

На правах рукописи

УДК 004.021; 004.932.2/72'1;
519.682.7; 811.93

КОЗЫРЕВА
Алина Владимировна

**АЛГОРИТМЫ И ПРОГРАММНЫЙ ИНСТРУМЕНТАРИЙ
ДЛЯ ЗАДАЧИ ПОЗИЦИОНИРОВАНИЯ МОБИЛЬНЫХ УСТРОЙСТВ
НА ОСНОВЕ ДАННЫХ ОТ ВСТРОЕННОЙ ФОТОКАМЕРЫ**

05.13.11 – математическое и программное обеспечение
вычислительных машин, комплексов и компьютерных сетей

АВТОРЕФЕРАТ
диссертации на соискание ученой степени
кандидата физико – математических наук

Новосибирск 2010

Работа выполнена в Институте систем информатики
им. А.П. Ершова СО РАН

Научный руководитель: Мурзин Федор Александрович,
кандидат физико – математических наук

Официальные оппоненты: Резник Александр Львович,
доктор технических наук

Пестунов Игорь Алексеевич,
кандидат физико – математических наук

Ведущая организация: Институт вычислительной математики и
математической геофизики СО РАН

Защита состоится 17 декабря 2010г. в 14 ч. 30 мин. на заседании
диссертационного совета ДМ 003.032.01 в Институте систем информатики им.
А.П. Ершова Сибирского отделения РАН по адресу:
630090, г. Новосибирск, пр. ак. Лаврентьева, 6.

С диссертацией можно ознакомиться в читальном зале ИСИ СО РАН
(г. Новосибирск, пр. ак. Лаврентьева, 6).

Автореферат разослан 15 ноября 2010 г.

Ученый секретарь
Диссертационного совета,

к.ф.– м.н.



Мурзин Ф.А.

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Актуальность проблемы

Диссертационная работа посвящена исследованиям, возникшим в связи с реализацией ряда программных средств для локализации положения мобильного устройства в пространстве.

Определение положения любого механизма самим механизмом – сложная проблема. В качестве мобильного устройства могут выступать и телефоны, и КПК, и даже роботы. В общем случае, рассматривается любое устройство, способное перемещаться в пространстве, как самостоятельно, так и с чьей либо помощью, а также это устройство должно обладать «зрением», т.е. иметь возможность каким-либо образом воспринимать окружающее пространство.

Местоположение устройства можно определять различными способами. Например, приемник, использующий технологию GPS (Global Position System) принимает сигналы со спутника, на основе которых приемник вычисляет свои координаты относительно планеты Земля. Далее, на основе двух и более камер можно построить стереоскопическую систему для воспроизведения окружающих трехмерных объектов. Некоторые современные модели телефонов оснащены датчиками перемещения в пространстве. На основе этих датчиков работает множество приложений для мобильных телефонов: игры, управление дополнительными функциями телефона (звук, управление музыкальным плеером и т.д.), управление приложениями телефона (например, смена ориентации дисплея с альбомного на ландшафтный при повороте телефона) и многое другое.

В данной работе рассматривается вариант замены механического устройства на его программную реализацию с использованием видеокамеры телефона. На данный момент практически все мобильные телефоны имеют камеру, в отличие от «гироскопа», что дает возможность таким мобильным устройствам запускать различные приложения. Кроме того, гироскоп — это физическое устройство. И как любое устройство оно занимает некоторое место. Так как телефоны — это достаточно небольшие устройства, то вопрос использования места внутри стоит достаточно остро. При наличии

подходящего программного инструментария, занимаемое гироскопом место можно использовать для чего-то иного или же просто уменьшить объем и вес телефона.

Область использования разработанной библиотеки достаточно широка: игровая индустрия, управление самим устройством, а также бизнес-приложений для него, и такие более «экзотические функции», как защита жестких дисков мобильного устройства при падении.

Так как библиотека разработана на языке C++, то ее можно легко портировать под другие устройства и использовать не только в мобильных телефонах.

Цель диссертационной работы

Цель работы – разработка алгоритмов и создание программных средств, расширяющих интеллектуальные возможности мобильных устройств, и повышение эффективности процессов обработки данных в этих устройствах. Под данными здесь понимаются фотографические изображения.

Основные задачи состояли в создании алгоритмов и программных систем, позволяющих осуществлять следующие функции.

