

МГУ им.Ломоносова
Факультет ВМК кафедра ММП

ЛАБОРАТОРНАЯ РАБОТА №1

Изучение и освоение методов обработки и сегментации изображений
в рамках курса "Обработка и распознавание изображений"

Батшева Анастасия 317 группа

Москва
2021

Содержание

1	Формулировка задания	2
2	Входные данные	2
3	Описание метода и экспериментальная часть	2
3.1	Сегментация	2
3.2	Классификация	6
4	Итоги	8
5	Программная реализация	9
5.1	Выводы	9

1 Формулировка задания

Формулировка задания для класса **Expert**:

Реализовать программу для работы с набором игровых фишек "Тримино". На вход даётся 4 изображения фишек тримино на неоднородном фоне. Необходимо сегментировать и классифицировать фишки.

2 Входные данные

Фотографии в формате bmp размером (,):



3 Описание метода и экспериментальная часть

Поскольку решаются две задачи: **сегментация** и **классификация**, то методы описываются последовательно.

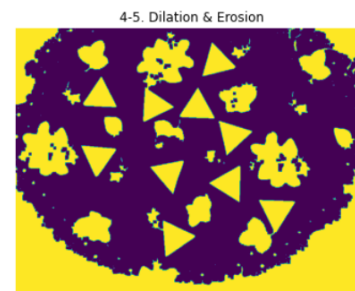
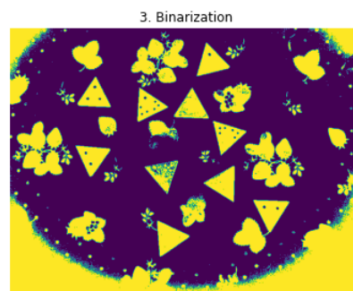
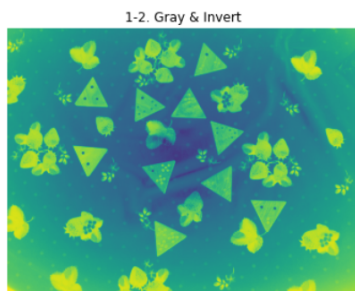
3.1 Сегментация

Сначала вычисляются правильные маски для двух изображений из исходных (по одному из каждой пары, на пестром фоне их две пары) на основе чисто математических методов.

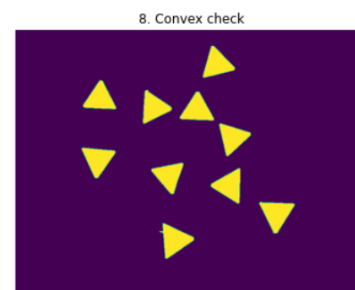
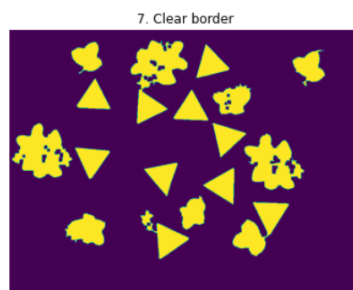
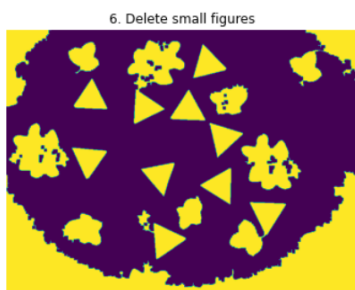
Заметим, что искомые триминошки - это треугольники. На это и будем опираться. Треугольник - хорошая фигура, так как она является выпуклой. Стало быть, разница между выпуклой оболочкой и фигурой должна быть 0.

Вычисление масок:

1. Перевод в полутоновое изображение
2. Вычисление порогового значения по Оцу для последующей бинаризации
3. Бинаризация
4. Dilation для заполнения внутренних разрывов и проколов
5. Erosion для возврата правильных форм



6. Отбор фигур площадью меньше 2500 - экспериментально вычисленная площадь триминошек
7. Очистка от краевых шумов, связанных с падением освещенности
8. Поэлементная проверка на соответствие выпуклой оболочке
9. Исключение неподходящих областей



Параметры методов подбираются экспериментально. На выходе мы должны иметь корректную маску для одного или более изображений.

