

В.Г.Сиротин, А.В.Коновалов
БИБЛИОТЕКА ПРОГРАММ ДЛЯ ПРЕДСТАВЛЕНИЯ И ОБРАБОТКИ
ГРАФИЧЕСКИХ ДАННЫХ МЕТОДОМ КВАДРОДЕРЕВЬЕВ

В настоящей работе описываются назначение, основные функциональные возможности, структура и некоторые особенности реализации библиотеки программ для представления и обработки графических (картографических) данных методом квадродеревьев.

Разработка рассматриваемой библиотеки программ ведется в рамках создания в Лаборатории машинной графики ВЦ СО АН СССР географической информационной системы АЛЬБА [1]. Рабочее название библиотеки - LibALBA. Библиотека программ LibALBA является функциональным ядром системы АЛЬБА, обеспечивающим ее ключевые возможности. Основной объем входной информации для системы поступает с цифровых карт местности. Результатами работы системы АЛЬБА являются ответы на запросы о характеристиках и особенностях обрабатываемого системой участка местности, располагающихся на нем природных объектах и искусственных сооружениях.

В качестве базового представления для хранения и обработки графических (картографических) данных в системе АЛЬБА использованы структуры типа квадродеревьев.

I. Неформальная постановка задач

Абстрагируясь от географической специфики, можно каждый из географических объектов отнести к одному из четырех основных геометрических типов: точечному, линейному, площадному либо к классу однозначных поверхностей вида $z = f(x, y)$. С этой точки зрения основные геометрические запросы к системе АЛЬБА могут быть сформулированы следующим образом:

1. Проведение теоретико-множественных операций (объединения, пересечения и разности). Важной разновидностью таких операций является селекция (отбор) объектов, попадающих в данную область (накрываемых данным площадным объектом). В результате выполнения этих операций генерируются новые объекты.

2. Поиск объекта из заданного класса, ближайшего к данной точке (объекту), с возможностью вычисления расстояния между ними и ограничениями на область поиска.

3. Вычисление метрических и топологических характеристик данного объекта (площади, периметра, числа компонент связности и т.д.).

4. Определение значения функции $f(x, y)$ в данной точке и нормали к поверхности (рельефу), определяемой данной функцией.

5. Определение на множестве пересекающихся линейных объектов (планарном графе) всех вершин, доступных из некоторой точки, принадлежащей одному из линейных объектов.

6. Генерализация (отbrasывание мелких объектов, сглаживание линейных объектов и границ площадных объектов).

Как видно из этого перечня, задачи, решаемые программами библиотеки LIBALBA, носят достаточно общий для машинной графики характер. В предметных терминах запросы I-6 могут быть проинтерпретированы самым различным образом. Так, например, с помощью комбинации запросов первого и четвертого типов можно определить все участки лесомассивов, располагающихся в определенном диапазоне высот над уровнем моря.

Особенностью системы АЛЬБА является то, что она относится к числу оперативных географических информационных систем, т.е. ответы на запросы типа I-6 должны генерироваться с максимально возможной быстротой.

Считается, что на носителях исходной информации, обрабатываемых системой АЛЬБА, объекты разных типов могут храниться следующим образом:

1) точечные - в виде геометрически неупорядоченного файла с записями постоянной длины. Каждая запись в файле содержит координаты точки и ее атрибуты;

2) линейные - в виде списка независимых линейных сегментов (кривых). Каждый сегмент в свою очередь представляет собой ломаную из отрезков прямых линий либо последовательность цепных кодов;

3) площадные - либо в виде многосвязных плоских фигур, задаваемых своей границей - линейным объектом, либо в виде растра точек. При этом в ряде случаев растра может явно не задаваться, а генерироваться в виде хаотической последовательности троек (i, j, k) , где (i, j) - координаты точки, а k - номер площадного объекта, которому она принадлежит;

4) однозначные функции двух переменных (рельеф) могут быть заданы на цифровых картах местности в виде:

- а) матрицы высот,
- б) неупорядоченного набора отметок высот,
- в) триангуляционной неоднородной сетки высот,
- г) совокупности линий одинакового уровня.

В случаях б), в) исходная информация представляет собой точечный объект, а в г) - линейный.

2. Обоснование выбранного подхода к представлению обрабатываемых данных

Выбор формы представления в памяти ЭВМ данных, подлежащих обработке, - одна из центральных проблем при проектировании любой системы обработки данных. Задача выбора формы представления данных по своей сути является неформальной. Выбор зависит, как минимум, от следующих факторов:

- форматов представления входных данных и требований к представлению результирующих данных,
- требований на скорость выполнения основных операций,

- ограничений, накладываемых техническими средствами.

Лавинообразный поток публикаций на эту тему, сообщений о все более изощренных алгоритмах и структурах, повышающих основные показатели обработки, чрезвычайно расширяют возможный выбор. Как правило, с каждым классом структур данных связана своя группа алгоритмов их обработки. При этом при решении одних задач предпочтительнее одни структуры, а при решении других - другие. Так, например, при использовании топологических структур подсчет периметра площадного объекта - тривиальная задача, а нахождение пересечения двух площадных объектов представляет определенные трудности. При использовании же структур данных растрового типа ситуация обратная: пересечение объектов с алгоритмической точки зрения задача тривиальная, а оптимальное решение задачи вычисления периметра сопряжено с рядом затруднений.

В системе АЛЬБА в качестве базового представления для хранения и обработки графических (карографических) данных выбраны квадрограммы. Этот выбор определяют следующие соображения:

- единство представления всех типов данных и выполнения практических необходимых операций;
- возможность единственно и одновременно для данных всех типов управлять точностью (детальностью) представления;
- относительно высокая скорость операций, которые, как представляется, будут наиболее часто выполняться при работе системы АЛЬБА.

3. Структура библиотеки

Библиотека программ LibALBA представляет собой слабо структурированную совокупность программ и файлов. Связь между модулями библиотеки осуществляется через файл данных. Соглашения о форматах этих файлов, а также об используемых системах координат и порядке инициализации являются единственными связывающими факторами для программ библиотеки.

Все программы библиотеки LIBALVA можно условно разбить на:

- 1) модуль обработки точечных объектов,
- 2) модуль обработки линейных объектов,
- 3) модуль обработки площадных объектов,
- 4) модуль обработки однозначных функций двух переменных (рельефов),
- 5) модуль общих функций,
- 6) модуль визуализации,
- 7) файловую подсистему.

Модули библиотеки могут быть использованы автономно.

Модуль обработки точечных объектов имеет структуру, показанную на рис. I (на рис. I-5 жирным шрифтом выделены программные компоненты). В результате построения структуры типа квадродерева все исходные объекты будут некоторым образом за- кодированы и упорядочены по значению так называемой "квадро- координаты". Операция геометрического доступа (см. рис. 2, верхний правый блок) позволяет "геометрически последователь- но" просматривать объекты и определять их значения, если та- ковые имеются.

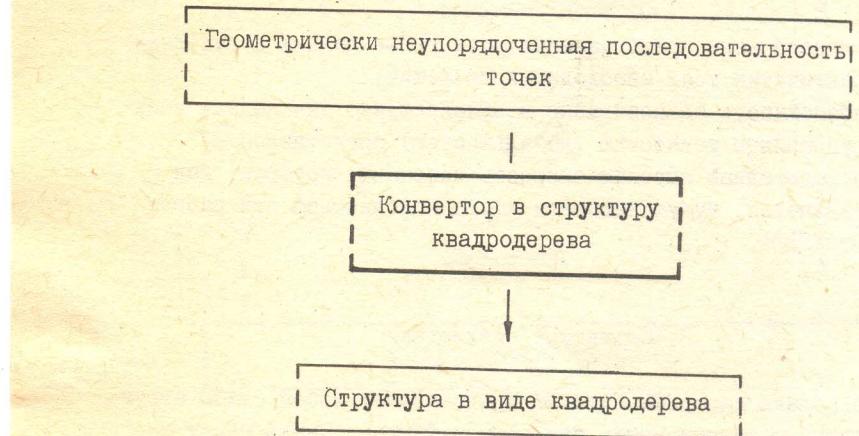


Рис. I. Структура модуля обработки точечных объектов

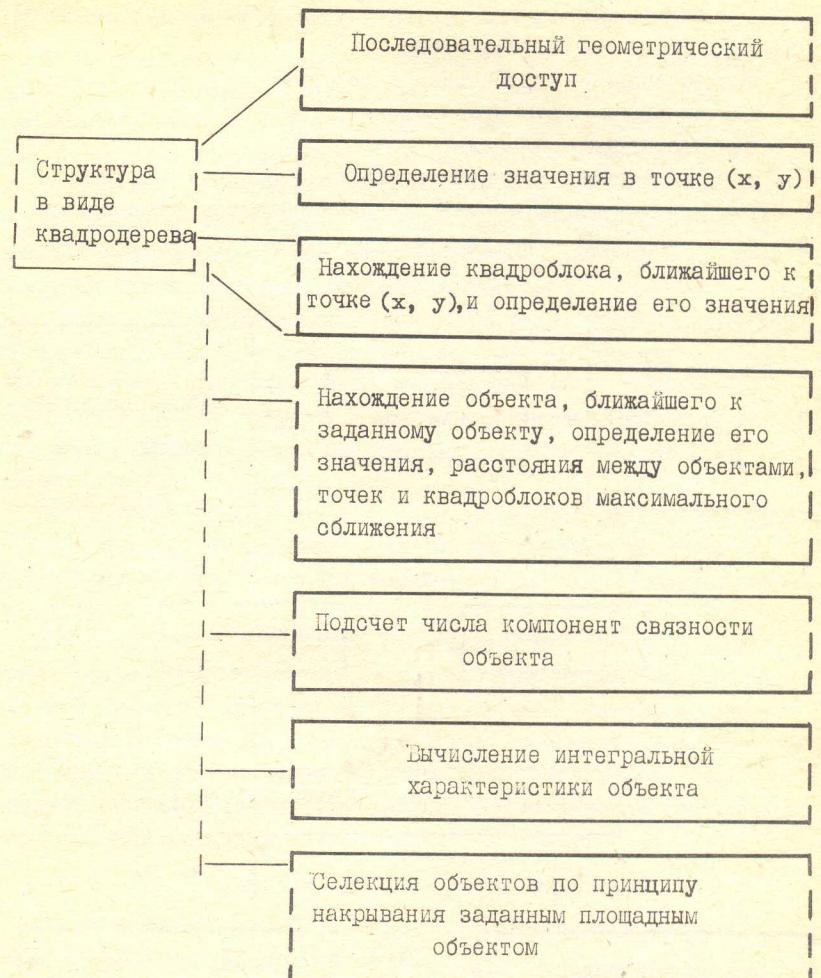


Рис. 2. Структура модуля общих функций

Под интегральной характеристикой (см. рис. 2) для объектов различных типов подразумевается:

- для точечных - число точек в объекте;
- для линейных - суммарная длина;
- для площадных - площадь;
- для однозначных функций (рельефов) - среднее значение.

