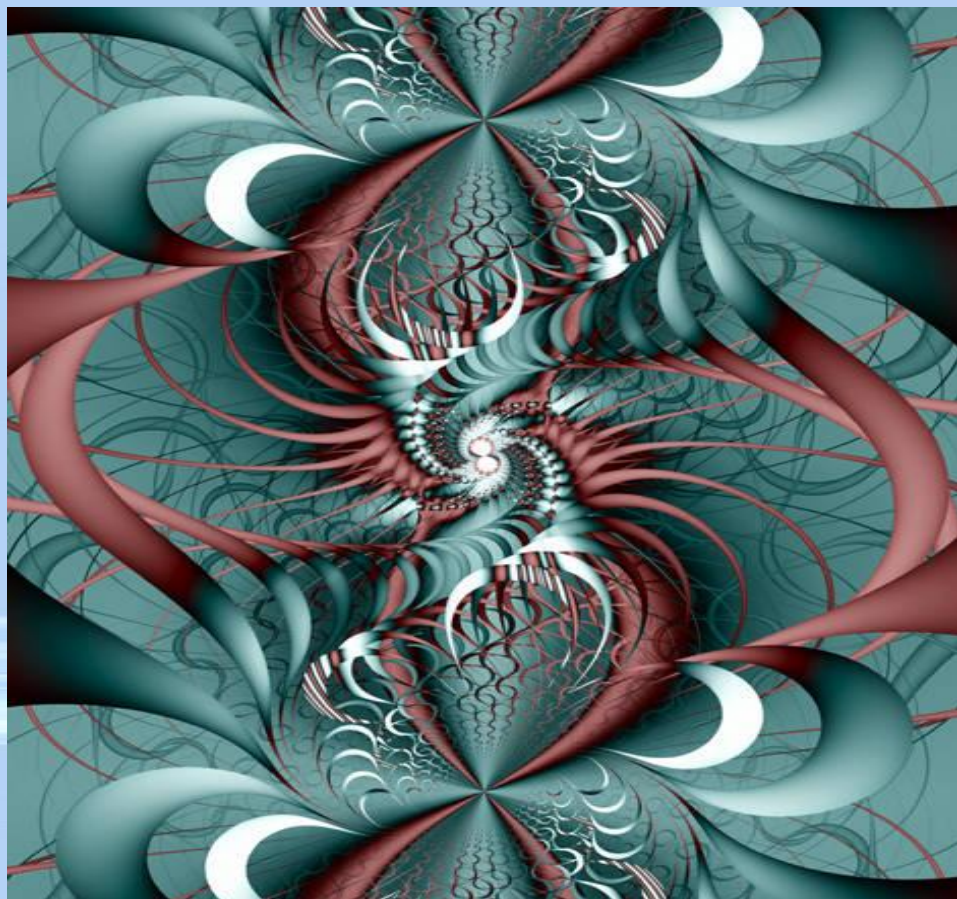


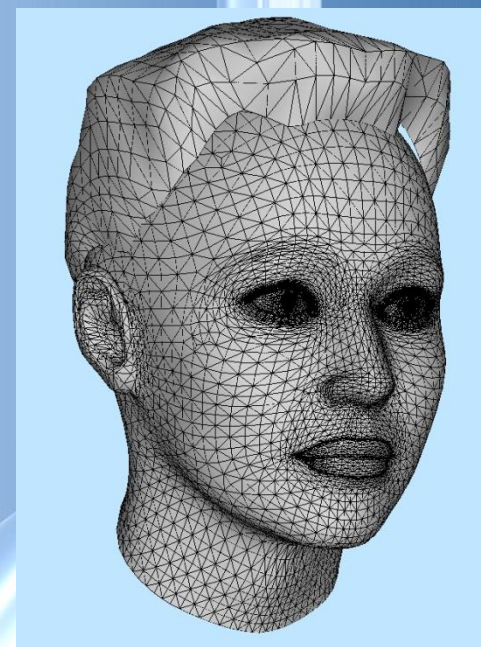


Математические основы компьютерной графики





Компьютерная графика – это к настоящему времени достаточно развитая и многосторонняя отрасль знания. Существует аппаратное и программное обеспечение для получения разнообразных изображений – от простых иллюстраций и чертежей до реалистических образов естественных объектов. Спектр использования компьютерной графики чрезвычайно широк: реклама и различного рода презентации, компьютерные игры и мультипликация, малые и монументальные формы дизайна, компьютерная живопись и архитектура, медицина, проектно-конструкторские разработки, административное управление и т.д. и т.п. Особое значение приобретает компьютерная графика в науке и образовании, становясь фактически новым инструментарием в процессе получения знаний.





История

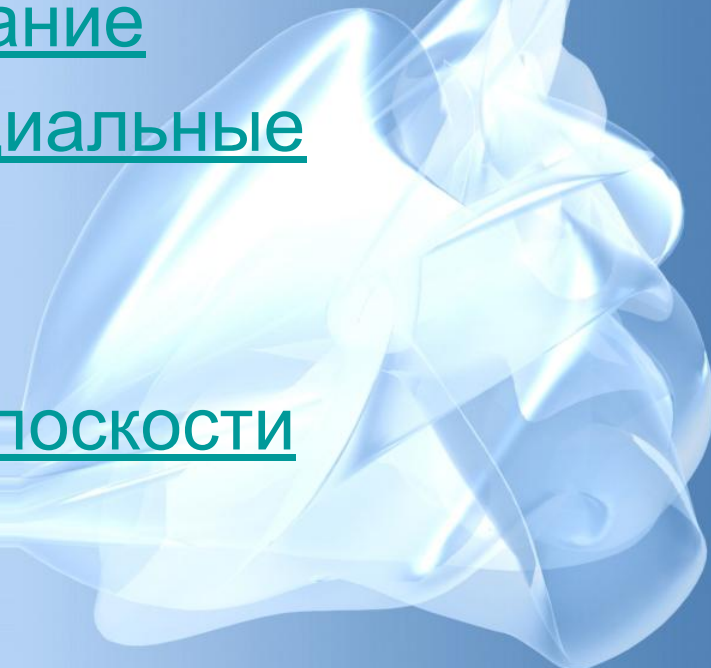
- Первые вычислительные машины не имели отдельных средств для работы с графикой, однако уже использовались для получения и обработки изображений. Программируя память первых электронных машин, построенную на основе матрицы ламп, можно было получать узоры.
- В 1961 году программист С. Рассел возглавил проект по созданию первой компьютерной игры с графикой. Создание игры «Spacewar» («Космические войны») заняло около 200 человеко-часов. Игра была создана на машине PDP-1.
- В 1963 году американский учёный Айвен Сазерленд создал программно-аппаратный комплекс Sketchpad, который позволял рисовать точки, линии и окружности на трубке цифровым пером. Поддерживались базовые действия с примитивами: перемещение, копирование и др. По сути, это был первый векторный редактор, реализованный на компьютере. Также программу можно назвать первым графическим интерфейсом, причём она являлась таковой ещё до появления самого термина.
- В середине 1960-х гг. появились разработки в промышленных приложениях компьютерной графики. Так, под руководством Т. Мофетта и Н. Тейлора фирма Itek разработала цифровую электронную чертёжную машину. В 1964 году General Motors представила систему автоматизированного проектирования DAC-1, разработанную совместно с IBM.
- В 1968 году группой под руководством Константинова Н.Н. была создана компьютерная математическая модель движения кошки. Машина БЭСМ-4, выполняя написанную программу решения дифференциальных уравнений, рисовала мультфильм «Кошечка», который для своего времени являлся прорывом. Для визуализации использовался алфавитно-цифровой принтер.

