

Построение трехмерных моделей по стереоизображениям с учетом распараллеливания

А.Н. Волкович

В работе описываются процедуры автоматического построения трехмерной модели по стереоизображениям. Описываются алгоритмы построения карт диспаратности, обсуждаются подходы к их параллельной реализации и приводятся результаты вычислительного эксперимента по сравнению эффективности выполнения последовательной и параллельной реализаций алгоритма.

Введение

В настоящее время использование компьютерных трёхмерных моделей для визуализации объектов широко распространено. Построение моделей вручную сопряжено с большими затратами, а использование специальной аппаратуры для сканирования трёхмерных объектов не всегда возможно.

Одним из альтернативных подходов в построении трехмерных моделей реальных объектов является подход, основанный на использовании нескольких цифровых изображений [1,2]. Выделяют два подхода для получения трёхмерной информации по изображениям: подход, использующий некоторые характерные признаки изображённых объектов, и подход, использующий плотные карты диспаратности. Алгоритмы построения плотной карты диспаратности позволяют для каждой точки одного изображения найти соответствующую ей точку на втором изображении, либо определить, что такой точки нет [3,4].

1. Построение трехмерной модели по стереоизображениям

1.1. Подготовка изображений

Входной информацией для алгоритма определения множества сопряжённых точек на паре изображений служат лишь сами изображения. На начальном этапе работы алгоритма выделяются множества точек на изображениях, среди которых находятся предполагаемые соответствия путём сравнения их окрестностей.

Сопряженные точки могут задаваться вручную или отыскиваться автоматически при помощи алгоритма автоматического нахождения сопряжённых точек [5].

Выравнивание изображений позволяет получить более простую эпиполярную геометрию в том смысле, что фундаментальная матрица для преобразованных изображений имеет вид:

$$F = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{bmatrix}$$

Для выполнения выравнивания пары изображений к каждому изображению применяется специальное проективное преобразование. Проективные преобразования выбираются так, чтобы соответствующие эпиполярные линии были параллельны оси x . В качестве выравнивающего преобразования можно использовать преобразование, которое переносит эпиполус z в бесконечно удалённую точку. При этом преобразование определяется не однозначно, чем можно воспользоваться для уменьшения внесённых проективных искажений. В результате получим, что сопряжённые точки на изображениях будут иметь кроме равных y координат также и близкие значения x -координат [1].

1.2. Построение плотной карты диспаратности

Методы построения плотной карты диспаратности можно разделить на две группы: локальные методы, которые работают с небольшими окрестностями точек для нахождения соответствий (например, блочный метод), и глобальные, которые работают со строками изображения (динамическое программирование) или с изображением целиком (метод максимального потока в графе). Локальные методы могут быть достаточно эффективны, однако они чувствительны к локально-неоднозначным регионам (например, регионам с однородной текстурой). Глобальные методы менее чувствительны к таким регионам, однако они более ресурсоёмки.

Простейший локальный алгоритм – блочный алгоритм. Он определяет диспаратность, сравнивая небольшой регион (блок) вокруг точки первого изображения с последовательностью таких же регионов на втором изображении в некоторой области поиска[6].

Существует также алгоритм, использующий динамическое программирование для построения плотной карты диспаратности. Этот алгоритм работает с парами соответствующих строк на изображении, рассматривая их независимо от остальных строк, в результате чего на карте диспаратности могут появиться характерные горизонтальные штрихи[7].

Третий подход к построению карты диспаратности состоит в сведении задачи к нахождению максимального потока в графе. В построенном специальным образом графе находится минимальный разрез, который и определяет искомые значения диспаратности [8].

1.3 Реконструкция трехмерной модели

В общем случае процесс реконструкции по множеству пар сопряжённых точек можно описать следующими шагами:

1. вычисление фундаментальной матрицы;
2. вычисление матрицы камер по фундаментальной матрице;
3. вычисление точки в пространстве, изображением которой эти точки являются.

Когда для реконструкции используются только лишь изображения, возникает неоднозначность реконструкции: пространственные координаты могут быть определены лишь с точностью до проективного преобразования.

Таким образом, без какой-либо информации о параметрах камер и реконструируемого объекта можно выполнить реконструкцию с точностью до проективного преобразования. Используя дополнительную информацию можно сократить неоднозначность.[1]

2. Построение трехмерной модели по стереоизображениям

2.1. Подготовка изображений

Обработка изображений большого разрешения требует значительных временных затрат. Отсюда возникает необходимость поиска способов оптимизации и ускорения выполнения алгоритмов. Одним из подходов, дающих значительные результаты в увеличении скорости выполнения алгоритмов, является распараллеливание.