Структура модуля обработки линейных объектов показана на рис. 3, а площадных - на рис. 4.

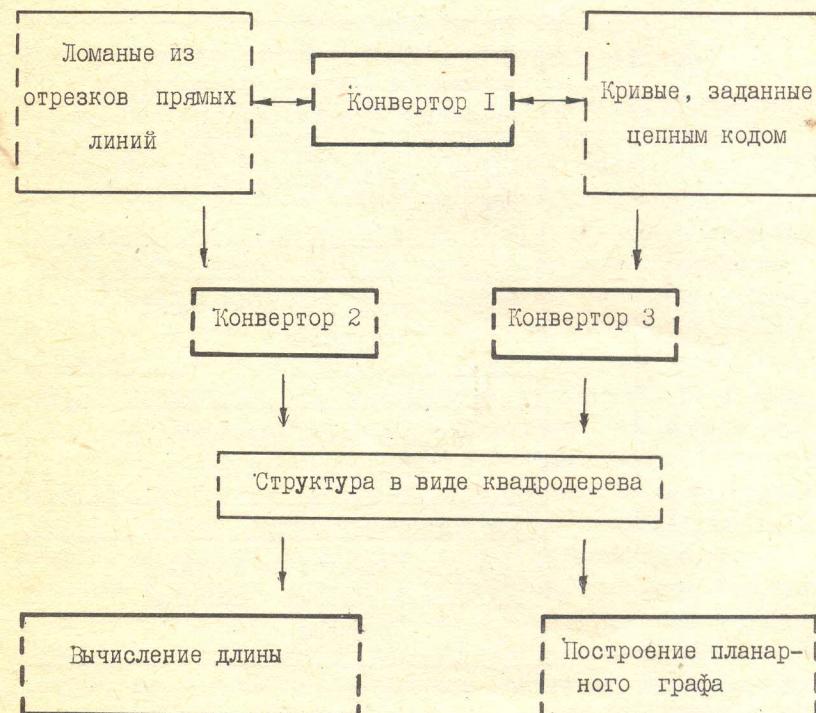


Рис. 3. Структура модуля обработки линейных объектов

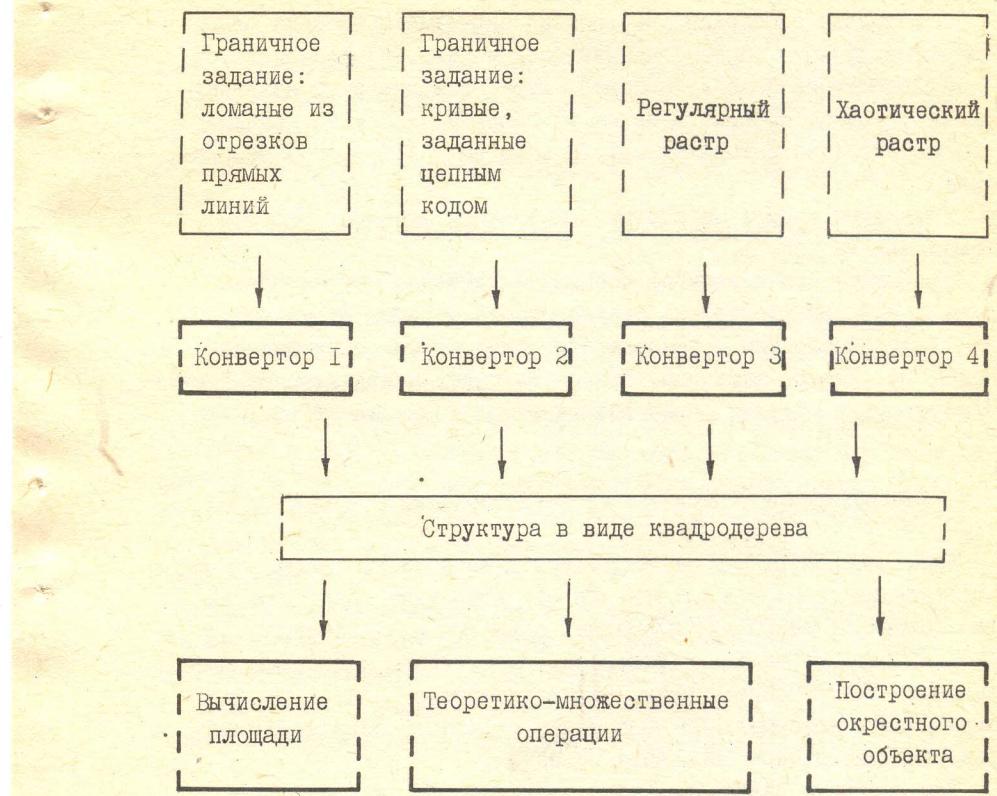


Рис. 4. Структура модуля обработки площадных объектов

В результате операции построения окрестного объекта относительно данного (правый нижний блок на рис. 4) генерируется новый площадной объект, который "шире" исходного на величину D , т.е. точки, которые в исходном объекте лежали на границе, в новом объекте находятся внутри него на расстоянии D от границы.

Структура модуля обработки однозначных функций (рельефа) приведены на рис. 5.

Напомним, что наряду со специфическими для случая однозначных функций (рельефов) операциями полученные структуры могут быть обработаны с помощью программ модуля общих функций (рис. 2). Данное замечание относится также к структурам, генерируемым в модулях обработки линейных и площадных объектов.

4. Квадродеревья и способы их представления в памяти ЭВМ

Рамки данной статьи не позволяют дать полного формального определения процессов построения структур типа квадродерева для точечных, линейных, площадных объектов и однозначных функций двух переменных (рельефов). Достаточно подробные сведения по этому вопросу можно найти в работах [2, 3]. Мы же ограничимся неформальным пояснением процесса построения квадродерева на примере площадных объектов.

Для простоты изложения будем предполагать, что исходное изображение задано на квадрате размером $2^3 \times 2^3$ и является двуградационным (рис. 6).

Основная идея метода построения квадродерева для плоской фигуры заключается в следующем. Исходная область подвергается рекуррентной декомпозиции на четыре подквадрата до тех пор, пока не встретится однородно заполненный квадрат либо не исчерпается разрешающая способность.

В соответствии с этим правилом, поскольку исходный квадрат на рис. 6 не является однородным, разделим его на четыре подквадрата. Они будут иметь следующие цвета:

- 1) белый - северо-западный (СЗ),
- 2) черный - юго-западный (ЮЗ),
- 3) неоднородные (серые) - северо-восточный (СВ) и юго-восточный (ЮВ).

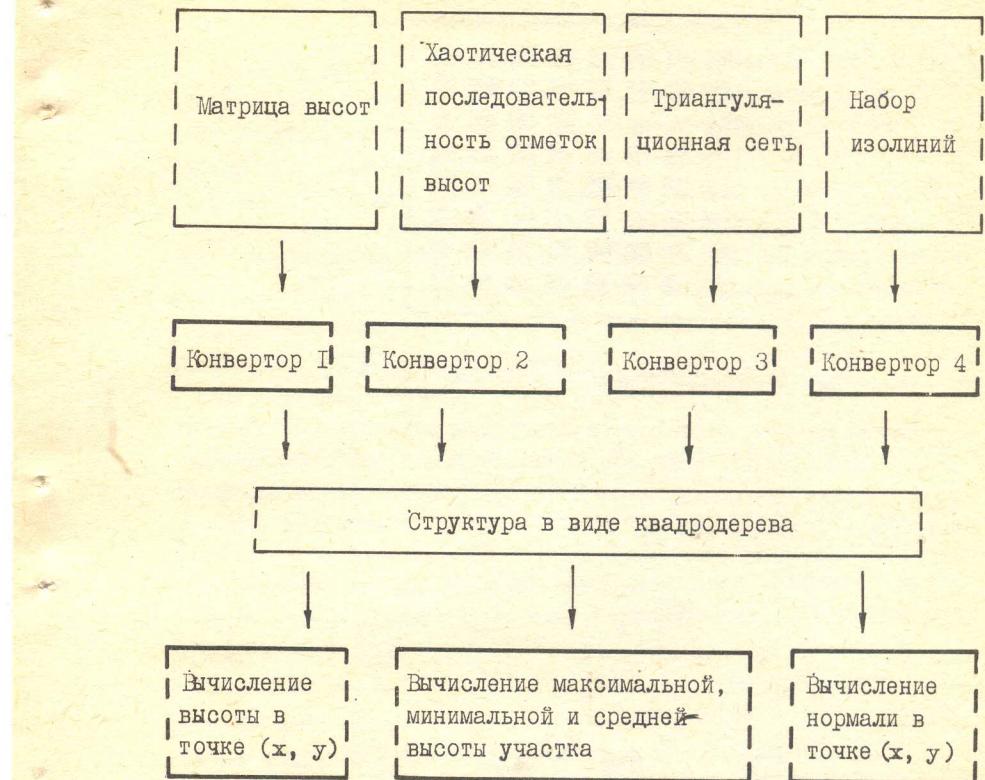


Рис. 5. Структура модуля обработки однозначных
поверхностей (рельефов)

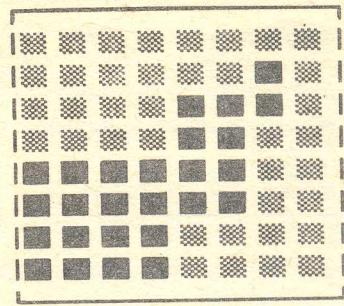


Рис. 6. Исходное двуградационное изображение

Обозначим эти подквадраты следующим образом: СЗ-0, СВ-1, ЮЗ-2, ЮВ-3. Тогда, если обозначить через I исходное изображение, структура квадродерева после первого разбиения будет выглядеть так, как на рис. 7.

