2.5. ПРЕОБРАЗОВАНИЯ НА ПЛОСКОСТИ

При построении изображений часто приходится иметь дело с ситуациями, когда общее изображение (рисунок) включает в себя целый ряд компонент (подрисунков), отличающихся друг от друга только местоположением, ориентацией, масштабом, т.е. отдельные подрисунки обладают значительным геометрическим сходством.

В этом случае целесообразно описать один подрисунок в качестве базового, а затем получать остальные требуемые подрисунки путем использования операций преобразования.

С помощью операций преобразования можно выполнять следующие действия: 1) перемещать рисунки из одного места экрана в другое; 2) создавать рисунок из более мелких элементов (составных частей); 3) добавлять к существующему рисунку новые элементы; 4) увеличивать размер рисунка для улучшения его наглядности или отображения более мелких деталей; 5) уменьшать размер рисунка для внесения, например, поясняющих надписей или
отображения на экране новых рисунков; 6) создавать движущиеся изображения.

Все изменения рисунков можно выполнить с помощью трех базовых операций: 1) переноса (перемещения) изображения; 2)масштабирования (увеличения или уменьшения размеров) изображения; 3) поворота изображения (употребляют также термины вращение, изменение ориентации).

Для реализации перечисленных операций используется аппарат линейных преобразований, хорошо разработанный в математике и просто реализуемый программно.

2.5.1 ЛИНЕЙНОЕ ПРЕОБРАЗОВАНИЕ

Линейное преобразование на плоскости - это такое отображение плоскости в себя, при котором прямая переходит в прямую.

Произвольная точка с координатами (X,Y) переходит в результате линейного преобразования в точку с координатами (X1,Y1) в соответствии с выражениями:

$$X1 = A*X + B*Y + C$$
$$Y1 = D*X + E*Y + F, \qquad (2.5.1)$$

где A,B,C,D,E,F - коэффициенты данного преобразования, однозначно его определяющие.

Последовательное применение двух линейных преобразований можно заменить одним третьим, эквивалентным, которое называется их произведением.

Формулы линейного преобразования (2.5.1) можно записать в матричной форме:

$$(X1, Y1, 1) = (X1, Y1, 1) * \begin{pmatrix} A & D & 0 \\ B & E & 0 \\ C & F & 1 \end{pmatrix} = (X, Y, 1) * \mathrm{M} \qquad (2.5.2)$$

где через M обозначается матрица преобразования.

Для двух последовательно выполняемых линейных преобразований можно записать следующее выражение:

$$(X2,Y2,1)=(X1,Y1,1)*M2=(X,Y,1)*M1*M2=(X,Y,1)*M \qquad (2.5.3)$$

где X,Y - координаты исходной точки;

X1,Y1 - координаты точки после первого преобразования;

X2,Y2 - координаты точки после второго преобразования;

M1,M2,M - матрицы соответственно первого, второго и результирующего преобразований.

Поскольку в общем случае операция умножения матриц не является коммутативной, то в общем случае и два последовательных линейных преобразования некоммутативны.

Если определитель матрицы преобразования отличен от нуля, то такое преобразование будет являться аффинным. При аффинном преобразовании плоскость не может вырождаться в линию или точку, параллельные прямые переводятся в параллельные и всегда имеется обратное преобразование. На практике, как правило, имеем дело с аффинными преобразованиями; такое преобразование может быть представлено суперпозицией трех преобразований: переноса, масштабирования, поворота.

2.5.2 ПЕРЕНОС ИЗОБРАЖЕНИЯ

Перенос изображения заключается в перемещении отображенного объекта из одного места экрана в другое место. Перенос изображения позволяет построить рисунок в произвольном месте экрана и затем перенести его в другую, требуемую, часть экрана. При этом можно изменить компоновку рисунка или создать единый рисунок из набора готовых элементов.

Для переноса точки из позиции с координатами (X,Y) в позицию с координатами (X1,Y1) надо к координате X добавить DX, а к координате Y - DY единиц, причем DX = X1-X, DY = Y1-Y.

