Модел. и анализ информ. систем. Т. **20**, № **2** (2013) 129–138
©*Благовещенский И.А., Демьянков Н.А.,* 2013

УДК 004.932

Технологии и алгоритмы для создания дополненной реальности [1]

Благовещенский И.А., Демьянков Н.А.

ООО "Ареал", AREALIDEA,
150044 г. Ярославль, Полушкина роща, 16, строение 67а, офис №2-6

e-mail: ivan.blagoveshchenskiy@arealsoft.ru, nikolay.demyankov@arealsoft.ru

получена 3 марта 2013

Ключевые слова: дополненная реальность, компьютерное зрение, feature detection, маркеры, OpenCV

В данной статье авторами рассматривается понятие дополненной реальности, а также возможные методы её создания. Вначале даётся короткая историческая справка о том, откуда пошло понятие "дополненная реальность кем оно было введено и что означает. Выделяются два принципа её построения: на основе маркера и без него. В статье мы уделяем основное внимание первому подходу. Для анализа видеопотока и поиска на нём объектов определённого типа используются методы и алгоритмы научной дисциплины под названием "компьютерное зрение". В рамках статьи авторы приводят короткое описание и характеристику только двух из них: генетические алгоритмы и feature detection & description. Для программной реализации описываемых алгоритмов может быть использована одна из приведённых библиотек компьютерного зрения: OpenCV или AForge.NET. Обе они дают широкие функциональные возможности в области обработки изображений и поиска объектов. В качестве завершения приводится пример построения дополненной реальности при помощи OpenCV. Основное внимание уделяется вопросам проецирования 3D модели на изображение-маркер. Данный пример может быть взят за основу создания собственного фреймворка дополненной реальности.

1. Введение в дополненную реальность. Существует множество различных определений того, что такое "дополненная реальность". Считается, что данный термин был предложен работавшим на корпорацию Boeing исследователем Томом Коделом в 1990 году. В 1994 году Пол Милгром и Фумио Кисино [8] описали Континуум Виртуальность-Реальность (англ. Milgram's Reality-Virtuality Continuum) — пространство между реальностью и виртуальностью, между которыми расположены дополненная реальность (ближе к реальности) и дополненная виртуальность (ближе к виртуальности). Более простое определение дал исследователь Рональд Азума [2] в 1997 году. Он определил дополненную реальность как систему, которая:

[1]Статья создана при финансовой поддержке Министерства образования и науки. Контракт №14.514.11.4029 от 10.08.2012 г.

- совмещает виртуальное и реальное;
- взаимодействует в реальном времени;
- располагается в трехмерном пространстве.

Дополненная реальность (англ. augmented reality, AR) является определенной разновидностью виртуальной реальности (англ. virtual reality, VE). Технологии виртуальной реальности полностью погружают человека в синтетическую среду. Будучи погруженным, он не может видеть окружающий его реальный мир. В отличие от виртуальной реальности, дополненная реальность позволяет человеку ощущать реальный мир вместе с виртуальными объектами, наложенными на окружающую обстановку. Таким образом, AR интегрируется и дополняет настоящий мир вместо того, чтобы полностью его заменить.

2. Маркеры дополненной реальности. Можно выделить два главных принципа построения дополненной реальности:

- на основе маркера;
- на основе координат местоположения пользователя.

Безмаркерные технологии зачастую применяется в мобильных устройствах, и строятся посредством специальных датчиков: акселерометр, гироскоп, магнетометр, GPS-приёмник. Подобный подход в рамках нашей статьи рассматриваться не будет. Основное внимание мы уделим построению дополненной реальности с помощью маркеров и алгоритмов компьютерного зрения.

Под маркером понимается объект, расположенный в окружающем пространстве, который находится и анализируется специальным программным обеспечением для последующей отрисовки виртуальных объектов. На основе информации о положении маркера в пространстве, программа может достаточно точно спроецировать на него виртуальный объект, от чего будет достигнут эффект его физического присутствия в окружающем пространстве. Используя дополнительные графические фильтры и высококачественные модели, виртуальный объект может стать практически реальным и трудно отличимым от остальных элементов интерьера или экстерьера.

