

Отчет по лабораторной работе на тему "Изучение и освоение методов обработки и сегментации изображений"

Висков Василий, группа 317

Апрель 2020

Постановка задачи

Ставится задача сегментации композиции геометрических фигур (треугольников и кругов разных цветов) при неоднородном освещении на неровной поверхности. Целью работы является получение алгоритма кодировки треугольников с помощью координат центра и числа объектов в трех кластерах кругов, находящихся около углов.

Описание данных

Данные представляют из себя три 8-bit RGB-фотографии деревянных фишек игрового набора Тримино, две из них выполнены с использованием вспышки, одна - без нее. Идентификация каждой фишке осуществляется за счет кругов, расположенных одноцветными группами различных мощностей в каждом из углов. Фишки на изображениях таковы, что не найдется ни одной пары треугольников, в углах которых расположены попарно равномощные группы, что дает возможность рассматривать кодировку фишек без учета цвета кругов.

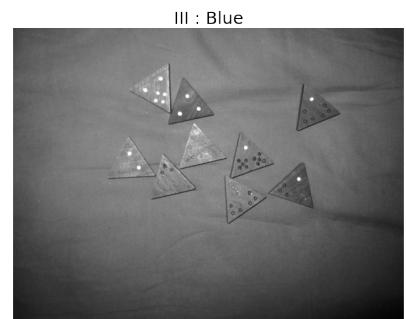
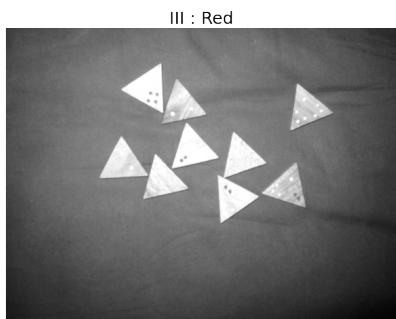
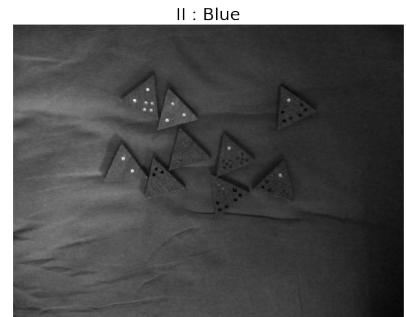
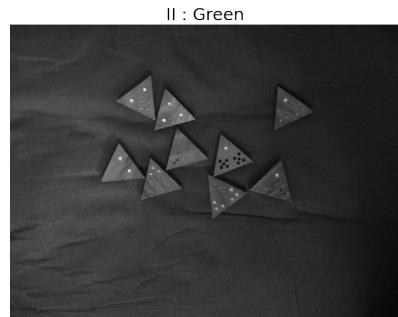
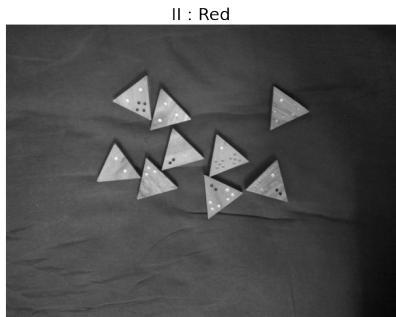
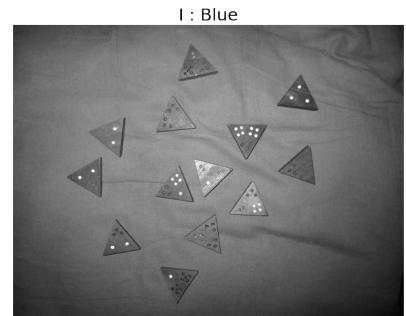
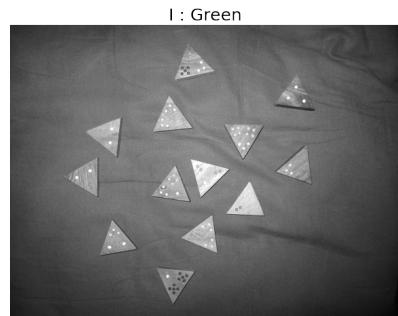
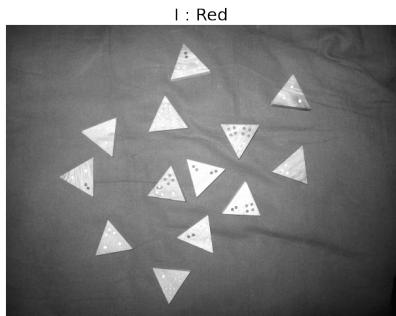