Матрица преобразования M для операции переноса имеет следующий вид:

$$M=\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ DX & DY & 1 \end{pmatrix} \qquad (2.5.4)$$

Подставляя ее в (2.5.2), получим

$$\begin{pmatrix} X1 & Y1 & 1 \end{pmatrix} = \begin{pmatrix} X & Y & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ DX & DY & 1 \end{pmatrix}$$

или X1 = X+DX, Y1 = Y+DY.

Положительное значение DX означает перемещение точки вправо по горизонтали, отрицательное - влево; положительное значение DY - перемещение вниз по вертикали, отрицательное - вверх.

Необходимо задавать такие значения DX, DY, чтобы после преобразования точка оставалась в пределах экрана, иначе она высвечиваться не будет. Не следует задавать слишком малые значения DX, DY (DX < 0,5, DY < 0,5), так как в этом случае точка повторно высвечивается на старом месте.

Перенос рисунка из одной области экрана в другую эквивалентен переносу всех точек рисунка и последующему высвечиванию соединяющих их линий. Для исключения искажения рисунка все точки необходимо переместить на одинаковое расстояние, т.е. использовать одну и ту же матрицу преобразования.

Для фигур, обладающих симметрией, или с границами, вычисляемыми с помощью уравнений, при переносе изображения нет необходимости добавлять смещение к координатам всех точек. Например, для переноса окружности достаточно перенести ее центр и вычертить ее, зная радиус. Таким же образом осуществляется перенос эллипса.

При перемещении текстовой информации величина перемещения иногда задается в единицах позиций печати, а не в количестве точек растра.

В этом случае величина горизонтального перемещения текста (заданная позициях печати) определяется путем деления величины DX (заданной количеством точек растра) на число элементов растра, приходящихся на один символ по горизонтали. Величина вертикального переноса - делением величины DY на число элементов растра, приходящихся на один символ по вертикали. Если од-
ному символу соответствует прямоугольник размером 8x8 точек, то номер строки, в которой располагается текст, изменяется на величину DY/8, а номер позиции в строке - на DX/8, если же одному символу соответствует прямоугольник 8x12, то интересующие смещения будут равны DX/8 и DY/12.

В заключение заметим, что операцию переноса удобно применять при построении графиков, когда на экране независимо друг от друга надо разместить несколько графиков.

2.5.3 МАСШТАБИРОВАНИЕ ИЗОБРАЖЕНИЯ

При создании изображения на экране дисплея может возникнуть необходимость изменения его размеров с целью повышения его наглядности, для вставки созданного изображения в уже имеющийся рисунок. Размер рисунка можно изменить, если умножить все расстояния между точками на некоторую постоянную величину (коэффициент масштабирования). Если коэффициент масштабирования больше единицы, то рисунок увеличивается, если меньше единицы - уменьшается.

Наряду с коэффициентом масштабирования для выполнения масштабирования надо указать новое положение рисунка (после выполнения масштабирования). Новое положение рисунка определяется центром масштабирования - некоторой центральной точкой, относительно которой выполняется масштабирование.

В качестве такой точки может быть выбрана центральная точка рисунка, точка, лежащая на границе рисунка, точка, лежащая вне рисунка и даже вне экрана.

Масштабирование может быть однородным (коэффициенты масштабирования вдоль осей абсцисс и ординат одинаковы), в этом случае пропорции рисунка сохраняются. Масштабирование может быть и неоднородным (коэффициенты вдоль осей абсцисс и ординат различны), в этом случае пропорции рисунка изменяются.

Неоднородное масштабирование может применяться при подборе пропорций рисунка, в процессе конструирования и проектирования при подборе размеров объекта.

Матрица преобразования при масштабировании имеет вид

$$M=\begin{pmatrix} KX & 0 & 0 \\ 0 & KY & 0 \\ 0 & 0 & 1 \end{pmatrix} \qquad (2.5.5)$$

Подставляя ее в (2.5.2), получим:

$$\begin{pmatrix} X1 & Y1 & 1 \end{pmatrix} = \begin{pmatrix} X & Y & 1 \end{pmatrix} \begin{pmatrix} KX & 0 & 0 \\ 0 & KY & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

или X1 = X*KX, Y1 = Y*KY,

где KX - коэффициент масштабирования по оси абсцисс;

KY - коэффициент масштабирования по оси ординат.

