

УДК 004.932

Д.В. Жук¹, А.В. Тузиков²

РЕКОНСТРУКЦИЯ ТРЕХМЕРНОЙ МОДЕЛИ ПО ДВУМ ЦИФРОВЫМ ИЗОБРАЖЕНИЯМ

Рассматривается задача восстановления трехмерной модели объекта по паре цифровых изображений, обсуждаются алгоритмы решения ее подзадач, включающих автоматическое нахождение сопряженных точек на изображениях, выравнивание изображений, построение плотной карты диспаратности и триангуляцию. Предлагаются модификации некоторых алгоритмов.

Введение

В настоящее время широко распространено использование компьютерных трехмерных моделей для визуализации объектов. Построение моделей вручную сопряжено с большими затратами, а применение специальной аппаратуры для сканирования трехмерных объектов не всегда возможно, так как глубина сканирования такими устройствами ограничена.

В работе приводится способ получения трехмерной модели по паре цифровых изображений, на которых один и тот же объект снят с немного различающихся позиций, а также предлагаются модификации алгоритмов, используемых на различных этапах решения задачи реконструкции, которые позволяют в некоторых случаях улучшить конечный результат реконструкции. Такой способ получения модели не накладывает ограничений на размеры объекта: это может быть как небольшой объект, так и земной ландшафт. В качестве примера продемонстрирована возможность восстановления рельефа местности по перекрывающимся аэрофотоснимкам, основанная на реализации описываемых алгоритмов. В результате испытания данной реализации на синтетических изображениях были получены незначительные отклонения реконструированной модели от исходной, используемой для формирования изображений.

Задача восстановления трехмерной модели сцены состоит в определении пространственных координат точек на поверхностях объектов, изображенных на снимках. В качестве исходных данных используются два цифровых изображения и небольшое количество опорных точек (т. е. точек, для которых известны их пространственные координаты), применяемых для устранения неоднозначности реконструкции. В результате обработки исходных данных формируется трехмерная модель сцены в действительном масштабе.

Можно выделить два подхода для получения трехмерной информации по изображениям. Первый подход использует некоторые характерные признаки изображенных объектов (такие, как границы объектов, угловые точки и т. д.), а второй – построение плотных карт диспаратности. Недостатком первого подхода является ограниченность множества точек, для которого восстанавливается трехмерная структура. Такой метод дает приемлемые результаты, когда реконструируемые объекты имеют достаточно простую форму, что встречается, например, в задачах реконструкции архитектурных сцен. В случае более сложных объектов этот метод не дает необходимого уровня детализации для получения высоко реалистичной модели и, как правило, применяется на этапе подготовки изображений для обработки их алгоритмами построения плотной 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.

Процесс реконструкции трехмерной модели по двум изображениям можно представить в виде следующих шагов:

1. Определить на изображениях некоторое число сопряженных точек. Этот шаг может быть выполнен как вручную, так и автоматически.
2. Используя сопряженные точки, выполнить выпрямление изображений.
3. Построить плотную карту диспаратности для пары выпрямленных изображений.

4. Произвести триангуляцию.

Далее рассмотрим вышеперечисленные действия подробнее.

1. Определение сопряженных точек

Входной информацией для алгоритма определения множества сопряженных точек на паре изображений служат лишь сами изображения. На начальном этапе работы алгоритма выделяются множества точек на изображениях, среди которых находятся предполагаемые соответствия путем сравнения их окрестностей. Кроме сходства окрестностей, найденные пары точек должны также удовлетворять эпиполярным ограничениям, для проверки которых в ходе работы алгоритма вычисляется и уточняется фундаментальная матрица \mathbf{F} . Фундаментальная матрица – это вырожденная ($\text{rank } \mathbf{F} = 2$) матрица размерности 3×3 , которая содержит в себе всю информацию об эпиполярной геометрии между двумя проекциями. Для любой пары сопряженных точек $\mathbf{x} \leftrightarrow \mathbf{x}'$, заданных в однородных координатах $\mathbf{x} = (x_1, x_2, 1)$, $\mathbf{x}' = (x'_1, x'_2, 1)$, фундаментальная матрица удовлетворяет уравнению $\mathbf{x}'^T \mathbf{F} \mathbf{x} = 0$. Это дает возможность найти некорректно идентифицированные пары сопряженных точек. Для нахождения фундаментальной матрицы достаточно иметь семь пар сопряженных точек [1].