Фундаментом для формирования и редактирования изображений служат **математические и алгоритмические основы компьютерной графики** (в первую очередь, это относится к изображениям, закодированным в векторной форме). Изучению именно таких основ компьютерной графики, используемых при обработке графических данных для вывода изображения на растровое графовыводящее устройство, а также последующего его видоизменения.



Математические основы компьютерной графики :

- Преобразования в двухмерном пространстве
- Преобразования в трехмерном пространстве
- Аффинное проецирование
- Перспективное проецирование
- Стереографическая и специальные перспективные проекции
- Масштабирование в окне
- Нахождение параметров плоскости





Преобразования в двумерном пространстве

- Преобразования в двумерном пространстве используются в разнообразных случаях: чтобы отдельные части объекта можно было описывать в различных координатных системах; чтобы типовые и повторяющиеся части можно было располагать в произвольных положениях на чертеже и в пространстве, в том числе с использованием циклов; чтобы без повторной кодировки можно было получать симметричные части объекта; для направленной деформации фигур, тел и их частей; для изменения масштаба чертежа, построения проекций пространственных образов... С аналитической точки зрения преобразования — это пересчет значений координат.

- Преобразование точки**

- Точка на плоскости представляется двумя координатами: $[x \ y]$. Матрица преобразования точки выглядит так:

$$\begin{vmatrix} a & b \\ c & d \end{vmatrix}$$

- Нижеследующее преобразование точки через квадратную матрицу; здесь $x_n = xa + yc$ и $y_n = xb + yd$ — новые координаты точки после преобразования:

$$\begin{vmatrix} x_n & y_n \\ a & b \\ c & d \end{vmatrix}$$

- Преобразование фигуры**

- Если представить фигуру как совокупность точек, то можно провести и ее преобразование. В следующем примере задано четыре точки: $A(0, 0)$, $B(1, 0)$, $C(1, 1)$, $D(0, 1)$, каждая из которых после преобразования перейдет соответственно в $A^*(0, 0)$, $B^*(a, b)$, $C^*(a + c, b + d)$, $D^*(c, d)$:
- Геометрически это соответствует деформации фигуры:

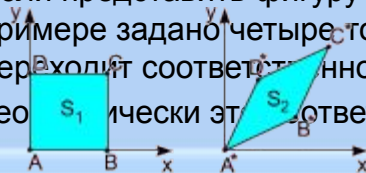


рис. 1.1

- При этом площадь новой фигуры равна площади старой фигуры, умноженной на детерминант матрицы преобразования: $S_2 = S_1 * |ad - bc|$.

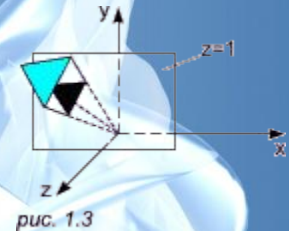
$$\begin{vmatrix} 0 & 0 \\ 0 & 1 \end{vmatrix} \quad \begin{vmatrix} 0 & 0 \\ c & d \end{vmatrix}$$

Однородные координаты. Операции в них

- Любая система координат, в которой представление точки в двумерном (трехмерном) пространстве задается при помощи трех (четырех) координат (P_1, P_2, P_3, P_4)), называется системой однородных координат. Вообще, для n -мерного пространства число однородных координат должно быть на единицу больше: $n + 1$.
- Применение однородных координат в общем случае позволяет устранять аномалии, возникающие при работе в декартовых координатах, и представлять сложные преобразования в виде произведения нескольких матриц.
- Геометрическая интерпретация на случай двумерного пространства: введение третьей координаты, равной единице, можно трактовать как переход в трехмерное пространство, в котором разрешено работать только в плоскости $z = 1$. Следует представлять себе, что экран компьютера (картинная плоскость, плоскость изображения) находится в плоскости $z = 1$:



- В случае выхода рисунка за сечение $z = 1$ рисунок возвращается принудительно в данное сечение — для того, чтобы были возможны последующие операции;
- Такая операция называется нормализацией однородных координат:



- Общий вид преобразования**
- $$\begin{bmatrix} X & Y & 1 \end{bmatrix} * \begin{bmatrix} \dots & p \\ \dots & q \end{bmatrix} = \begin{bmatrix} X^* & Y^* & Z^* \end{bmatrix} = \begin{bmatrix} \frac{X^*}{Z^*} & \frac{Y^*}{Z^*} & 1 \end{bmatrix}$$
- $$\begin{bmatrix} x & y & 1 \end{bmatrix} * \begin{bmatrix} a & b & p \\ c & d & q \\ m & n & s \end{bmatrix} = \begin{bmatrix} ax+cy+m & bx+dy+n & px+qy+s \end{bmatrix} = \begin{bmatrix} \frac{ax+cy+m}{px+qy+s} & \frac{bx+dy+n}{px+qy+s} & 1 \end{bmatrix}$$