Рис. 1: RGB-слои картинок

Метод решения

Задача решается классическими методами распознавания образов, решение строится на применении предобработки изображений и использовании точечных, пространственных, алгебраических и морфологических операций.

Первым шагом решения является применение CLAHE (Contrast Limited Adaptive Histogram Equalization) [1]. В дальнейшем для сегментирования кругов на изображении будет применяться медианное сглаживание с окном размера 5x5.

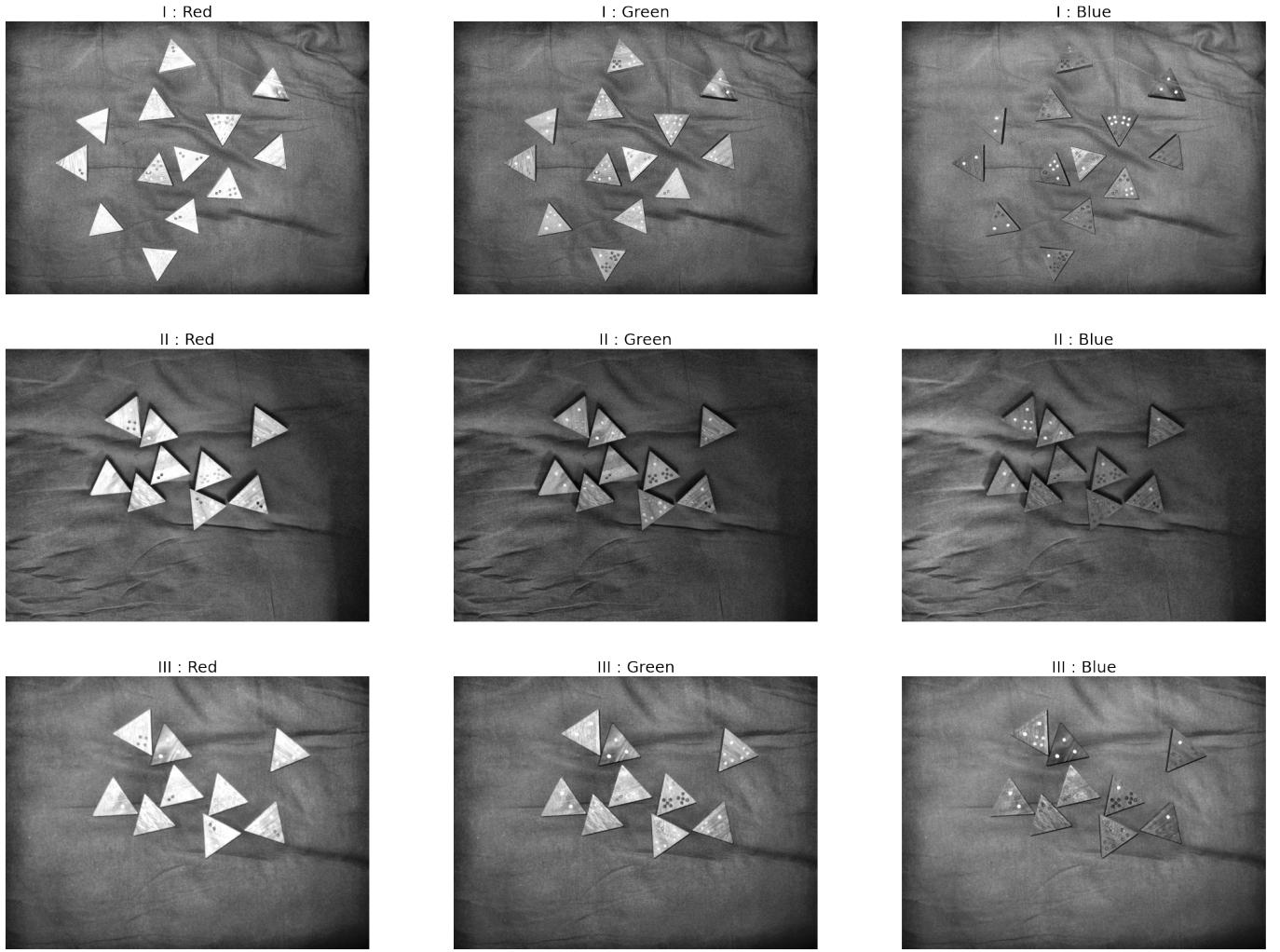


Рис. 2: Применение CLAHE к исходным изображениям

Вторым шагом решения является сегментирование фишек и получение бинарной маски размера исходного изображения, содержащей единицы в тех позициях пикселей, где изображен треугольник. Маска строится с использованием бинаризации разности слоев R, G и бинаризации разности R, B.

Порог бинаризации p подбирается с помощью модели двух классов $\{0, 255\}$. Пусть $D_1(p) = \frac{\sum_{k=0}^p k*H(k)}{\sum_{k=0}^p H(k)}$, $D_2(p) = \frac{\sum_{k=p+1}^{255} k*H(k)}{\sum_{k=p+1}^{255} H(k)}$ - средние яркости каждого класса. Решающее правило выглядит следующим образом: $g(x) = 255 * \mathbf{1}[x > \frac{D_1 + D_2}{2}]$. Алгоритм поиска порога заключается в поиске ближайшего значения на дискретной сетке от 0 до 255 к решению уравнения $p = \frac{D_1(p) + D_2(p)}{2}$.

Далее для каждой бинарной маски применяется дилатация с ядром креста 3×3 ¹, алгоритма замыкания площади [2] и последовательной эрозии с повторением:

- с ядром креста 3×3 один раз;
- с ядром пирамиды-«треугольника» 5×7 один раз;
- с ядром пирамиды-«треугольника» 5×5 один раз;
- с ядром квадрата 3×3 5 раз;

После две маски суммируются и обрезаются.

¹Все рассматриваемые ядра можно найти в файле *methods.py*

Третьим шагом является выделение связных компонент на маске, которые соответствуют каждой фишке, и подсчет основных характеристик этих компонент:

- центроиды, которые будут соответствовать центрам фишек;
- площади самих компонент и их выпуклых оболочек;

Величины площадей потребуются нам для фильтрации маски от случайных лишних связных компонент, не соответствующих фишкам. Последним шагом является вырезание всех треугольников для дальнейшей работы с ними по отдельности.

Четвертым шагом является построение масок для выделения кругов на треугольниках: применяется аддитивная модель масок, независимых при построении, но зависимых при сегментировании, так как разные маски могут выделять одни и те же круги. Специфика самих масок основывается на цвете кругов и освещении на изображении. Рассмотрим последовательно каждую маску:

- суммируются разности исходных слоев G, B и R, B, применяется априорный порог 254 для бинаризации, после применяется алгоритм размыкания (эрозия с ядром пирамиды «треугольника» 2x3, дилатация с ядром круга 5x5);
- суммируются разности исходных слоев R, G и CLAHE-слоев R, G, применяется априорный порог 254 для бинаризации, после применяется алгоритм размыкания (эрозия с ядром пирамиды-«треугольника» 2x3, дилатация с ядром круга 5x5);
- к разности исходных слоев G, R применяется гамма-коррекция с параметром 5, применяется бинаризация, где порогом выбирается 50-е максимальное значение интенсивности слоя, после применяется алгоритм размыкания (эрозия с ядром пирамиды-«треугольника» 2x3, дилатация с ядром круга 5x5);
- строится min-max-масштабирование с домножением на 255 (в дальнейшем будем называть просто min-max) произведения исходного слоя G, гамма-коррекции R с параметром 0.1 и гамма-коррекции B с параметром 0.1, после этого произведение бинаризуется с порогом 0, применяется CLAHE-слою B и разности исходных R, G, после строится из первого слоя вычитается второй, к этой разности применяется гамма-коррекция с параметром 2, получившееся изображение бинаризуется с порогом 254, после применяется алгоритм размыкания (эрозия с ядром креста 3x3, дилатация с ядром круга 5x5);
- суммируются гамма-коррекции с параметром 0.5 разностей исходных R, B и R, G и разности CLAHE-слоев R, G, а также CLAHE-слоя R, изображение бинаризуется с порогом 20, картинка инвертируется с обрезкой фона треугольника, после применяется алгоритм размыкания (эрозия с ядром пирамиды-«треугольника» 2x3, дилатация с ядром круга 5x5);
- суммируется разности исходных слоев G, B и R, B, применяется гамма-коррекция с параметром 0.5, изображение бинаризуется с порогом 254, после всем прилегающим (имеют хотя бы один общий угол) к ненулевым пикселям присваивается значение интенсивности 255, после применяется алгоритм размыкания (эрозия с ядром креста 3x3, дилатация с ядром креста 3x3);
- суммируются разность CLAHE-слоев R, B и разность исходных R, G, изображение инвертируется с обрезкой фона, бинаризуется с порогом 254, после применяется алгоритм размыкания (эрозия с ядром креста 3x3, дилатация с ядром креста 3x3);

- разность исходных R, B бинаризуется с порогом 1, инвертируется с обрезкой фона, после применяется алгоритм размыкания (эрозия с ядром креста 3x3, дилатация с ядром круга 5x5);
- строится min-max произведение: суммы медианно сглаженного слоя B и разности R, G, суммы R и разности G, B, суммы G и разности B, R (назовем WM); после строится разность дилатации маски треугольника с ядром круга 5x5 и самой маски, после к этому изображению применяется дилатация с ядром круга 5x5 (назовем deleted, представляет из себя утолщенную обе стороны границу маски); строится min-max сумма границ WM, которые строятся двумя способами:
 - к WM применяется гауссовское сглаживание с ядром размера 7x7 и дисперсией 1, из него вычитается WM;
 - из дилатации WM с ядром креста 3x3, повторенным 2 раза, вычитается WM;

из этой суммы вычитается deleted, изображение бинаризуется с порогом 128, после применяется алгоритм размыкания (эрозия с ядром пирамиды-«треугольника» 2x3, дилатация с ядром круга 7x7);

Аддитивность модели заключается в суммировании всех построенных масок.

Начало **пятого шага** аналогично третьему шагу, только теперь поиск связанных компонент и центров производится локально, для масок с кругами и маски с одним треугольником.

После для полученных сегментов-кругов применяется алгоритм поиска углов Харриса [3], который выдает множество потенциальных углов, среди которых необходимо найти 3 точки, наилучшим образом подходящие на эту роль. Для этого сначала ищутся евклидовы расстояния от всех точек до центра, значения эти сортируются по убыванию, в качестве первого угла выбирается наиболее удаленная от центра точка. Дальше, рассматривая их в порядке приближения к центру, последовательно ищутся такие две точки, которые удалены друг от друга и от уже взятой в качестве угла точки на расстояние, большее порога (в качестве порога взята величина, примерно равная стороне треугольника, вычисляется он динамически по формуле $a = \sqrt{3} * R - 20$, где R - расстояние от центра до точки, взятой первой в качестве угла).

Найдя углы, строится координатная ось с началом координат в центре треугольника, где углам и центрам кругов приписываются радиус-векторы, далее каждый круг относят к тому углу, косинусная близость между радиус-векторами которых максимальна.

В итоге мы получаем центры всех треугольников на картинке и их кодировки в виде триплета, состоящего из величин количества кругов, относящихся к каждому углу.