Зачастую в роли маркера выступает лист бумаги с некоторым специальным изображением. Тип рисунка может варьироваться достаточно сильно и зависит от алгоритмов распознания изображений. Вообще говоря, множество маркеров достаточно широко: ими могут быть и геометрические фигуры простой формы (например, круг, квадрат), и объекты в форме прямоугольного параллелепипеда, и даже глаза и лица людей.

3. Компьютерное зрение. Теория компьютерного зрения (англ. computer vision) является основополагающей для развития технологий дополненной реальности, и прежде всего в области использования маркеров. Основное направление данной дисциплины – это анализ и обработка изображений (в том числе и видеопотока). Алгоритмы компьютерного зрения позволяют выделять ключевые особенности на изображении (углы, границы области), производить поиск фигур и объектов в реальном времени, выполнять 3D реконструкцию по нескольким фотографиям и многое другое.

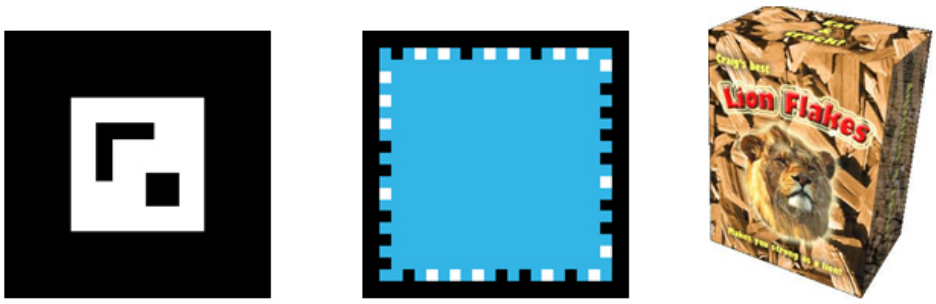

Рис. 1. Примеры маркеров

В области дополненной реальности алгоритмы компьютерного зрения используются для поиска в видеопотоке специальных маркеров. В зависимости от задачи в качестве маркера могут выступать как специально сформированные изображения, так и лица людей. После нахождения маркера в видеопотоке и вычисления его местоположения, появляется возможность построения матрицы проекции и позиционирования виртуальных моделей. С помощью них можно наложить виртуальный объект на видеопоток таким образом, что будет достигнут эффект присутствия. Основная сложность как раз и состоит в том, чтобы найти маркер, определить его местоположение в кадре и спроецировать соответствующим образом виртуальную модель.

За последнее десятилетие была создана большая теоретическая база в сфере обработки изображений и поиска на нём различных объектов (см., например, [3] – [7], [9] – [11] и приведенную там литературу). Прежде всего, это касается методов контурного анализа, template matching, feature detection и генетических алгоритмов. С точки зрения построения дополненной реальности зачастую используются последние два подхода. Дадим небольшие пояснения по каждому из них.

Генетические алгоритмы. Генетические алгоритмы – это эвристические алгоритмы поиска, используемые для решения задач оптимизации и моделирования путём случайного подбора, комбинирования и вариации искомых параметров с использованием механизмов, напоминающих биологическую эволюцию.

В компьютерном зрении они используются для поиска объекта некоторого заданного класса на статическом изображении или видеопотоке. Вначале необходимо провести обучение алгоритма при помощи двух различных наборов изображений:

1. “Хорошие” – содержат нужный объект.

2. “Плохие” – ложные изображения без искомого объекта.

При этом для обучения используется большое число изображений, и чем их больше – тем лучше будет работать сам алгоритм. Для каждой картинки производится выделение различных ключевых особенностей: границы, линии, центральные элементы.

По ним производится построение статистической модели, которая затем и используется для поиска объекта на изображении.