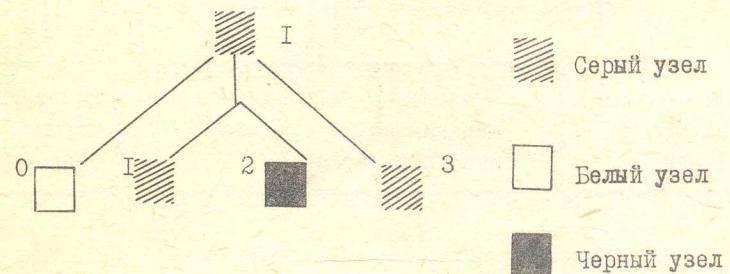


Рис. 7. Структура квадродерева после первого разбиения

Линии, соединяющие квадраты на рис. 7, соответствуют связям типа "предок-потомок".

Продолжая аналогичным образом декомпозицию для серых узлов, получим квадродерево, изображенное на рис. 8.

Номер узла-потомка внутри узла-отца приписывается справа к номеру отца, и, таким образом, получаем код, однозначно идентифицирующий любой из блоков, возникающих в процессе рекурсивного разбиения. Этот код называют квадрокодом, а блок — квадроблоком.

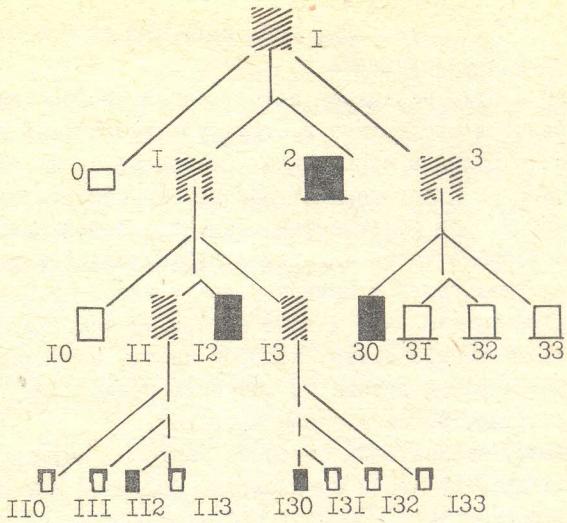


Рис. 8. Квадродерево для изображения с рис. 6

Исходя из перечня операций, реализуемых программами библиотеки LibALBA, оказалось целесообразным использовать для хранения квадродеревьев подход, получивший в настоящее время наименование "метод линейного квадродерева". Основная идея этого подхода состоит в том, чтобы не хранить о квадродереве никакой другой информации, кроме пар:

{ квадрокод, значение квадроблока },
идентифицирующих терминальные черные узлы. При этом в случае двуградационных изображений значение квадроблока отсутствует. Список записей с квадрокодами квадроблоков и их значениями упорядочивается по величине квадрокода. С целью упростить реализацию под квадрокод отводится поле фиксированной длины, в которое можно записать столько символов, какова глубина квадродерева. В случае, если квадрокод имеет меньшую длину (т.е. квадроблок больше, чем блок минимального разрешения), он дополняется справа символами – заполнителями, отличными от 0, 1, 2, 3. Так, например, квадродерево, изображенное на рис. 8, будет описано как

112, 12X, 130, 2XX, 30X,
где X - символ-заполнитель.

В связи с этим возникает естественный вопрос: каким образом оказывается отсутствие в структуре указателей на узлы и информации о белых и серых узлах квадродерева на эффективность обработки квадродерева? Как это ни парадоксально на первый взгляд, для большинства операций, перечисленных в п.2, удалось подобрать такие алгоритмы, эффективность которых от исключения указателей и нетерминальных узлов не снижается. Это объясняется тем, что данные алгоритмы используют тот же способ обхода квадродерева, который неявно возникает в результате упорядочивания дерева по значению квадрокода: "вначале от корня к листьям, затем - слева направо".

Программы, реализующие основные запросы, осуществляют доступ к квадродереву через специальный файловый комплект служебных программ. Этот комплект является составной частью библиотеки LibALBA. Основными операциями комплекта являются следующие:

- поместить запись с заданными квадрокодом и значением в файл квадродерева и объявить ее текущей,
- считать запись с заданным квадрокодом из файла и объявить ее текущей,
- считать следующую (предыдущую) за текущей запись и объявить ее текущей,
- установить указатель текущей записи на начало (конец) файла,
- удалить запись с заданным квадрокодом из файла и объявить текущей следующую за ней запись.

Последовательно обращаясь к перечисленным функциям, можно строить макрофункции более высокого уровня. Например, можно построить функцию определения цвета квадроблока, примыкающего к данному блоку с северной стороны.

5. Заключение

В географической информационной системе АЛЬБА в качестве базового представления для хранения и обработки графических (картографических) данных использованы структуры типа квадротеревьев. Генерация по исходным данным квадротеревьев и их обработка в системе АЛЬБА производится с помощью программ библиотеки LibALBA. В настоящее время заканчивается реализация первой версии библиотеки на языке Фортран для ЭВМ типа IBM PC/AT.

Метод квадротеревьев на сегодняшний день является одним из наиболее мощных и общих при решении задач обработки графической информации с помощью ЭВМ. Несмотря на большое число публикаций на эту тему, метод во многих вопросах применяется на эвристическом уровне. В задачу авторского коллектива наряду с практической реализацией библиотеки LibALBA входит проведение необходимых исследований по обеспечению корректности и оцениванию точности получаемых результатов, а также проведение широкого спектра численных экспериментов.

Л и т е р а т у р а

1. Сиротин В.Г. Географическая информационная система АЛЬБА //Разработка и использование персональных ЭВМ: Тез. докл. междунар. симпз."INFO - 89".- Минск, 1989.- С. 720-725.
2. Александров В.В., Горский Н.Д. Представление и обработка изображений: рекурсивный подход.- Л.: Наука, 1985.
3. Samet H. The quadtree and related hierarchical data structures // ACM Comput. Surveys. - 1984. - Vol. 16, N 2. - P. 187-260.

В.И.Торшин

ИНТЕРАКТИВНАЯ ГРАФИЧЕСКАЯ СИСТЕМА
ДЛЯ ПОДГОТОВКИ ТРАНСПОРТНЫХ ОБСТАНОВОК

Настоящая статья посвящена некоторым вопросам проектирования и реализации проблемно-ориентированных интерактивных графических систем (ИГС) на примере одной из них, предназначеннай для подготовки и модификации транспортных обстановок (ИГС ТО). Цель создания данной ИГС - обеспечение экспертной системы электронной навигации*) исходными данными.

I. Объекты транспортных обстановок

Первостепенной задачей любой проблемно-ориентированной интерактивной графической системы является определение объектов моделирования, которые в данной ИГС будут обрабатываться, преобразовываться и т.п. В ИГС ТО транспортной обстановкой будем называть совокупность стационарных и динамических объектов, составляющих информационную среду для задач электронной навигации. Транспортная обстановка включает в себя следующие стационарные объекты: рельеф участка опорной поверхности, сеть допустимых перемещений, а также динамические объекты, т.е. маршруты движения, множества динамических элементов.

*) Сиротин В. Г., Витяев Е. Е. Проект экспертной системы электронной навигации.- Наст. сб.- 92-107.
122

Перечисленные объекты в ИГС ТО моделируются следующим образом. Рельеф участка опорной поверхности представлен матрицей высот, элементы которой задают значение высоты в соответствующем узле прямоугольной равномерной сетки.

Сеть допустимых перемещений представлена графом. В дальнейшем будем его называть графом возможных перемещений (ГВП), который представляет собой совокупность вершин и рёбер (дуг), связывающих вершины, т.е.

$$ГВП = \{V, D\},$$

где V - множество вершин, D - множество рёбер.

$$V = \{x_i, y_i, z_i, \varepsilon_i, k_i\},$$

где (x_i, y_i, z_i) - координаты вершины, ε_i - радиус габаритного круга, k_i - внешний контур вершины. Внешний контур определяется множеством узлов, представляющих собой вершины многоугольника, охватывающего вершину.

Маршруты представляются ориентированными ломаными линиями и составляются из рёбер графа возможных перемещений. Маршруты не могут иметь разрывов и разветвлений, т.е. имеют одну начальную и одну конечную точку - вершину ГВП; из начальной точки маршрута можно по составляющим его ломанным пройти в конечную точку.

Динамический элемент представляет собой материальную точку, которая может размещаться на опорной поверхности и перемещаться по одному из маршрутов с заданной скоростью. Другими словами, динамический элемент (ДЭ) определяется шестёркой чисел $(x_i, y_i, z_i, v_x, v_y, v_z)$, где (x_i, y_i, z_i) - координаты ДЭ; (v_x, v_y, v_z) - вектор скорости.

Динамические элементы могут составлять группы, а те, в свою очередь, - классы и т.д., т.е. над динамическими элементами может строиться некоторая древовидная иерархическая структура, нижний уровень которой составляют динамические элементы. Узлы этой структуры будем называть множествами динамических элементов (МДЭ) уровня N , где N - уровень иерархии узла. В частности, динамические элементы будем иногда назы-

вать множествами динамических элементов нулевого уровня. На рис. I представлено множество динамических элементов третьего уровня и все составляющие его подмножества.

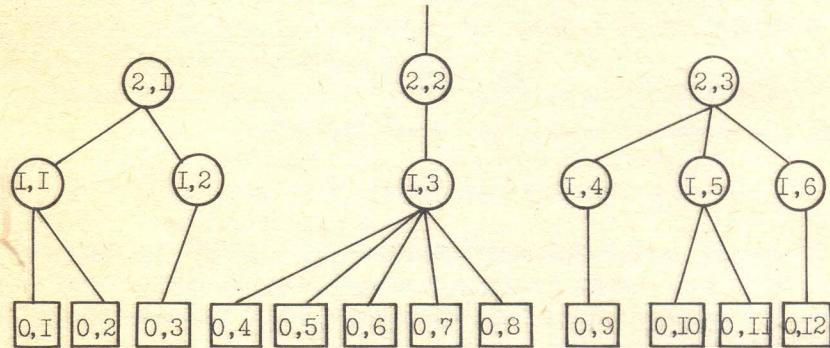


Рис. I

Динамические элементы не могут входить одновременно в два МДЭ более высокого уровня.