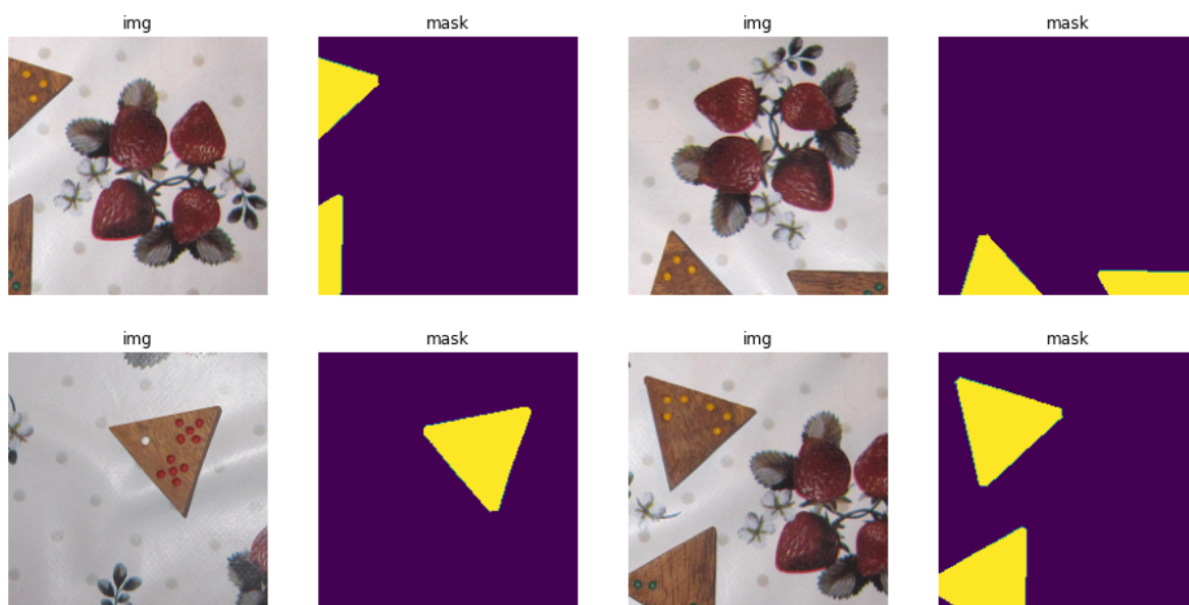
После этого создаётся датасет для нейросети посредством аугментации.

Тренировка нейросети:

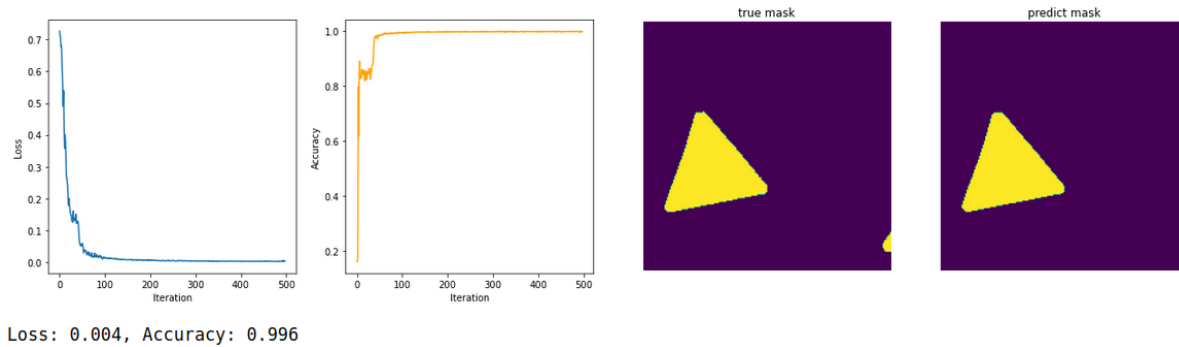
1. Исходные изображение и маска случайно разбиваются на n квадратных "вырезок" размерами 200×200
2. Каждый квадрат испытывает поворот на 0, 90, 180, 270 градусов
3. Результат так же изменяется по яркости
4. Результат инвертируется

В результате на выходе имеем датасет около тысячи пар (изображение-маска). После этого на датасете обучается **UNet**, собранный из блоков предобученного **vgg13**.

