

ПЛЕНАРНЫЕ ДОКЛАДЫ

УДК 004.42:004.92

А. В. Ананько, А. И. Калько

Учреждение образования «Барановичский государственный университет», Барановичи, Республика Беларусь

РАСПОЗНАВАНИЕ КОНТУРОВ ОБЪЕКТОВ И ОБРАЗОВ НА ИЗОБРАЖЕНИИ

Введение. Компьютерное зрение (иначе техническое зрение) — теория и технология создания машин, которые могут производить обнаружение, отслеживание и классификацию объектов. Как научная дисциплина, компьютерное зрение относится к теории и технологии создания искусственных систем, которые получают информацию из изображений [1].

Применение технологий компьютерного зрения позволяет создавать интересные и полезные программные решения, способные наглядно демонстрировать как конечный результат обработки изображений, так и промежуточные, без дополнительных трудностей.

Зачастую в исследовательских целях необходимо получить относительный размер объектов на изображении, больших заданного начального значения: размер в пикселях может быть в дальнейшем использован для расчёта действительных размеров, например в квадратных миллиметрах. Реализовать поиск контуров объектов и расчёт их площади возможно с помощью библиотеки компьютерного зрения OpenCV.

Основная часть. Задача работы — спроектировать и создать приложение для определения формы и размеров объектов на изображении позволяющее определять процентное отношение размера найденных объектов к общей площади изображения. Поиск объектов на изображении происходит по следующему критерию: выбираются те объекты, площадь которых больше заданной пользователем.

В процессе разработки были использованы следующие решения:

1. Java. Ключевой особенностью языка является то, что его код сначала транслируется в специальный байт-код, независимый от платформы, затем выполняющийся виртуальной машиной JVM (Java Virtual Machine). Подобная архитектура обеспечивает кроссплатформенность и аппаратную переносимость программ на Java, благодаря чему подобные программы без перекомпиляции могут выполняться на различных платформах [2].

2. JavaFX. Платформа на основе Java для создания приложений с насыщенным графическим интерфейсом. Может использоваться как для создания настольных приложений, запускаемых непосредственно из-под операционных систем, так и для интернет-приложений, работающих в браузерах, и для приложений на мобильных устройствах. Данная платформа послужила основой создания графического интерфейса разрабатываемого программного продукта.

3. OpenCV. Библиотека алгоритмов обработки изображений, компьютерного зрения, а также численных алгоритмов общего назначения. Исходный код библиотеки доступен (за исключением отдельных небольших модулей) и распространяется под открытой лицензией BSD [3].

4. IntelliJ IDEA. Интегрированная среда разработки программного обеспечения для многих языков программирования, в частности Java, JavaScript, Python, разработанная компанией JetBrains. Среда доступна в двух редакциях. Community Edition является полностью свободной версией, доступной под лицензией Apache 2.0, в ней реализована полная поддержка JavaSE, Kotlin, Groovy, Scala, а также интеграция с наиболее популярными системами управления версиями. В редакции Ultimate, доступной под коммерческой лицензией, реализована поддержка JavaEE, UML-диаграмм, подсчёт покрытия кода, а также поддержка других систем управления версиями, языков и фреймворков [4].

Структура разработанного приложения представлена на диаграмме классов (рисунок 1).

Основным, представляющим весь базовый функционал приложения, классом является «LargeObjects», содержащий все необходимые поля и методы для обработки изображений и поиска объектов на них с помощью библиотеки OpenCV [5].

Класс «MainSceneController» является классом-контроллером основной сцены JavaFX-приложения и обеспечивает работу пользовательского интерфейса программы. Класс «AboutSceneController» отвечает за работу окна «О программе». Класс «ProjectUtils» является вспомогательным для более удобного написания кода. Класс «ObjectSizeApp», необходим для непосредственного запуска JavaFX-приложения.