2. Графические представления объектов

Графические образы объектов, описанных выше, разрабатывались в следующих ограничениях изобразительного характера: количество основных цветов - 16; количество смешанных цветов - 256; разрешающая способность экрана - 640 x 350. Кроме того, в ряде случаев накладывается ограничение, связанное с необходимостью изменять часть изображения в реальном времени, не затрагивая остального изображения. Это ограничение обусловлено значительной трудоёмкостью переформирования изображения на большой части экрана.

Для матрицы высот (рельефа) используется несколько графических представлений. В качестве основного представления ис-

пользована совокупность областей (необязательно односвязных), залитых некоторым цветом. Заливаемые области и цвет выбираются в соответствии с некоторым разбиением заданного диапазона высот на уровни (шкалу уровней) и таблицей соответствия уровней и базовых цветов для их изображения. Шкала уровней определяет интервалы высот, которые необходимо одним и тем же цветом выделять при формировании изображения рельефа. Выделение достигается применением различных цветов и оттенков в цветовом клине. Использование различных шкал уровней и соответствующих им раскрасок позволяет получать изображения рельефа, отражающие различные его свойства. Фактически представлением матрицы высот является карта изолиний, в которой линиями являются границы областей, окрашенные в различные цвета. Кроме основного графического представления матрицы высот в ИГС используется два дополнительных, представляющих собой разрезы вдоль строки или столбца матрицы. Полученные разрезы раскрашиваются в соответствии со шкалой уровней и цветовым клином.

Для графа возможных перемещений используется также два графических представления. Основное представление – это множество кругов (вершин ГВП), радиуса, равного габаритному, ломанных линий, соединяющих вершины, и внешних контуров вершин, составленных из отрезков прямых линий. В качестве дополнительного графического образа используется разрез рельефа вдоль ребра ГВП вертикальными плоскостями.

Полученный разрез раскрашивается, исходя из шкалы уровней и таблицы соответствия высот и цветов.

Маршруты движения изображаются путём выделения цветом начальной вершины и рёбер ГВП, заканчивающихся стрелкой, которая указывает направление маршрута. Кроме того, изображается разрез рельефа вдоль маршрута, раскрашенный, как и в случае ребра ГВП.

Множества динамических элементов обладают двумя качествами, которые необходимо отображать на экране дисплея: во-первых, это древовидная структура МДЭ, во-вторых, – расположение элементов множества на маршрутах.

Для работы с МДЭ в ИГС ТО использованы два графических образа. Первый – изображение древовидной структуры (см. рис. I) в виде дерева. Путь прямой визуализации дерева на экране дисплея в связи с существенными ограничениями на размеры поля экрана, отводимого под МДЭ, оказался неприемлемым. Поэтому для изображения деревьев было реализовано два способа. Первый заключается в визуализации узла-отца и всех его связей с узлами-сыновьями (рис. 2).

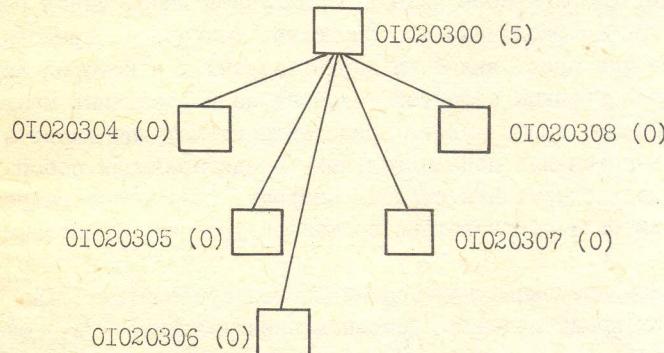


Рис. 2

По причине малой информативности такого изображения каждый узел снабжается идентификатором. По идентификатору можно установить место узла в иерархии множества динамических элементов. Так, по идентификатору узла-отца (см. рис. 2) 0I020300 (5) можно выяснить, что: узел-отец содержит пять узлов-сыновей, является МДЭ первого уровня и имеет номер, равный трём, узел-отец является подмножеством МДЭ второго уровня с номером два и МДЭ третьего уровня с номером один.

Второй способ заключается в отображении всех связей внутри МДЭ, но изображение дерева структуры как бы положено набок. На рис. 3 изображена структура МДЭ с рис. I. Понятно, что оба варианта обладают достоинствами и недостатками. Так, первый способ не отражает всех связей в МДЭ, зато легко размещается на ограниченных полях экрана дисплея. Второй способ более ин-

Формативен, но может иметь большую длину, что приводит к известным издержкам, связанным с организацией окон видимости.

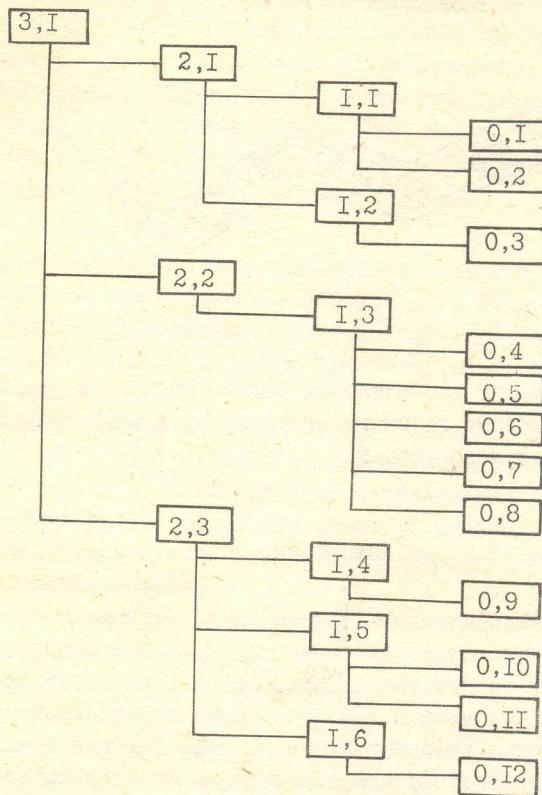


Рис. 3

Вторым графическим образом множества динамических элементов является его изображение на маршруте в виде области, ограниченной прямыми линиями и дугами окружностей и залитой некоторым цветом. Поскольку МДЭ имеет иерархическую структуру, представляют интерес и составляющие его подмножества. Поэтому внутри описанной выше области возможно появление меньших по размеру аналогичного вида областей, заливаемых цветом, отлич-

ным от цвета МДЭ более высокого уровня. На рис. 4 представлено размещённое на маршруте МДЭ, структура которого приведена на рис. I.

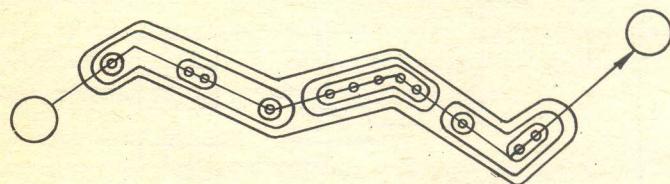


Рис. 4

На маршрутах могут изображаться только множества динамических элементов, которые не имеют предков, остальные же - в составе МДЭ более высокого уровня.

3. Интерфейс

При создании интерактивных систем одним из важных требований является "дружественность" интерфейса. В это понятие вкладываются довольно обширный список требований к способам организации интерфейса человека и ЭВМ, временными характеристиками, способам отображения информации и т.п. Разрабатывая предлагаемую интерактивную графическую систему, мы попытались построить интерфейс дружественным, придерживались при этом следующих соображений:

- Управление ИГС осуществляется с помощью меню. Выбор операции и операндов обеспечивается путём манипуляций с различными наборами команд меню и изображениями объектов.

- Минимизация управляющих (функциональных) кнопок алфавитно-цифровой клавиатуры. Пользователь может обходиться при работе с ИГС набором из следующих клавиш: +, -, /, *, Pg Up, Pg down, стрелки ↑, →, ↓, ←, ТАБУЛЯЦИЯ.

- Широкое использование для указания и выбора тех или иных объектов и характеристик различного вида графических знакомаркеров. В описываемой ИГС это - стрелки, кресты, инверсиро-

ванные прямоугольники, вложенные цветные квадраты, цветные прямоугольные рамки.

- Использование звуковых сигналов, дублирующих реакцию ИГС на различные ситуации во время выполнения операций. Практически любая фиксация объекта, которому предстоит дальнейшая обработка, сопровождается звуковым сигналом. Различные ошибочные действия оператора также вызывают кроме символьной реакции звуковую.

- Минимизация переформирований изображения основного объекта для данного этапа работы с ИГС. Во всех случаях, когда необходим переход к другому набору операций по работе с прежним объектом, когда изображение не изменяется или изменяется мало, перерисовка изображения не производится. Новый же набор команд, как правило, расшифровывающий одну из операций с объектом, визуализируется на экране дисплея, частично перекрывая кнопки предыдущего светового меню. По завершении работы с дополнительным меню его изображение стирается и восстанавливается прежнее меню. При этом прежнее меню уже содержит соответствующие изменения, связанные с работой с дополнительным меню.

- Второстепенность цифровой информации и вспомогательной, т.е. вся дополнительная и цифровая информация индицируется на периферийных участках экрана и не создает каких-либо помех для работы с объектом и основным его изображением.

Для воплощения вышеперечисленных соображений при реализации ИГС были использованы средства базовой графики^{*)}, позволяющие выводить цветные линии, заливать прямоугольные области, выводить символьную информацию, а также вводить целые, вещественные числа и тексты, коды клавиш. И, кроме того, сохранять и восстанавливать заданные прямоугольные области на экране дисплея.

^{*)} Ткачев Ю.А. Комплекс графических программ GRAPH'ER. - Наст. сб.- С. 20-59.