- Операция смещения**

$$\begin{bmatrix} x & y & 1 \end{bmatrix} * \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ m & n & 1 \end{bmatrix} = \begin{bmatrix} x+m & y+n & 1 \end{bmatrix}$$

- Матрица преобразования содержит в себе константы m и n , под действием которых точка смещается на m единиц вдоль оси x и на n единиц — вдоль оси y :

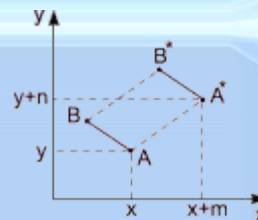


рис. 1.4



- Операция масштабирования**

$$\begin{vmatrix} x & y & 1 \end{vmatrix} * \begin{vmatrix} a & 0 & 0 \\ 0 & d & 0 \\ 0 & 0 & 1 \end{vmatrix} = \begin{vmatrix} ax & dy & 1 \end{vmatrix}$$

- За счет коэффициентов a и d матрицы преобразования происходит увеличение (или уменьшение) значения координат точки (x, y) в a и d раз по осям x и y соответственно:

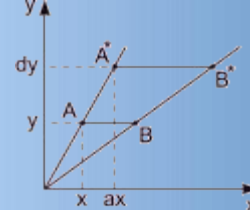


рис. 1.5

- Общее полное масштабирование**

$$\begin{vmatrix} x & y & 1 \end{vmatrix} * \begin{vmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & s \end{vmatrix} = \begin{vmatrix} x & y & s \end{vmatrix} = \begin{vmatrix} \frac{x}{s} & \frac{y}{s} & 1 \end{vmatrix}$$

- В данном случае при $s < 1$ будет происходить увеличение значения координат точки (x, y) в s раз; при $s > 1$ мы получим обратный эффект — уменьшение значения координат (x, y) в s раз.

- Поворот на угол q**

$$\begin{vmatrix} x & y & 1 \end{vmatrix} * \begin{vmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{vmatrix} = \begin{vmatrix} x\cos\theta - y\sin\theta & x\sin\theta + y\cos\theta & 1 \end{vmatrix}$$

- Здесь q — угол, на который требуется повернуть точку (x, y) . Обратите внимание: поворот происходит относительно точки $(0, 0)$ декартовой системы координат против часовой стрелки!

- Отображение или зеркалирование**

- Зеркалирование относительно прямой $y = x$ (рис. 1.6a):

$$\begin{vmatrix} x & y & 1 \end{vmatrix} * \begin{vmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{vmatrix} = \begin{vmatrix} y & x & 1 \end{vmatrix}$$

- Зеркалирование относительно прямой $x = 0$ (рис. 1.6b):

$$\begin{vmatrix} x & y & 1 \end{vmatrix} * \begin{vmatrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{vmatrix} = \begin{vmatrix} -x & y & 1 \end{vmatrix}$$

- Зеркалирование относительно прямой $y = 0$ (рис. 1.6c):

$$\begin{vmatrix} x & y & 1 \end{vmatrix} * \begin{vmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{vmatrix} = \begin{vmatrix} x & -y & 1 \end{vmatrix}$$

- Зеркалирование относительно начала координат (рис. 1.6d):

$$\begin{vmatrix} x & y & 1 \end{vmatrix} * \begin{vmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{vmatrix} = \begin{vmatrix} -x & -y & 1 \end{vmatrix}$$

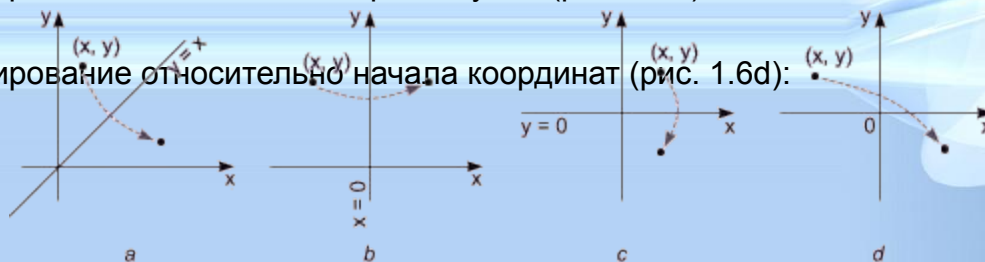


рис. 1.6



- **Поворот фигуры вокруг произвольной точки (m, n) на произвольный угол а**
- Чтобы провести любое сложное преобразование, необходимо разложить его на базовые операции. Поворот фигуры вокруг произвольной точки (m, n) на произвольный угол а состоит из трех базовых операций:
1) перенос фигуры на вектор A(-m, -n) для совмещения точки (m, n) с началом координат; 2) поворот фигуры на угол а; 3) перенос фигуры на вектор A'(m, n) для возвращения ее в исходное положение. Так как фигуру можно представить набором точек, то операции 1) - 3) можно выполнять последовательно для каждой точки. Покажем это на примере.
- Пусть мы хотим повернуть треугольник с координатами A(x, y), B(x₁, y₁), C(x₂, y₂) вокруг точки D(m, n) на угол а. Пусть P_{-s} — матрица переноса точки на вектор A(-m, -n), V_a — матрица поворота на угол а, P_s — матрица переноса точки на вектор A'(m, n).
- Итак, мы имеем все данные, необходимые для проведения сложного преобразования первой точки A(x, y):

$$\begin{aligned} \begin{vmatrix} x & y & 1 \end{vmatrix} * P_{-s} * V_a * P_s &= \begin{vmatrix} x & y & 1 \end{vmatrix} * \begin{vmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -m & -n & 1 \end{vmatrix} * \begin{vmatrix} \cos\alpha & \sin\alpha & 0 \\ -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{vmatrix} * \begin{vmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ m & n & 1 \end{vmatrix} = \\ &= \begin{vmatrix} x & y & 1 \end{vmatrix} * \begin{vmatrix} \cos\alpha & \sin\alpha & 0 \\ -\sin\alpha & \cos\alpha & 0 \\ -m(\cos\alpha-1)+n\sin\alpha & -n(\cos\alpha-1)-m\sin\alpha & 1 \end{vmatrix} \end{aligned}$$

- Точно такие же преобразования необходимо провести для оставшихся двух точек треугольника, подставляя соответствующие их координаты взамен x и y (последовательность операций см. на рис. 1.7). Таким образом, сложная операция разбивается на простейшие и задается произведением соответствующих матриц преобразования, причем порядок, в котором перемножаются матрицы, существенно определяет результат.