- Калибровку камеры мобильного устройства.
- Быстрый поиск (локализация) фрагмента фотографического изображения в другом изображении.
- Определение изменения положения мобильного устройства в пространстве.

Научная новизна работы

В результате выполненных исследований автором предложены методы решения задачи калибровки камеры мобильного устройства с минимальным участием человека и максимальным использованием предварительных знаний о камере.

Проведенные исследования алгоритмов выделения контуров на

изображениях и алгоритмов поиска «особенностей» на изображениях позволили адаптировать их для использования на мобильных устройствах.

В работе также предложен способ сопоставления «особенностей» двух изображений для дальнейшего расчета матрицы вращения и вектора сдвига, который дает представление о характере перемещения мобильного устройства в пространстве.

Проведенный анализ матрицы вращения позволил предложить алгоритм расчета матрицы, основанный на ее представлении в виде произведения матриц косинусов углов, а также с использованием пар особенностей двух изображений.

В рамках решения поставленной задачи разработан программный пакет, предназначенный для решения поставленной задачи, обладающий следующими функциями:

- калибрование камеры мобильного устройства;
- поиск соответствий на двух изображениях;
- расчет характера движения мобильного устройства на основе полученных изображений.

Практическая значимость

Практическая ценность работы заключается в возможности решения задачи позиционирования мобильного устройства программными методами. Проведенные исследования позволили разработать новый подход к проблеме позиционирования мобильного устройства. В работе представлены методы, которые реализуют поэтапное решение задачи с использованием алгоритмов обработки изображений. Область использования приложения практически так же широка, как и область использования гироскопа. Это может быть игровая индустрия, которая всегда занимала большой пласт в развитии компьютерных технологий. Кроме этого, приложение может быть широко применено в управлении самим устройством, а также бизнес-приложений для него. Например, набор номера, изменение громкости входящего звонка, управление музыкальным плеером и многое другое. Для тех, кто часто пользуется интернетом со своего телефона будет полезной функция смены ориентации дисплея с альбомного на ландшафтный. А могут быть и совсем «экзотические

функции», такие как защита жестких дисков телефона при падении. Кроме того, разработанную библиотеку можно использовать не только в мобильных телефонах, а также в области робототехники или камерах слежения. К разработке, рассмотренной в диссертационной работе, проявила интерес корпорация IBM.

Апробация работы

Результаты работы докладывались и обсуждались на:

- VII международной конференции памяти акад. А.П. Ершова, "Перспективы систем информатики", рабочий семинар "Наукоемкое программное обеспечение", Новосибирск 2009;
- Конференции - конкурсе работ студентов, аспирантов и молодых ученых "Технологии Microsoft в теории и практике программирования", Россия, Новосибирск, 2007;
- XLIV международной научной студенческой конференции "Студент и научно-технический прогресс", Россия, Новосибирск, 2006;
- Научном семинаре в IBM Russian System and Tehnology Lab, Россия, Москва, Май 2007;
- Научном семинаре в IBM Deutschland Research and Development Lab, Германия, Бёблинген, Июль 2010.

Публикации

По результатам диссертации опубликовано 6 работ, включая 1 статью в рецензируемом научном журнале из списка ВАК, 2 статьи в тематическом сборнике, 2 статьи в сборниках трудов международных научных конференций и 1 тезисы.

Структура и объем диссертации

Диссертационная работа состоит из введения, шести глав, заключения и списка литературы. Содержание работы изложено на 111 страницах. Список литературы содержит 76 наименований. В работе содержится 12 таблиц, 22 рисунка и графика, полученных в результате расчетов на ЭВМ.

Во **введении** обосновывается актуальность темы исследований и приводится краткое содержание работы.

В **первой главе** рассказывается о том, что такое смартфон, и чем он отличается от телефона или портативного компьютера. Также обоснован выбор ОС Symbian в качестве основной системы для разрабатываемого приложения.

Смартфоны имеют определенные характеристики, которые отличают их от стандартного персонального компьютера:

1. Ограниченные ресурсы;
2. Надежность;
3. Специфический пользовательский интерфейс;
4. Библиотечная поддержка особенностей смартфонов;
5. Поддержка программ от стороннего разработчика.