4. Связь объектов моделирования ИГС с изображением

С точки зрения проектирования ИГС представляет интерес организация взаимодействия с объектами моделирования посредством изображения.

Связь объекта и его изображения в интерактивной системе обязательна, поскольку в противном случае необходимо устраивать пути доступа к объекту и его частям (параметрам) каким-либо другим способом, обычно менее эффективным. Для большинства объектов, с которыми имеет дело ИГС, связь между объектом и его изображением тривиальна. Так, для графа возможных перемещений, например, каждой вершине графа соответствует на изображении цветной круг или квадрат, каждому ребру графа — ломаная линия, соединяющая пару вершин. Примером более сложной связи может служить связь между множеством динамических элементов и изображением его структуры вторым способом (см. рис. 3).

Эта связь организована следующим образом. Для МДЭ строится графический сегмент, содержащий описание изображения его структуры в терминах элементов изображения.

В качестве элементов изображения структуры МДЭ используются графические объекты:

- а) - МДЭ без предков и потомков;
- б) - МДЭ, имеющее потомков;
- в) - МДЭ, имеющее предка и потомков;
- г) - МДЭ, имеющее предка и не имеющее потомков;
- д) - узел связи между МДЭ разных уровней с ответвлением к потомку;
- е) - связь между МДЭ разных уровней;
- ж) - узел связи от предка к последнему потомку данного уровня.

Изображение структуры множества динамических элементов строится уже по этому описанию, и по любому элементу от а) до г) описания однозначно определяется соответствующее ему МДЭ.

Организация такой связи демонстрирует, кроме того, пример организации взаимодействия человека с объектом моделирования посредством изображения путём построения некоторой промежуточной модели объекта. Собственно взаимодействие строится с промежуточной моделью, а все необходимые изменения в основной модели вычисляются по промежуточной, с использованием средств связи, например, корреляционных таблиц.

Следует отметить, что использование промежуточных моделей и каких-либо сложных соответствий между моделью объекта и его изображением эффективно, если объект достаточно сложный и постоянные перевычисления нового состояния объекта и его изображения требуют много ресурсов.

5. Некоторые приёмы организации выполнения операций ИГС

Выполнение функций ИГС организовано по следующей схеме: выбор объекта (операндов), выбор операции и выполнение операции, т.е. в ИГС постоянно решаются задачи определения операции и operandов.

Для решения этих задач используются указатели, т.е. некоторые выборки параметров объектов или операций, изменяя значения которых можно "перемещаться" в некоторой области допустимых значений объекта или среди множества объектов. Так, например, для светового меню, представляющего собой упорядоченный набор описаний операций (и/или объектов), указателем является номер кнопки меню. Как и объекты, указатели имеют некоторый образ на экране дисплея – графический и/или алфавитно-цифровой. Так, для множества динамических элементов используется несколько указателей.

Первый включает номер уровня МДЭ, не имеющего предков, и номер МДЭ в уровне. Его образом на экране является цифровая индикация номера уровня и номера собственно МДЭ, а также графическая – в виде выделения красным цветом квадрата, отображающего МДЭ, не имеющее предков. Второй также включает номер

уровня МДЭ и номер МДЭ внутри уровня, но только для тех множеств, которые являются подмножествами МДЭ, не имеющего предков, т.е. этот указатель изменяется внутри множества, задаваемого первым указателем. Образом второго указателя является также цифровая индикация номеров уровня и множества и графическая – в виде креста, размещаемого в центре квадрата, отражающего МДЭ (см. рис. I). Третий указатель используется для выбора уровня МДЭ, внутри которого будет производиться выбор множества без предков или какого-либо подмножества.

Образом на экране данного указателя является индикация номера уровня и номера множества, а также инверсированный прямоугольник с номером выбранного уровня.

Четвёртый указатель используется для выбора множества внутри одного из МДЭ уровня, заданного третьим указателем. Образ на экране – цифровая индикация номеров уровня и МДЭ, а также графическая – в виде выделенного цветом квадрата, отображающего это множество.

6. Организация работы с памятью

Программные системы, предназначенные для подготовки данных, обычно имеют дело с большим количеством информации, причём нередко имеющей различную структуру. Каждая программная система в той или иной степени решает проблему ограниченности ресурсов и сохранения информации. Описываемая ИГС использует метод сегментации данных. Сегментация заключается в формировании некоторого множества типов структур данных. Например, для представления в ЭВМ множества вершин в графе возможных перемещений используется структура, представляющая собой набор записей, состоящих из координат вершины (x, y, z), радиуса габаритного круга и ссылки на сегмент, содержащий описание внешнего контура вершины. Такой сегмент получает некоторый тип и снабжается набором операций, обеспечивающих доступ как к отдельным записям сегмента, так и к элементам записей. В ряде случаев этот набор дополняется различными операциями обработки данных. Примером дополнительных операций могут слу-

жить операция сдвига всех вершин или масштабирования габаритного радиуса и т.п.

Представленные в виде сегментов данные ИГС размещаются на магнитных носителях или других средствах массовой памяти, откуда производится их считывание в оперативную память ЭВМ, когда для выполнения операции требуется информация этих сегментов. При недостатке оперативной памяти сегменты, не требующиеся для выполнения запрошенной операции, могут откачиваться во внешнюю память. В зависимости от наличных ресурсов внешняя память (массовая) или память для сегментов может быть нескольких типов в соответствии со скоростью доступа. Так, например, можно перемещать сегменты вначале на электронный диск, затем в файл прямого доступа на диск и, наконец, в списоковый файл на диск. Такая организация памяти обеспечивает значительную независимость ИГС от количества обрабатываемой информации.

В предлагаемой ИГС реализована одноуровневая система циркуляции сегментов, т.е. оперативная память — списоковый файл на диске.

7. Технологические аспекты реализации ИГС

При проектировании ИГС важным этапом является этап разработки оформления экрана дисплея. Оформление экрана связано с планировкой экрана, размещением различных полей для вывода графической и символьной информации, выбором цветовых сочетаний, размещением пояснительных надписей и т.п.

Эта работа проводится в несколько итераций и требует некоторых эстетических усилий. Окончательный вид экрана при работе с ИГС устанавливается обычно уже во время работы системы. Поэтому важно иметь возможность раздельной подготовки оформления экрана и собственно ИГС. При реализации ИГС ТО все оформительские работы были выполнены с помощью графического редактора, разработанного В. А. Дебеловым.

Было подготовлено 13 файлов, содержащих полное оформление экрана, включая кнопки меню, названия подсистем, расположение полей и т.п. Кроме того, подготовлено пять файлов, содержащих

описание вспомогательных меню. Графические образы для некоторых указателей были подготовлены также по этой технологии. Использование таких изображений указателей и вообще любых и изображений, которые могут перемещаться по экрану, опирается на средства сохранения изображения прямоугольных областей экрана, т.е. перед выводом указателя на экран производится сохранение накрываемой им области, а при смещении – эта накрытая область экрана восстанавливается.

Другой технологический аспект реализации ИГС ТО связан с установлением соответствия между программными полями ИГС и полями на экране дисплея. Для этой цели использована система опорных точек изображения – аттачеров, т.е. выделяется система точек, позволяющая получить основные размеры и расположения элементов изображения. Так, для ИГС ТО аттачёрами помечались углы полей экрана, предназначенных для вывода информации ИГС (графической, текстовой, числовой). Установка аттачёров производилась одновременно с подготовкой оформительских изображений экрана.

8. Реализация ИГС

Интерактивная графическая система по подготовке транспортных обстановок реализована в виде четырёх независимых интерактивных программ. Связь между этими программами поддерживается с помощью общих файлов. ИГС ТО реализована на ЭВМ IBM PC AT в операционной среде MS DOS. Примерный объём системы в операторах Фортрана составляет 30 тыс.

С.А.Упольников

ВИЗУАЛИЗАЦИЯ РЕЛЬЕФА МЕСТНОСТИ

Введение

Задача визуализации рельефа местности решалась в рамках создания экспертной системы электронной навигации [1]. Суть этой задачи заключается в построении на экране дисплея изображения, моделирующего вид местности "из окна".

Выбор ракурса вида осуществляется моделированием "полета" над местностью и управляется набором параметров, характеризующих положение и ориентацию наблюдателя. Для удобства выбора ракурса в режиме диалога строится дополнительный "вид сверху" на сцену, включающую наблюдателя и участок местности. Для повышения реалистичности изображения учитывается внешний источник освещения - Солнце . На изображении рельефа местности выделяются с помощью теней участки, невидимые для другого наблюдателя, также произвольно расположенного в пространстве.

Статья посвящена описанию конкретных методов, алгоритмов и технических приемов, использованных при решении сформулированной выше задачи. Реализация выполнена в виде библиотеки Фортран-программ для РС с графическим адаптером EGA.

I. Система проецирования

Исходный участок местности задается в виде файла высот фиксированных точек и файла признаков, характеризующих при-

надлежность этих точек конкретному типу местности (болото, озеро, горы и т.п.). Содержимое каждого файла представляет собой прямоугольную матрицу чисел.

Поверхность наблюдения моделируется следующим образом. Рассматривается бесконечная плоскость — море, на которой выделен прямоугольник. Над этим прямоугольником в соответствии с матрицей высот определяется однозначная функция, по которой и строится поверхность конкретного участка местности. Размеры участка и размерность матрицы высот являются исходными данными.

I.I. Рельеф

Пусть в пространстве задана прямоугольная правая декартова система координат XYZ с единицами измерения по осям в метрах; XYZ — глобальная система координат. На плоскости XH рассмотрим прямоугольник со сторонами, лежащими на осях X, Y , и с координатами одной из вершин (RX, RY) , где RX, RY — размеры исходного участка. На прямоугольнике строится равномерная по каждой из осей прямоугольная сетка с числом вершин $N \times M$, где N, M — размерность исходной матрицы высот. Каждая из прямоугольных ячеек разбивается на два треугольника диагональю, параллельной прямой $(x + y = 0)$. Построенная сетка (рис. I.) и матрица высот однозначно определяют триангулированную поверхность.