рис. 1.7

$$\begin{vmatrix} x & y & 1 \end{vmatrix} * \begin{vmatrix} 1 & 0 & p \\ 0 & 1 & q \\ 0 & 0 & 1 \end{vmatrix} = \begin{vmatrix} x & y & px+qy+1 \end{vmatrix} = \begin{vmatrix} \frac{x}{px+qy+1} & \frac{y}{px+qy+1} & 1 \end{vmatrix}$$

- **Центральное проектирование (перспектива)**

$px + qy + 1 = H$ — плоскость.



Преобразования в трехмерном пространстве

- На прошлой лекции мы изучили операции, которые можно выполнять с точкой на плоскости. Такие же операции имеются и в трехмерном пространстве. Отличие здесь небольшое: точка задается не двумя, а тремя координатами, и при работе в однородных координатах матрицы преобразований/операций будут состоять не из трех, а из четырех столбцов.

- **Операция смещения**

$$\begin{vmatrix} x & y & z & 1 \end{vmatrix} * \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ m & n & k & 1 \end{vmatrix} = \begin{vmatrix} x+m & y+n & z+k & 1 \end{vmatrix}$$

- **Операция масштабирования**

$$\begin{vmatrix} x & y & z & 1 \end{vmatrix} * \begin{vmatrix} a & 0 & 0 & 0 \\ 0 & d & 0 & 0 \\ 0 & 0 & e & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix} = \begin{vmatrix} ax & dy & ez & 1 \end{vmatrix}$$

- **Общее полное масштабирование**

$$\begin{vmatrix} x & y & z & 1 \end{vmatrix} * \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & s \end{vmatrix} = \begin{vmatrix} x & y & z & s \end{vmatrix} = \begin{vmatrix} \frac{x}{s} & \frac{y}{s} & \frac{z}{s} & 1 \end{vmatrix}$$

- **Матрицы поворота вокруг осей x, y, z на угол α**

- R_x

R_y

R_z

$$\begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha & \sin\alpha & 0 \\ 0 & -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

$$\begin{vmatrix} \cos\alpha & 0 & -\sin\alpha & 0 \\ 0 & 1 & 0 & 0 \\ \sin\alpha & 0 & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

$$\begin{vmatrix} \cos\alpha & \sin\alpha & 0 & 0 \\ -\sin\alpha & \cos\alpha & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$





- **Поворот тела вокруг точки (m, n, k) на угол a**
- Разобьем данную операцию на базовые (простейшие), а именно: 1) перенос тела на вектор A(-m, -n, -k) для совмещения точки (m, n, k) с началом координат; 2) поворот тела на угол a; 3) перенос тела на вектор A'(m, n, k) для возвращения его в исходное положение. Представим тело набором точек (вершин тела) и выполним операции 1) - 3) с каждой из них; в матричной форме это представляется следующим образом (R(a) — матрица поворота вокруг оси x, y или z):

$$\begin{vmatrix} x & y & z & 1 \end{vmatrix} * \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -m & -n & -k & 1 \end{vmatrix} * R(a) * \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ m & n & k & 1 \end{vmatrix}$$

- **Зеркалирование**
- **Замечание:** ниже представлены всего лишь три матрицы зеркалирования, на самом деле их больше. Предлагаем читателю самостоятельно построить недостающие матрицы.

- Зеркалирование относительно оси z:

$$\begin{vmatrix} 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix} \quad \begin{vmatrix} -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

- Зеркалирование относительно плоскости x = 0:

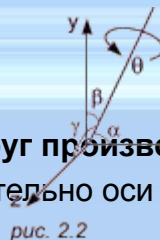
$$\begin{vmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{vmatrix}$$

- Зеркалирование относительно начала координат:

•

- **Вращение тела на угол q вокруг произвольной оси, проходящей через точку (0, 0, 0)**

- Здесь a — угол наклона относительно оси x, b — угол наклона относительно оси y, g — угол наклона относительно оси z.





Аффинное проецирование

- **Геометрия аффинная**
 - Аксонометрическая (прямоугольная) проекция
 - Ортографическая проекция
 - Ортогональная проекция
 - Диметрическая проекция
 - Изометрическая проекция
 - Косоугольная проекция
 - Проекция Кабине
 - Проекция Кавалье
- **Геометрия перспективная**
 - Перспективная проекция
 - Одноточечная проекция
 - Двухточечная проекция
 - Трехточечная проекция
- Аффинная геометрия является чертежным средством; в этой геометрии используется параллельное проецирование, которое осуществляется пучком параллельных прямых (см. рис. 3.1, слева). Детерминант матрицы преобразований в аффинной геометрии равен нулю. Геометрия перспективная является художественным средством, в ней отсутствуют параллельные линии и используется центральное проецирование, при котором все линии сходятся у горизонта в одну точку (см. рис. 3.1, справа); за счет того, что одна, две или три компоненты четвертого столбца матрицы преобразований не равны нулю, ее детерминант также не равен нулю.
- Наиболее распространены аффинные двумерные и трехмерные преобразования. Их основные геометрические свойства: прямые линии после преобразования остаются прямыми, параллельные — параллельными, плоскости остаются плоскостями и параллельные плоскости — параллельными. Вычерчивание трехмерных объектов, независимо от того на бумаге ли это происходит или на экране дисплея, осуществляется при помощи двумерных проекций. В плоской проекции каждая точка предмета проецируется определенным образом на плоскость проекции, и ее образ называется точкой проекции. Если линии проекции, соединяющие точки предмета с соответствующими точками проекции, параллельны, то мы имеем плоскую параллельную проекцию. Если же линии проекции сходятся в одной общей точке, то получаемое изображение называется центральной проекцией, или перспективным изображением предмета.
- Рассмотрим далее несколько видов аксонометрических проекций. Заметим, что среди них выделяются прямоугольные (ортогональные) проекции — те, у которых проецирующие лучи перпендикулярны к плоскости изображения. Слова «аксонометрическая» и «прямоугольная» часто используют как синонимы.



