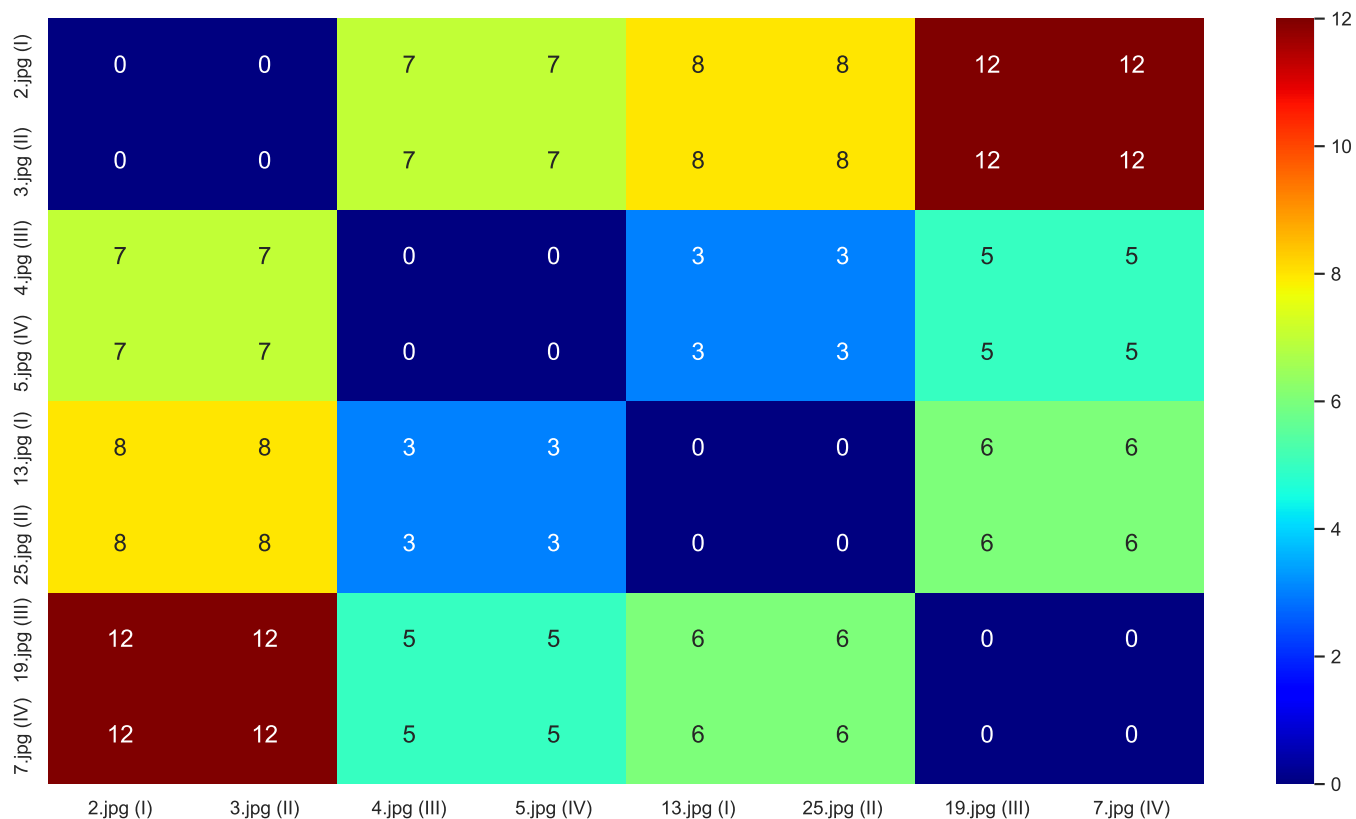


Рис. 16. Квадраты расстояний между векторами степеней вершин скелетов

6 Выводы

Итак, была реализована программа с графическим интерфейсом для классификации изображений моделей графов, построенных из магнитной головоломки.

Эта программа правильно классифицирует все обучающие изображения, соответствующие уровню Intermediate с помощью построения скелета.

Таким образом, была решена поставленная задача.

Список литературы

- [1] [Видеозаписи лекций Местецкого Л.М.](#)
- [2] Местецкий Л.М. Непрерывная морфология бинарных изображений: фигуры, скелеты, циркуляры. – М.: ФИЗМАТЛИТ, 2009.
- [3] Гонзалес Р., Вудс Р. Цифровая обработка изображений. М., Техносфера, 2006.
- [4] T. Y. Zhang, C. Y. Suen. A fast parallel algorithm for thinning digital patterns. Communications of the ACM, March 1984, Volume 27, Number 3.
- [5] [Discrete skeleton evolution \(DSE\). Wikipedia.](#)
- [6] [DSE-skeleton-pruning. Github.](#)