Примером использования данного подхода может служить алгоритм распознания лиц и глаз на видеопотоке. Постепенно обучая алгоритм, можно добиться вы-

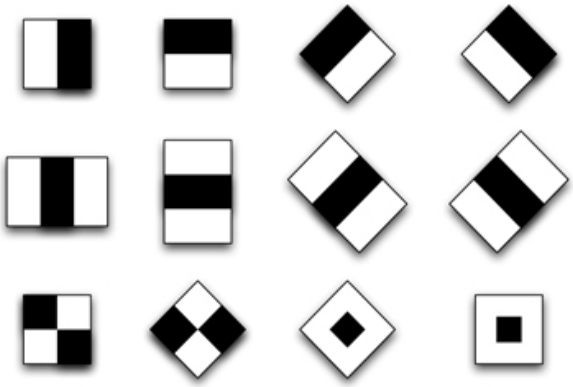

Рис. 2. Примитивы Хаара, используемые в алгоритме

соких результатов нахождения заданного класса объектов. Однако необходимость обучения как раз и делает использование генетических алгоритмов достаточно проблематичным. Для их хорошей работы требуется существенное число различных изображений (как “хороших”, так и “плохих”), и построение классификатора для каждого объекта может занимать продолжительное время.

Feature detection. Концепция feature detection в компьютерном зрении относится к методам, которые нацелены на вычисление абстракций изображения и выделения на нем ключевых особенностей. Данные особенности могут быть как в виде изолированных точек, так и кривых или связанных областей. Не существует строгого определения того, что такое ключевая особенность изображения. Каждый алгоритм понимает под этим своё (углы, грани, области и т.п.).

Зачастую для поиска маркеров используются алгоритмы, которые выполняют поиск и сравнение изображений по ключевым точкам. Ключевая точка – это некоторый участок картинки, который является отличительным для заданного изображения. Что именно принимается за данную точку – напрямую зависит от используемого алгоритма.

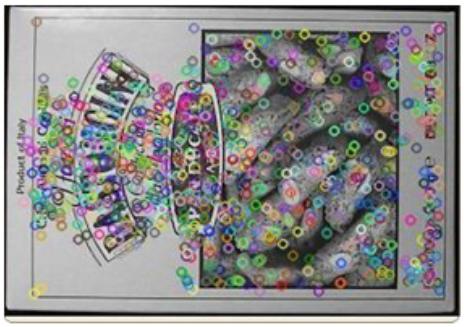

Рис. 3. Пример ключевых точек на изображении

Для их нахождения и последующего сравнения используются три составляющие:

- Детектор (англ. feature detector) – осуществляет поиск ключевых точек на изображении.

- Дескриптор (англ. descriptor extractor) – производит описание найденных ключевых точек, оценивая их позиции через описание окружающих областей.
- Матчер (англ. matcher) – осуществляет построение соответствий между двумя наборами точек.

Вначале при помощи детектора производится поиск ключевых точек шаблонного (искомого) изображения. Полученные точки затем описываются посредством дескриптора. Данная информация сохраняется в отдельный файл (или базу данных), чтобы не выполнять данный процесс повторно. При обработке видеопотока с целью поиска заданного шаблона описанный процесс выполняется для каждого кадра (за исключением сохранения данных). Для установления соответствия между ключевыми точками и дескрипторами применяется матчер.

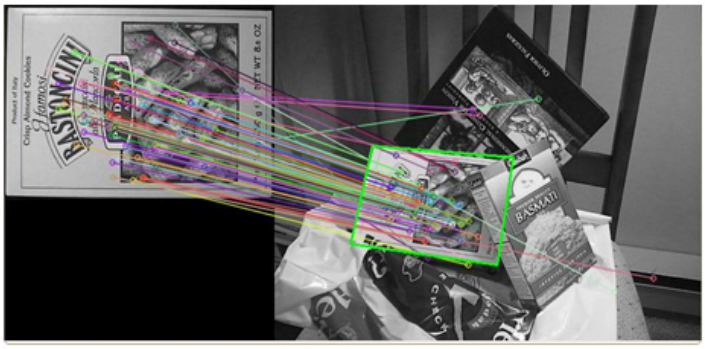

Рис. 4. Соответствия между точками шаблона и тестируемого изображения

Естественно предположить, что разные алгоритмы работают с разной скоростью и эффективностью. В условиях применения их для построения дополненной реальности необходимо использовать только те, которые показывают высокую скорость работы при достаточно хорошем качестве отслеживания позиций ключевых точек. В противном случае мы можем получить заметные отставания у снимаемых видеоданных.

