

## Технология OpenCV

Гришин А.Н.; Орёл ГТУ; e.mail: [alextank@mailorel.ru](mailto:alextank@mailorel.ru)

Едва ли не полстолетия многих занимал вопрос: “Может ли компьютер думать?”. Теперь пришло время задаться новым вопросом - “Может ли компьютер чувствовать?” Одно из пяти человеческих чувств - зрение. Слово “видеть” означает не только восприятие тех или иных визуальных образов, в это понятие включается и сложный мыслительный процесс, во время которого человеческий мозг активизирует накопленный опыт, стараясь еще и осознать увиденное. Водителю автомобиля хватает лишь мгновенного взгляда на дорогу, чтобы оценить качество ее покрытия, скорость впереди идущей машины, боковым зрением зафиксировать тех, кто движется справа и слева, и решить, что делать дальше. Компьютеру все эти действия надо запрограммировать.

Компьютерное зрение, которое разрабатывается для быстрого и безопасного передвижения автомобилей-роботов, наверняка пригодится и обычным машинам с водителями - чтобы избежать столкновений, при перестроении, при маневрировании в «пробках». Эта задача с каждым днем приобретает все большую актуальность, поскольку сегодня в автокатастрофах из-за невнимательности или ошибок водителей гибнет больше людей, чем от рака, причем их число будет только расти. Компьютерное зрение может помешать развитию этой опасной тенденции. Прежде конструкторы пытались построить компьютерное зрение на основе радаров или аналогичных систем, использующих отражение света, но с их помощью автомобиль мог замечать только крупные препятствия и не мог определить, где он едет. Опыт показал, что для решения этой проблемы необходимы видекамеры, изображение с которых будет считываться и анализироваться в реальном времени.

Несколько лет назад разработчики программного обеспечения из корпорации Intel объявили о выпуске пакета исходных кодов программ для реализации стереоскопического компьютерного "зрения", вошедших в библиотеку Open Source Computer Vision Library (OpenCV). Программный код библиотеки может применяться в самых разных приложениях - от обеспечения безопасности до исследований космоса. Алгоритмы, реализованные в OpenCV, были созданы в научно-исследовательском центре Intel в Нижнем Новгороде и позднее были использованы в технологии IPP (Intel Integrated Performance Primitives), которую разработали в научно-исследовательских центрах Intel в Сарове и Нижнем Новгороде. Intel IPP, а также компиляторы Intel использовались при создании программного обеспечения для автомобилей-роботов. Надо заметить, что компиляторы Intel создавались с участием российских специалистов, работающих в научно-исследовательских центрах Intel в Москве, Новосибирске и Сарове.

Библиотека Open CV представляет собой инструментарий, который насчитывает более 500 функций обработки и анализа изображений для создания приложений компьютерного “зрения”, в том числе средств взаимодействия человека с компьютером. Новые программы позволяют с помощью двух компьютерных видеокамер получать не только изображение, но и данные о глубине объектов, что дает возможность использовать существующие приложения для таких задач, как распознавание жестов, слежение за объектами и распознавание лиц. Выпущенный пакет программ включает в себя дополнительный интерфейс, позволяющий импортировать функции библиотеки OpenCV в MATLAB - один из самых популярных программных пакетов для исследования и быстрой разработки прототипов программ. Рост быстродействия процессоров, снижение цен на видеокамеры и десятикратный рост пропускной способности захвата видео, создают возможность для выполнения алгоритмов компьютерного “зрения” в реальном времени на стандартных ПК. Библиотека компьютерного “зрения” Intel призвана стимулировать развитие этого направления, предоставляя исходные коды для широкого спектра функций компьютерного “зрения” и обработки изображений. Компьютерное зрение автомобиля-робота строится на сочетании сразу нескольких “органов зрения”: в процессоры компьютеров-«водителей» поступает информация не только с нескольких видеокамер, но и с нескольких лазерных дальномеров. Лазеры мгновенно определяют расстояние до более или менее крупных предметов на пути движения машины, и потом эти сведения используются при восстановлении картинка с видеокамер.

Библиотека OpenCV активно используется большим количеством компаний (Intel, IBM, Microsoft, SONY, Siemens...) и исследовательскими центрами (Stanford, MIT, CMU, Cambridge, INRIA...). Основные функциональные возможности библиотеки OpenCV представлены ниже:

- основная обработка изображения: фильтры изображения, морфология, пирамиды, преобразования цветных мест, геометрические преобразования, гистограммы;
- продвинутая обработка изображения и извлечение признаков: обнаружение угла, осторожный датчик края, преобразование расстояния, водораздел;
- анализ формы и вычислительная геометрия: моменты изображения, контуры, триангуляция (разбивка на треугольники) и мозаика;
- анализ движения: оптический поток, прослеживание объекта;
- шаблоны движения, фильтры, калибровка камеры, Eriplar – геометрия обнаружение объекта (классификатор), продвинутое прослеживание ( видео модуль наблюдения);

Извлечение текста из видео:

- каждая буква имеет единственный цвет и гладкую форму;
- буквы организованы в словах: горизонтально-выровненные группы;
- все буквы в слове (и все подзаголовки в видео структуре) имеют тот же самый цвет;

- буквы и слова сохраняют свои формы поперек нескольких структур, пребывают в том же самом положении или движутся гладко;
- сортировка контуров координат у ограниченного прямоугольника для идентификации слова;
- использование `cvDrawContours` + `cvAvgStdDev` для проверки цвета.