Операционные системы для смартфонов должны учитывать данные характеристики. В общем виде архитектура мобильных ОС мало отличается от ОС настольных компьютеров. Основа системы – ядро – включает в себя основные функции системы, поддерживающие работу устройства, на котором система запущена. Программная надстройка над ядром: сервисы ОС – обслуживают ядро системы, реализуя такие функции, как управление приложениями, памятью, настройки системы, мобильная связь и т.д. Следом идут сервисы и библиотеки системы, которые могут использовать приложения, находящиеся на последнем уровне.

Исходя из условий поставленной задачи, были проведены исследования существующих операционных систем для мобильных телефонов по следующим критериям:

- Распространенность смартфонов на базе данной ОС;
- Наличие API для разработки приложений;
- Основной язык разработки приложений для данной ОС;
- Работа с фотокамерой в данной ОС;
- Формат изображений, поступающий с камеры устройства и методы работы с ними.

Распространенность смартфонов. По данным статистической компании Gartner за три последних года и первую половину текущего года (2010) наиболее популярными являются смартфоны на базе ОС Symbian. Данные устройства занимают больше половины рынка. В последние несколько лет становятся популярными смартфоны под управлением ОС iPhone (с 2010 iOS). За 2010 год также резко возросло распространение смартфонов с ОС Android. В то же время устройства под управлением Windows Mobile все менее популярны среди пользователей.

Наличие API для разработки приложений. Каждая из рассмотренных систем имеет набор API для разработки приложений той или иной степени гибкости.

Основной язык разработки приложений для данной ОС. Операционные системы Symbian и iOS позволяют разработчикам писать приложения на языках группы C. Для Symbian это Symbian C++, для iOS – Objective-C. Остальные системы требуют разработки на языках Java, Ruby, Perl. Для поставленной задачи C-языки имеют преимущество, так как одна из их характеристик — это скорость работы, что существенно для данной работы.

Работа с фотокамерой в данной ОС. Работа с камерой во всех ОС имеет сходный характер. Изначально требуется настроить камеру, после чего есть возможность получить изображение сцены в память телефона. ОС Symbian имеет также методы работы с автоматическим фокусом.

Формат изображений, поступающий с камеры устройства и методы работы с ними. Все рассмотренные ОС, кроме Symbian, получают изображения в формате jpeg. В Symbian изображения с камеры поступают в формате bitmap. Тесты показали, что, не смотря на то, что изображения в формате jpeg занимают мало памяти, работа с изображениями в формате bitmap предпочтительнее по причине скорости работы приложений с данным форматом. Так как доступ к таким изображениям можно осуществлять на битовом уровне, избегая дополнительного вызова методов.

На основе проведенного исследования ОС была выбрана система Symbian, которая подошла по всем рассматриваемым критериям.

Во второй и третьей главах описаны основные алгоритмы поиска

особенностей на изображениях и методы их сопоставления.

Для решения поставленной задачи нет необходимости в полном сравнении двух изображений, достаточно рассмотреть лишь некоторую их часть. В качестве таких «частей» могут выступать небольшие участки изображения, которые легко отличимы от своей окрестности. Однако в общем случае для всех точек изображения это невозможно.

Определение. Точечная особенность изображения – это такая точка изображения, окрестность которой отличается от окрестности близлежащих точек по выбранной мере, т.е.

$$\{\forall x : |x' - x| < r \rightarrow \rho(\Omega_x, \Omega_{x'}) > \varepsilon\},$$

где Ω_x – окрестность точки x , называемая окном поиска, а $\rho(\Omega_x, \Omega_{x'})$ – функция близости окрестностей по некоторой мере.

Так называемой точечной особенностью сцены должна соответствовать точечная особенность изображения. Обратное неверно: существуют такие особые точки изображения, которым не соответствует никакие особые точки сцены. Такие точки называются ложными особенностями сцены. Отделение ложных особенностей от настоящих является одной из основных проблем при отслеживании движения камеры или при реконструкции трехмерной сцены.

Большинство детекторов точечных особенностей работают сходным образом: для каждой точки изображения вычисляется некоторая функция от ее окрестности. Точки, в которых эта функция достигает локального максимума, очевидно можно отличить от всех точек из некоторой ее окрестности.

Существует целый набор функций, которые можно использовать для обнаружения точечных особенностей. Чаще всего для задач отслеживания точек сцены применяются функции, находящие в изображении структуры, похожие на уголки (corners). Детекторы, использующие такие функции, называются детекторами углов. Именно они чаще всего применяются для решения задач отслеживания точечных особенностей сцены.