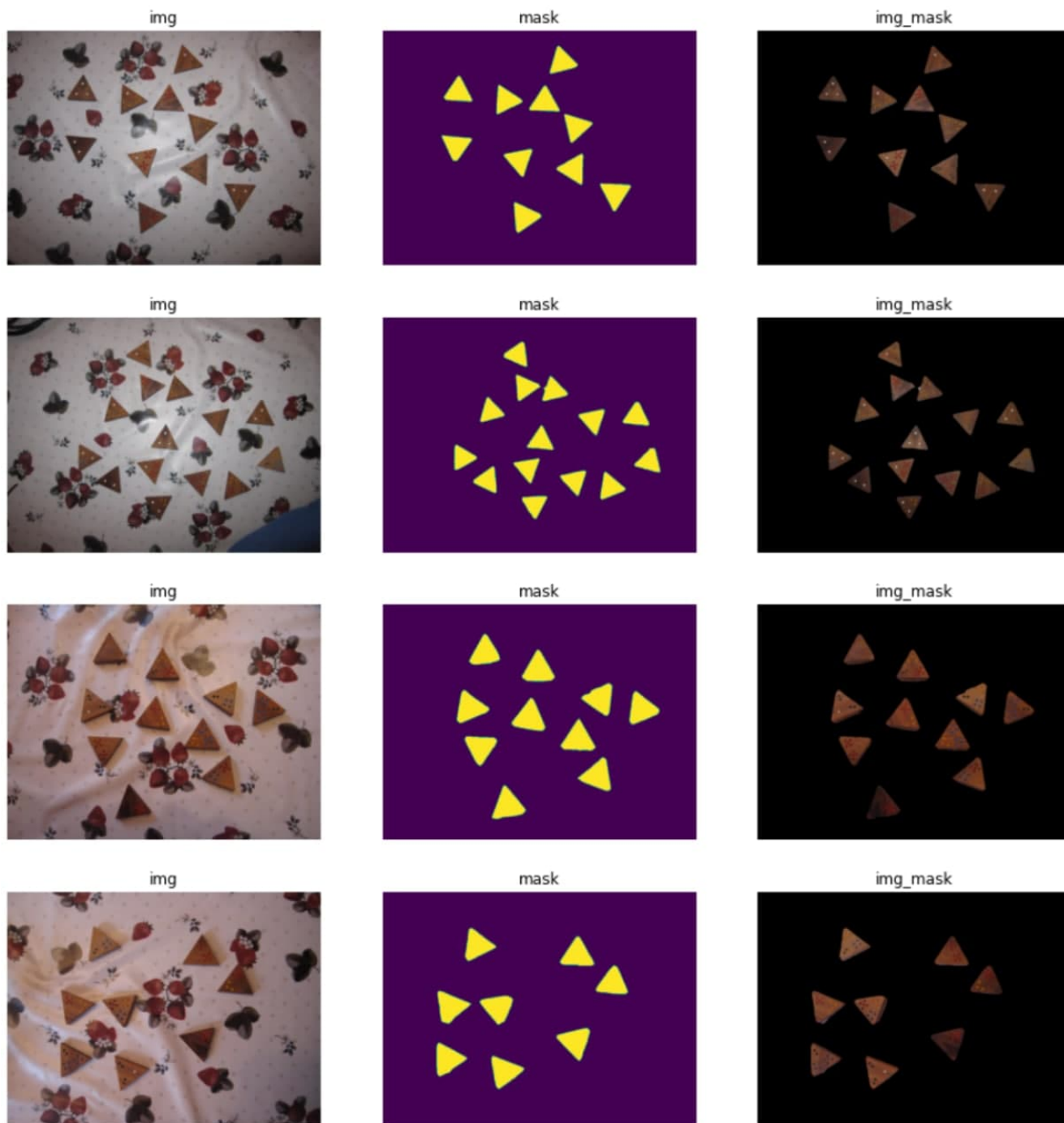
Примеры предикатов:



Процесс обучения сетки:



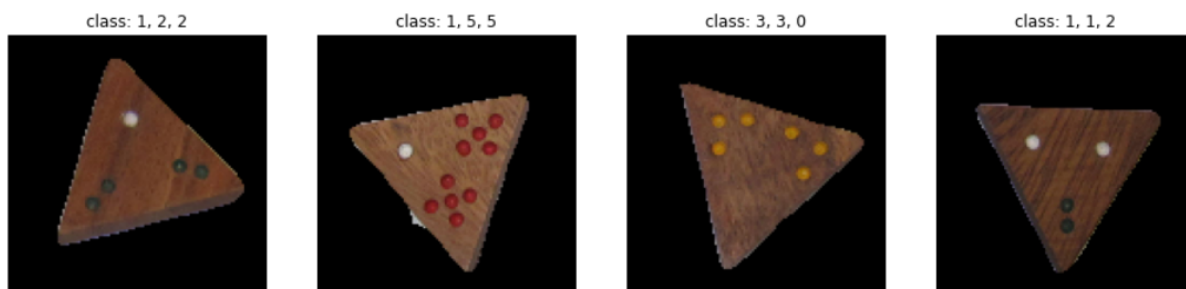
Итоги нейросетевой сегментации:



Видно, что результат соответствует ожиданиям и дает довольно высокое качество. Проблемы, возникающие с сетью связаны с очень маленьким числом данных, так что между плохим результатом сети и переобучением я сделала ставку на второе, учитывая, что степень обобщения всё равно достаточная для хороших результатов на произвольных картинках.

