

Лабораторная работа №1

Янаков Дмитрий, 317 группа

31 марта, 2022

1 Постановка задачи

Требуется разработать и реализовать программу для работы с изображениями карточек игрового набора "Геометрика". Этапы работы программы:

1. Считывание и вывод (по желанию) на экран входных изображений;
2. Обработка изображений. В неё входят:
 - (а) Подсчёт количества карт;
 - (б) Выделение фигур на карточках;
 - (в) Определение типа фигуры - многоугольник или фигура с гладкой границей;
 - (г) Для многоугольника определяется количество вершин, а также является ли он выпуклым.
3. Вывод на экран изображений с результатом обработки.

2 Описание данных

Входом программы являются изображения с карточками в формате JPG. Примеры таковых изображены на [рис. 1](#):



Рис. 1: Примеры входных изображений

Карточки могут быть как изолированны, так и наложены друг на друга. Фон для изолированных карточек - либо белый, либо пестрый; для наложенных - только белый.

3 Метод решения

Решение разделено на два этапа. Каждый из них представляет собой некоторую последовательность действий: первый выделяет на изображении фигуры, а второй - определяет их свойства.

Разберём работу каждого из этапов на примере [рис. 1а](#).

3.1 Первый этап

Для начала изображение переводится в полутонаовый вид ([рис. 2a](#)), чтобы впоследствии с ним было удобнее работать. Далее, к нему применяется оператор Кэнни для выделения всех границ ([рис. 2b](#)).

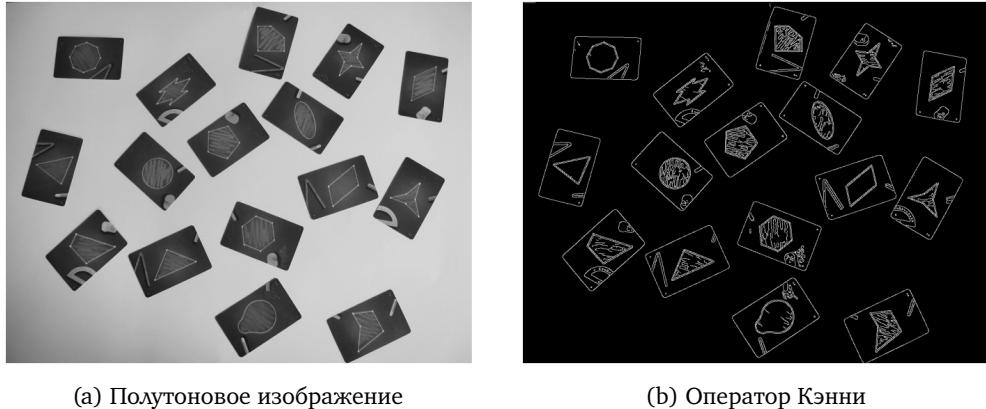


Рис. 2

Можно заметить, что на карточках помимо фигурок, есть дополнительные предметы: ластики, линейки и т.д. Для того, чтобы их выделить, переведем исходное изображение в HSV-пространство ([рис. 3a](#)) и применим к нему фильтрацию (то есть оставим цвета, которые удовлетворяют некоторому заданному диапазону). Результат показан на [рис. 3b](#).

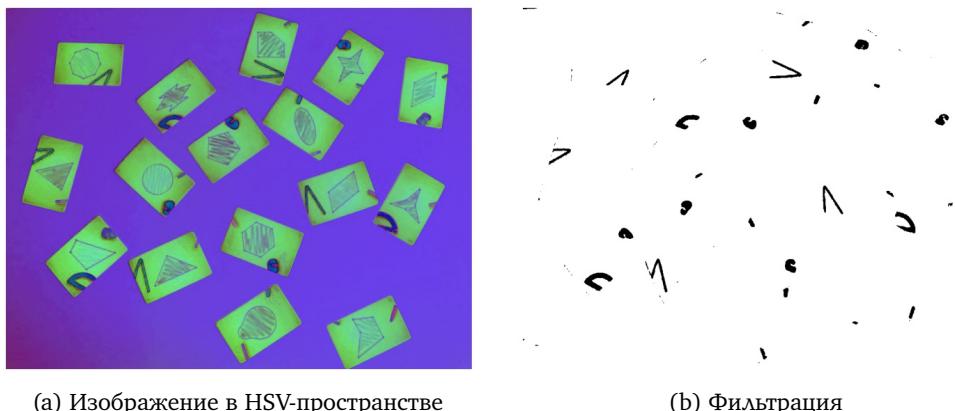
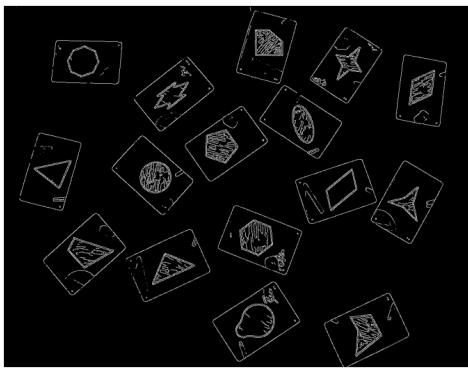