Определение. Угол — пересечение двух ребер или точка, для которой существуют две различные по направлению доминанты в локальной окрестности рассматриваемой точки. Это означает, что точечная особенность может быть углом, но так же она может быть и изолированной точкой с высокой интенсивностью, концы линий на изображении или точкой на кривой где кривизна достигает своего локального экстремума. На практике так называемые угловые детекторы находят особенности в целом лучше, чем отдельные углы. Как результат, если необходимо найти на изображении только углы, то необходимо проводить анализ найденных особенностей на предмет реальности угла.

По результатам тестирования наиболее часто используемых угловых алгоритмов был выбран алгоритм Headly, как алгоритм, показавший наилучшее время.

С точки зрения распознавания и анализа объектов на изображении наиболее информативными являются не значения яркостей объектов, а характеристики их границ – контуров. Другими словами, основная информация заключена не в яркости отдельных областей, а в их очертаниях. Задача выделения контуров состоит в построении изображения именно границ объектов и очертаний однородных областей.

Определение. Контур изображения – совокупность точек, в окрестности которых наблюдается скачкообразное изменение функции яркости.

Так как при цифровой обработке изображение представлено как функция целочисленных аргументов, то контуры представляются линиями шириной, как минимум, в один пиксель. При этом может возникнуть неоднозначность в определении линии контура. Если исходное изображение, кроме областей постоянной яркости, содержит участки с плавно меняющейся яркостью, то введенное определение контура остается справедливым, однако при этом не гарантируется непрерывность контурных линий: разрывы контуров будут наблюдаться в тех местах, где изменение функции яркости не является достаточно резким. С другой стороны, если на «кусочно-постоянном» изображении присутствует шум, то, возможно, будут обнаружены «лишние» контуры в точках, которые не являются границами

областей. При разработке алгоритмов выделения контуров нужно учитывать указанные особенности поведения контурных линий. Специальная дополнительная обработка выделенных контуров позволяет устранять разрывы и подавлять ложные контурные линии.

Тестирование трех различных типов контурных алгоритмов: градиентного, отслеживающего, сканирующего — показало, что сканирующий алгоритм наиболее эффективен для решения поставленной задачи. Окончательное решение об основном алгоритме для поиска особенностей будет сделано позже, после рассмотрения алгоритмов сопоставления особенностей и отбора ложных особенностей.

На следующем шаге необходимо найти соответствия найденным ранее «точкам» на втором изображении. Есть два возможных пути:

- найти особенности на втором изображении и сопоставить их с найденными особенностями первого изображения;
- искать на втором изображении в координатах особенностей первого изображения точки наиболее подходящие по выбранной мере.

На основе анализа существующих способов нахождения пар совпадений (сопряженных пар точек) был разработан новый алгоритм, сочетающий в себе характеристики отслеживающих и сопоставляющих типов алгоритмов. Найденные особенности первого и второго изображений сопоставляются в области слежения. Данный вариант целесообразен по следующей причине. Мобильное устройство перемещается в пространстве, и набор особенностей получаемых изображений будет постоянно меняться — часть особенностей будет выходить за границы изображений, и вместо них будут появляться новые, а часть особенностей будет «подавляться» новыми более яркими особенностями. Таким образом, «особенности» должны быть известны на обоих изображениях независимо от выбранного варианта сопоставления.

В качестве меры, по которой происходит сопоставление окрестностей точек, можно выбрать различные функции: от простого сравнения яркостей или цветов в выбранных областях до кросс-корреляции для полной идентификации областей. Все три рассмотренных варианта (корреляция,

свертка и сравнения яркостей) показали близкие результаты, но алгоритм корреляции был немного быстрее, хоть и сложнее в реализации.

В тестировании алгоритмов сопоставления участвовали особенности найденные угловым детектором Headly и сканирующим алгоритмом для определения контуров. Установлено, что для задачи отбора «некачественных» особенностей контурные алгоритмы показали лучшие результаты, чем алгоритмы, базирующиеся на угловых детекторах. А именно, если «концы» контура имеют пары на втором изображении, то и оставшиеся точки контура должны иметь пары на втором изображении находящиеся на отрезке, соединяющем пары для «концов».

После того как найдены особенности на двух изображениях, можно определить, как изменилось положение камеры. Об алгоритмах решающих эту задачу рассказано в **четвертой главе**.

