

## Метод поиска объектов на изображении с помощью контурного анализа по заданным характеристикам их контуров

*Национальный аэрокосмический университет им. Н. Е. Жуковского «ХАИ»*

Рассмотрена область применения метода контурного анализа в подсистемах распознавания образов. Разработан алгоритм нахождения объектов с использованием метода контурного анализа. Изучены функции библиотеки OpenCV, необходимые для реализации поставленной задачи. Разработана информационная подсистема подсчета объектов на изображении.

**Ключевые слова:** контур, изображение, контурный анализ, информационная подсистема, функция поиска контуров, библиотека OpenCV.

### Введение

Одним из важнейших методов распознавания графических образов является контурный анализ. Данный метод актуален в наше время и широко используется в различных сферах. Например, существуют программы по распознаванию текста и символов на изображении [1]. Такие программы очень часто встречаются в информационных системах производственного и хозяйственного назначения. В основном их используют для преобразования отсканированного изображения. Кроме распознавания текста на изображении существует возможность распознавания символов с помощью Web-камеры [2]. Данный метод контурного анализа (КА) применяется в детектировании номеров машин [3], т.е. может быть использован фирмами охраны [4]. При контурном анализе, на изображении можно подсчитывать количество объектов, например, разрабатываемая программа подсчета количества картошки на изображении. Такой метод рассматриваемого анализа может использоваться в сельскохозяйственной сфере.

КА значительно расширяет кругозор специалиста, позволяя с единых позиций подходить к обработке как акустических, радиотехнических и оптических сигналов, так и радиолокационных, телевизионных, оптических и других видов изображений. Важная роль анализа контуров подчеркивается в целом ряде оригинальных и обобщающих работ по распознаванию и обработке зрительных образов [5].

### Применение КА на основе методов библиотеки OpenCV

При решении задач контурного анализа с помощью библиотек OpenCV [6] нельзя создать универсальную программу, которая бы решала не один ряд задач. Поэтому для решения каждой задачи создается свой алгоритм. Описанный алгоритм рассчитан на решение задачи подсчета объектов, имеющих видимую границу на изображении.

Алгоритм включает в себя следующие шаги:

- создание переменных для хранения изображения. Используется структура `IplImage`;
- объявление и инициализация переменной указателя на имя файла с изображением;
- загрузка изображения. Используется функция `cvLoadImage`;