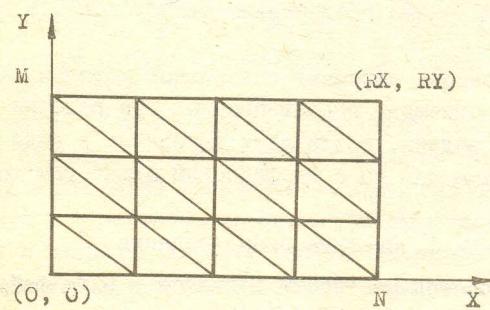


Рис. I.

Чтобы в поверхности, моделирующей местность, не было разрывов, по границе области добавляются вертикальные четырехугольные грани, опирающиеся на плоскость XY, которые также делятся на треугольники. Они образуют так называемый срез. Пример поверхности рельефа показан на рис. 2.

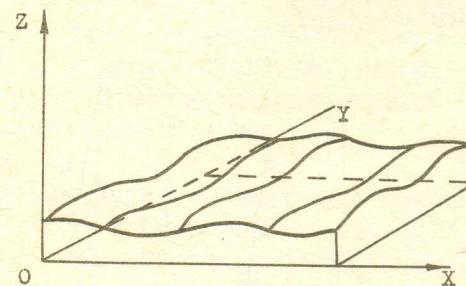


Рис. 2.

В дальнейшем для построения поверхности рельефа предполагается использовать билинейную интерполяцию.

В описываемой реализации размерность матрицы высот равнялась 64×32 и под нее отводился массив в оперативной памяти.

I.2. Наблюдатель

Пусть в начальный момент времени наблюдатель расположен в точке наблюдения $e_n = (nx, ny, nz)$ и главная ось зрения направлена в точку исследования $e_i = (ix, iy, iz)$. Координаты задаются в глобальной системе XYZ. Перед наблюдателем (между точкой наблюдения и точкой исследования) перпендикулярно главной оси зрения на расстоянии d от наблюдателя расположена экранная плоскость.

Локальная система координат наблюдателя $X'Y'Z'$ в начальный момент времени (при инициализации системы) строится следующим образом. Начало координат – точка пересечения главной оси зрения и экранной плоскости. Ось z' совпадает с главной осью зрения и направлена к наблюдателю. Ось y' направлена параллельно проекции оси Z глобальной системы на экранную

плоскость (проецирование осуществляется параллельно главной оси зрения). Ось x' определяется однозначно, так как локальная система – прямоугольная правая декартова. Единицы измерения локальной системы – метры. Естественно предполагается, что в начальный момент времени главная ось зрения не перпендикулярна плоскости XY . На рис. 3 иллюстрируется построение локальной системы координат.

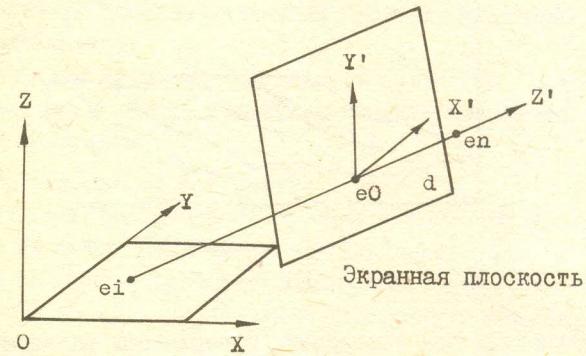


Рис. 3

Локальная система координат полностью определяется точкой начала e_0 и тремя единичными векторами, образующими ее базис e_x , e_y , e_z (все координаты определяются в глобальной системе). При каждом изменении положения и ориентации наблюдателя эти параметры пересчитываются и используются для построения изображений.

I.3. Управление наблюдателем

Для управления ориентацией наблюдателя используются дополнительные параметры: углы поворота, крена и тангажа.

Угол поворота – угол между положительным направлением оси Y глобальной системы и проекцией вектора ($-e_z$) на плоскость XY . По предположению, в начальный момент времени этот угол существует. Угол крена – угол между проекцией оси Z глобальной системы на экранную плоскость и вектором (e_y). В начальный момент этот угол равен нулю. Угол тангажа – угол

между плоскостью XY глобальной системы и вектором (ez). Эти параметры насчитываются в начальный момент времени и модифицируются при каждом изменении ориентации наблюдателя.

Управление ориентацией осуществляется заданием приращений этих параметров. Пересчет базисных векторов локальной системы ex, ey, ez выполняется только с использованием приращений. Поворот, крен и тангаж - вращения базисных векторов вокруг осей Y', z' и X' локальной системы соответственно. Независимый пересчет базисных векторов обеспечивает корректность локальной системы координат при любых "виражах", выполняемых наблюдателем.

Управление положением наблюдателя осуществляется заданием либо его координат в глобальной системе, либо относительного смещения вдоль текущей оси z' локальной системы координат.

Для удобства управления наблюдателем в режиме диалога формируется дополнительный "вид сверху", на котором при любых изменениях параметров наблюдателя схематично отображаются его ориентация, положение относительно рельефа и сектор обзора. Дополнительный вид формируется в "темпе нажатия клавиши".

2. Система отображения

Прямое моделирование вида из окна предполагает следующую процедуру обработки. На экранной плоскости выделяется прямоугольная область, называемая окном. Из точки en ("глаз" наблюдателя) через каждую точку окна проводится луч. Ищется ближайшее пересечение этого луча с поверхностью, моделирующей местность. Возможны следующие случаи:

- пересечения нет (точка окна соответствует небу);
- пересечение с плоскостью XY (точка окна соответствует морю);
- пересечение с рельефом (точка окна соответствует видимому элементу рельефа).

Затем каждая точка окна интерпретируется на экране дисплея подходящим образом.

Учитывая ряд факторов, для построения изображения, моделирующего вид из окна, выбрана система обратного проецирования элементов поверхности, которая описывается ниже.

Рассматривались следующие факторы:

- число элементов модели местности существенно меньше числа элементов изображения (пикселей); в данной модели это соотношение порядка 1:100;
- относительная ресурсоемкость процедуры пересечения луча с поверхностью ограничивает возможности использования программ в режиме диалога;
- однозначность поверхности, моделирующей местность, позволяет использовать простейшие методы удаления невидимых элементов поверхности;
- геометрическая простота элементов поверхности (треугольные грани) дает возможность применить специальные методы клиппирования и заливки.

2.1. Преобразование проецирования

Метод обратного проецирования состоит в том, что рассматриваются лучи из точки наблюдения в точку поверхности и ищется их пересечение с экранной плоскостью. Затем выполняется клиппирование по окну на экранной плоскости и удаление невидимых точек.

Пересечение лучей с экранной плоскостью эквивалентно обратному перспективному проецированию точек поверхности на эту плоскость. Учитывая, что элементы поверхности — треугольники, достаточно проецировать только их вершины.

Преобразование проецирования выполняется в два шага: аффинное преобразование координат точки поверхности из глобальной в локальную систему координат; перспективное проецирование на экранную плоскость.

Если ввести обозначение для вектора столбца — $\langle \rangle$ и для матрицы — $[]$, то аффинное преобразование координат точки $t = (x, y, z)$ из глобальной в координаты $t' = (x', y', z')$ локальной системы выполняется по формуле

$$\langle t' \rangle = [\langle ex \rangle, \langle ey \rangle, \langle ez \rangle] \times \langle \langle t \rangle - \langle e0 \rangle \rangle.$$

Локальная система координат индуцирует на экранной плоскости систему координат $x'y'$. Перспективное проецирование

точки $t = (x', y', z')$ из локальной системы координат в координаты экранной плоскости (xs, ys) , расположенной на расстоянии d от наблюдателя, выполняется по следующим формулам:

$$xs = x' \times d / (d - z'),$$

$$ys = y' \times d / (d - z').$$

Необходимым условием для корректного выполнения этого преобразования является $z < 0$. Естественно, что данное условие выполняется далеко не всегда, поскольку локальная система координат может быть ориентирована произвольно. Для его соблюдения перед преобразованием проецирования выполняется клипирование элемента поверхности. Преобразование клипирования рассматривается в п. 3.

2.2. Экранное преобразование

При преобразовании клипирования в качестве окна на экранной плоскости (плоскость z' локальной системы) рассматривается прямоугольник со сторонами, параллельными координатным осям, и с центром в точке с координатами $(0, 0)$. Размеры окна определяются по размерам прямоугольной области, выделенной на экране дисплея для построения изображения вида. Параметры этой области являются исходными данными и задаются в левосторонней (начало в левом верхнем углу) экранной системе координат, стандартной для дисплеев IBM PC.

Если $(i1, j1, i2, j2)$ – параметры области на экране, заданные в пикселях ($i1 < i2$ и $j1 < j2$), то размеры окна клипирования rx , ry в единицах локальной системы определяются по формулам

$$rx = (i2 - i1)/xm,$$

$$ry = (j2 - j1)/ym.$$

Экранное преобразование точки (xs, ys) в пиксельные координаты (i, j) экранной системы выполняется по следующим формулам:

$$i = [(i_2 + i_1)/2 + xs \times xm],$$

$$j = [(j_2 + j_1)/2 - ys \times ym].$$

В данных выражениях [] – знак взятия целой части числа; xm , ym – коэффициенты пересчета из пикселей в метры, зависящие от разрешения дисплея и его размеров. Например, для стандартного экрана IBM PC с графическим адаптером EGA (640x350) они равны примерно 2 560 и 1 940.

3. Формирование основного вида

Для формирования изображения поверхности местности был применен алгоритм упорядоченного отображения граней. Возможность его применения объясняется тем, что поверхность является однозначной относительно плоскости XY и изображение строится на экране растрового дисплея, т.е. цветовые характеристики пикселя на экране соответствуют последней из отображенных в него точек поверхности. Общая логическая схема алгоритма построения изображения включает: формирование изображений моря и неба, выбор очередного элемента рельефа, клиппирование элемента рельефа, проецирование и визуализацию элемента рельефа.

3.1. Построение фона

Предполагается, что все точки рельефа не ниже уровня моря. Следовательно, если наблюдатель расположен выше уровня моря, то для него никакая точка рельефа не может быть закрыта морем. Поэтому море и небо изображаются, в первую очередь, в виде фона для рельефа.