Трехмерная точка сцены \mathbf{M} и двумерная точка изображения \mathbf{m} связаны следующим равенством:

$$\mathbf{M} = K\mathbf{m},$$

где K – так называемая калибровочная матрица камеры или матрица внутренних параметров камеры. Данная матрица имеет следующий вид:

$$K = \begin{bmatrix} f/w & \gamma & u_0 \\ 0 & f/h & v_0 \\ 0 & 0 & 1 \end{bmatrix},$$

где $(u_0 \ v_0)$ – сдвиг системы координат в фотоприемнике, γ — коэффициент асимметрии между осями, f – фокусное расстояние камеры, а w и h — масштабные коэффициенты по осям x и y соответственно.

Существует много алгоритмов позволяющих с разной долей участия пользователя определить матрицу параметров камеры. Например, Zhengyou Zhang в своей работе «Новая гибкая технология калибровки камеры» рассмотрел метод калибровки камеры с использованием плоского шаблона с 2-3 различных позиций. Характер движения камеры при этом не обязательно должен быть известен. Шаблон для калибровки выполнен в виде «шахматной доски». Прэтт в своем труде «Цифровая обработка

изображений» описал метод самокалибровки стационарной камеры. Данный метод итеративен и улучшает полученные значения калибровочной камеры с каждым новым изображением наблюдаемой сцены.

На основе анализа различных методов калибровки, а также исходя из требования к приложению использовать в минимальной степени помощь пользователя, был предложен следующий метод.

Во-первых, процесс расчета калибровочной матрицы вынесен за пределы всех алгоритмов, касающихся вычисления поворота и сдвига камеры, т.к. данный процесс проходит только один единственный раз по причине того, что внутренние параметры камеры не меняются, и вычислять их каждый раз заново не имеет смысла.

Для калибровки камеры нам потребуется помощь пользователя программы, так как провести все необходимые замеры, не используя внешнюю помощь, не представляется возможным. При первом запуске программы пользователю будет предложено провести калибровку путем перемещения мобильного устройства в пространстве по «квадрату» на основе указаний программы. Изображения при этом смартфон будет делать самостоятельно в углах данного «квадрата», размеры которого заранее известны. Так как движение в данном случае происходит без вращения, то должны соблюдаться следующие равенства:

$$\begin{cases} X_1 - U_1 \approx X_2 - U_2 \approx X_3 - U_3 \approx X_4 - U_4 \\ Y_1 - V_1 \approx Y_2 - V_2 \approx Y_3 - V_3 \approx Y_4 - V_4 \\ Z_1 - W_1 \approx Z_2 - W_2 \approx Z_3 - W_3 \approx Z_4 - W_4 \end{cases},$$

где $(X_i \ Y_i \ Z_i)$ – координаты точки в системе координат первого изображения, $(U_i \ V_i \ W_i)$ – координаты точки в системе координат второго изображения.

Откуда после небольших преобразований с использованием уравнения связи между двумерной точкой и трехмерной, можно получить значения пар f/w и f/h . Для оценки коэффициента γ и пары $(u_0 \ v_0)$ будем использовать знание того, что смартфоны обладают «классическими» камерами, что означает, что γ будет стремиться к нулю, так как асимметрия осей в камерах мобильных телефонов отсутствует, а координаты точки прохождения

фокусной прямой через оптическую плоскость (u_0, v_0) будут близки к значениям центральной точки изображений.

В процессе тестирования разрабатываемого приложения была замечена проблема изменения фокусного расстояния при изменении сцены. Ее решение рассмотрено в пятой главе данной работы. На данном этапе оценки калибровочной матрицы стоит упомянуть, что большинство алгоритмов оценивают коэффициенты f/w и f/h в паре. В данной работе на этапе калибровки камеры вычисляются значения w и h в явном виде, используя функции ОС Symbian. В процессе получения снимков будет уставлено использование автофокусировки. В последствии методом $GetFocalLengthInfoL(TInt & aMinFocalLength, TInt & aCurrentFocalLength, TInt & aMaxFocalLength)$ будет получено значение используемого фокусного расстояния, благодаря которому станут известны значения масштабных коэффициентов.