3.2 Классификация

После сегментации для каждой триминошки вычисляется класс:



Разумеется, здесь, как и в предыдущем пункте, можно воспользоваться нейросетями, однако, в этот раз не избежать ручной разметки, ибо все доступные способы мною были перепробованы и, из-за сложной структуры домино (дерево, бороздки, полоски, точки цвета почти как дерево - все это усложняло и не давало вычислить хотя бы маску для точек). Но ручная классификация меня не устроила. Вдобавок, этот курс подразумевает скорее математические методы анализа изображений. И в следующей секции объясняется БЕЗсетевой метод определения класса.

Идентификатор класса - число точек у каждого угла триминошки. Поэтому сначала для каждой фишки определяется три области вблизи углов: окружность с радиусом, равным расстоянию от вершины угла до центра треугольника (что то же самое, что радиус описанной окружности). Геометрически это являет собой как раз нужную область - регион возле угла площадью одна треть от исходной. Согласно логике расположения точек, такая область целиком содержит точки одного угла и идентифицирует 1 из 3 классов фигуры.

Выделение областей классификации:

1. Определение центра треугольника
2. Вычисление расстояния каждой точки треугольника до центра с помощью градиента распределения расстояний у круга соответствующего размера
3. Поиск самых удалённых (и различных) областей от центра - углов
4. Вычисление координат углов
5. Вычисление нужного радиуса для охватывающей окружности
6. Уточнение захватывающей области: берется сектор окружности с углом 60 градусов и нужным направлением к центру тримино (так исчезают тени и куски скатерти, случайно захваченные маской)
7. Определение области "вблизи угла"
8. Разбиение исходной фигуры на три области вблизи каждого угла



Заметим, что все точки имеют расцветку, соответствующую классу, чем мы и будем пользоваться в дальнейшем. Это означает, что на определенном канале (в зависимости от цвета) точки будут видны хорошо. Например, на изображении видны яркие желтые точки и темные синие:

Определение класса для каждой области

1. Разделение по каналам
2. Снижение контраста фона за счет приведения к значению медианы цвета деревяшки (так мы весь фон заменяем на средний цвет дерева)
3. Бинаризация по цвету фона плюс некоторый порог для отсева неравномерности фона
4. Сужение рассматриваемой области за счет действия маски "область вокруг точек"
5. Отсев соли и перца медианным фильтром
6. Поиск точек-кружков методом `skimage.blobs_log` с нужным порогом



И вот итоговая классификация для 1 фотографии. Остальные в ноутбуке.



4 Итоги

Как видно из примеров, качество методов сегментации и классификации очень высоко. Во-первых, даже безнейронные методы могут показывать и показывают отличный результат для выделения необходимых фигур даже из пестрого фона просто на основе свойств фигур и фона.

Во-вторых, Unet, отлично справляется с детекцией фигур даже в тех местах, где чисто математические методы не справились (там, где триминошки сильно заезают на клубнику на скатерти). Но сеть привязана к объему выборки, что, в нашем случае мало выполнимо. Поэтому, будь выборка больше, качество было бы выше.

Сеть в задачах сегментации, на мой взгляд, необходима, ибо без нее каждый метод необходимо подстраивать по параметрам под каждую картинку в частности. В поисках же более общего решения не остается других путей, кроме как использовать сети.

5 Программная реализация

Архив содержит 3 библиотеки, 1 ноутбук и сет картинок.

Библиотеки:

1. **utils** - содержит функции из блока "**Сегментация**":
 - (a) skimage-методы обработки изображений: dilation, erosion, opening, бинаризация и др.
 - (b) Генерация датасета на основе вычисленных масок
 - (c) Классы для адаптации под тип входных данных сети: Dataset, ToTensor, Normalize
2. **net** - содержит реализацию сети UNet
3. **netprocess** - содержит функции процессов обучения и постобработки из блоков

5.1 Выводы

В данном отчете представлены результаты лабораторной работы № 1. Взяты сложные данные класса **Expert**, обработка производилась с помощью как нейронных, так и чисто математических методов. Было задействовано множество функций библиотек skimage и pytorch, а так же функций собственного производства. Учитывая многогранность и сложность подхода, качество результата и отчета, искренне надеюсь на дополнительные баллы.