Для повышения скорости работы алгоритмов feature points detection применяются различные способы фильтрации точек, дабы минимизировать их число и отсеять совсем плохие сочетания. Таким образом, можно добиться не только повышения скорости работы алгоритмов, но и качества трекинга маркеров.

4. Библиотеки компьютерного зрения. За прошедшее десятилетие различными организациями и научными сообществами было создано несколько библиотек компьютерного зрения. Поскольку вопрос производительности является достаточно острым в данной дисциплине, то большинство из них написаны на языке C++. Помимо хорошей скорости выполнения это также даёт возможность их применения на различных платформах.

OpenCV. В настоящее время OpenCV (англ. Open Source Computer Vision Library) является, наверное, самой известной и развитой библиотекой компьютерного зрения. В её состав входят модули:

- обработки изображений;

- построения простых пользовательских интерфейсов, загрузки/сохранения видео- и аудиоданных;
- анализа движения и отслеживания объектов (оптический поток, шаблоны движения, устранение фона);
- калибровки камеры, поиска стереосоответствий и элементов обработки трехмерных данных;
- поиска, анализа и сравнения ключевых точек изображений (Feature Detection and Description);
- детектирования объектов на изображении (вейвлеты Хаара, HOG и т. д.);
- методы и модели машинного обучения (SVM, деревья принятия решений и т.д.).

OpenCV выпускается под лицензией BSD с открытым исходным кодом, в результате чего её разрешено использовать как для академических целей, так и для коммерческих. Разработка библиотеки началась в 1999 году и продолжается по сей день. Многие проекты, затрагивающие компьютерное зрение, пользуются именно данной библиотекой. Стоит отметить, что немалое число библиотек дополненной реальности так или иначе используют реализации алгоритмов, созданных в OpenCV.

AForge.NET. AForge.NET является C# фреймворком с открытым исходным кодом, созданным для разработчиков и исследователей в области компьютерного зрения и искусственного интеллекта. Основным идеологом и разработчиком данного проекта является один человек (Андрей Кириллов), однако на протяжении всего периода свой вклад вносили разработчики из различных стран мира.

Фреймворк включает в себя следующие компоненты:

- AForge.Imaging – обработка изображений и набор различных фильтров;
- AForge.Vision – набор методов и алгоритмов компьютерного зрения;
- AForge.Video – обработка видеопотока;
- AForge.Neuro – построение и работа с нейронными сетями;
- AForge.Genetic – набор генетических алгоритмов;
- AForge.Robotics – специальный набор методов для применения в области робототехники;

и ряд других.

На основе AForge.NET была разработана библиотека Gratf, используемая для построения дополненной реальности. Она написана на языке C# и обладает хорошей переносимостью на различные платформы.

Помимо указанных выше, существует ряд других кроссплатформенных библиотек компьютерного зрения. Среди них можно выделить ROS (Robot Operating

System) – библиотека с открытым исходным кодом, применяемая для создания программного обеспечения роботов. VXL, Integrating Vision Toolkit, ViSP - C++ фреймворки с набором модулей по обработке и анализу изображений, видеопотока, поиска шаблонов и объектов, классификаторами и многим другим. Список подобных библиотек достаточно широк, и немалое число из них построены на основе OpenCV. По этой причине перечислять их не имеет смысла.

5. Построение дополненной реальности. В основе данного пункта лежит статья [1]. Для решения поставленной задачи мы воспользуемся описанной выше библиотекой OpenCV. С помощью неё будет производиться поиск маркера на видеопотоке с последующим отображением виртуального объекта. Непосредственно сам процесс поиска мы описывать не будем, поскольку для этого используются алгоритмы feature point detection, описанные выше. Прежде всего, нас интересует вопрос построения 3D пространства по найденной 2D гомографии.