Для расчета матрицы поворота и вектора сдвига введем для каждого положения камеры свою стандартную систему координат. Пусть первому положению соответствует система координат $O'X'Y'Z'$, а второму – $O''X''Y''Z''$. Пусть вектор $\mathbf{M}' = (X, Y, Z)^T$ характеризует координаты некоторой точки M трехмерного пространства в системе первой камеры, а вектор $\mathbf{M}'' = (U, V, W)^T$ - во второй системе. Тогда легко показать, что связь между векторами \mathbf{M}' и \mathbf{M}'' задается соотношением

$$\mathbf{M}'' = \mathbf{R}\mathbf{M}' + \mathbf{t}.$$

\mathbf{R} и \mathbf{t} называются внешними параметрами камеры и характеризуют собой поворот камеры и вектор сдвига соответственно.

Существует множество систем оценки матрицы \mathbf{R} и вектора \mathbf{t} . Мы будем предполагать, что в каждый момент времени происходит либо сдвиг, либо вращение камеры. Это даст нам возможность уменьшить сложность вычислений и, как результат, уменьшить время работы приложения.

Предварительно необходимо проверить выполняется ли для любых четырех точек-особенностей первого изображения и соответствующим им особенностям второго изображения следующее равенство:

$$\begin{cases} X_1 - U_1 \approx X_2 - U_2 \approx X_3 - U_3 \approx X_4 - U_4 \\ Y_1 - V_1 \approx Y_2 - V_2 \approx Y_3 - V_3 \approx Y_4 - V_4 \\ Z_1 - W_1 \approx Z_2 - W_2 \approx Z_3 - W_3 \approx Z_4 - W_4 \end{cases} .$$

Если оно выполнимо, то характер движения камеры можно описать как движение вдоль одной из осей. Если равенство не выполняется, то имеет место вращение мобильного телефона. В данном случае необходимо производить расчет матрицы поворота. Рассмотрим одну из систем:

$$\begin{cases} X_1 = r_{11}U_1 + r_{12}V_1 + r_{13}W_1 + t_1 \\ X_2 = r_{11}U_2 + r_{12}V_2 + r_{13}W_2 + t_1 \\ X_3 = r_{11}U_3 + r_{12}V_3 + r_{13}W_3 + t_1 \\ X_4 = r_{11}U_4 + r_{12}V_4 + r_{13}W_4 + t_1 \end{cases}$$

Применяя формулы связи двумерных и трехмерных координат точки, а также теорию однородных координат получаем

$$A \begin{pmatrix} r_{11} \\ r_{12} \\ r_{13} \\ t_1 \\ -1 \end{pmatrix} = 0, \quad A = \begin{pmatrix} (u_1 - u_0)/\alpha & (v_1 - v_0)/\alpha & 1 & 1 & (x_1 - u_0)/\alpha \\ (u_2 - u_0)/\alpha & (v_2 - v_0)/\alpha & 1 & 1 & (x_2 - u_0)/\alpha \\ (u_3 - u_0)/\alpha & (v_3 - v_0)/\alpha & 1 & 1 & (x_3 - u_0)/\alpha \\ (u_4 - u_0)/\alpha & (v_4 - v_0)/\alpha & 1 & 1 & (x_4 - u_0)/\alpha \end{pmatrix}$$

Решив данную систему, будут известны значения r_{11} и r_{12} . Далее, исходя из формулы записи матрицы вращения R через косинусы и синусы углов поворота, можно найти угол поворота вокруг оси x . Аналогичным образом находятся углы по оси y и z .

В **пятой главе** приведены дополнения и упрощения, благодаря которым, алгоритмы работают быстрее и эффективнее.

В процессе тестирования разрабатываемого приложения была замечена проблема изменения фокусного расстояния при изменении сцены. Для решения данной проблемы возможны два варианта:

1. Вычислять фокусное расстояние при каждом поступившем изображении.
2. Принять фокусное расстояние за константу.

Оба варианта имеют свои недостатки. На изображениях с разным фокусным расстоянием могут быть абсолютно разные особенности. Часть особенностей сцены будет в фокусе и, как правило, расплывчато, в этом случае стандартные алгоритмы нахождения особенностей не справляются с задачей. Таким образом, можно получить два абсолютно разных набора особых точек двух изображений, на основе которых будет затруднительно вычислить поворот и сдвиг. Второй вариант, всегда использовать одно и то же фокусное расстояние также не удобен, так как один раз пользователь может запустить программу и самым оптимальным будет расстояние очень большое, практически бесконечность, а в другой раз оптимальным может оказаться микро-расстояние. Данную проблему можно решить путем вынесения оценки фокусного расстояния в предварительный анализ сцены. Этот вариант подходит в наибольшей степени, и плохо работает только в случае, когда фокусное расстояние меняется очень резко в течение короткого времени работы приложения.