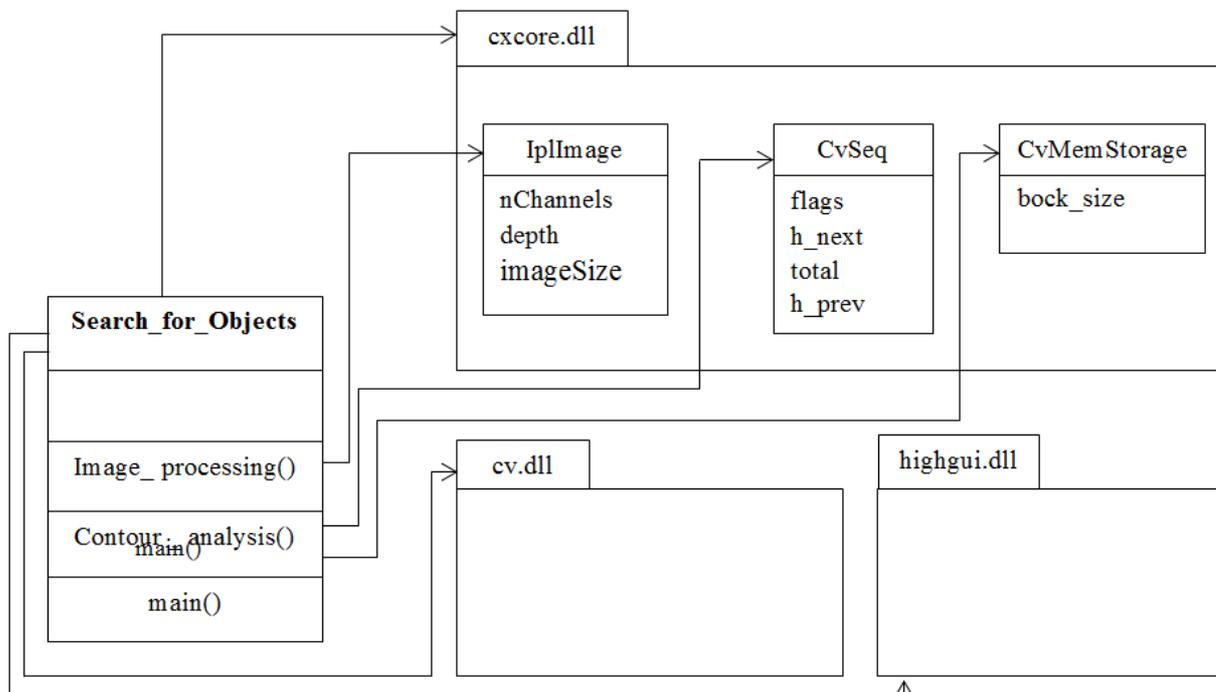
- проверка загрузки изображения. Используется типовая конструкция `assert` языка C++;
- создание переменных `storage` (контейнер для хранения всех контуров, тип `CvMemStorage`), `first_contour` (указатель на первый контур в структуре `CvSeq`);
- предварительная обработка изображения для корректного обнаружения необходимых объектов, а именно правильный перевод его в черно-белое изображение. Следовательно, при нахождении контура в `OpenCV` важно учитывать, что будет черным, а что белым цветом на изображении, так как черные пятна на изображении распознаются как наличие объекта, и именно на этих объектах находится контур, а белая область изображения считается «дырой» (отсутствие чего-либо). Для таких преобразований можно использовать ряд стандартных функций, а также разбиение изображения на каналы и преобразовать каждый канал с помощью разных функций. Вид преобразования определяется экспериментально, делая опыты на нескольких изображениях с различными расположениями одного и того же объекта;
- использование функции поиска контуров `cvFindContours`;
- удаление лишних контуров, если размер объекта имеет какое то определенное значение (зная, что объект по длине контура не может быть больше или меньше какого-либо значения). Возможность использования цикла, который организуется по списку, хранящемуся в структуре `cvSeq` и доступ к каждому контуру можно получить с помощью ссылок на элементы, `h_next`, `h_prev`;
- использование функции прорисовки контура `cvDrawContours()` для подтверждения корректной работы функции поиска контуров;
- создание окон, в которых будут отображаться изображения с использованием функции `cvNamedWindow`;
- загрузка изображений с использованием функции `cvShowImage()`;
- применение функции `cvWaitKey()` (позволит закрыть программу по нажатию клавиши);
- освобождение ресурсов с использованием функции `cvReleaseImage()`;
- уничтожение созданных окон с помощью функции `cvDestroyAllWindows()`.

Важнейшей функцией в решении такой задачи является `cvFindContours`. Синтаксис функции: `CVAPI(int) cvFindContours (CvArr* image, CvMemStorage* storage, CvSeq** first_contour, int header_size CV_DEFAULT(sizeof(CvContour)), int mode CV_DEFAULT (CV_RETR_LIST), int method CV_DEFAULT (CV_CHAIN_APPROX_SIMPLE), cvPoint offset CV_DEFAULT (cvPoint(0,0))`. Параметры функции: `image` – исходное 8-битное одноканальное изображение (ненулевые пиксели обрабатываются как 1, а нулевые — 0); `storage` – контейнер найденных контуров; `first_contour` – указатель на первый найденный контур; `header_size` – размер заголовка последовательности, который равен или больше результата функции `sizeof (CvChain)`, если выбран параметр `method CV_CHAIN_CODE`, и больше либо равен `sizeof (CvContour)` – в противном случае; `mode` – режим поиска: `CV_RETR_EXTERNAL` – находятся только критические внешние контуры, `CV_RETR_LIST` – находятся все контуры, и параметр помещает их в список, `CV_RETR_CCMP` – находятся все контуры, и записывают их в иерархию с двумя уровнями: верхний уровень – внешние границы компонентов, второй уровень – границы отверстий, `CV_RETR_TREE` – находятся все контуры и записывается полная иерархия вложенных контуров в виде дерева; `method` – метод аппроксимации, в разрабатываемой подсистеме выбран `CV_CHAIN_APPROX_SIMPLE` (сжимает горизонтальные, вертикальные, и

диагональные доли); offset – смещение, с которым каждая точка контура сдвинута относительно начала координат изображения.

### Разработка информационной подсистемы поиска и подсчёта объектов на изображении

При разработке информационной подсистемы подсчета объектов на изображении используют структуру, показанную на рисунке.



Структура информационной подсистемы

Разрабатываемая подсистема с именем «Search\_for\_Objects» зависит от трех динамически подключаемых библиотек OpenCV: «cxcore.dll», «cv.dll», «highgui.dll».