Библиотека полностью написана на стандартном языке C++ без каких либо его изменений. Приведён пример исходного кода программы для распознавания лица:

```
#include <cv.h>
#include <highgui.h>
#include <math.h>
int main( int argc, char** argv ) {
    CvPoint center;
    double scale=-3;
    IplImage* image = argc==2 ? cvLoadImage(argv[1]) : 0;
    if(!image) return -1;
    center = cvPoint(image->width/2,image->height/2);
    for(int i=0;i<image->height;i++)
    for(int j=0;j<image->width;j++) {
        double dx=(double)(j-center.x)/center.x;
        double dy=(double)(i-center.y)/center.y;
        double weight=exp((dx*dx+dy*dy)*scale);
        uchar* ptr = &CV_IMAGE_ELEM(image,uchar,i,j*3);
        ptr[0] = cvRound(ptr[0]*weight);
        ptr[1] = cvRound(ptr[1]*weight);
        ptr[2] = cvRound(ptr[2]*weight); }
    cvSaveImage( "copy.png", image );
    cvNamedWindow( "test", 1 );
    cvShowImage( "test", image );
    cvWaitKey();
    return 0; }
```



Технология OpenCV была использована в некоторых машинах на гонках машин – роботов, которая прошла 8 октября 2005 года в пустыне Мохаве (штат Невада). Утром 8 октября на старт соревнования под названием «Большой вызов-2005» вышли 23 автомобиля. Им предстояло преодолеть дистанцию в 212 километров по пересеченной местности. Только пять из них добрались до финиша, и быстрее всех - за 6 часов 53 минуты 58 секунд – это сделал «Туарег».

Внедорожники - роботы двигались совершенно самостоятельно, без какого-либо вмешательства людей (даже по радио). По условиям соревнований, за два часа до старта команды ученых и инженеров, подготовивших эти автомобили, получили компакт-диски с маршрутом, который ввели в электронную память машин. При этом маршрут указывал лишь общее направление гонки, а с преодолением ям, косогоров и прочих препятствий естественного и искусственного происхождения (о создании дополнительных преград позаботились организаторы) компьютеры и управляемые ими внедорожники должны были справиться самостоятельно.

Чтобы машины могли ориентироваться на местности, их создатели применили самые разные технологии - радары и лазерные сканеры, ультразвуковые локаторы и видекамеры. Чемпион «Большого вызова-2005», которого подготовили к гонке в Стэнфордском университете, и показавший второй результат автомобиль Sandstorm (подготовленный специалистами из Института робототехники при университете им. Карнеги Меллона) использовали оптимизированные библиотеки Intel Integrated Performance

Primitives (IPP) и OpenCV. Используемые в этих библиотеках алгоритмы применяющиеся для статистической обработки изображений, помогли искусственному интеллекту обеих машин преодолеть трассу с препятствиями.

Координаты при движении автомобилей-роботов, участвовавших в гонке «Большой вызов-2005», определялись с помощью GPS-приемников, а окружающая обстановка обследовалась лазерными дальномерами, радарами, стереокамерами и системой монокулярного зрения. Информация о текущем положении сверялась с цифровой картой. В компьютере, управлявшем победителем гонки, это происходило в среднем 10 раз в секунду, а пиковые показатели достигали 100 раз. Это самый высокий результат среди участников гонки, именно он и сыграл решающую роль в достижении победы.

Победитель гонки - внедорожник «Фольксваген-Туарег» и показавший второй результат автомобиль Sandstorm использовали оптимизирующие библиотеки (IPP) и OpenCV. Именно эти библиотеки помогли обеим машинам преодолеть всю трассу за минимальное время. Члены команды Стэнфордского университета заявили, что получили прирост производительности в 50% после перевода своего компьютерного кода на компиляторы Intel. Они особо отметили надежность компьютерного оборудования Intel: за весь период подготовки к гонке и во время самой гонки не было ни одного отказа.

### **Список литературы**

1. БАБР\_RU-Сибирь - Региональный информационный центр новости Иркутск [Электронный ресурс] /<http://babr.ru/?pt=news&event=v1&IDE=25575> - 2005.
2. Intel® Open Source Computer Vision Library (Documentation) [Электронный ресурс] /<http://groups.yahoo.com/group/OpenCV/> - 2005.