В ОС Symbian получить фокусное расстояние можно процедурой объекта «камера» *GetFocalLengthInfoL(TInt & aMinFocalLength, TInt & aCurrentFocalLength, TInt & aMaxFocalLength)*. Благодаря этому при каждом запуске программы, известно текущее фокусное расстояние. Более того, процедурой *void SetFocusDistance (TInt aDistance)* устанавливается фиксированное значение фокусного расстояния с отменой автофокусировки. Таким образом, с известным фокусным расстоянием будет известна полностью калибровочная матрица, актуальная для данного момента запуска приложения.

Имеет смысл через определенный промежуток времени изменить фокусное расстояние. Таким образом, мы будем обновлять значение не на каждом изображении, но каждые 10-15 снимков. Что позволит нам иметь четкое изображение без больших затрат на память и время.

Так же в процессе тестирования разработанного приложения было отмечено, что количество найденных особенностей зависит от яркости сцены. Так изображения с небольшими перепадами яркости имеют небольшое количество «особенностей» в отличие от изображений с полным набором значений яркостей.

Для решения данной проблемы предложено следующее решение. На этапе предварительного анализа сцены происходит расчет средневзвешенного уровня яркости пикселей изображения, который получается путем умножения каждого уровня яркости на число пикселей данного уровня, а затем делится на общее число уровней яркости. На основе данного средневзвешенного уровня яркости устанавливается порог, по которому алгоритм нахождения особенностей признает тот или иной перепад яркостей за особенность.

Кроме предварительного анализа, были внесены следующие изменения в непосредственную работу с изображениями.

Работа с небольшим изображением позволяет, во-первых, уменьшить влияние шума на изображении, во-вторых, позволяет рассматривать физически более крупные участки сцены. Обычно участок 10x10 точек 1-мегапиксельного снимка соответствует сцене 30x30 сантиметров. Участок того же размера, но на снимке 400x300, соответствует сцене 260x260 сантиметров. Кроме того, время обработки такого изображения значительно уменьшается, что является положительным моментом в данном исследовании. Дополнительный плюс в использовании небольших изображений в том, что память мобильных устройств сильно ограничена. В результате очень затруднительно хранить в памяти два больших изображения размером 1200x1600, приходится тратить дополнительное время на размещение изображения в постоянной памяти и работу с ним. Таким образом удобнее, когда с камеры поступают изображения размером 240x320.

Также есть смысл разделить изображение на непересекающиеся блоки и выполнять поиск особенностей в каждом блоке отдельно. Данный подход имеет несколько положительных моментов:

- найденные «особенности» будут располагаться в разных частях изображения, а не сгруппированы в одной наиболее яркой части;
- найденные особенности могут иметь разные значения яркости, только потому, что в данном блоке данная точка действительно максимально отличается от окрестности, хотя в сравнении с

- хотя ОС Symbian не приветствует работу многопоточных приложений, в будущем использование многоядерных процессоров в мобильных телефонах будет повсеместным, что даст разработанному приложению дополнительный временной выигрыш.

В **шестой главе** речь идет об области применения созданной библиотеки. Так же представлено сравнение с существующим физическим устройством «гироскопом» - устройством, способным измерять изменение углов ориентации связанного с ним тела относительно инерциальной системы координат, как правило, основанное на законе сохранения вращательного момента (момента импульса).

На данный момент некоторые модели телефонов оснащены тем или иным видом гироскопа. Фотокамерой же оснащены практически все модели телефонов. При этом условии практическая ценность разработанной библиотеки высока, благодаря ней пользователь может пользоваться всеми теми преимуществами, что дает гироскоп. Программное решение имеет такой недостаток как невозможность решить поставленную задачу при быстром и резком перемещении телефона в пространстве. Однако, гироскоп — это физическое устройство. И как любое устройство оно занимает некоторое место. Так как телефоны — это достаточно небольшие устройства, то вопрос с использованием места внутри стоит достаточно остро. При наличии приложения место, занимаемое гироскопом можно использовать для чего-то иного или же просто уменьшить объем и вес телефона. Кроме того, гироскопы, как правило, создаются сторонними компаниями и за них компаниям-производителям мобильных телефонов приходится платить, что в конечном итоге может сказаться на цене мобильного телефона.