Под гомографией понимается матрица построения взаимно-однозначного отображения проективного пространства на себя, переводящего точки в точки, прямые в прямые и сохраняющего отношение инцидентности точек и прямых, а также двойное отношение любой четверки коллинеарных точек. Она позволяет приводить изображения к единой перспективе и геометрии.

Для построения 3D пространства нам необходимо знать 2 матрицы: внутреннюю (intrisic matrix) и внешнюю (extrinsic matrix). Затем, воспользовавшись методами OpenGL, можно без каких-либо проблем нарисовать виртуальный объект поверх маркера.

Внутренняя матрица (или матрица проекции) состоит из параметров используемой камеры: фокального расстояния по двум осям (f_x, f_y) и координат центра фокуса (c_x, c_y).

$$\begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$$

Процесс нахождения параметров камеры называется ее калибрацией. Для его проведения необходимо сделать снимки калибровочного шаблона (шахматной доски) с помощью используемой камеры, перенести фотографии на компьютер и произвести подсчёт параметров матрицы. С целью получения как можно более точных параметров необходимо соблюдение следующих условий:

- Шаблон шахматной доски распечатывается на чистом листе формата A4.
- Лист шаблона должен лежать ровно на поверхности, края не должны быть загнутыми, желательно отсутствие каких-либо изгибов.
- Размер снимков с камеры должен быть приведен к размеру кадра видеопотока. В некоторых случаях возможна ситуация, когда разрешение фотографий выше разрешения видео. Поэтому перед калибровкой необходимо уменьшить снимки до нужного размера.
- Количество снимков шаблона должно быть не меньше 10. Выполнить их нужно с разных ракурсов. Чем больше будет снимков, тем более точные будут получены параметры матрицы проекции, а эта точность далее будет влиять на наличие/отсутствие сдвигов при построении 3D объектов.

Внешняя матрица (или матрица модели) – это матрица преобразований модели посредством растяжения, поворота и переноса. Она позволяет однозначно задать положение объекта в пространстве.

$$\begin{bmatrix} r_1 1 & r_1 2 & r_1 3 & t_1 \\ r_2 1 & r_2 2 & r_2 3 & t_2 \\ r_3 1 & r_3 2 & r_3 3 & t_3 \end{bmatrix}$$

Диагональные элементы отвечают за растяжение модели. Остальные элементы r – за поворот объекта в пространстве. Элементы t отвечают за перенос. В целом структура данной матрицы может варьироваться в зависимости от уравнений преобразований координат, но описанные ключевые элементы в ней присутствуют всегда.

Вычисление её производится путём определения четырёх пар точек объекта и соответствующего его положения в кадре. Точки положения в кадре – это вершины четырехугольника, описывающего (ограничивающего) объект в кадре. Получить данные точки можно, если подействовать на крайние точки шаблона гомографией.

Относительно точек самого объекта стоит отметить несколько моментов:

- Точки объекта задаются в 3D, а точки на кадре в 2D. Соответственно, если задать им ненулевое значение z, то начало координат на z будет сдвинуто относительно плоскости объекта на кадре.

Рис. 5. Положение объекта на маркере с направлениями осей координат

- Точки объекта задаются таким образом, чтобы было удобно работать в 3D пространстве, а именно: начало координат находится прямо по центру шаблона, а единица длины равняется половине меньшей стороны. В этом случае мы исключаем зависимость от конкретных размеров шаблона в пикселях, а 3D пространство будет отмасштабировано по меньшей стороне.

Сконструированные матрицы затем используются для вычисления вектора поворота и переноса с последующим занесением в матрицу модели. Стоит отметить,

что процесс вычисления матрицы модели занимает несколько миллисекунд и существенно не влияет на скорость работы программы. Основная нагрузка идёт в области анализа видеопотока и поиска на нём шаблонного изображения.