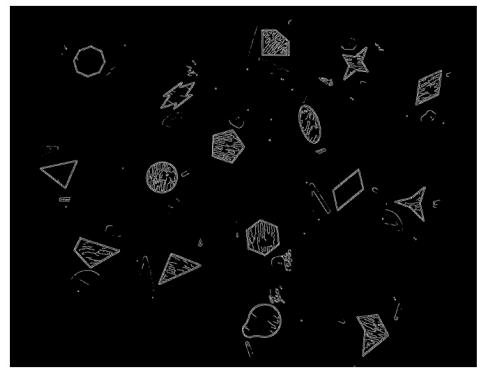
Рис. 3

Удалив найденные предметы ([рис. 4a](#)), избавимся также от контуров самих карточек. Для этого необходимо будет выделить почти всю их площадь (обрезав немного по краям) и применить полученную маску к изображению, которое будет получено после удаления отфильтрованных предметов. Тем самым, останется только внутренность карточек.

Чтобы выделить карточки, применяется следующая последовательность морфологических операций: дилатация и замыкание с диском в качестве примитива (для того, чтобы обеспечить полный контур у всех карточек). Далее идет заполнение всех внутренностей полученных границ, а затем применяется эрозия (также с диском в качестве примитива), чтобы убрать "склейки" карточек и немного уменьшить их площадь. В результате получаем [рис. 4b](#).



(a) После удаления лишних предметов

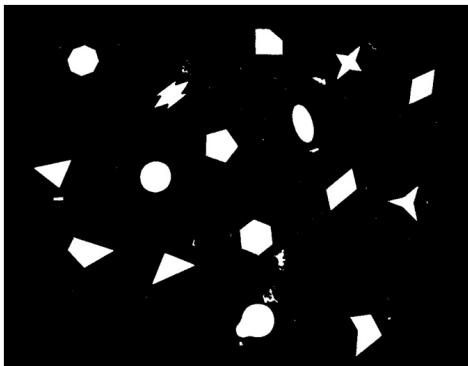


(b) После удаления контуров карточек

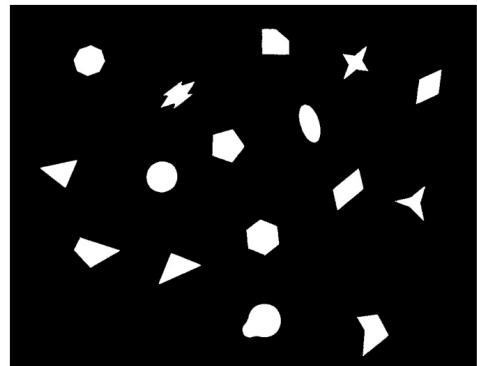
Рис. 4

Далее, к полученному изображению, применяется аналогичная последовательность операций (разве что меняется радиус диска), за исключением замыкания. В итоге остаются фигуры и некоторый шум, который не получилось убрать при фильтрации ([рис. 5a](#)). Для того чтобы избавиться от него, удалим из изображения объекты, чья площадь меньше заданной фиксированной величины. Причем эта величина достаточна велика, чтобы классифицировать именно фигурки.

Таким образом, мы полностью выделили фигуры на изображении и первый этап работы программы завершён ([рис. 5b](#)).