Описание программной реализации

Программный код разбит на модули:

- process.py - основная часть, содержит реализацию большинства шагов, описанных выше;
- methods.py - содержит все методы преобразования изображений;
- preprocess.py - содержит все методы предобработки изображений;
- classes.py - содержит реализации всех операций над изображениями;
- utils.py - содержит реализации всех вспомогательных функций;

- ViskovIPR1.ipynb - содержит результаты экспериментов и используемые при реализации алгоритмов источники, а также невключенные в основной пул модулей функции преобразования и обработки изображений (функции сверток с операторами Кирша, Превита, Лапласа, которые не показали хороших результатов);

Некоторые алгоритмы были взяты из библиотек cv2 (дилатация, эрозия, предобработка) и skimage (сегментация по маскам, поиск центров и площадей).

Эксперименты

Несколько результатов работы алгоритма на разных этапах приведены ниже.

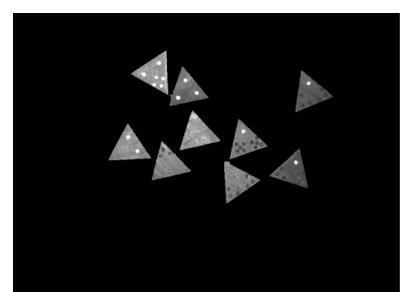
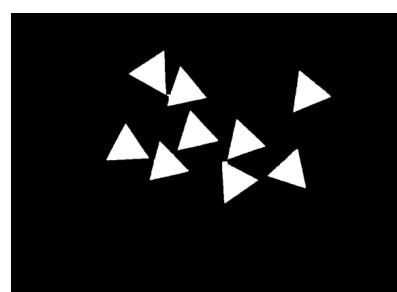
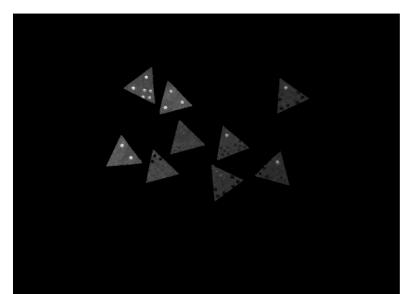
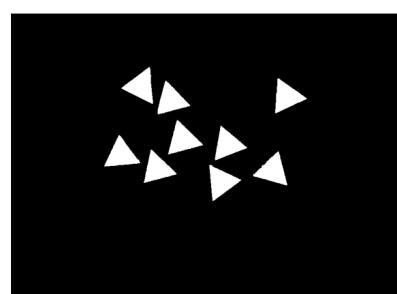
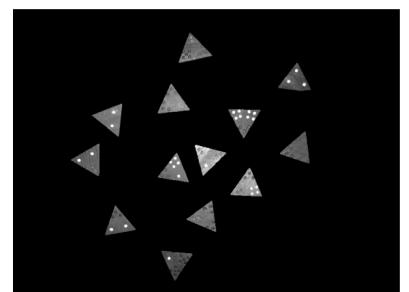
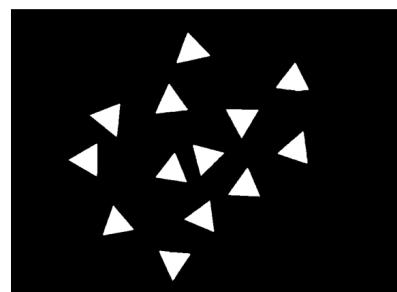