- **АксонOMETрическая ортографическая проекция**
- Смещает изображение по оси z на n единиц и проецирование происходит в плоскости $z = 0$:

$$\begin{bmatrix} x & y & z & 1 \end{bmatrix} * \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & n & 1 \end{bmatrix} = \begin{bmatrix} x & y & n & 1 \end{bmatrix}$$

- **АксонOMETрическая ортогональная проекция**
- Проецирующие лучи перпендикулярны к плоскости изображения, и плоскость проекции совпадает с одной из координатных плоскостей.
- Пример: выполним поворот куба (рис. 3.2, слева) относительно оси x на угол 90° и спроецируем его на плоскость $z = 0$ (рис. 3.2, справа). Обратите внимание, что точки, которые после поворота на куба на 90° будут лежать на проецирующих лучах (пары A и C , B и D и т. д.), после преобразования займут одинаковое положение (пары A'' и C'' , B'' и D'' и т. д.):

Запись в матричной форме:

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

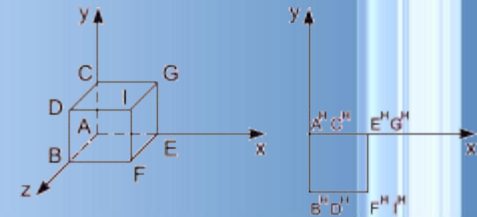


рис. 3.2

- **АксонOMETрическая диметрическая проекция**
- В этом случае были специально подобраны коэффициенты,
- дающие минимальные искажения какого-либо изображения:

$$\begin{bmatrix} 0.93 & 0.13 & -0.35 & 0 \\ 0 & 0.93 & 0.35 & 0 \\ 0.38 & -0.33 & 0.36 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 0.71 & 0.41 & -0.58 & 0 \\ 0 & 0.82 & 0.58 & 0 \\ 0.71 & -0.41 & 0.58 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

- **АксонOMETрическая изометрическая проекция**
- В общем смысле дает проекцию с изменением масштаба:

- Надо заметить, что любая проекция имеет свои достоинства и недостатки (искажение размеров изображения по одной из осей, неодинаковость сокращения размеров по разным осям, заслонения точек друг другом), поэтому каждая из них используется там, где это более удобно: ортографическая — только для переноса и



Перспективное проецирование

- На фотографиях, картинах, экране изображения кажутся нам естественными и правильными. Эти изображения называют перспективными. Свойства их таковы, что более удаленные предметы изображаются в меньших масштабах, параллельные прямые в общем случае не параллельны. В итоге геометрия изображения оказывается достаточно сложной, и по готовому изображению сложно определить размер тех или иных частей объекта.
- Обычная перспективная проекция — это центральная проекция на плоскость прямыми лучами, проходящими через точку — центр проецирования. Один из проецирующих лучей перпендикулярен к плоскости проецирования и называется главным. Точка пересечения этого луча и плоскости проекции — главная точка картины.
- Существует три системы координат. Обычно программист работает и держит данные о геометрических объектах в мировых координатах. Для повышения реалистичности при подготовке к выводу изображения на экран данные об объектах из мировых координат переводят в видовые координаты. И только в момент вывода изображения непосредственно на экран дисплея переходят к экранным координатам, которые представляют собой номера пикселей экрана.
- Первые две системы могут использоваться в многомерных системах координат, но последняя только в двухмерной. Операции являются необратимыми, то есть из двухмерной картинки-проекции невозможно восстановить трехмерное изображение.

- **Матрица общего перспективного преобразования**

$$\begin{vmatrix} a & x & x & p \\ x & d & x & q \\ x & x & e & r \\ m & n & L & s \end{vmatrix}$$

- В этой матрице элементы a, d, e отвечают за масштабирование, m, n, L — за смещение, p, q, r — за проецирование, s — за комплексное масштабирование, x — за вращение.
- **Одноточечное проецирование на плоскость $z = 0$**

$$\begin{vmatrix} x & y & z & 1 \end{vmatrix} * \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & r \\ 0 & 0 & 0 & 1 \end{vmatrix} = \begin{vmatrix} x & y & 0 & rz+1 \end{vmatrix} = \begin{vmatrix} \frac{x}{rz+1} & \frac{y}{rz+1} & 0 & 1 \end{vmatrix}$$

- Суть этого проецирования такова: чем глубже находится предмет, тем больше становится значение z -координаты и знаменателя $rz + 1$, и, следовательно, тем мельче выглядит предмет на плоскости проекции. Выполним несложные выкладки и поясним их графически:



- уравнение $x'/F = x/(F + z_{\text{пр}})$ равносильно: $x' = xF/(F + z_{\text{пр}}) = x/(1 + z_{\text{пр}}/F) = x/(1 + rz_{\text{пр}})$, где $r = 1/F$, F — фокус.

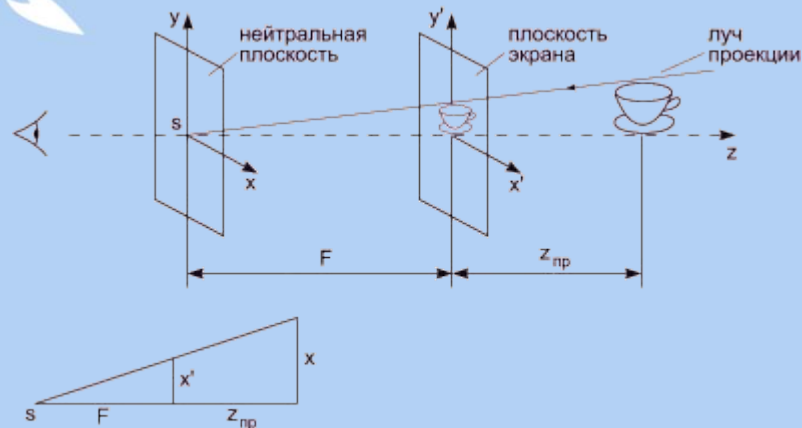


рис. 4.1

- Для того, чтобы точки, лежащие на линии, параллельной оси z , не терялись друг за другом, используется одноточечное проецирование на линию (см. матрицу преобразования и рис. 4.2); исчезла z -координата, но, поскольку дальние предметы стали более мелкими, чем такие же близкие, у зрителя появляется ощущение глубины. Запомните: это первый способ передачи глубины на плоскости!

