

САНКТ - ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Математико-механический факультет

Кафедра системного программирования

Тема

**Распознавание движения человека по ряду
изображений**

Курсовая работа студента 361 группы

Ромашкина Амира Сергеевича

Научный руководитель,

Аспирант...../А.Вахитов/

/Подпись/

Санкт-Петербург

2010

Содержание

Введение.....	3
Общая информация о компьютерном зрении.....	5
Алгоритмы распознавания движения.....	8
Решение проблемы тени.....	10
Принцип нахождения высоты объекта.....	11
Заключение.....	12
Список используемой литературы.....	13

Введение

Мало кто будет спорить, что системы видеонаблюдения сегодня устанавливает любая уважающая себя компания или государственное учреждение. Посредством компьютерного видения можно получить полную информацию об объектах, находящихся или находившихся в поле зрения камер.

Имеется очень широкий спектр задачи компьютерного видения, и одной из основополагающих является задача определения движений объектов.

Сегодня детектирование движения используется повсеместно, в любой точке земного шара. Например, требуется считать, сколько человек вошло в магазин, а сколько вышло. Узнавать в режиме реального времени, что никто не пробрался на закрытую территорию и т.д.

Для постановки задачи детектирования движения нужно вначале решить ответить на несколько вопросов:

- 1) Со скольких камер ведется наблюдение?
- 2) Требуется ли самообучение программы в процессе наблюдения?

Видеонаблюдение может вестись как с одной камеры, так и с нескольких одновременно. Ситуация нескольких камер гораздо более надежна, поэтому в жизни эта ситуация встречается гораздо чаще, чем «плоский» вариант с одной камерой. С другой стороны она более трудоемка, поэтому было принято решение начать изучение компьютерного видения с «плоского» случая.

Камера может находиться как внутри здания с минимальным фоновым визуальным шумом, так и на улице, где фон будет состоять из вечно колышущих деревьев, меняющегося положения солнца и соответственно теней. Как нетрудно заметить, видео наблюдение на улице на порядок сложнее. Для решения этих проблем приходится использовать самообучающиеся алгоритмы детектирования движения, которые подстраиваются под изменяющиеся условия внешней среды.

Как только от программы требуется сопоставить данные с камеры с реально существующими объектами или понятиями (рост объекта, длина пройденного пути, траектория пути и т.д.), то следующим на очереди вопросом, на который нужно ответить постановщику задачи является вопрос:

3) Откалибрована ли камера?

Калибровка камеры - это процесс соотнесения идеальной математической модели камеры с конкретным физическим устройством, а также определение ориентации и расположения камеры в системе координат сцены. Этот вопрос эквивалентен вопросу «Будет ли программа работать, если камеру подвинуть?» В общем, даже в жизни предварительная калибровка камеры может вполне быть удобнее, а главное надежнее, чем программная реализация калибрования.

Задача моей курсовой работы - познакомиться с общими принципами и методами детектирования движения и модернизировать, дополнить имеющийся метод. В процессе распознавания движения появляется ряд проблем, которые нужно решать.

Первая из проблем - фоновый визуальный шум, появляющийся даже в закрытых помещениях из-за дефектов видеоряда или мерцания искусственного света, например.

Также серьезной проблемой является тень, мешающая определить настоящий рост движущегося человека. Для избавления от тени был использован метод, который будет подробнее описан ниже.

Общая информация о компьютерном зрении

Компьютерное зрение (компьютерное видение) – это теория и технология создания искусственных систем, которые получают информацию из изображений. Видеоданные могут быть представлены множеством форм, таких как видеопоследовательность, изображения с различных камер или трехмерными данными с медицинского сканера.

Область компьютерного зрения может быть охарактеризована как молодая и разнообразная. Даже хотя существуют более ранние работы, можно сказать, что только с конца 1970-х началось интенсивное изучение этой проблемы, когда компьютеры смогли управлять обработкой больших наборов данных, таких как изображения.

Однако эти исследования обычно начинались с других различных областей, и, следовательно, нет стандартной формулировки проблемы компьютерного зрения. Также, и это даже более важно, нет стандартной формулировки того, как должна решаться проблема компьютерного зрения. Вместо этого, существует масса методов для решения различных строго определенных задач компьютерного зрения, где методы часто зависят от задач и редко могут быть обобщены для широкого круга применения.

Многие из методов и приложений все ещё находятся в стадии фундаментальных исследований, но все большее число методов находит применение в коммерческих продуктах, где они часто составляют часть более большой системы, которая может решать сложные задачи (например, в области медицинских изображений или измерения и контроля качества в процессах изготовления).