(a) Фигурки с оставшимся шумом



(b) Фигурки без шума

Рис. 5

3.2 Второй этап

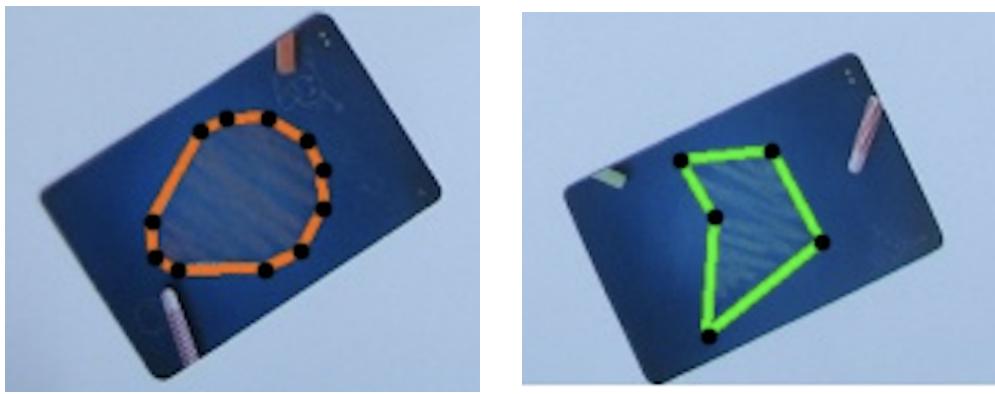
На втором этапе мы приближаем фигуры многоугольниками и с помощью этого определяем гладкость, выпуклость и количество вершин.

Чтобы определить гладкость, приблизим каждую фигуру на изображении многоугольником, максимально возможное расстояние которого до заданной фигуры равно 1% от периметра исходной фигуры ([рис. 6a](#)). В таком случае количество определяемых вершин будет больше, что позволит задать дополнительные условия для выявления гладкости, так как на гладких фигурах будет определяться больше точек.

Для определения точного количества вершин, мы ослабим ограничение до 2%. Это позволит лучше аппроксимировать фигуры ([рис. 6b](#)).

И, наконец, выпуклость определяется следующим образом - по полученным вершинам строится выпуклая оболочка: если она совпадает с исходной фигурой, то фигура выпуклая, иначе невыпуклая.

Таким образом, мы сегментируем все фигурки на изображении и определяем их свойства.



(a) Приближение гладкой фигуры

(b) Приближение многоугольника

Рис. 6

4 Программная реализация

Программа реализована на языке программирования Python с использованием Jupyter Notebook. Для корректной работы необходимо разместить файл `geometrika.ipynb` и папку `Samples` с тестовыми изображениями в одной директории.

Библиотеки, использованные при реализации:

- numpy для матричных вычислений;
- openCV для работы с изображениями;
- scikit-image и scipy для морфологических операций над изображениями;
- matplotlib для работы с графиками.

Чтобы получить результат, необходимо запустить все ячейки в файле. Изображения с нанесенной на них разметкой будут находиться в той же директории с названием `output_for_НАЗВАНИЕ_ВХОДНОГО_ФАЙЛА`, однако результат можно также посмотреть в ноутбуке (рис. 7).

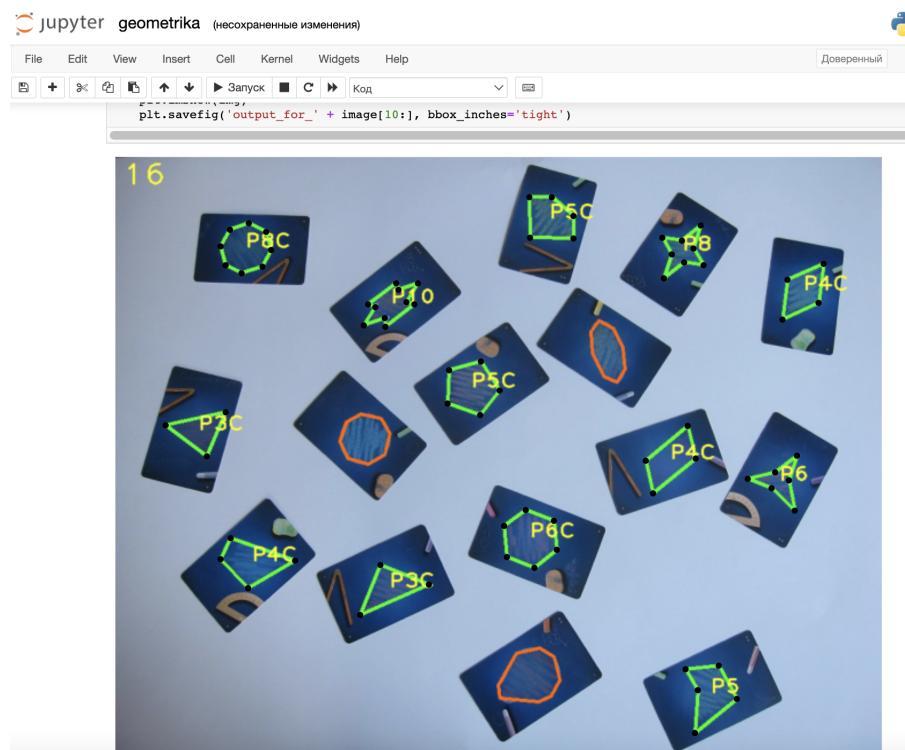


Рис. 7: Скриншот результата работы программы

Отметим, как интерпретировать вывод:

- в левом верхнем углу полученного изображения отображено общее количество карточек;
- гладкие фигуры обведены оранжевым цветом;
- многоугольники обведены зелёным цветом (об этом также свидетельствует буква "Р" на фигуре);
- число идущее после буквы "Р" означает количество вершин в многоугольнике, а буква "С" (если она присутствует) означает, что многоугольник выпуклый.