Динамическая библиотека «cxcore.dll» содержит в себе такие структуры, как IplImage, CvSeq, CvMemStorage. Данные структуры содержат определенные поля, к которым обращаются разработанные функции в информационной подсистеме. Функция «Image\_processing» содержит в себе весь набор переменных, стандартных функций и структур, необходимых для обработки изображения. Структура IplImage, к которой обращается рассматриваемая функция, позволяет выделить память под изображение (размер памяти хранится в поле imageSize структуры IplImage). Структура IplImage несет в себе информацию о количестве битов изображения (nChannels) и глубину цвета в битах (depth). Кроме того разрабатываемая функция содержит в себе набор стандартных функций, позволяющих перевести изображение в нужный формат и обработать его так, чтобы контуры интересующих нас объектов были четко выделены.

Функция «Contour \_ analysis» содержит в себе все необходимые данные для проведения системой контурного анализа. Данная функция обращается к структурам CvSeq и CvMemStorage. Структура CvSeq содержит в себе набор полей, который позволяет работать с данными, записанными в список, об этом

свидетельствуют поля: `h_next` – перейти к следующей записи и `h_prev` – вернуться к предыдущей. Однако структура `CvSeq` позволяет разработчику работать не только со списками, но и с древовидной структурой данных. Для этого в `CvSeq` предусмотрены поля `v_next` и `v_prev`, которые позволяют передвигаться вниз структуры. Поле `total` используется для хранения информации о контурах, в частности их длины. Поле `flags` хранит указатель на первый контур в структуре `CvSeq`. `CvMemStorage` используется для хранения всех найденных контуров.

Функция «`main`» является стандартной, и она включает в себя всю работу с отображением и уничтожением изображений в памяти компьютера.

### Выводы

В ходе исследования контурного анализа был разработан алгоритм нахождения объектов на изображении, а также информационная подсистема, которая позволяет находить и подсчитывать объекты. В данной подсистеме используется изображение размером 980\*550 пикселей, средняя длина контура составляет 100 пикселей, а длина удаляемых контуров равна 28 пикселей. Таким образом, программа в общем количестве насчитала 1949 контуров, однако с помощью очистки контуров получили результат – 94 объекта. Полученное значение равно количеству объектов на изображении.

### Список литературы

1. Распознавание образов для программистов [Электронный ресурс]/ Кручинин А. – Режим доступа: <http://blog.vidikon.com/?cat=1&paged=8> – 05.01.2010.
2. Применение математического аппарата контурного анализа для распознавания образов с web – камеры для 1С [Электронный ресурс]/ Смирнов С. – Режим доступа: <http://infostart.ru/public/266244> - 19.03.2014;
3. Детектирование автомобильных номеров на основе контурного анализа [Электронный ресурс] – Режим доступа: <http://mp3car.ru/blog/ComputerVision/15.html> - 26.10.2013;
4. Система распознавание автомобильных номеров [Электронный ресурс]/ - Режим доступа: <http://avtonomerok.com> – 28.05.2014;
5. Техносфера “Многомерный гиперкомплексный контурный анализ и его приложения к обработке изображений и сигналов” [Электронный ресурс]/ Леухин, А. Н. – Режим доступа: <http://tekhnosfera.com/mnogomernyy-giperkompleksnyy-konturnyy-analiz-i-ego-prilozheniya-k-obrabotke-izobrazheniy-i-signalov> - 2004;
6. Обработка и распознавание изображений в системах автоматизированного проектирования / Е. А. Дружинин, О. К. Погудина, И. Н. Бабак, А. В. Губарев. - Х.: ХАИ, 2011. – 51 с.

**Рецензент:** д.т.н., проф. зав. каф. Е. А. Дружинин. Национальный аэрокосмический университет имени Н. Е. Жуковского «ХАИ», Харьков.

Поступила в редакцию 29.05.2014.

## **Метод пошуку об'єктів на зображенні за допомогою контурного аналізу за заданими характеристиками їхніх контурів**

Розглянуто область застосування методу контурного аналізу в підсистемах розпізнавання образів. Розроблено алгоритм знаходження об'єктів з використанням методу контурного аналізу. Вивчено функції бібліотеки OpenCV, необхідні для реалізації поставленого завдання. Розроблено інформаційну підсистему підрахунку об'єктів на зображенні.

**Ключові слова:** контур, зображення, контурний аналіз, інформаційна підсистема, функція пошуку контурів, бібліотека OpenCV.

## **Search method of objects in the image with contour analysis on specified characteristics of their contours**

The applications region the method of contour analysis in subsystems recognition was investigated. Algorithm for finding objects has been developed. It have been based on the method of contour analysis. Author has studied OpenCV library functions, which necessary to accomplish the task. Information subsystem count objects in the image have been developed

**Keywords:** contour, image, contour analysis, information subsystem, function FindContours, library OpenCV.