В большинстве практических применений компьютерного зрения компьютеры предварительно запрограммированы для решения отдельных задач, но методы, основанные на знаниях, становятся все более общими.

Некоторые примеры типичных задач компьютерного зрения представлены ниже.

Распознавание.

Классическая задача в компьютерном зрении, обработке изображений и машинном зрении это определение содержат ли видеоданные некоторый характерный объект, особенность или активность. Эта задача может быть достоверно и легко решена человеком, но до сих пор не решена удовлетворительно в компьютерном зрении в общем случае: случайные объекты в случайных ситуациях.

Существующие методы решения этой задачи эффективны только для отдельных объектов, таких как простые геометрические объекты (например, многогранники), человеческие лица, печатные или рукописные символы, автомобили и только в

определенных условиях, обычно это определенное освещение, фон и положение объекта относительно камеры.

В литературе описано различное множество проблем распознавания:

Распознавание: один или несколько предварительно заданных или изученных объектов или классов объектов могут быть распознаны, обычно вместе с их двухмерным положением на изображении или трехмерным положением в сцене.

Идентификация: распознается индивидуальный экземпляр объекта. Примеры: идентификация определенного человеческого лица или отпечатка пальцев или автомобиля.

Обнаружение.

Видеоданные проверяются на наличие определенного условия. Например, обнаружение возможных неправильных клеток или тканей в медицинских изображениях. Обнаружение, основанное на относительно простых и быстрых вычислениях, иногда используется для нахождения небольших участков в анализируемом изображении, которые затем анализируются с помощью приемов, более требовательных к ресурсам, для получения правильной интерпретации.

Существует несколько специализированных задач, основанных на распознавании текстов, например:

Поиск изображений по содержанию: нахождение всех изображений в большом наборе изображений, которые имеют определенное содержание. Содержание может быть определено различными путями, например, в терминах схожести с конкретным изображением (найдите мне все изображения похожие на данное изображение), или в терминах высокоуровневых критериев поиска, вводимых как текстовые данные (найдите мне все изображения, на которых изображено много домов, которые сделаны зимой и на которых нет машин).

Оценка положения: определение положения или ориентации определенного объекта относительно камеры. Примером применения этой техники может быть содействие руке робота в извлечении объектов с ленты конвейера на линии сборки.

Оптическое распознавание знаков: распознавание символов на изображениях печатного или рукописного текста, обычно для перевода в текстовый формат, наиболее удобный для редактирования или индексации (например, ASCII).

Движение.

Несколько задач, связанных с оценкой движения, в которых последовательность изображений (видеоданные) обрабатываются для нахождения оценки скорости каждой точки изображения или 3D сцены. Примерами таких задач являются:

Определение трехмерного движения камеры

Слежение, т.е. следование за перемещениями объекта (например, машин или людей)

Восстановление сцены

Даны два или больше изображения сцены, или видеоданные. Восстановление сцены имеет задачей воссоздать трехмерную модель сцены. В простейшем случае, моделью может быть набор точек трехмерного пространства. Более сложные методы воспроизводят полную трехмерную модель.

В рамках данной курсовой работы я занимаюсь только движением и распознаванием объекта (человека).