Описанный выше подход можно применить ко всем библиотекам компьютерного зрения в плане построения дополненной реальности. С практической точки зрения он выглядит несколько затруднительным ввиду необходимости проведения калибровки камеры для получения корректной матрицы проецирования. Одним из способов решения данной проблемы является создание набора параметров различных камер; в зависимости от используемой камеры будет производиться загрузка необходимых параметров.

Развивая данное направление, можно вполне построить свой собственный фреймворк дополненной реальности. Разрабатывая различные способы фильтрации точек, можно добиться того, что маркер будет находиться даже при сильном угле наклона и большом удалении. Это позволит получить более реалистичную картинку и даст возможность пользователям рассмотреть модель со всех сторон на разном расстоянии.

Список литературы

1. *Степанов А.В.* 2D->3D in Augmented reality // http://habrahabr.ru/post/139429/, 2012 (Stepanov A.V. 2D->3D in Augmented reality // http://habrahabr.ru/post/139429/, 2012 [in Russian]).

2. *Azuma R.* A Survey of Augmented Reality // Presence: Teleoperators and Virtual Environments, 1997. P. 355–385.

3. *Amit Y.* 2D Object Detection and Recognittion: Models, Algorithms and Networks. The MIT Press, 2002.

4. *Bay H., Tuytelaars T., L. Van. Gool.* Surf: Speed up robust features // European Conference on Computer Vision, 2006. P. 404–417.

5. *Hartley R., Zisserman A.* Multiple view geometry in computer vision. 2nd edition. Cambridge University Press, 2004.

6. *Lepetit V., Fua P., Pilet J.* Point Matching as a Classification Problem for Fast and Robust Object Pose Estimation // Conference on Computer Vision and Pattern Recognition, 2004.

7. *Mikolajczyk K., Schmid C.* An Affine Invariant Interest Point Point Detector // European Conference on Computer Vision, 2002. P. 128–142.

8. *Milgram P., Kishino A. F.* Taxonomy of Mixed Reality Visual Displays // IEICE Transactions on Information and Systems. 1994. E77-D(12). P. 1321–1329.

9. *Moreels P., Perona P.* Evaluation of Features Detectors and Descriptors based on 3D objects // IJCV, 2006.

10. *Obeysekera M.* Affine Reconstruction from multiple views using Singular Value Decomposition / School of Computer Science and Software Engineering, The University of Western Australia, 2003.

11. *Wuest H., Vial F., Stricker D.* Adaptive line tracking with multiple hypotheses for augmented reality // Proceedings of the Fourth IEEE and ACM International Symposium on Mixed and Augmented Reality, 2005. P. 62–69.

Technologies and Algorithms for Building the Augmented Reality

Blagoveshchenskiy I.A., Demyankov N. A.

Areal LTD, AREALIDEAtm,
Polushkina roshcha, 16, building 67a, office 2-6, Yaroslavl, 150044, Russia

Keywords: augmented reality, computer vision, feature detection, markers, OpenCV

This article is about the Augmented Reality technology itself and its current implementations. In the first part of the article the authors give a short historical reference to the origins of the name "augmented reality", by whom it was introduced and what it means. Later in the article two major approaches to building AR are described. The first one is based on the usage of a marker, and the second one is marker-free. The first approach is examined in detail. In order to analyze video stream and recognize known objects in it, algorithms of the Computer Vision are used. The authors give a short description and the main characteristics only of two of them: genetic algorithms and feature detection & description. For a programmatic implementation of those algorithms one can use special libraries like OpenCV and AForge.NET, also mentioned in the article. Both of them give vast functional capabilities in image processing and object recognition. At the end of the article is given an example of creating AR using the OpenCV library. Main attention is payed to the problem of making projection of a 3D model on the marker's plane. This example can be used as the foundation for a custom AR framework.

Сведения об авторах:
Благовещенский Иван Александрович,
ООО "Ареал", AREALIDEA, генеральный директор;
Демьянков Николай Андреевич,
ООО "Ареал", AREALIDEA, программист