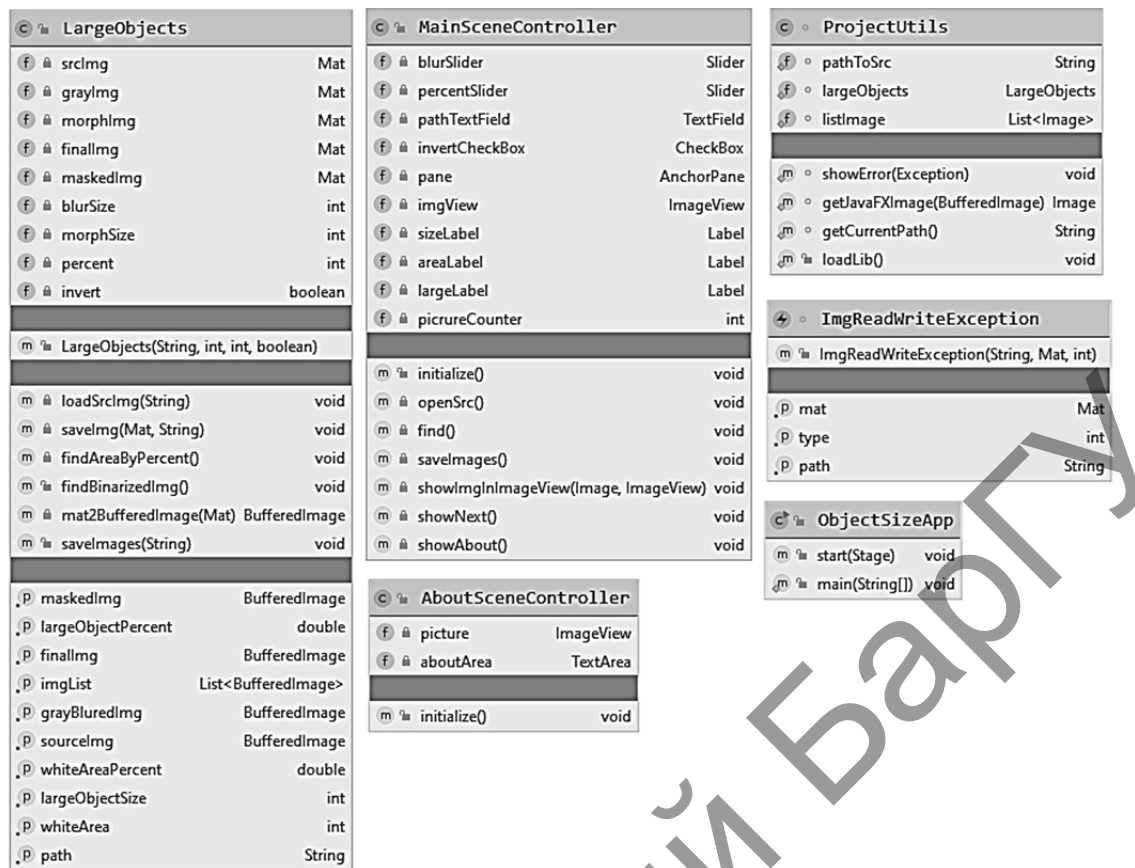


Рисунок 1 — Диаграмма классов

Общий алгоритм работы приложения состоит в следующем: загрузка исходного изображения, его обработка с целью улучшения распознавания краёв отдельных объектов и выделение объектов, размер которых превышает указанный.

Работа класса «LargeObjects» основывается на использовании методов и атрибутов классов библиотеки OpenCV преобразования исходного изображения, его сглаживания, инверсии и других преобразований для получение наиболее точного контура значимых по размеру объектов [6]:

- `Imgcodecs.imread(...)` — чтение изображений из файла в переменную типа `Mat` (объект-матрица пикселей обрабатываемого изображения);
- `Imgproc.cvtColor(...)` — конвертация цветового режима изображения (в данном случае приведение к оттенкам серого);
- `Imgproc.medianBlur(...)` — медианное размытие изображения (метод используется для сглаживания текстур объектов);
- `Mat.convertTo(...)` — увеличение контрастности изображения;
- `Imgproc.threshold(...)` — создание чёрно-белого изображения (целью является бинарное разделение объектов — белые и фона — чёрный);
- `Imgproc.dilate(...)`, `Imgproc.erode(...)`, `Imgproc.morphologyEx(...)` — сглаживание границ объектов для достижения большей точности;
- `Imgproc.findContours(...)` — поиск контуров объектов на изображении (матрице). В качестве одного из параметров указывается `Imgproc.RETR_EXTERNAL` который позволяет из иерархии найденных контуров (массив координат точек, формирующих края однородных областей изображения) сохранить только внешние;
- `Imgproc.fillConvexPoly(...)` — вывод найденных контуров в виде закрасненных многоугольников на матрицу изображения;
- `Imgcodecs.imwrite(...)` — сохранение матрицы как изображения в файл;
- `Core.addWeighted(...)` — получение изображения из двух с указанием параметра прозрачности для каждого (используется для полупрозрачного выделения подходящих объектов на исходном изображении).

Библиотеки и среда исполнения, необходимые для работы программы поставляются вместе в самораспаковывающемся архиве. Основной функционал приложения представлен в главном окне (рисунок 2). При выборе файла, изображение показано в главном окне приложения, выведены размеры этого изображения.

Далее необходимо выбрать силу размытия текстуры объектов для более точного определения их границ и задать размер искомых объектов в процентах по высоте. Если фон изображения светлый, то не-

обходимо поставить галочку в поле «Инvertировать изображение». После чего необходимо нажать на кнопку «Поиск объектов заданного размера».

По окончании обработки в середине окна выводится исходное изображение с полупрозрачной маской голубого цвета выделяющей объекты искомого размера. Справа под изображением выводится суммарная площадь искомым объектов в пикселях и процент, занимаемый этими объектами от всего изображения (рисунок 3). Из приведённого рисунка следует, что эти величины равны 1070436 пикселей и 47,57 %. Над кнопкой «Поиск объектов заданного размера» можно увидеть, чему равен минимальный размер искомым объектов (здесь он составляет 90 000 пикселей или 4 %).