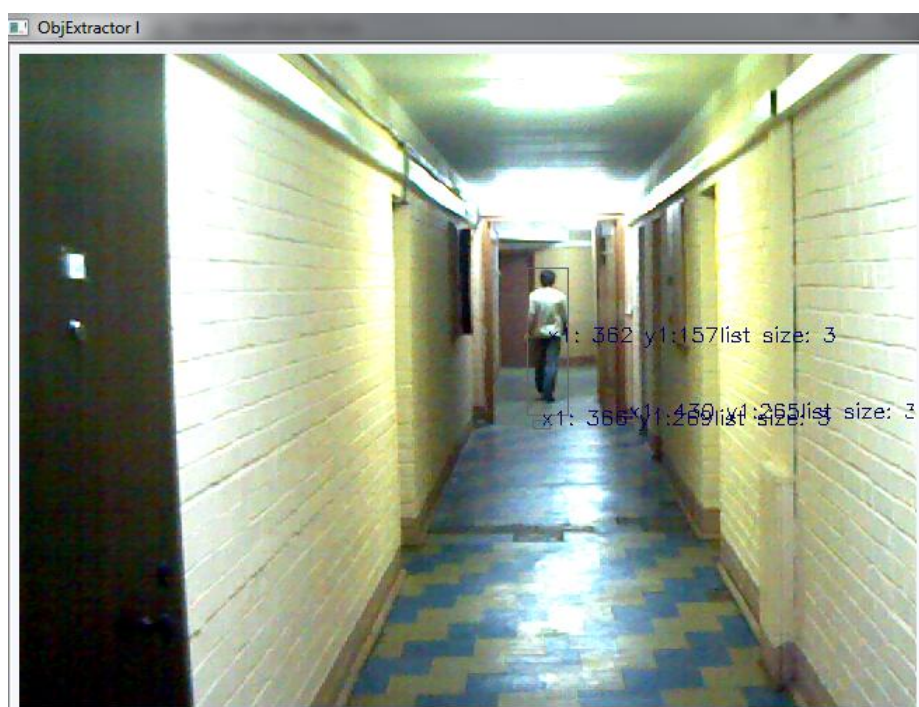


Рис.1 Object extractor. Выделяет движущийся объект.

OpenCV (Open Source Computer Vision) — библиотека алгоритмов компьютерного зрения, обработки изображений и численных алгоритмов общего назначения с открытым кодом. Реализована на C/C++.

Существуют «обертки» это библиотеки и над другими языками, но из соображений скорости работы программы был выбран язык C++.

Разработка проходила в среде разработки Visual Studio 2008.

Алгоритмы распознавания движения

Существуют различные подходы для обнаружения движения в непрерывном потоке видео. Все они основаны на сравнении текущего кадра видео с одним из предыдущих кадров или с чем-то, что мы будем называть фоном. В этой статье я попытаюсь описать некоторые из наиболее распространенных подходов.

Один из самых распространенных подходов является сравнение текущего кадра с предыдущим. Это полезно в сжатия видео, когда необходимо оценить изменения, и писать только изменения, а не весь кадр. Но это не самый лучший способ для обнаружения движения.

На этом этапе мы получим изображение с белые пиксели на месте, где текущий кадр отличается от предыдущего кадра на указанное значение порога. Это уже можно считать пикселей, и если его количество будет больше, чем предопределенные сигнализации уровне мы можем сигнал о движении события.

Но большинство камер делают шумные изображения, так что мы получим движения в таких местах, где нет никакого движения вообще. Об этой проблеме я писал выше.

Существует другой подход. Есть возможность сравнить текущий кадр не с предыдущей, но с первого кадра видеопоследовательности. Таким образом, если бы не было объектов в начальный кадр, сравнение текущего кадра с первого 1 даст нам целый движущийся объект, независимо от его скорости движения. Но этот подход имеет большой недостаток - то, что случится, если существует, например, автомобиль на первый кадр, а потом он исчез? Да, мы будем всегда иметь движение, обнаруженное на месте, где автомобиль был. Конечно, мы можем продлить начальный кадр иногда, но это не даст хороших результатов в тех случаях, когда мы не можем гарантировать, что первый кадр будет содержать только статический фон. Но, может быть обратная ситуация. Если я поставлю картину на стене в комнате? Я получу движения обнаружить только начальный кадр, будут возобновлены.

Наиболее эффективные алгоритмы основаны на создании так называемого фона сцены и сравнения каждого текущего кадра с фоном.

По завершении процесса заполнения кадра черными и белыми точками начинается процесса выделения объектов. Сгустки белых пикселей алгоритм объединяет в один объект и выделяет его на экране прямоугольником.

В рамках работы я занимался избавлением от шума рекурсивным алгоритмом, который считает размер «пятна», и если оно размером меньше какой-то небольшой

константы(15,например) то затирает пятно черным и, то есть, считает, что там движения нет.

Решение проблемы тени

Очевидная проблема при выделении человека – его тень. Простой алгоритм распознавания движения тут же реагирует на смену освещения пикселя и тем самым считает тень – частью движущегося объекта, что неправильно.

Вначале для решения этой задачи я попробовал перевести традиционное цветовое пространство RGB в пространство HSV, имеющее одним из координат – яркость. Но это не возымело нужного действия. После этого студентом 5ого курса был подсказан алгоритм преобразования модели RGB в новое также трехмерное пространство, которое не реагирует на смену яркости пикселя.