- А теперь для примера отобразим куб, стороны которого равны единице (см. рис. 4.3), на плоскость $z = 0$; пусть на оси z фокус $F = -10$ и, следовательно, $r = -1/10 = -0.1$:

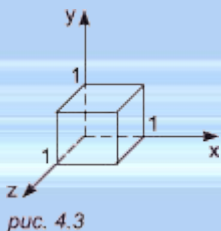


рис. 4.3

$$\begin{vmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 \end{vmatrix} * \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -0.1 \\ 0 & 0 & 0 & 1 \end{vmatrix} = \begin{vmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0.9 \\ 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0.9 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0.9 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0.9 \end{vmatrix} = \begin{vmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 1.11 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 1.11 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 \\ 1.11 & 1.11 & 0 & 1 \end{vmatrix}$$

$$\begin{vmatrix} x & y & z & 1 \end{vmatrix} * \begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & r \\ 0 & 0 & 0 & 1 \end{vmatrix} = \begin{vmatrix} x & y & z & 1 \\ rz+1 & rz+1 & rz+1 & 1 \end{vmatrix}$$

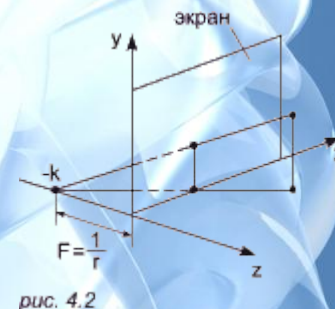


рис. 4.2



- В плоскости экрана после преобразования мы будем иметь следующее изображение:

- Обратите внимание: если бы центр тяжести куба находился в точке $(0, 0, 0)$, куб выглядел бы более «традиционно»:

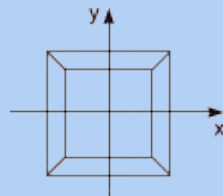


рис. 4.5



рис. 4.4

- Двухточечное проецирование**

- Если проекция двухточечная (например, по $p \neq 0$ и $q \neq 0$), то имеются две точки схода на соответствующие оси. Обратите внимание: так как по z в данном случае реализуется параллельное проецирование, то удвоения контура куба на экране (x, y) нет. Меняя p и q , мы регулируем точку схода:

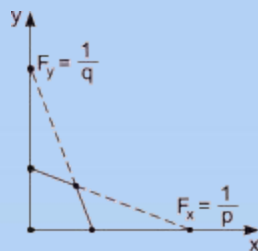
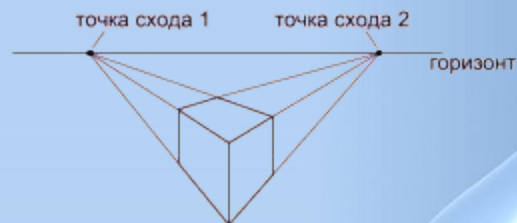


рис. 4.6



- Трехточечное проецирование по p, q, r**

- В данном случае $p \neq 0, q \neq 0, r \neq 0$, и проекция будет иметь следующий вид:

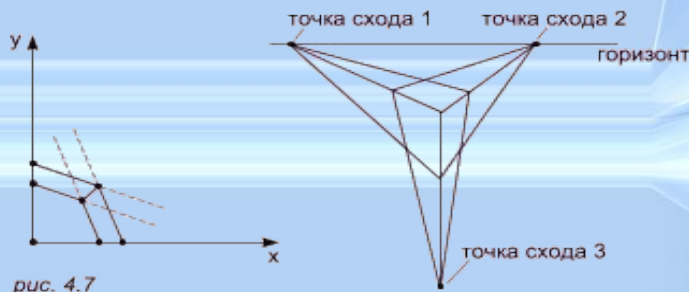
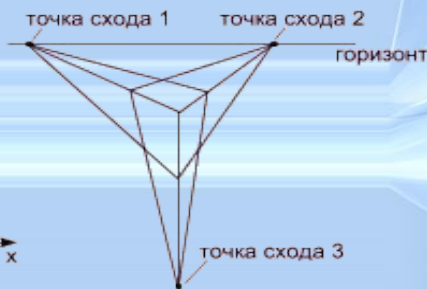
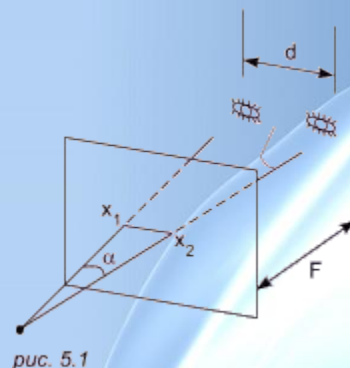


рис. 4.7



Стереографическая и специальные перспективные проекции

- Чтобы помочь человеку осознать, что на экране перед ним — именно трехмерное изображение (то есть восстановить третью координату из двумерного изображения), используются различные приемы: нанесение штриховки, окраска, дымка, уменьшение размеров с глубиной, тени, стерео... Рассмотрим подробнее последний способ.
- Разберемся, каким же образом мы определяем объем, глубину в изображении? Для этого надо понять, как и с помощью чего мы «чувствуем» третье измерение? Оказывается, человек чувствует глубину по напряжению аккомодационной мышцы глаза. Чем ближе к глазу находится объект, тем сильнее напрягается эта мышца и тем сильнее она выгибает глазной хрусталик. Таким образом, очень сильное напряжение аккомодационной мышцы сообщает мозгу о том, что предмет находится очень близко.
- Так как глаза человека находятся на некотором расстоянии друг от друга, один и тот же предмет они видят немного по-разному. На рис. 5.1 точка x_1 — это изображение предмета на плоскости проекции для правого глаза, x_2 — для левого, F — расстояние от глаз до плоскости проекции (около 50 см.), d — расстояние между глазами (около 5 см.), $\alpha = \arctg(d/F)$ — стереоугол (около 5.71°):
- Таким образом, чтобы создать для глаз эффект стерео, необходимо сформировать два изображения — для левого и правого глаза. Ниже даны соответствующие матрицы:



- левый глаз правый глаз
- | | |
|--|---|
| $\begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1/F \\ F/20 & 0 & 0 & 1 \end{vmatrix}$ | $\begin{vmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1/F \\ -F/20 & 0 & 0 & 1 \end{vmatrix}$ |
|--|---|
- Обратим внимание на некоторые элементы матриц. Значение $-1/F$ говорит о том, что проецирование идет по оси z ; третий столбец матриц состоит из одних нулей — это указывает на то, что проецирование идет на плоскость $z = 0$; $F/20 = d/2$ и $-F/20 = -d/2$ задают смещение по оси x для левого и правого глаза соответственно.
- Существуют различные способы рассматривания полученных изображений. Первый способ применяется в виртуальных шлемах: изображение с помощью крохотных жидкокристаллических дисплеев подается отдельно на каждый глаз. Другой способ — анаглифический. В этом случае картинка для левого глаза окрашивается в синий цвет, а для правого — в красный; зритель надевает специальные фильтры-очки с синими и красными стеклами — и одним глазом видит только первое изображение, а другим — только второе. Мозг наблюдающего воспримет такую картину как единое объемное стереоизображение.



- **Специальная перспективная проекция на сферу («рыбий глаз»)**
- В отличие от классической перспективы, в перспективной проекции на сферу каждую часть изображения лучше рассматривать перпендикулярно к сфере, то есть при просмотре различных частей проекции голова наблюдателя или рисунок перемещаются. Такое рассматривание можно назвать фрагментарным.

- $$z_a = z_{пр} * F$$
$$C = 2F/[z_a + \sqrt{x_{пр}^2 + y_{пр}^2 + z_a^2}]$$
$$x' = x_{пр} * C$$
$$y' = y_{пр} * C$$

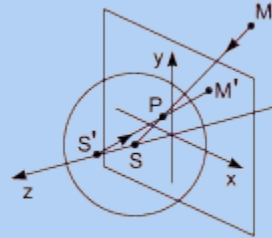


рис. 5.2

- Взгляните на рис. 5.2. Луч, выходящий из точки M, есть луч проецирования на сферу в точке P. Точка S является центром проекции. Луч, выходящий из точки S' и проходящий через P и M', — это луч перепроецирования со сферы на плоскость.
- **Специальная перспективная проекция на цилиндрическую поверхность**
- Проекция на цилиндрическую поверхность позволяет показывать объекты с очень большими углами зрения по горизонтали, вплоть до круговой панорамы. Вычерчиваются проекции в виде разверток на обычных графических устройствах.

- $$z'_{пр} = z_{пр} + F$$
$$y'/y = (F + z_{пр})/F, y = y'F/(F + z_{пр}) = y'F/z'_{пр}$$
$$x' = F * \arctg(x_{пр}/z_a)$$
$$y' = F * y/\sqrt{x_{пр}^2 + z_{пр}^2}$$
- На рис 5.3 S — центр проекции (точка зрения), z — главная ось.

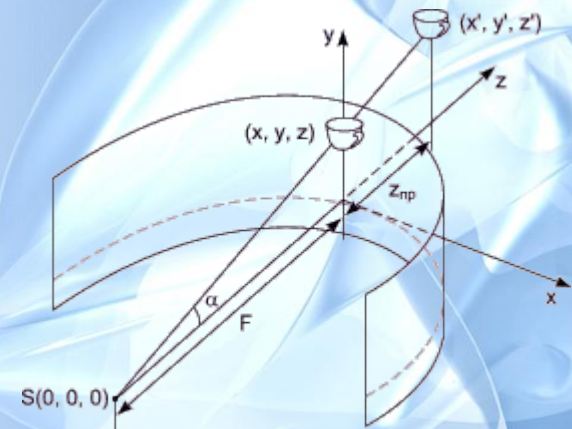
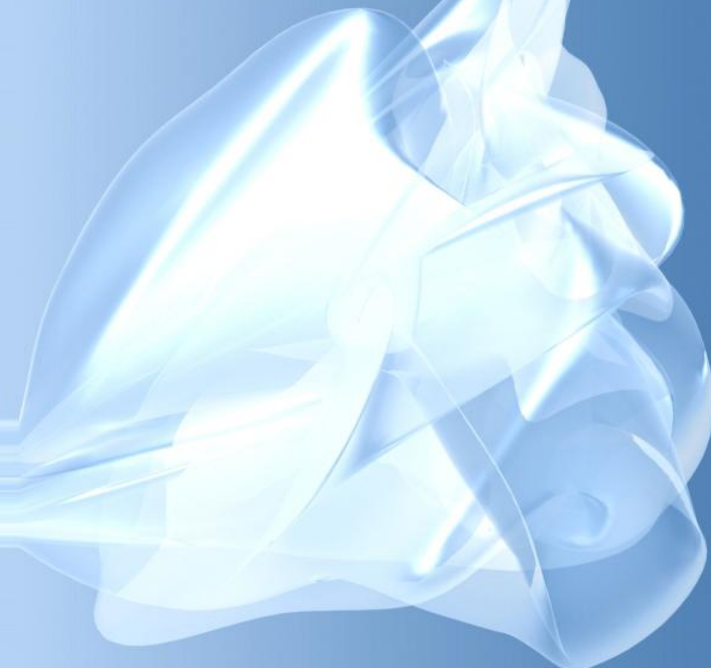


рис. 5.3



Масштабирование в окне

- **Масштабирование**
- Для того, чтобы выполнить масштабирование, необходимо:
- задать или определить координаты размера изображения: x_{\min} , y_{\min} , x_{\max} , y_{\max} и координаты области вывода: X_{\min} , Y_{\min} , X_{\max} , Y_{\max} ;
- вычислить коэффициенты масштабирования:
$$f_x = (X_{\max} - X_{\min}) / (x_{\max} - x_{\min}),$$
$$f_y = (Y_{\max} - Y_{\min}) / (y_{\max} - y_{\min}),$$
$$f_x = f_y = \min(f_x, f_y)$$
- произвести вычисление текущих координат экрана:
$$x_э = X_{\min} + f_x * (x_p - x_{\min}),$$
$$y_э = Y_{\min} + f_y * (y_p - y_{\min}),$$
где $x_э$, $y_э$ — экранные координаты, x_p , y_p — реальные координаты.