Нажатием кнопки можно переключиться последовательно между видами, полученными в процессе обработки (рисунок 4): вид с полупрозрачной маской, вид маски, вид размытого изображения в оттенках серого и вид исходного изображения — и сохранить их на диск.

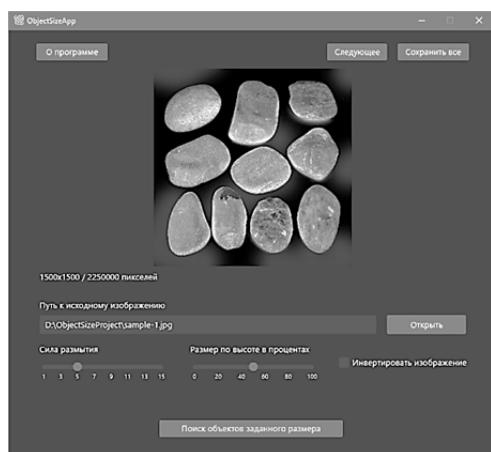


Рисунок 2 — Вид главного окна программы

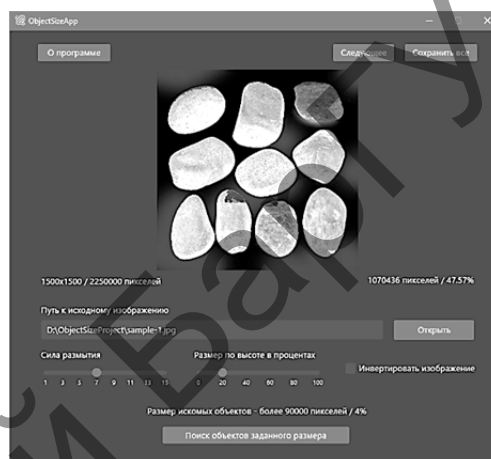


Рисунок 3 — Результат обработки

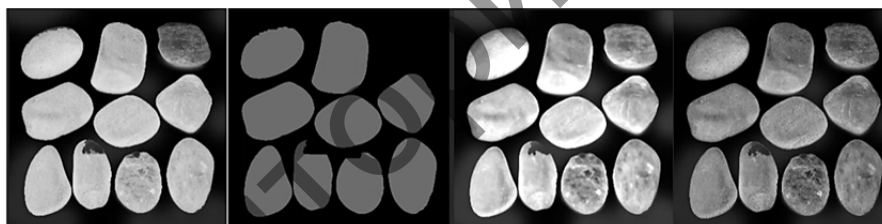


Рисунок 4 — Виды изображения

Параметры силы размытия и инvertирование изображения выбираются пользователем индивидуально для каждого изображения. Корректная работа приложения не гарантируется, если изображение имеет сложную текстуру или множество объектов находится рядом друг с другом.

Заключение. В результате выполнения работы было разработано приложение для распознавания и фильтрации объектов «Камень» с помощью библиотеки алгоритмов компьютерного зрения OpenCV на языке Java с применением платформы JavaFX. В качестве среды программирования была использована IntelliJ IDEA Ultimate 2020.1. Проектирование и разработка программного продукта потребовала подробного изучения документации и дополнительной литературы по работе с библиотекой OpenCV, платформы JavaFX и углубленным ознакомлением с возможностями, и особенностями языка программирования Java. Разработанное программное обеспечение может быть использовано в разных сферах жизни человека, например, при исследовании размеров объектов, на изображениях, полученных с микроскопа.

Список цитируемых источников

1. Computer vision [Электронный ресурс]. — Режим доступа: https://en.wikipedia.org/wiki/Computer_vision . — Дата доступа: 05.04.2020.
2. Java | Введение [Электронный ресурс]. — Режим доступа: <https://metanit.com/java/tutorial/1.1.php> . — Дата доступа: 05.04.2020.
3. Библиотека OpenCV [Электронный ресурс]. — Режим доступа: <http://oop.afti.ru/materials/32> . — Дата доступа: 05.04.2020.
4. IntelliJ IDEA [Электронный ресурс]. — Режим доступа: https://ru.wikipedia.org/wiki/IntelliJ_IDEA . — Дата доступа: 05.04.2020.
5. Калько, А. И. Распознавание автомобильных номеров с использованием нейронной сети / А. И. Калько, О. И. Наранович // Содружество наук. Барановичи-2016 : материалы XII Междунар. науч.-практ. конф. молодых исследователей, 19—20 мая 2016 г., Барановичи, Респ. Беларусь : в 3 кн. / редкол. : А. В. Никишова (гл. ред.) [и др.]. — Барановичи : РИО БарГУ, 2016. — Кн. 2. — С. 82—84.
6. Наранович, О. И. Система идентификации ручного почерка : [монография] / О. И. Наранович, А. И. Калько. — LAP LAMBERT Academic Publishing RU, 2018. — 44 с.