Алгоритм автоматического нахождения сопряженных точек состоит из следующих шагов.

1. Выделить на каждом изображении множество точек, которое затем можно будет использовать для построения множества предполагаемых сопряженных пар. Например, в качестве такого множества можно использовать угловые точки [2].

2. Для каждой особой точки на первом изображении найти соответствующую ей особую точку на втором изображении, сравнивая некоторые их окрестности. Аналогичным образом находятся соответствия для каждой особой точки на втором изображении. В случае когда соответствующая точка оказывается уже занята другим соответствием, выбирается та пара, у которой сходство окрестностей больше. Часто в качестве меры для определения степени сходства окрестностей используется нормированная кросс-корреляция в силу ее инвариантности к масштабированию и сдвигу значений интенсивности на изображениях. В работе [3] авторы предложили применение инвариантных характеристик (инвариантов моментов и информационных полей) для построения мер сходства фрагментов изображений на базе известных блочных мер сходства. Указанные характеристики инвариантны к преобразованию вращения, поэтому их применение может дать лучшие по сравнению с классическими мерами сходства результаты в случае, когда при съемке второго изображения произошло вращение камеры. Рассматриваемые в работе [3] характеристики также могут применяться для построения исходного множества точек алгоритма.

3. Для определения ложных совпадений, полученных на шаге 2, воспользуемся алгоритмом RANSAC, где будем применять эпиполярные ограничения. Шаги 3.1, 3.2 выполняются в цикле. Для определения необходимого числа итераций можно воспользоваться формулами, приведенными в работе [1].

3.1. Случайным образом выбрать семь пар точек из множества, полученного на шаге 2, и вычислить по ним фундаментальную матрицу \mathbf{F} [1]. Если в результате нахождения \mathbf{F} по семи парам точек будут получены три действительных решения, необходимо произвести оценку каждого из них.

3.2. Для каждой предполагаемой сопряженной пары вычислить расстояние d_{\perp} , отражающее насколько точно пара удовлетворяет эпиполярному ограничению.

3.3. Подсчитать количество пар, согласующихся с \mathbf{F} (для которых d_{\perp} меньше некоторого порогового значения).

Выбрать матрицу \mathbf{F} , с которой согласуется наибольшее число пар, и отбросить все остальные точки. Если таких решений несколько, выбрать решение с наименьшим среднеквадратичным отклонением согласующихся пар.

4. Вычислить оценку значения \mathbf{F} , используя все оставшиеся пары точек.

5. В окончательное множество сопряженных точек включаются те точки, которые попадают в окрестность соответствующей эпиполярной линии, определяемой матрицей \mathbf{F} , полученной на шаге 4.

Кроме множества сопряженных точек, этот алгоритм находит также и фундаментальную матрицу, которая будет использоваться на шаге выравнивания изображений.

2. Выравнивание изображений

Процесс выпрямления заключается в таком преобразовании изображений, в результате которого эпиполярные ограничения принимают наиболее простой и удобный для использования вид, т. е. все эпиполярные линии параллельны оси абсцисс и соответствующие эпиполярные линии на обоих изображениях имеют одинаковые ординаты.

Рассмотрим метод, с помощью которого находится пара проективных преобразований для выравнивания изображений [4]. При этом из множества пар допустимых выравнивающих преобразований выбирается та пара, которая вносит минимальное искажение.

Для обеспечения параллельности эпиполярных линий оси абсцисс проективное преобразование должно отображать эпиполлюс в бесконечно удаленную точку $(1, 0, 0)^T$. Для обеспечения минимального искажения, вносимого преобразованием, потребуем, чтобы в окрестности некоторой заданной точки \mathbf{x}_0 преобразование было близким к преобразованию вращения и сдвига. Выбор точки \mathbf{x}_0 определяется исходя из структуры изображения, однако для большинства случаев можно использовать центр изображения.