Нахождение параметров плоскости

- **Нахождение плоскости по точкам**

- Пусть уравнение $ax + by + cz + d = 0$ описывает плоскость. Переменная d является свободной, ей можно придать любое значение, положим ее равной единице: $d = 1$. Известно, что плоскость задается, причем однозначным образом, тремя точками (x_1, y_1, z_1) , (x_2, y_2, z_2) , (x_3, y_3, z_3) пространства (эти точки, вообще говоря, не должны одновременно находиться на одной прямой):

$$x_1 a + y_1 b + z_1 c = -1$$

$$x_2 a + y_2 b + z_2 c = -1$$

$$x_3 a + y_3 b + z_3 c = -1$$

- Решив эту систему уравнений относительно неизвестных a , b и c , мы найдем уравнение плоскости. Покажем это на примере. Найдем уравнение плоскости, проходящей через три точки: $A(1, 0, 0)$, $B(0, 1, 0)$, $C(1, -1, 1)$. Рис. 7.1 иллюстрирует нашу задачу:

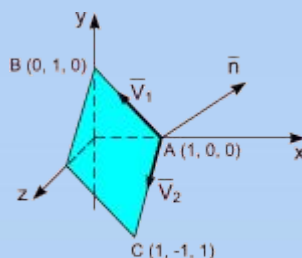


рис. 7.1

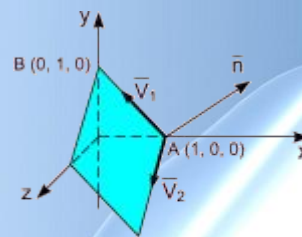


рис. 7.1

- Представим систему уравнений в виде произведения матриц: первая из них содержит координаты точек A , B и C , вторая — неизвестные a , b и c , третья — правые части уравнений:

$$\begin{vmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & -1 & 1 \end{vmatrix} \cdot \begin{vmatrix} a \\ b \\ c \end{vmatrix} = \begin{vmatrix} -1 \\ -1 \\ -1 \end{vmatrix}$$

- Отсюда находим $a = -1$, $b = -1$, $c = -1$ и подставляем их в уравнение плоскости: $-1x - 1y - 1z + d = 0$ или $x + y + z = d$. Подставим в это уравнение плоскости любую точку, например, $(1, 0, 0)$ и найдем d : $d = 1$. Окончательно уравнение данной плоскости выглядит так: $x + y + z - 1 = 0$.

- Представим систему уравнений в виде произведения матриц: первая из них содержит координаты точек A , B и C , вторая — неизвестные a , b и c , третья — правые части уравнений:
- Отсюда находим $a = -1$, $b = -1$, $c = -1$ и подставляем их в уравнение
- плоскости: $-1x - 1y - 1z + d = 0$ или $x + y + z = d$. Подставим в это
- уравнение плоскости любую точку, например, $(1, 0, 0)$ и найдем d : $d = 1$. Окончательно уравнение данной



- **Метод определения плоскости по нормали**

- Если имеется возможность определить n — вектор нормали к плоскости, — то можно найти и уравнение этой плоскости; n находится как векторное произведение векторов $V_1\{x_1, y_1, z_1\}$ и $V_2\{x_2, y_2, z_2\}$, лежащих в определяемой плоскости. В координатной форме это запишется следующим образом (значок \times обозначает векторное произведение векторов; i, j, k — единичные векторы осей x, y и z):

- Для простого примера, приведенного выше, мы будем иметь следующее:

$$V_1\{0 - 1, 1 - 0, 0 - 0\} = V_1\{-1, 1, 0\},$$

$$V_2\{1 - 1, -1 - 0, 1 - 0\} = V_2\{0, -1, 1\},$$

- Вычисленные коэффициенты при i, j, k подставляем в уравнение плоскости вместо a, b и c : $1x + 1y + 1z + d = 0$. Взяв произвольную точку плоскости и подставив ее координаты вместо x, y, z , найдем d : $C(1, -1, 1)$, $d = -1 * 1 - 1 * (-1) - 1 * 1 = -1$. Итак, окончательное уравнение плоскости примет следующий вид: $x + y + z - 1 = 0$.

- **Метод Ньюэла**

- Этот метод эквивалентен определению нормали в каждой вершине многоугольника посредством векторного произведения прилежащих ребер и усреднения результатов.

- Пусть a, b, c, d — коэффициенты уравнения плоскости, тогда:

$$a = S(y_i - y_j) * (z_i + z_j)$$

$$b = S(z_i - z_j) * (x_i + x_j)$$

$$c = S(x_i - x_j) * (y_i + y_j)$$

$$d = -(ax_n + by_n + cz_n).$$

- Суммирование под знаком суммы ведется по i , которое всюду изменяется от 1 до n , n — число вершин многоугольника. Переменная j принимает значение 1 при $i = n$, в противном случае $j = i + 1$. Между точками находим среднюю плоскость, проходящую через все точки. В методе происходит как бы вычисление «центра масс».

- Попробуем найти коэффициенты для плоскости, изображенной на рис. 7.1:

$$a = (y_1 - y_2)(z_1 + z_2) + (y_2 - y_3)(z_2 + z_3) + (y_3 - y_4)(z_3 + z_4) + (y_4 - y_1)(z_4 + z_1) = \\ = (0 - 1)(0 + 0) + (1 - 0)(0 + 1) + (0 - (-1))(1 + 1) + (-1 - 0)(1 + 0) = 2,$$

$$b = \dots = 2,$$

$$c = \dots = 2,$$

$$d = -(2x_4 + 2y_4 + 2z_4) = -(2 - 2 + 2) = -2.$$

Получаем плоскость $x + y + z - 1 = 0$. Тот же самый результат будет, если вычислить нормали каждой из четырех