Первоначально область экрана, отведенная для построения изображения, заливается цветом моря. Затем на экранной плоскости вычисляется линия горизонта – линия пересечения экранной плоскости и плоскости, параллельной плоскости XY глобальной системы и расположенной на высоте наблюдателя. Если эта линия не существует или не пересекает окна на экранной плоскости, то в зависимости от ориентации наблюдателя вся выделенная область экрана заливается цветом неба или остается без изменения. Если пересечение существует, то соответству-

щая часть выделенной области заливается цветом неба. Для анализа используется алгоритм клиппирования отрезка по прямоугольнику. Построение линии горизонта иллюстрируется на рис. 4.

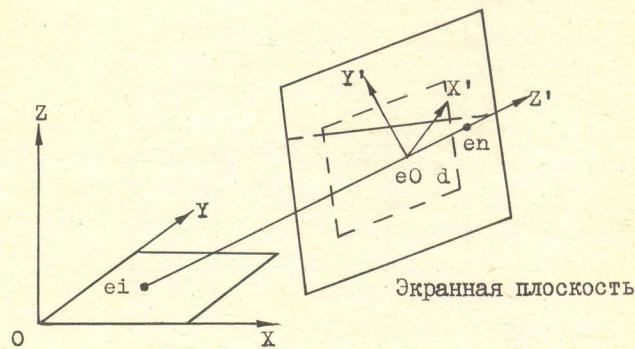


Рис. 4

3.2. Выбор элемента рельефа

Поверхность рельефа считается непрозрачной. Порядок выбора элементов рельефа (треугольных граней) определяется следующим правилом: выбранный элемент не может закрывать от наблюдателя элементы, выбираемые после него.

Определим нормали для элементов поверхности рельефа. На области задания рельефа (сеточная область на плоскости XY) введем ориентацию элементов против часовой стрелки, если смотреть из положительного полупространства (по Z) на эту плоскость, как показано на рис. 5. Ячейка с левым нижним углом в точке (i, j) содержит два ориентированных треугольника: $(i, j), (i + 1, j), (i, j + 1)$ и $(i + 1, j), (i + 1, j + 1), (i, j + 1)$.

Эта ориентация однозначно индуцирует ориентацию граней рельефа, включая и грани среза. Если две грани имеют общее ребро, то порядок его вершин меняется на противоположный при переходе с одной грани на другую. Такая ориентация называется согласованной. Нормаль для каждой грани определяется по

правилу "правого винта" и по ее ориентации и может быть вычислена как результат векторного произведения двух подряд идущих ребер, рассматриваемых как векторы. Результирующий вектор нормируется.

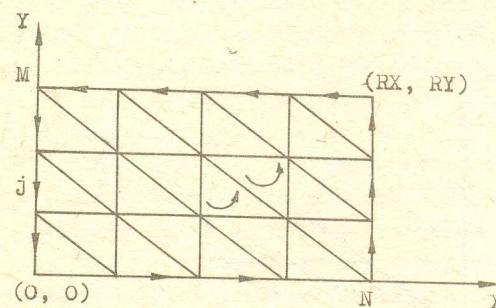


Рис. 5

Будем считать, что грань невидима, если проекция вектора ее нормали на вектор из точки этой грани в точку наблюдения отрицательна. Проекция вычисляется как скалярное произведение этих векторов. Для построения вектора в точке наблюдения можно брать любую точку грани, например ее вершину, так как нулевая проекция соответствует тому, что точка наблюдения лежит в плоскости этой грани. Границы, не являющиеся невидимыми, будем считать видимыми.

Учитывая однозначность и непрерывность поверхности рельефа, можно доказать корректность введенного определения невидимости, т.е. точка пересечения луча от наблюдателя с внутренностью невидимой грани не может быть ближайшей к наблюдателю на этом луче и, следовательно, видимой для него. Предполагается, что z -координата точки наблюдения $en = (x, y, z)$ больше высоты местности в точке (x, y) .

Можно также доказать, что если две грани имеют общее ребро, согласованно ориентированы и обе пересекаются некоторой прямой по внутренним точкам, то проекции нормалей к граням, индуцированные этой ориентацией, на данную прямую будут противоположно направленными.

Из последних двух утверждений следует, что порядок выбора граней, имеющих общее ребро, несущественен, так как если обе грани видимы, то они не могут закрыть друг друга, а невидимые грани не отображаются. Поэтому достаточно выбирать правильно ячейки сетки (пары граней).

Порядок выбора ячеек сетки полностью определяется проекцией точки наблюдения на плоскость $z = 0$ и ее расположением относительно области задания рельефа. Пусть ячейка идентифицируется левым нижним углом. Сетка с $N \times M$ вершинами содержит $(N - 1) \times (M - 1)$ ячеек. Определим новые ячейки с номерами $(i, 0), (i, M), (0, j), (N, j)$, где $i = 1, \dots, N; j = 1, \dots, M$, для граней среза рельефа. Ячейки выбираются по строкам. Порядок выбора строк ячеек определяется y -координатой точки наблюдения: $(y < 0) - M - 1, M - 2, \dots, 0; (y > RY) - 1, 2, \dots, M; (0 < y < RY) - 1, 2, \dots, j(y), M - 1, \dots, j(y) + 1$. Аналогично определяется порядок выбора ячеек в строке в зависимости от x -координаты точки наблюдения: $(x < 0) - N - 1, N - 2, \dots, 0; (x > RX) - 1, 2, \dots, N; (0 < x < RX) - 1, 2, \dots, i(x), N - 1, \dots, i(x) + 1$. Один из возможных вариантов порядка выбора ячеек иллюстрируется на рис. 6.

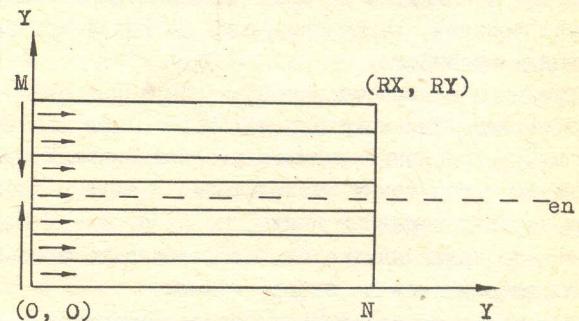


Рис. 6

3.3. Клиппирование элементов рельефа

Клиппирование граней выполняется по следующим причинам:

- непосредственное применение перспективного проецирования для наблюдателя, расположенного и ориентированного произвольно, может вызвать непредсказуемые результаты (переполнение, деление на нуль, искажение результата);
- должен быть контроль "врезания" наблюдателя в рельеф;
- необходима оптимизация программ обработки (элементы рельефа, не попадающие в окно, не обрабатываются).

Общая схема клиппирования включает следующие этапы: контроль врезания; обработку граней, не попавших или частично попавших в пирамиду видимости; клиппирование при визуализации по окну на экране. Последний этап выполняется в используемых базовых графических программах [2] и здесь не рассматривается.

Контроль врезания выполняется следующим образом. Анализируется каждая из четырех вершин окна, построенного на экранной плоскости. Если ее z-координата меньше z-координаты точки поверхности рельефа или моря, лежащей непосредственно под ней, то считается, что произошло врезание и выводится предусмотренное "пустое" изображение. Затем формируется специальная таблица, в которой каждой вершине рельефа сопоставляется признак, характеризующий ее положение относительно пирамиды видимости.

Пирамида видимости определяется пятью ориентированными плоскостями, как показано на рис. 7. При построении таблицы анализируется конус видимости, образованный плоскостями, проходящими через точку наблюдателя, и затем экранная плоскость. Если вершина попала в конус, но не попала в пирамиду видимости, то также считается, что произошло врезание. Этот контроль выполняется до выбора граней.

Обработка по пирамиде видимости производится (в соответствии с построенной таблицей) после выбора очередной ячейки для каждой ее грани. В таблице каждой вершине присвоен логический код, определяющий положение ее проекции относительно прямоугольной области на экранной плоскости. Возможные

области расположения проекции показаны на рис. 8. Для каждой грани выполняется простейший логический анализ признаков, приписанных ее вершинами, и грани, не попавшие в поле видимости, отбраковываются.

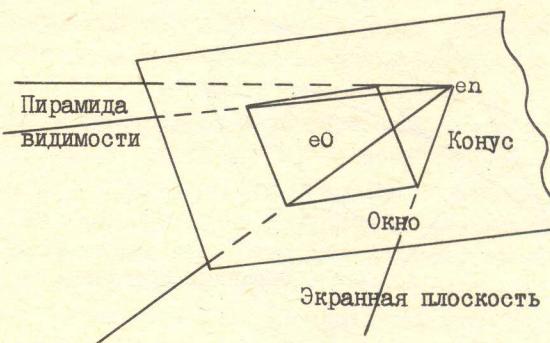


Рис. 7

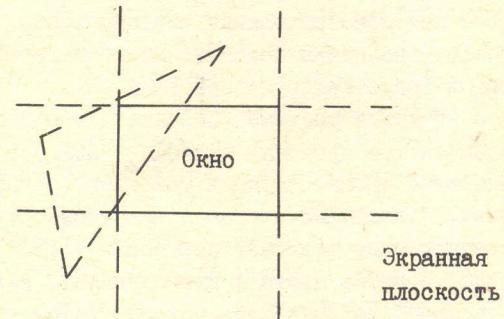


Рис. 8

После анализа грани по ее нормали и выполнения аффинного преобразования в локальную систему координат вершины грани, расположенные со стороны наблюдателя относительно экранной плоскости, "сносятся" на нее. Это обеспечивает корректность

выполнения перспективного преобразования, но вызывает искажение проекции рельефа на границах экранной области. Эта ситуация возникает, в частности, при моделировании "полета" среди гор и иллюстрируется на рис. 9.

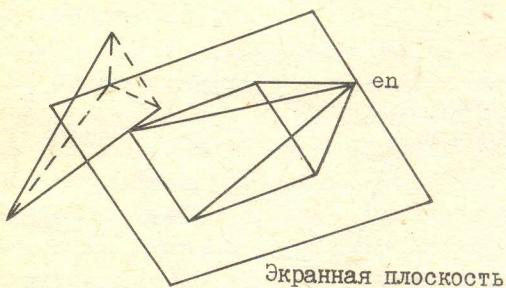


Рис. 9

3.4. Отображение элемента рельефа

Если грань не была отбракована при анализе ее видимости по нормали и при преобразовании клипирования, то она отображается. После выполнения перспективного и экранного преобразований грань представляет собой треугольник с вершинами, заданными в экранной системе координат (в пикселях).

