Конспект

Гит-репозиторий — это облачное хранение вашего проекта на сервере (например, на сервере Гитхаба, но можно и на другом).

У каждого программиста может быть сколько угодно репозиториев, по одному на каждый проект. А можно вести все проекты в одном репозитории, но тогда это превратится в мешанину. Но каждый имеет право на мешанину.

В репозитории могут храниться:

• файлы с исходным кодом проекта;

• базы данных;

• картинки и графики;

• текстовые файлы;

• и всё остальное, что нужно проекту для работы.

Коммит (commit) — основной объект разработки, в котором хранятся все изменения кода за итерацию. По сути, это список со всеми актуальными изменениями и ссылка на предыдущую версию коммита. У каждого коммита есть атрибуты: имя, дата создания, автор и комментарии к текущей версии (например, «Создал страницу courses.html» при разработке сайтов с видеокурсами).

Ветка (branch) — указатель на коммит с определенными изменениями. Например, два разработчика взяли коммит, и каждый из них сделал свои изменения в коде, создав по новому коммиту («Создал страницу coursеs.html c личным кабинетом» и «Создал страницу courses.html со свободным доступом на курсы»). Так в проекте появились две ветки с разным кодом: разработчик может выбрать, над каким коммитом ему работать дальше.

Основной веткой проекта, как правило, считается ветка main или master — разработчики создают новые ветки на ее основе. Также можно создать неограниченное количество веток, чтобы вносить новые изменения, не мешая основному проекту.

OpenCV

**OpenCV — библиотека алгоритмов компьютерного зрения, обработки изображений и численных алгоритмов общего назначения с открытым кодом.**

Также, можно сказать что OpenCV – это набор типов данных, функций и классов для обработки изображений алгоритмами компьютерного зрения.

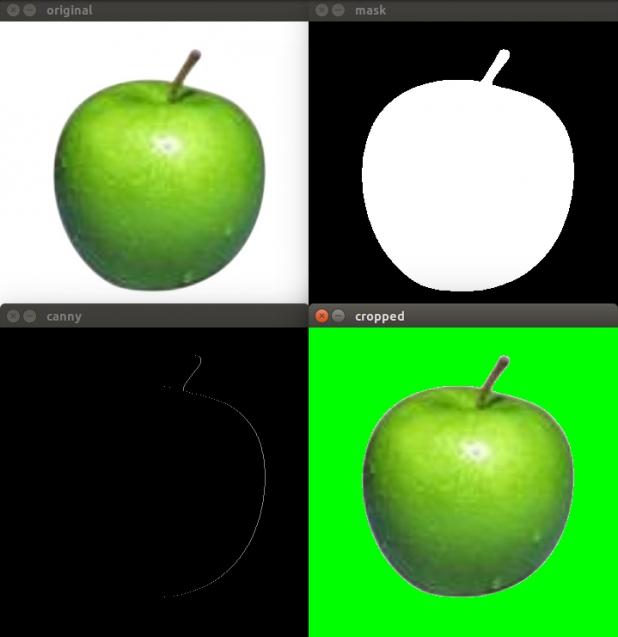
Основные модули библиотеки:  
  
**cxcore** — ядро  
\* содержит базовые структуры данных и алгоритмы:  
— базовые операции над многомерными числовыми массивами  
— матричная алгебра, математические функции, генераторы случайных чисел  
— Запись/восстановление структур данных в/из XML  
— базовые функции 2D графики  
  
**CV** — модуль обработки изображений и компьютерного зрения  
— базовые операции над изображениями (фильтрация, геометрические преобразования, преобразование цветовых пространств и т. д.)  
— анализ изображений (выбор отличительных признаков, морфология, поиск контуров, гистограммы)  
— анализ движения, слежение за объектами  
— обнаружение объектов, в частности лиц  
— калибровка камер, элементы восстановления пространственной структуры  
  
**Highgui** — модуль для ввода/вывода изображений и видео, создания пользовательского интерфейса  
— захват видео с камер и из видео файлов, чтение/запись статических изображений.  
— функции для организации простого UI (все демо приложения используют HighGUI)  
  
**Cvaux** — экспериментальные и устаревшие функции  
— пространств. зрение: стерео калибрация, само калибрация  
— поиск стерео-соответствия, клики в графах  
— нахождение и описание черт лица  
  
**CvCam** — захват видео  
— позволяет осуществлять захват видео с цифровых видео-камер

Поиск объектов через определение их контуров средствами OpenCV

Множество объектов на изображении можно опознать по их форме, например, печати в сканах документов, как правило, круглые, развороты паспорта четырёхугольные и т.д. Зная простые свойства искомого объекта и отсутствие этих свойств у других объектов изображения, можно свести задачу детектирования на изображении к задаче поиска объекта, чей контур содержит в себе какое-то количество углов. Говоря проще: хочешь найти на фотографии стола книгу – найди что-то с четырьмя сторонами.

Пример нахождения контуров объекта.

У нас есть изображение яблока, мы хотим выделить его и поместить на другое изображение с зеленым фоном.

Простейшим решением является сначала обнаружить края яблока с помощью **Canny** фильтра, а затем найти контуры функцией **findContours** в OpenCV и создать маску с помощью **drawContours**: