

ПРИМЕНЕНИЕ КОНТУРНОГО АНАЛИЗА ДЛЯ НАХОЖДЕНИЯ ВНЕШНИХ ГРАНИЦ СЛОЖНОГО ОБЪЕКТА

Чжао Юньсян, Цапко И.В.
Томский политехнический университет
991546822@qq.com

Введение

Компьютерное зрение – это вид деятельности, в котором для извлечения данных из визуальной информации применяются статистические методы и используются модели, построенные с помощью физики, геометрии теории обучения. Цель дисциплины компьютерного зрения заключается в анализе изображений, полученных с помощью датчиков и на основе этого в формировании полезных выводов при сравнении объектов и сцен реального мира. При этом предварительно необходимо построить некоторое описание или модель этих объектов на основе изображения [1].

Компьютерное зрение, одна из самых востребованных областей на данном этапе развития всеобщих цифровых компьютерных технологий.

Применения компьютерного зрения

В умении распознавать человек, все, пока оставляет компьютер далеко позади. Машина преуспела лишь в определённых задачах — например, в распознавании номеров или машинописного текста. Успешно распознавать разнородные объекты и случайные сцены (разумеется, в условиях реальной жизни, а не лаборатории) компьютеру всё ещё очень трудно.

Среди многочисленных сфер применения компьютерного зрения можно выделить самую перспективную на текущий момент – это дополненная реальность. Данная технология получает широкое применение в реализации различных мобильных приложений и позволяет накладывать виртуальные элементы на картину реального мира. Так, например, можно получить информацию о доме, направив на него камеру телефона или планшета [2].

Еще одним из наиболее важных применений является обработка изображений в медицине. Исходные изображения, полученные при помощи ангиографии, рентгенографии, томографии, ультразвуковых исследований и микроскопии обрабатываются для извлечения из них полезной информации и последующей постановки медицинского диагноза пациентам. В качестве примера подобной информации можно назвать обнаружение злокачественных образований, опухолей, изменение кровотока или размеров органов.

Широкое применение компьютерное зрение нашло и в робототехнике. Главными задачами, решаемыми в этой области, являются распознавание, обнаружение и идентификация

объектов для последующей оценки его положения относительно устройства захвата, а также для объезда препятствий при автономной навигации.

Программа определения границ объекта

Данная работа посвящена разработке программы, позволяющей определить контур изображения насекомого, полученного в результате компьютерной томографии (рисунок 1).



Рисунок 1. Исходное изображение

Для реализации программы на языке программирования C++ использовалась библиотека компьютерного зрения OpenCV [3].

OpenCV — это свободно распространяемая библиотека алгоритмов компьютерного зрения, обработки изображений и численных алгоритмов общего назначения с открытым кодом. Реализована на C/C++, также разрабатывается для Python, Matlab, Java, Ruby, Lua и для других языков.

При разработке программы были использованы следующие функции:

1. Обработка изображения — Свёртка.

В OpenCV операция свёртки при помощи ядра kernel реализуется функцией cvFilter2D():

```
CVAPI(void) cvFilter2D( const CvArr*  
src, CvArr* dst, const CvMat*  
kernel, CvPoint anchor  
CV_DEFAULT(cvPoint(-1,-1)) );
```

где kernel — ядро свёртки (массив из одного канала из элементов типа float);

src — исходное изображение;

dst — изображение для сохранения результата.

Данная операция позволяет сгладить исходное изображение и тем самым очистить его от шумовых составляющих.

2. Пороговое преобразование изображения.

Эта операция осуществляется при помощи функции:

```
CVAPI(void) cvInRangeS( const CvArr*  
src, CvScalar lower, CvScalar upper,  
CvArr* dst );
```

где *src* — исходный массив;
lower — скаляр с нижней границей (включая);
upper — скаляр с верхней границей (не включая);
dst — массив для хранения результата.

Функция проверяет, что элемент массива лежит между двух скаляров:

```
dst[idx] = lower <= src[idx] < upper
```

Результат обработки изображения приведен на рисунке 2.

Нахождение контуров и операции с ними.

Контурный анализ — это один из важных и очень полезных методов описания, распознавания, хранения поиска графических образов/объектов.

Контур — это внешние очертания (обвод) объекта/предмета.

Для поиска контуров используется функция `cvFindContours()`:

```
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)))
```

где *image* — исходное 8-битное одноканальное изображение (ненулевые пиксели обрабатываются как 1, а нулевые — 0);

storage — хранилище памяти для хранения данных найденных контуров;

first_contour — указатель, который будет указывать на первый элемент последовательности, содержащей данные найденных контуров;

header_size — размер заголовка элемента последовательности;

mode — режим поиска, может быть представлен одной из следующих констант:

`CV_RETR_EXTERNAL` (0) — ищет только крайние внешние контуры;

`CV_RETR_LIST` (1) — ищет все контуры и размещает их списком;

`CV_RETR_CCOMP` (2) — ищет все контуры и размещает их в виде 2-уровневой иерархии;

`CV_RETR_TREE` (3) — ищет все контуры и размещает их в иерархии вложенных контуров.

Отобразить найденные контуры можно с помощью функции `cvDrawContours()`:

```
CVAPI(void) cvDrawContours( CvArr  
*img, CvSeq* contour, CvScalar  
external_color, CvScalar hole_color,  
int max_level, int thickness  
CV_DEFAULT(1), int line_type  
CV_DEFAULT(8), CvPoint offset  
CV_DEFAULT(cvPoint(0,0)));
```

Пример работы программы изображен на рисунке 3.



Рисунок 2. Изображение после порогового преобразования

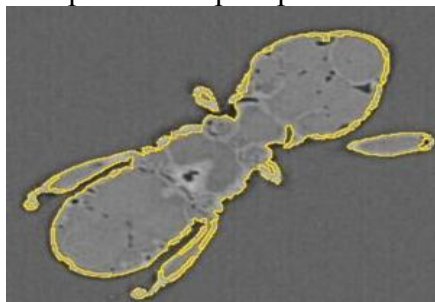


Рисунок 3. Контур изображения

Полученный контур позволяет получить такие характеристики объекта как его площадь, периметр, а также сравнить контур с шаблонными значениями.

Заключение

В результате был проведен контурный анализ изображения насекомого и получено очертание объекта.

В процессе работы, мы познакомились с библиотекой `OpenCV`, изучили несколько функций, таких как: сравнение контуров; детектор границ Кенни (`Canny`); морфологические преобразования и т.д.

Список использованных источников

1. Компьютерное зрение [Электронный ресурс]. — URL: <https://ru.wikipedia.org/> (дата обращения 28.09.2016).

2. Цапко И.В., Власов А.В. Выделение объектов на изображениях методом поиска границ регионов // Автоматизация. Современные технологии. — 2015. № 9. С. 33-38.

3. `OpenCV шаг за шагом` [Электронный ресурс]. — URL: <http://robocraft.ru/page/opencv/> (дата обращения 28.09.2016).