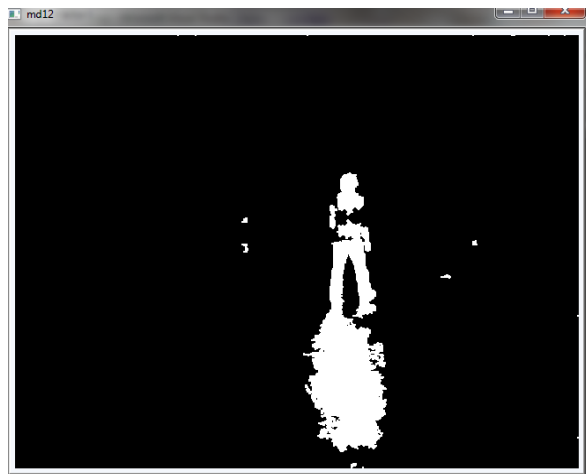


Рис 2. Motion Detector working RGB

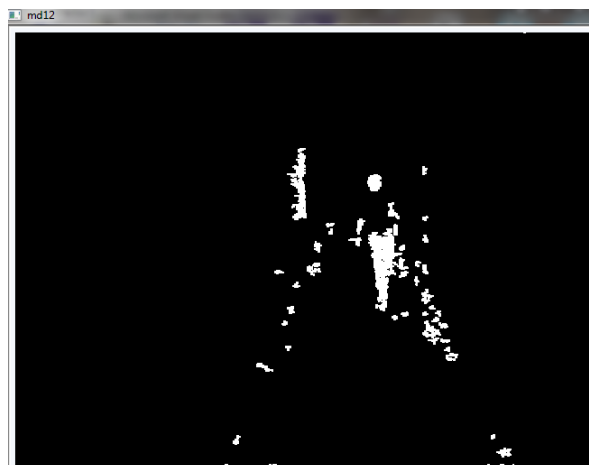


Рис 3 Motion Detector working with new shadow-non-sensitive color space

Как видно из рисунков, несмотря на то, что тень все-таки убралась, сильно пострадало качество детектирования движения точек, увеличился фоновый шум, с которым не справляется «уборщик шума» из-за больших размеров шумовых «пятен».

Принцип нахождения высоты движущегося объекта

Для того чтобы корректно определить высоту объекта камера должна быть откалибрована. Это делается с помощью нескольких замеров объекта известной высоты (например, палки) с места съемки. Находятся (x,y) – координаты палки и высота ее в пикселях на экране. Также можно запоминать реальное расстояние до палки, чтобы построить математическую модель плоскости, по которой будет идти объект. Но для нашей задачи этого не требовалось.

Изначально предполагалось вычислять высоту объектов линейно. То есть найти линейную зависимость между высотой реальной палки координат и высотой в пикселях и интерполировать в остальные точки. Но опытным путем было выяснено, что так не достигается нужная точность.

Было принято решение брать полный многочлен 2-ой степени от x и y . $F(x,y) = a*x^2+b*x*y+c*y^2+d*x+e*y+f$. Этот многочлен вычисляется в координатах точки на экране, в которой находится палка (например, левый нижний угол палки). Далее в формулу $H^2 = P^2 * F(x, y)$ подставляем H – реальную высоту палки и P – высоту палки в пикселях на экране в конкретной точке (x,y) . Сделав не менее 6 замеров, находим коэффициенты многочлена и можем сосчитать высоту объекта в зависимости от его высоты в пикселях. Естественно, этот метод не абсолютно точный.

Заключение

По сути, новизна проделанной работы минимальна, так как все, чем я занимался в рамках работы, уже было создано ранее. Главной целью курсовой работы было ознакомление с основными принципами и методами решения задач компьютерного зрения.

В процессе работы мной были изучена библиотека OpenCV. Я познакомился с основными методами детектирования движения, а также дополнил, частично исправил существующую реализацию одного из методов.

Но многое нужно еще исправлять и дополнять в данной программе, как например то, что после реализации игнорирования тени упало качество общего детектирования движения.

Список используемой литературы

- [1] «Определение движущихся объектов по серии изображений и измерение их размеров», Ираклий Мерабашвили, Дипломная работа (2010)
- [2] http://ru.wikipedia.org/wiki/Компьютерное_зрение
- [3] http://www.codeproject.com/KB/audio-video/Motion_Detection.aspx
- [4] Y. Weiss. Deriving intrinsic images from image sequences. In ICCV01 (2001), Pages: 68-75
- [5] Y. Matsushita and K. Nishino, "Illumination normalization with time-dependent intrinsic images for video surveillance," IEEE Trans. Pattern Anal. Mach. Intell., 26 (2004), pp. 1336-1348