5 Эксперименты

Попробуем применить алгоритм к изображению на [рис. 1b](#). Результат показан на [рис. 8](#).

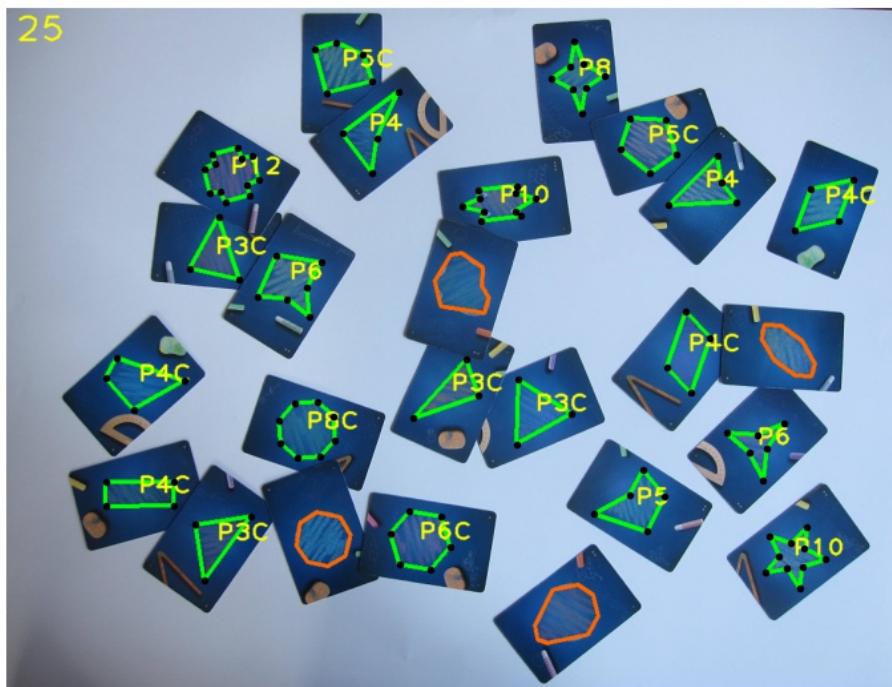


Рис. 8: Результат работы программы на IMG_4

Как видно, алгоритм корректно подсчитал количество карточек, сегментировал фигуры, а также определил их свойства.

Однако есть внешние факторы, которые могут повлиять на результат работы. Такими являются: чёткость изображения, засветченность, а также слабое освещение. Примеры неверных сегментаций в таких случаях показаны на [рис. 9](#).

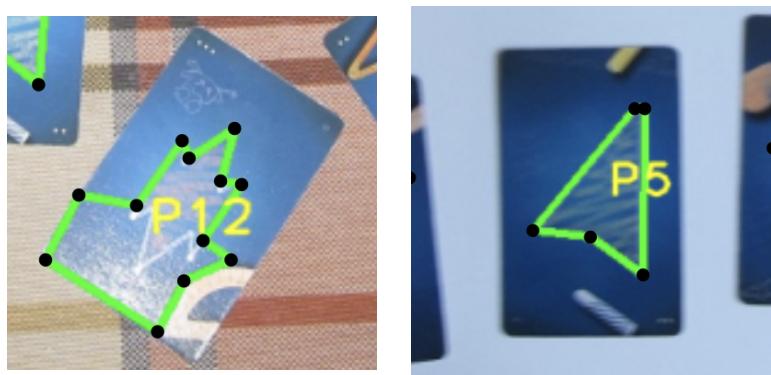


Рис. 9: Примеры неверных сегментаций

6 Выводы

Итак, по результатам данной лабораторной работы была получена программа для сегментации фигурок на карточках игрового набора "Геометрика". Подход, использованный при реализации соответствует обучению без учителя, поскольку подбор гиперпараметров основывался только на предоставленных изображениях. С одной стороны это позволяет не тратить время на процедуру обучения, но с другой стороны, чтобы получить высокое качество сегментации, может потребоваться заново настроить гиперпараметры.