В работе исследовалось распараллеливание алгоритма на основе динамического программирования.

Для параллельной реализации алгоритма динамического программирования для построения плотных карт диспаратности на многопроцессорных параллельных вычислительных системах необходимо произвести выделение независимых блоков алгоритма, которые могут исполняться на различных вычислительных узлах.

Были рассмотрены различные возможности распараллеливания алгоритма построения плотной карты диспаратности на основе метода динамического программирования, определены их сильные и слабые стороны.

Первый подход предполагает распараллеливание алгоритма путем пропуска строк. Таким образом, при наличии двух вычислительных узлов, первый вычислительный узел обрабатывает только нечетные строки, а второй только четные. Данный подход максимально сбалансировано использует машинное время, так как сложность обработки соседних строк, в общем случае,

близка. Однако при данном подходе возрастает объем используемой памяти, по сравнению с последовательной реализацией алгоритма, в число раз, равное числу потоков, что может значительно снижать производительность как вычислительного узла, так и сети, соединяющей узлы многопроцессорной вычислительной системы.

На Рисунок 1 проиллюстрирован данный подход. Здесь показаны изображения загруженные в вычислительные узлы, а обрабатываемые строки отмечены заливкой.

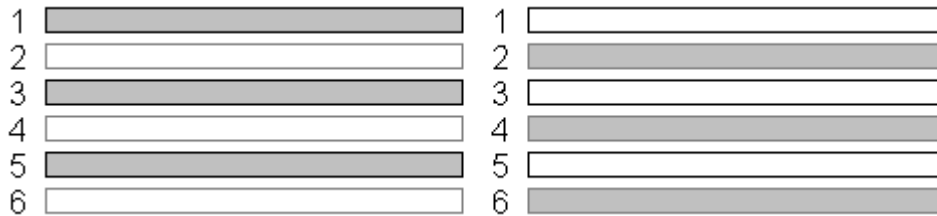


Рисунок 1 Обработка изображений первым способом на двухпроцессорной системе

Второй способ предполагает разделение выровненных изображений на несколько частей. Тогда каждый вычислительный узел обрабатывает только «свою» часть изображения, что снижает затраты памяти в узле, а объем передаваемой по сетям информации сводится к размеру, сравнимому с размером одной пары изображений. Данный подход позволяет снизить ресурсозатраты вычислительной системы, по сравнению с первым подходом, однако возникает разбалансировка загрузки вычислительных узлов за счет неоднородности вычислительной сложности различных участков изображений.

На Рисунок 2 проиллюстрирован данный подход. Здесь видно, что затраты памяти значительно снижаются, также указывается и производится обработка всех строк выделенных на вычислительный узел (показаны заливкой).



Рисунок 2 Обработка изображений вторым способом на двухпроцессорной системе

Третий подход, объединяющий сильные стороны рассмотренных методов, заключается в применении к выровненным изображениям специального фильтра, который позволяет преобразовать изображения таким образом, чтобы при дальнейшем разделении изображений на части, в каждую часть попадали строки номер которых кратен номеру вычислительного узла. Таким образом достигается максимально сбалансированное использование вычислительных узлов при низких затратах памяти.

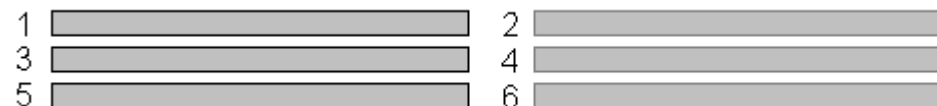


Рисунок 3 Обработка изображений третьим способом на двухпроцессорной системе

Аналогичную логику распараллеливания предполагается использовать для блочного алгоритма построения карт диспаратности, с отличием в том, что в блочном методе будут обрабатываться не строки, а полосы по высоте равные высоте обрабатываемого блока.

Однако данный подход заведомо не подходит для метода основанном на поиске максимального потока в графе. Данный подход предполагает обмен информацией между обрабатываемыми строками изображения, поэтому в случае применения третьего метода значительно

возрастет трафик обмена данных между узлами, что значительно снизит эффективность распараллеливания. Здесь хорошо применим второй подход к распараллеливанию задачи, однако он приведет к значительно разбалансированной загрузке вычислительных узлов. Поэтому возникает объективная необходимость поиска критерия оценки сложности обработки частей изображения, для его разделения на максимально равные по сложности части. Задача поиска критерия сложности поставлена автором как первоочередная для дальнейших исследований.

2.2. Определение эффективности распараллеливания.

Для определения эффективности выполнения алгоритма в последовательной и параллельной реализации был проведен вычислительный эксперимент. В ходе проведения эксперимента сравнивалось время выполнения алгоритма при параллельной и последовательной реализациях на различных вычислительных системах. Обработывались две пары изображений:

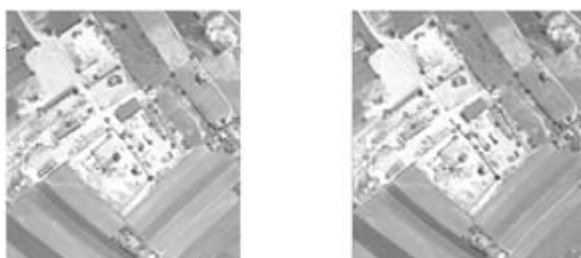


Рисунок 4 Стереопара: "местность"

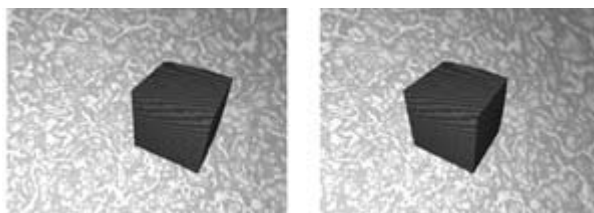


Рисунок 5 Стереопара: "куб внутри сферы"

Вычислительный эксперимент проведен на персональном компьютере на базе процессоров Intel Pentium 4 3.06 GHz с режимом Hyper Threading и Intel Core Duo 2.6 GHz.

Для получения информации об отношении скорости расчетов в последовательном и параллельном режимах произведено измерение времени выполнения последовательно реализованного алгоритма, а затем и параллельно реализованной версии. С целью получения наиболее объективных данных проведено десять замеров и вычислено среднее значение скорости выполнения.

Результаты вычислительного эксперимента приведены в Таблица 1.

Таблица 1. Результаты вычислительного эксперимента

	Линейное	Параллел.
Местность	6,92 сек.	4,5 сек.
Куб внутри сферы	1,83 сек.	1,1 сек.

Таким образом, можно говорить о том, что параллельная реализация алгоритма позволяет увеличить его производительность на двуконвейерных или двуконвейерных системах на 30-40% по сравнению с их последовательной реализацией.

Заключение

Дальнейшая изыскательская работа позволит оптимизировать процесс построения трехмерных моделей по стереоизображениям. Программная реализация и внедрение алгоритмов построения трехмерной модели сцен по стереоизображениям с использованием возможностей параллельных систем позволит в значительной степени автоматизировать географо-геодезические работы и работы по обработке данных полученных при дистанционном зондировании Земли, а так же работы по оперативному построению трехмерных моделей реальных объектов [9]. Предполагается возможное внедрение параллельных методов в систему «Стереок» разрабатываемую в рамках программ Союзного государства «Космос-СГ» и «Космос-НТ», а также программы «СКИФ».

Литература

1. Hartley R., Zisserman A. Multiple View Geometry in Computer Vision. - Cambridge University Press, 2001. - 624 p..
2. Fougeras O., Luong Q.-T. The Geometry of Multiple Images. - The MIT Press, 2001. - 646 p.
3. Scharstein D., Szeliski R. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms // International Journal of Computer Vision. - 2002. - Vol. 47. - № 1-3. - P. 7-42.
4. Тузиков А.В., Шейнин С.А, Жук Д.В.. Математическая морфология, моменты, стереобработка: избранные вопросы обработки и анализа цифровых изображений. Минск, Белорус. наука, 2006.-198с.
5. Borodach A, Tuzikov A. Automatic determination of matching points on two images. Proceedings of the 9th International Conference "Pattern Recognition and Information Processing", 22-24 May, 2007, Minsk, Belarus, vol. 1, 49-53.
6. Жук Д.В., Тузиков А.В. Реконструкции трехмерной модели по цифровым изображениям. // Информатика, 2006 - № 1. - С. 16-26.
7. Intille S., Bobick A., Large Occlusion Stereo // International Journal of Computer Vision, - 1999, - Vol. 33. - № 2. - P. 181-200.
8. Roy S., Stereo Without Epipolar Lines: A Maximum-Flow Formulation. International Journal of Computer Vision, vol. 34, n. 2/3, 1999, p. 147-161.
9. А.Н. Волкович, Д.В. Жук, А.В. Тузиков. Методы построения трехмерных моделей местности и их реализация для параллельных систем. Доклады 5-й международной конференции "Обработка информации и управление в чрезвычайных и экстремальных ситуациях", 24 – 26 октября, Минск, Беларусь, 2006, 100-104.