Рис. 3: Результат работы алгоритма сегментации фишек

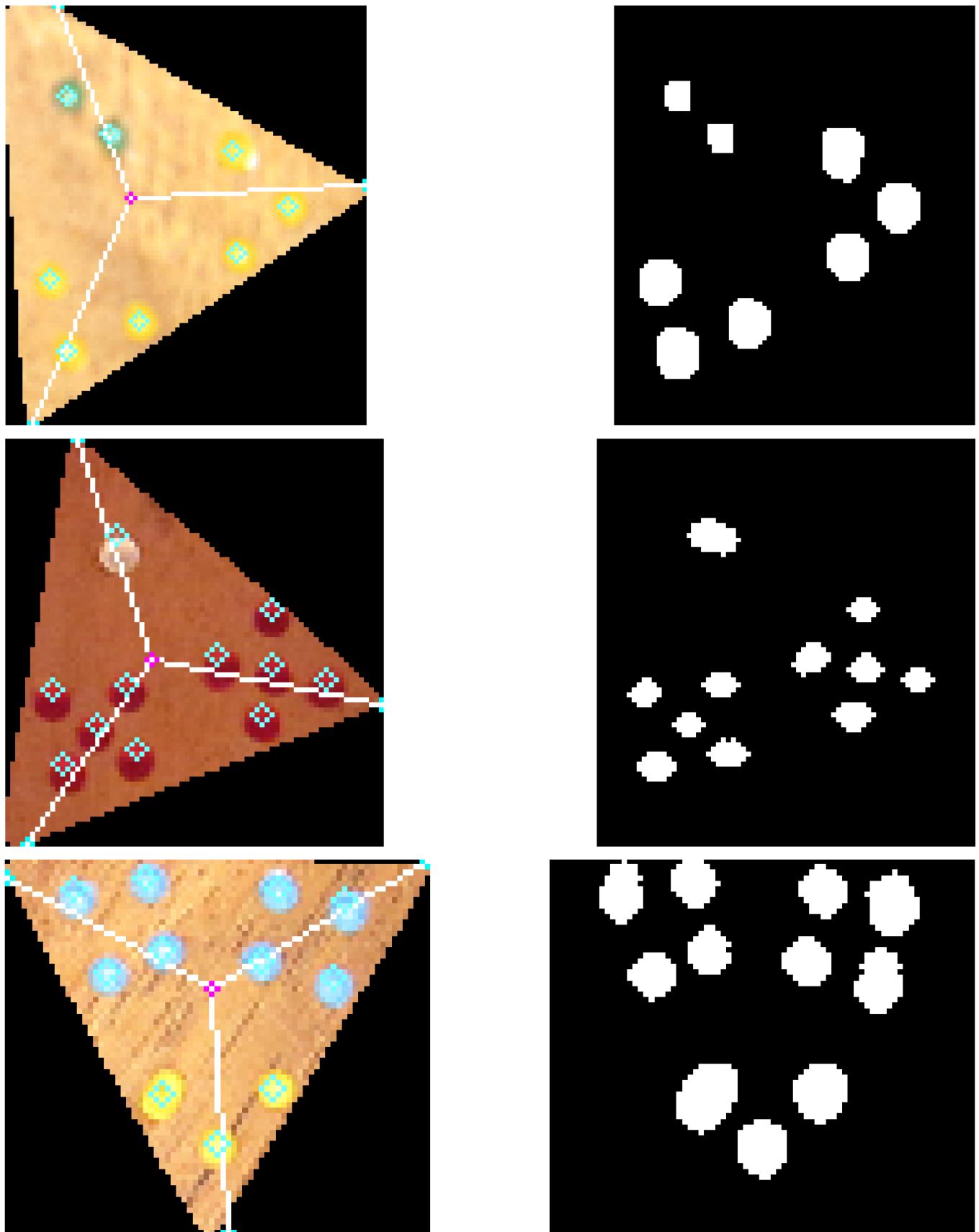


Рис. 4: Результат работы алгоритма сегментации кругов

Как видим, алгоритм сегментации треугольников выдает хорошие маски, позволяющие получить нужные связные компоненты с хорошей степенью точности. На рисунке 4 изображены результаты поиска центров треугольников и кругов, а также проведенные векторы от

центра к углам. В силу специфики выделения треугольников из исходного изображения найденные точки не всегда точно соответствуют углам фигур, но их некоторые аппроксимации позволяют точно решать задачу подсчета косинусной близости радиус-векторов и отнесения кругов к каждому из углов.

Выводы

- классические методы распознавания образов позволяют решить поставленную задачу, выдавая хорошие сегментные маски;
- улучшением предложенной модели может быть учет пересечений масок: при наличии пересечения масок для какого-то круга будет браться пересечение, а не объединение масок;

Источники

Список литературы

- [1] https://docs.opencv.org/master/d5/daf/tutorial_py_histogram_equalization.html
- [2] https://scikit-image.org/docs/dev/api/skimage.morphology.html#skimage.morphology.area_closing
- [3] https://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_feature2d/py_features_harris/py_features_harris.html