Визуализация или цветовая заливка грани осуществляются высвечиванием на экране дисплея пикселей, лежащих внутри треугольника, подходящим цветом и интенсивностью. Следовательно, визуализация включает перебор пикселей (развертку треугольника) и выбор цвета и интенсивности каждого пикселя.

Метод цветовой заливки образа грани рассматривается в следующем пункте. Для его использования необходима следующая информация: положение грани на рельефе с целью определения типа поверхности местности, к которой она относится, по файлу признаков; значения высот ее вершин и коэффициентов ослабления интенсивности в зависимости от положения источника освещения – Солнца.

Положение грани определяется координатами текущей ячейки и номером грани в ней. Высоты вершин вычисляются по матрице

высот. В качестве коэффициента ослабления используется косинус угла между вектором нормали к грани и вектором, направленным на источник параллельного освещения. Этот коэффициент изменяется от 1 – максимальная интенсивность, до 0 – минимальная интенсивность (если косинус отрицательный, коэффициент принимает значение нуль). Фактически, с помощью коэффициента моделируется простейший закон отражения – закон косинусов Ламберта [3].

3.5. Расчет теней

При формировании вида требовалось выделить (с помощью затенения) участки местности, невидимые из фиксированной точки пространства (для другого наблюдателя), для чего была реализована программа определения видимости произвольной точки наблюдения.

Алгоритм работы этой программы следующий. По двум точкам (наблюдатель и исследуемая точка) формируется отрезок, который клипируется по габаритному параллелепипеду рельефа. Если пересечение пусто, то точка видима, в противном случае ищется пересечение отрезка, полученного в результате клипирования с гранями рельефа.

Для оптимизации выбора граней используется следующий прием. Рассматривается проекция отрезка на плоскость XY и по аналогии с алгоритмом Брезенхема [3] последовательно выбираются ячейки сетки – области задания рельефа, через которые проходит проекция. Пример последовательности выбора ячеек приведен на рис. 10. Ищется пересечение отрезка с гранями выбираемых ячеек. Расчет теней выполняется только для узловых вершин рельефа. Формируется матрица теней, которая используется при визуализации граней.

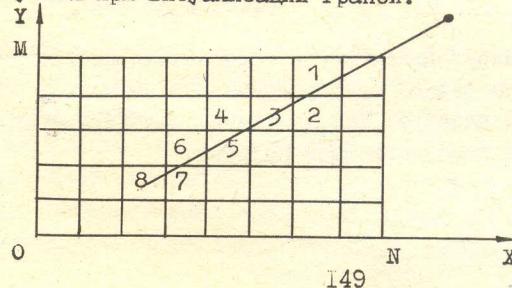


Рис. 10.

4. Цветовая закраска и тени

Графический адаптер EGA позволяет высвечивать на экране одновременно не более 16 различных оттенков (цвета и интенсивности) из палитры в 256 оттенков. Каждый оттенок определяется заданием интенсивностей базовых цветов RGB (красный, зеленый, синий). Интенсивность базового цвета может принимать значения 0, 1, 2, 3. Даже для построения условного изображения местности 16 оттенков недостаточно.

4.1. Смешение цветов

В используемом пакете базовых графических программ [2] при визуализации отрезка предусмотрен режим смешения двух цветов, который позволяет для закрашенных областей получить на экране одновременно до 128 псевдооттенков.

Псевдооттенок образуется при высвечивании пикселей некоторой области двумя фиксированными оттенками в шахматном порядке. Если i, j – координаты пикселя, то высвечивание выполняется следующим образом: i -четное, j -четное – оттенок 1; i -четное, j -нечетное – оттенок 2; i -нечетное, j -четное – оттенок 2; i -нечетное, j -нечетное – оттенок 1. Использование более сложных структур смешения вызывает явное проявление текстуры на закрашиваемой области.

В качестве базовых оттенков используется следующая палитра цветов:

(0, 0, 0) – черный цвет;

(1, 1, 1), (2, 2, 2), (3, 3, 3) – оттенки серого,

(1, 0, 0), (2, 0, 0), (3, 0, 0) – оттенки красного,

(0, 1, 0), (0, 2, 0), (0, 3, 0) – оттенки зеленого,

(0, 0, 1), (0, 0, 2), (0, 0, 3) – оттенки синего,

(1, 1, 0), (2, 2, 0), (3, 3, 0) – оттенки желтого.

Для подбора оттенков применяется специальная программа, которая отображает на экране одновременно все возможные смеси для фиксированной палитры. Каждый псевдоцвет характеризуется номерами двух цветов базовой палитры.

4.2. Таблица соответствия цветов

Выбор цвета грани осуществляется по таблице цветов, которая формируется на основе разделения местности на участки различного типа.

В исходной матрице признаков каждой вершине матрицы высот приписан один из следующих типов местности: населенный пункт, лес, озеро, болото, условная местность. Тип "условная местность" предполагает анализ вершин по высоте и подразделяется на следующие подтипы: зеленые луга, степи, лиственные леса, хвойные леса, каменистые горы, снежные горы. Для анализа по высоте задается шкала высот, характерная для конкретного рельефа.

Каждому типу местности в таблице соответствуют пять уровней интенсивности (зависимость от освещенности солнцем). Для выбора уровня интенсивности по коэффициенту задается шкала интенсивностей.

Каждый элемент таблицы содержит номера двух цветов базовой палитры, определяющих подобранный для этого элемента псевдоцвет. Кроме того, предусмотрены цвета для граней среза, моря и неба.

4.3. Заливка треугольника

Реализованы два способа заливки треугольников: однородная заливка и неоднородная. В обоих случаях развертка треугольников выполняется классическим способом – по строкам [3]. Для корректности визуального образа границ треугольника вычисление концов закрашиваемого отрезка очередной строки осуществляется генерацией точек граничных ребер по алгоритму Брезенхема [3].

При однородной заливке каждый треугольник закрашивается одним цветом. Этот цвет определяется по таблице цветов в зависимости от типа одной из вершин, значения ее высот и коэффициента ослабления интенсивности для этого треугольника. Для анализа используется вершина прямого угла треугольника – прообраза грани рельефа.

Неоднородная заливка выполняется следующим образом. Треугольник разбивается на три четырехугольные области, как по-

казано на рис. II. Каждой области приписывается тип соответствующей ей вершины. Если тип не является условной местностью, то вся область отображается соответствующим цветом. Если тип - условная местность, то область отображается по аналогии с методом Гуро [3], но изменяемым параметром является высота точки, соответствующей пикслю, в зависимости от которой и вычисляется ее цвет по таблице.

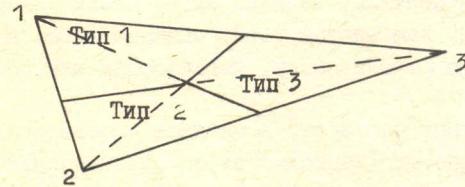


Рис. II.

4.4. Формирование теней

Формирование теней выполняется одновременно с заливкой треугольника. Определение того, что треугольник или одна из областей его разбиения является теневой, выполняется по соответствующей ей вершине из сформированной матрицы теней. Если заливается теневая область, то вместо выбранных для этой области цветов базовой палитры берутся цвета с меньшими интенсивностями. Интенсивность 1 заменяется на интенсивность 0; интенсивности 2, 3 - на интенсивность 1 того же цвета. Структура базовой палитры позволяет легко алгоритмизировать такой выбор.

Заключение

Реализация выполнена в виде библиотеки Фортран-программ. Эти программы вызываются из диалоговых подсистем системы электронной навигации [1]. Наряду с программами формирования изображения библиотека включает дополнительные программы: нахождение ближайшего видимого прообраза для точки на экране, определение расстояния от экрана до прообраза; пересчет ко-

ординат точки с экрана одного наблюдателя в экранные координаты другого наблюдателя и т.п.

Дальнейшее развитие программ предполагается вести по двум направлениям:

- оптимизации времени построения изображения (в текущем варианте оно составляет 30-50 с на РС АТ-286 с сопроцессором);
- повышению качества и реалистичности изображения за счет настраиваемого разбиения граней в зависимости от габаритов их образов на экране и использования специальных методов текстурирования при их заливке.

Л и т е р а т у р а

1. Сиротин В.Г., Витяев Е.Е. Проект экспертной системы электронной навигации// Наст. сб. - С. 92-107.
2. Ткачев Ю.А. Комплект графических программ GRAF'ER// Наст. сб. - С. 20-59.
3. Фоли Дж., Вэн-Дэм А. Основы интерактивной машинной графики. - М.: Мир, 1985. - Т. 2.

С о д е р ж а н и е

Предисловие	3
Дебелов В.А. Об одном подходе к разработке интерактивных графических программ	5
Ткачев Ю.А. Комплект графических программ GRAPH'ER . . .	20
Голубев В.М. Системы геометрических построений. Постанов- ки задач и методы их решения	60
Шулта Н.С. Система конструктивного геометрического моде- лирования твердых тел	77
Сиротин В.Г., Витяев Е.Е. Проект экспертной системы элек- тронной навигации	92
Сиротин В.Г., Коновалов А.В. Библиотека программ для представления и обработки графических данных ме- тодом квадродержевьев	108
Торшин В.И. Интерактивная графическая система для подго- товки транспортных обстановок	122
Упольников С.А. Визуализация рельефа местности	135

МАШИННАЯ ГРАФИКА
И ГЕОМЕТРИЧЕСКОЕ МОДЕЛИРОВАНИЕ

Сборник научных трудов
Под редакцией Александра Михайловича Мацокина

Позиция изд. плана № 1

Редактор Е.В.Блинова

Ответственный за выпуск В.Г.Сиротин

Обложка художника Д.К.Томашевского

Подписано в печать 19.07.90г.

МН 11004

Формат бумаги 60x84 1/16 Объем 9,6 п.л., 9 уч.-изд. л.

Тираж 350 экз. Цена 70 коп. Заказ № 256

Ротапринт ВЦ СО АН СССР, Новосибирск, 90