Предположим, что \mathbf{x}_0 находится в центре координат, а эпиполлюс $\mathbf{e}' = (f, 0, 1)^T$ лежит на оси ординат. Тогда преобразование для переноса эпиполлюса в точку $(1, 0, 0)^T$ имеет вид

$$\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -1/f & 0 & 1 \end{bmatrix}.$$

Для произвольной точки \mathbf{x}_0 и эпиполлюса \mathbf{e}' требуемое преобразование можно вычислить как $\mathbf{H}' = \mathbf{GRT}$, где \mathbf{T} – сдвиг для переноса \mathbf{x}_0 в начало координат, а \mathbf{R} – вращение относительно начала координат, отображающее эпиполлюс в точку $(f, 0, 1)^T$.

Требование совпадения ординат у соответствующих эпиполярных линий накладывает ограничение на пару преобразований: если \mathbf{I} и \mathbf{I}' – пара соответствующих эпиполярных линий на изображениях, то тогда должно выполняться условие $\mathbf{H}^{-T}\mathbf{I} = \mathbf{H}'^{-T}\mathbf{I}'$.

Таким образом, для получения пары согласованных преобразований можно вычислить преобразование \mathbf{H}' , как описано выше, а затем найти согласованное с ним преобразование \mathbf{H} , которое бы минимизировало функционал

$$\sum_i d(\mathbf{H}\mathbf{x}_i, \mathbf{H}'\mathbf{x}'_i)^2, \quad (1)$$

где $d(\mathbf{x}, \mathbf{x}')$ – расстояние между точками.

Для решения указанной задачи минимизации можно воспользоваться следующим результатом [1]. Пусть J и J' – пара изображений с фундаментальной матрицей $\mathbf{F} = [\mathbf{e}']_x \mathbf{M}$ и \mathbf{H}' – проективное преобразование изображения J' , отображающее эпиполлюс \mathbf{e}' в бесконечно удаленную точку $(1, 0, 0)^T$. В этом случае преобразование \mathbf{H} изображения J согласуется с преобразованием \mathbf{H}' тогда и только тогда, когда \mathbf{H} имеет вид $\mathbf{H} = \mathbf{H}_A \mathbf{H}_0$, где $\mathbf{H}_0 = \mathbf{H}' \mathbf{M}$, а \mathbf{H}_A – аффинное преобразование вида

$$\mathbf{H}_A = \begin{bmatrix} a & b & c \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Здесь используется разложение фундаментальной матрицы в произведение кососимметрической и невырожденной матриц:

$$\mathbf{F} = [\mathbf{e}']_{\times} \mathbf{M},$$

где \mathbf{F} – фундаментальная матрица, соответствующая паре изображений J и J' с эпицентрами \mathbf{e} и \mathbf{e}' соответственно; $[\mathbf{e}']_{\times}$ – кососимметрическая матрица

$$[\mathbf{e}']_{\times} = \begin{bmatrix} 0 & -e'_3 & e'_2 \\ e'_3 & 0 & -e'_1 \\ -e'_2 & e'_1 & 0 \end{bmatrix},$$

соответствующая вектору $\mathbf{e}' = (e'_1, e'_2, e'_3)^T$. Этот результат дает возможность переписать выражение (1) в виде

$$\sum_i d(\mathbf{H}_A \tilde{\mathbf{x}}_i, \tilde{\mathbf{x}}'_i)^2 \rightarrow \min ,$$

где $\tilde{\mathbf{x}}'_i = \mathbf{H}' \mathbf{x}'_i$, $\tilde{\mathbf{x}}_i = \mathbf{H}_0 \mathbf{x}_i$.

Полагая $\tilde{\mathbf{x}}_i = (\tilde{x}_i, \tilde{y}_i, 1)^T$, $\tilde{\mathbf{x}}'_i = (\tilde{x}'_i, \tilde{y}'_i, 1)^T$ и учитывая, что $(\tilde{y}_i - \tilde{y}'_i)^2$ – константа, получим следующую задачу минимизации с параметрами a , b и c :

$$\sum_i (a\tilde{x}_i + b\tilde{y}_i + c - \tilde{x}'_i)^2 \rightarrow \min ,$$

решение которой находится с помощью стандартных алгоритмов.

Результат применения данного выпрямляющего преобразования к изображениям (рис. 1) показан на рис. 2.



Рис. 1. Исходная пара изображений

Отметим, что данный метод выравнивания изображений неприменим, когда эпицентр находится на самом изображении или вблизи его границы. В этом случае можно воспользоваться методом, приведенным в работе [5].



Рис. 2. Выпрямленные изображения

3. Построение плотной карты диспаратности

Используя особые точки на изображении (например, угловые), можно получить лишь небольшое количество точек поверхности реконструируемых объектов. Для получения модели сцены по этим точкам можно воспользоваться интерполяцией, однако на практике такой метод не дает приемлемых результатов, так как теряются детали, присутствующие на исходных изображениях. В настоящее время разработан ряд алгоритмов, которые позволяют найти пары сопряженных точек практически для всех точек стереопары. Как правило, такие алгоритмы работают с выпрямленными парами изображений и используют ряд предположений об изображенных на них объектах. Приведем самые распространенные среди них:

1. Если изображение точки *A* находится слева от изображения точки *B* на одном изображении, тогда изображение *A* также будет левее изображения *B* на другом изображении (рис. 3, *а*). Данное предположение может нарушаться, например, когда узкий объект находится близко к камере (рис. 3, *б*). Это предположение позволяет использовать динамическое программирование для построения плотной карты диспаратности.

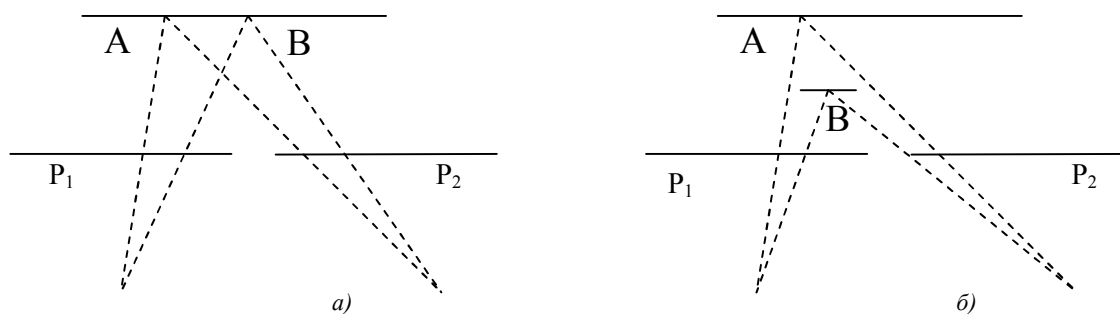


Рис. 3. Порядок следования соответствующих точек на изображениях: *а*) порядок следования изображений точек *A* и *B* сохраняется; *б*) для камеры P_1 изображение точки *A* предшествует изображению точки *B*, а изображение точки *B* предшествует изображению точки *A* для камеры P_2

2. Любая точка изображения входит не более чем в одну пару сопряженных точек.

Методы построения плотной карты диспаратности можно разделить на две группы: локальные, которые работают с небольшими окрестностями точек для нахождения соответствий (например, блочный метод), и глобальные, которые работают со строками изображения (дина-

мическое программирование) или с изображением целиком (метод максимального потока в графе). Локальные методы могут быть достаточно эффективны, однако они чувствительны к локально-неоднозначным регионам (например, с однородной текстурой). Глобальные методы менее чувствительны к таким регионам, однако они более ресурсоемки.

Простейший локальный алгоритм – блочный. Он определяет диспаратность, сравнивая небольшой регион (блок) вокруг точки первого изображения с последовательностью таких же регионов на втором изображении в некоторой области поиска. В качестве меры сходства блоков изображений часто используются сумма квадратов разностей интенсивностей

$$\sum_{u,v} (I_1(u,v) - I_2(u+d,v))^2, \quad (2)$$

сумма абсолютных разностей

$$\sum_{u,v} |I_1(u,v) - I_2(u+d,v)| \quad (3)$$

и нормированная кросс-корреляция. В формулах (2) и (3) применяются следующие обозначения: $I_k(u,v)$ – интенсивность пикселя (u,v) на k -м изображении, d – оцениваемое значение диспаратности. Суммирование ведется по некоторому прямоугольному окну.

Пример использования блочного алгоритма показан на рис. 4, *а*. На карте диспаратности можно видеть области, где значение диспаратности было определено неверно. Для устранения подобных ошибок может потребоваться изменение размеров окна либо увеличение числа исходных изображений.

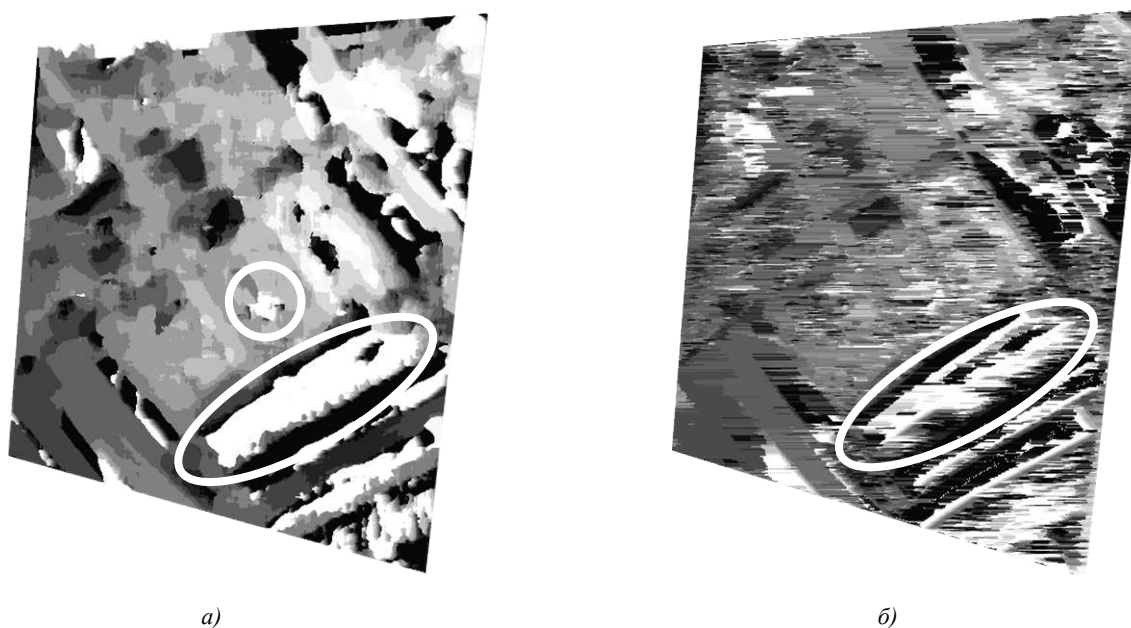


Рис. 4. Примеры карт диспаратности: *а*) построенных с помощью блочного алгоритма; *б*) с помощью алгоритма динамического программирования. Светлые участки соответствуют большему значению диспаратности, овалом отмечены области, где ошибка определения диспаратности достаточно велика

В работе [6] рассматривается предложенная авторами модификация блочного алгоритма, позволяющая учитывать параметры камер для уточнения карты диспаратности. В ходе работы этого алгоритма вместе с картой диспаратности производится реконструкция поверхности объекта. Для уточнения диспаратности в точке (x, y) для каждого значения диспаратности d' из $[d-\Delta d, d+\Delta d]$ по матрицам камер \mathbf{P} и \mathbf{P}' и паре точек $(x+d, y)$ и (x, y) восстанавливается точка \mathbf{X} . По точке \mathbf{X} и соседним с ней точкам реконструированной поверхности объекта

строится касательная плоскость Π , такая, что $\mathbf{X} \in \Pi$. Затем вычисляется проективное преобразование \mathbf{H}_Π , связывающее между собой изображения точек плоскости Π в камерах \mathbf{P} и \mathbf{P}' . В общем случае для камер

$$\mathbf{P} = \begin{bmatrix} \mathbf{U}^T \\ \mathbf{V}^T \\ \mathbf{W}^T \end{bmatrix}, \quad \mathbf{P}' = \begin{bmatrix} \mathbf{U}'^T \\ \mathbf{V}'^T \\ \mathbf{W}'^T \end{bmatrix},$$

где $\mathbf{U}^T, \mathbf{V}^T, \mathbf{W}^T$ и $\mathbf{U}'^T, \mathbf{V}'^T, \mathbf{W}'^T$ – векторы, соответствующие строкам матриц камер \mathbf{P} и \mathbf{P}' соответственно. Для плоскости Π , определенной вектором $\mathbf{\Pi} = (x, y, z, s)^T$ в однородных координатах, это преобразование имеет вид [7]

$$\mathbf{H}_\Pi = \begin{bmatrix} |\mathbf{U}', \mathbf{V}, \mathbf{W}, \mathbf{\Pi}| & |\mathbf{U}', \mathbf{W}, \mathbf{U}, \mathbf{\Pi}| & |\mathbf{U}', \mathbf{U}, \mathbf{V}, \mathbf{\Pi}| \\ |\mathbf{V}', \mathbf{V}, \mathbf{W}, \mathbf{\Pi}| & |\mathbf{V}', \mathbf{W}, \mathbf{U}, \mathbf{\Pi}| & |\mathbf{V}', \mathbf{U}, \mathbf{V}, \mathbf{\Pi}| \\ |\mathbf{W}', \mathbf{V}, \mathbf{W}, \mathbf{\Pi}| & |\mathbf{W}', \mathbf{W}, \mathbf{U}, \mathbf{\Pi}| & |\mathbf{W}', \mathbf{U}, \mathbf{V}, \mathbf{\Pi}| \end{bmatrix}.$$

На следующем шаге алгоритма производится вычисление блочной меры сходства с учетом преобразования \mathbf{H}_Π . Например, модифицируя меру сходства (2), получим

$$\sum_{u,v} (I_1(\mathbf{x}) - I_2(\mathbf{x}'))^2,$$

где $\mathbf{x} = (u, v, 1)$, $\mathbf{x}' = \mathbf{H}_\Pi \mathbf{x}$. Таким образом в ходе работы алгоритма происходит сравнение не двух одинаковых блоков, как в классическом блочном алгоритме, а блока и его образа, порожденного касательной плоскостью, которая аппроксимирует поверхность объекта. В качестве нового значения диспаратности в точке (x, y) выбирается то значение, для которого достигается наибольшее сходство блока и его образа. Данная модификация позволяет устранить либо снизить ошибку определения диспаратности в некоторых областях карты диспаратности.

Рассмотрим теперь 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 этого на карте диспаратности могут появиться характерные горизонтальные штрихи. Поскольку этот алгоритм обрабатывает всю строку целиком, то он, как правило, позволяет получить лучшие, по сравнению с блочным алгоритмом, результаты для областей со слабовыраженной текстурой.

В работе [8] вводится понятие изображения пространства диспаратности (DSI) – явное представление пространства, в котором ведется поиск соответствий. Обозначим через $I_R(x, i)$ и $I_L(x, i)$ интенсивность точки с координатами (x, i) на правом и левом изображениях соответственно, а через d – диспаратность. Тогда в общем виде DSI можно представить функцией

$$\text{DSI}_i^L(x, d) = |I_L(x, i) - I_R(x-d, i)|, \quad (4)$$

где $-d_{\max} \leq d \leq d_{\max}$, $0 \leq x+d < N$, N – ширина изображения. Одно из возможных представлений этой функции показано на рис. 5 (для каждой пары точек левой и правой строк с допустимой диспаратностью вычисляется стоимость соответствия).

Часто используется измененная формула (4), где вместо разности интенсивностей двух точек используется вычисляемая по некоторому окну мера сходства, например сумма абсолютных разностей (3) или нормированная кросс-корреляция. Такая модификация позволяет учитывать влияние соседних строк на изображении, однако это может привести к ухудшению результатов на границах объектов.

Для нахождения \mathbf{X} при помощи метода линейной триангуляции [1] система (5) преобразуется к виду

$$\begin{cases} \mathbf{x} \times (\mathbf{P}\mathbf{X}) = 0; \\ \mathbf{x}' \times (\mathbf{P}'\mathbf{X}) = 0, \end{cases}$$

где \times обозначает векторное произведение, что равносильно системе уравнений

$$\begin{bmatrix} \mathbf{x}\mathbf{p}^{3T} - \mathbf{p}^{1T} \\ \mathbf{y}\mathbf{p}^{3T} - \mathbf{p}^{2T} \\ \mathbf{x}'\mathbf{p}'^{3T} - \mathbf{p}'^{1T} \\ \mathbf{y}'\mathbf{p}'^{3T} - \mathbf{p}'^{2T} \end{bmatrix} \mathbf{X} = \mathbf{A}\mathbf{X} = \mathbf{0}. \quad (6)$$

Здесь \mathbf{p}^{iT} и \mathbf{p}'^{iT} – строки матриц \mathbf{P} и \mathbf{P}' ; $\mathbf{x} = (x, y, 1)$, $\mathbf{x}' = (x', y', 1)$. В качестве оценки \mathbf{X} принимается значение сингулярного вектора, соответствующего наименьшему сингулярному значению матрицы \mathbf{A} .

Когда для реконструкции используются только изображения (а именно множество сопряженных точек), возникает неоднозначность реконструкции: пространственные координаты могут быть определены лишь с точностью до проективного преобразования. Справедлива следующая теорема [1].

Теорема. Пусть $\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$ – множество сопряженных точек на двух изображениях и \mathbf{F} – фундаментальная матрица, единственным образом определяемая уравнением $\mathbf{x}_i'^T \mathbf{F} \mathbf{x}_i = 0$ для всех i . Пусть $(\mathbf{P}_1, \mathbf{P}'_1, \{\mathbf{X}_{1i}\})$ и $(\mathbf{P}_2, \mathbf{P}'_2, \{\mathbf{X}_{2i}\})$ – две реконструкции, полученные по сопряженным точкам $\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$. Тогда существует невырожденная матрица \mathbf{H} , такая, что $\mathbf{P}_2 = \mathbf{P}_1 \mathbf{H}^{-1}$, $\mathbf{P}'_2 = \mathbf{P}'_1 \mathbf{H}^{-1}$ и $\mathbf{X}_{2i} = \mathbf{H} \mathbf{X}_{1i}$ для всех i , кроме тех, для которых выполняется равенство $\mathbf{F} \mathbf{x}_i = \mathbf{x}_i'^T \mathbf{F} = 0$.

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. Так, например, указав на изображении проекции трех множеств параллельных прямых с разными направлениями, можно сократить неоднозначность до аффинного преобразования. В случае если известны внутренние параметры камер, реконструкция возможна с точностью до преобразования подобия.

Рассмотрим случай, когда известны n опорных точек (т. е. точек, для которых заданы пространственные координаты) $\{\mathbf{X}_{Ei}\}$ и их изображения $\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$. В данном случае, используя координаты сопряженных точек $\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$, можно восстановить их координаты в пространстве $\{\mathbf{X}_i\}$, которые, в силу приведенной выше теоремы, будут связаны с истинными координатами проективным преобразованием \mathbf{H} :

$$\mathbf{X}_{Ei} = \mathbf{H} \mathbf{X}_i, \quad i = 1, \dots, n. \quad (7)$$

Каждая пара $\mathbf{x}_i \leftrightarrow \mathbf{x}'_i$ дает три линейно-независимых уравнения относительно элементов \mathbf{H} . Таким образом, для решения системы (7) необходимо не менее пяти пар, причем никакие четыре опорные точки не должны быть копланарны. В случае $n \geq 6$ система (7) может не иметь точного решения из-за погрешностей измерения координат точек. В этом случае решение \mathbf{H} может быть получено путем минимизации алгебраической или геометрической ошибки [1].

Результат восстановления трехмерной модели по исходным изображениям (см. рис. 1) с использованием описанных алгоритмов показан на рис. 6. Для получения модели использовал-

ся блочный метод построения карты диспаратности, привязка модели к местности осуществлялась по восьми опорным точкам.

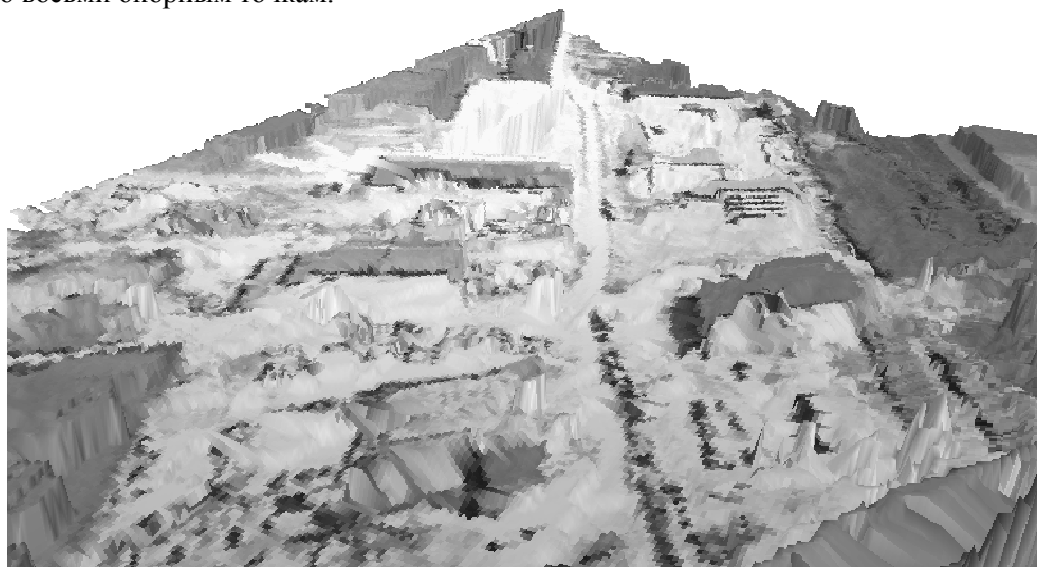


Рис. 6. Реконструкция рельефа местности по исходным изображениям рис. 1 (левый верхний угол)

Заключение

Реконструкция трехмерной модели по паре изображений с помощью информации, содержащейся на самих изображениях, позволяет получить детализированную модель реконструируемого объекта. Детализация модели достигается за счет использования плотных карт диспаратности для поиска сопряженных точек. В основном от выбора алгоритма построения плотной карты диспаратности зависит время, затрачиваемое на получение модели, а также точность реконструкции. Полученная таким образом модель может быть привязана к исходному объекту при помощи небольшого числа опорных точек.

Список литературы

1. Hartley R., Zisserman A. *Multiple View Geometry in Computer Vision*. – Cambridge University Press, 2001. – 624 p.
2. Harris C., Stephens M. A combined corner and edge detector // *Proc. 4th Alvey Vision Conference*. – Manchester, 1988. – P. 147–151.
3. Жук Д.В., Тузиков А.В. Построение плотной карты диспаратности для стереоизображений с использованием локально-инвариантных характеристик // Докл. Четвертой Междунар. конф. «Обработка информации и управление в чрезвычайных и экстремальных ситуациях». 29 ноября – 1 декабря 2004 г., Минск, Беларусь. – Мн.: ОИПИ НАН Беларуси, 2004. – С. 141–146.
4. Hartley R. Theory and practice of projective rectification // *International Journal of Computer Vision*. – Vol. 35. – № 2. – 1999. – P. 115–127.
5. Pollefeys M., Koch R., Van Gool L. A simple and efficient rectification method for general motion // *Proc. International Conference on Computer Vision*. – Corfu, Greece, 1999. – P. 496–501.
6. Zhuk D., Tuzikov A. Embedding camera parameters into block matching metrics for dense disparity map correction // *Pattern Recognition and Information Processing (PRIP'2005): Proceedings of the Eighth International Conference*. 18–20 May 2005, Minsk, Republic of Belarus. – Minsk: Propilei, 2005. – P. 334–336.
7. Fougeras O., Luong Q.-T. *The Geometry of Multiple Images*. – The MIT Press, 2001. – 646 p.
8. Intille S., Bobick A. Large occlusion stereo // *International Journal of Computer Vision*. – Vol. 33. – № 2. – 1999. – P. 181–200.

9. Scharstein D., Szeliski R. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms // International Journal of Computer Vision. – Vol. 47. – № 1–3. – 2002. – P. 7–42.

Поступила 05.10.05

¹*Белорусский государственный университет,
Минск, пр. Независимости, 4
e-mail: dzhuk@tut.by*

²*Объединенный институт проблем
информатики НАН Беларуси,
Минск, Сурганова, 6
e-mail: tuzikov@newman.bas-net.by*

D.V. Zhuk, A.V. Tuzikov

THREE-DIMENSIONAL MODEL RECONSTRUCTION USING TWO DIGITAL IMAGES

The paper surveys a process of a three-dimensional model reconstruction using a pair of digital images. The subtasks of the problem including automatic finding of corresponding points, image rectification, dense disparity map computation and triangulation are discussed. Algorithms used at each step of the reconstruction process and some their modifications are described.