Область применения разработанной библиотеки практически так же широка как область применения гироскопов: мобильные телефоны, робототехника, камеры слежения.

О ЛИЧНОМ ВКЛАДЕ АВТОРА

Все основные результаты диссертации автором получены самостоятельно.

ОСНОВНЫЕ РЕЗУЛЬТАТЫ

1. На основе анализа существующих способов нахождения пар совпадений (сопряженных пар точек) был разработан новый алгоритм, сочетающий в себе характеристики отслеживающих и сопоставляющих типов алгоритмов. Найденные особенности первого и второго изображений сопоставляются в области слежения. Установлено, что для задачи отбора «некачественных» особенностей контурные алгоритмы показали лучшие результаты, чем алгоритмы, базирующиеся на угловых детекторах.
2. Предложен алгоритм предварительного анализа сцены, позволяющий уменьшить общее время обработки изображений. В алгоритме оценивается общая яркость сцены, на основе которой устанавливается определенный порог, по которому алгоритм нахождения особенностей признает тот или иной перепад яркостей за особенность. В частности, в данном алгоритме сцена с нормально распределенной гистограммой имеет более высокий порог яркости, чем сцена со смещенной гистограммой.
3. Проведен анализ способов калибровки камеры с участием и без участия человека. На основе проведенного анализа предложен оригинальный алгоритм расчета матрицы калибровки камеры. А именно, при первичном расчете матрицы калибровки вычисляются масштабы вдоль горизонтальной и вертикальной координатных осей. В дальнейшем, окончательные значения элементов матрицы рассчитаются в зависимости от результатов предварительного анализа сцены. Анализ сцены также позволяет получать точное значение фокусного расстояния, которое считается фиксированным на период работы приложения.
4. Создана программная библиотека для смартфонов под управлением ОС Symbian, которая на основе снимков, поступающих с видеокамеры, производит расчет матрицы поворота и вектора сдвига.

Выработаны требования и определены границы применения разработанной библиотеки, на основе которых предложена модель допустимых упрощений, при которых разработанное приложение может работать быстрее и эффективнее, экономя процессорное время и память мобильного устройства. Проведено тестирование разработанной библиотеки на искусственных и экспериментальных данных.

Список публикаций

1. Козырева А.В. Алгоритмы позиционирования мобильного устройства на основе данных от встроенной фотокамеры // Известия Томского политехнического университета. – 2010. – Т. 316. – № 5. – С. 83–90.
2. Козырева А.В. Определение координат мобильного устройства в пространстве на основе изображений, получаемых от его видеокамеры // Проблемы интеллектуализации и качества систем информатики. – Новосибирск, 2006. – С. 113-131.
3. Козырева А.В. О некоторых способах калибровки видеокамеры // Проблемы интеллектуализации и качества систем информатики. – Новосибирск, 2006. – С. 131-141.
4. Козырева А.В. О задаче позиционирования мобильного устройства на основе изображений, полученных от установленной на нем фотокамеры // VII междунар. конф. памяти акад. А.П. Ершова, "Перспективы систем информатики", Рабочий семинар "Научное программное обеспечение", Новосибирск 2009, – С. 173-180.
5. Козырева А.В. О задаче позиционирования мобильного устройства в пространстве на основе изображений, получаемых от его видеокамеры // Технологии Microsoft в теории и практике программировании / Конференция-конкурс работ студентов, аспирантов и молодых ученых. – Новосибирск, 2007. – С. 121-122.

6. Козырева А.В. Определение координат мобильного устройства в пространстве на основе изображений, получаемых от его видеокамеры // Материалы XLIV международной научной студенческой конференции "Студент и научно-технический прогресс", Информационные технологии. - Новосибирск, 2006. – 1с.

Козырева А.В. 

Козырева А.В.

**АЛГОРИТМЫ И ПРОГРАММНЫЙ ИНСТРУМЕНТАРИЙ ДЛЯ
ЗАДАЧИ ПОЗИЦИОНИРОВАНИЯ МОБИЛЬНЫХ УСТРОЙСТВ**

Автореферат

Подписано в печать

Объем , уч.-изд. л.

Формат бумаги 60 · 90 1/16

Тираж экз.

Отпечатано в ЗАО РИЦ «Прайс-курьер»

630090, г. Новосибирск, пр. ак. Лаврентьева, 6, тел. 34-22-02

Заказ № 235