Применяя преобразование (2.5.5) ко всем точкам рисунка, получим рисунок, промасштабированный относительно начала координат. При этом, если KX > 1 и KY > 1, то рисунок увеличивается в размере и удаляется от начала координат; если KX < 1 и KY< 1, то рисунок уменьшается в размерах и приближается к началу координат.

МАСШТАБИРОВАНИЕ ОТРЕЗКА

Рассмотрим теперь различные варианты масштабирования отрезка как основного элемента при создании изображений.

Пусть отрезок задан координатами своих концов ($X_н$,$Y_н$) и ($X_к$,$Y_к$). Проведем масштабирование относительно одного из его концов, например, относительно точки ($X_н$,$Y_н$). В этом случае положение этой точки останется неизменным, а длина отрезка должна измениться в K раз, где K - коэффициент масштабирования (имеется в виду однородное масштабирование).

Длина исходного отрезка

$$L=\sqrt{(X_к - X_н)^2 + (Y_к - Y_н)^2}$$

Длина промасштабированного отрезка должна быть

$$L1=K*L=K\sqrt{(X_к - X_н)^2 + (Y_к - Y_н)^2} = \sqrt{(K*dX)^2 + (K*dY)^2}$$

где $dX=X_к-X_н$, $dY=Y_к-Y_н$ - величины проекций отрезка на координатные оси.

При фиксированных координатах $X_н$, $Y_н$ вычислим новые координаты другого конца отрезка:

$X_н' - X_н = K*dX = K*(X_к-X_н)$,

$X_к' = X_н + K*(X_к-X_н)$.

Аналогично $Y_к' = Y_н + K*(Y_к-Y_н)$.

Новые координаты точек в обозначении имеют знак штрих.

При неподвижном другом конце отрезка:

$X_к – X_н' = K*(X_к – X_н)$, $X_н' = X_к + K*(X_н – X_к)$,

$Y_к – Y_н' = K*(X_к – X_н)$, $Y_н' = Y_к + K*(Y_н – Y_к)$.

Таким образом, при определении новых координат конца отрезка к координате неподвижного конца надо прибавить проекцию исходного отрезка, увеличенного в K раз. При определении проекции отрезка на первом месте должна стоять координата конца, изменяющего свое положение.

Рассмотрим масштабирование отрезка относительно произвольной точки. Пусть концы отрезка имеют те же координаты ($X_н$,$Y_н$) и ($X_к$,$Y_к$), центр масштабирования - координаты (X_c,Y_c), а коэффициенты масштабирования вдоль координатных осей - kx, ky.

Как видно из предыдущего, отрезок может быть промасштабирован при одновременном масштабировании его горизонтальных и вертикальных проекций.

При масштабировании относительно произвольной точки эта точка остается неподвижной, а изменяются расстояния от концов отрезка до этой точки. Поэтому получим:

$$
\begin{aligned}
X_н' &= X_c + kx*(X_н - X_c) = X_н*kx + (1-kx)*X_c \\
Y_н' &= Y_c + ky*(Y_н - Y_c) = Y_н*ky + (1-ky)*Y_c \qquad (2.5.6) \\
X_к' &= X_c + kx*(X_к - X_c) = X_к*kx + (1-kx)*X_c \\
Y_к' &= Y_c + kx*(Y_к - Y_c) = Y_к*kx + (1-ky)*Y_c
\end{aligned}
$$

Вторые слагаемые в полученных выражениях являются константами, поэтому в программе их достаточно вычислить один раз.

Если угол наклона масштабируемой прямой меняться не должен, то коэффициенты масштабирования вдоль координатных осей должны совпадать, т.е. kx = ky.

Часто приходится масштабировать отрезок относительно его середины, в этом случае координаты центра масштабирования будут иметь следующие значения:

$$X_c = \frac{X_н + X_к}{2} \quad , \quad Y_c = \frac{Y_н + Y_к}{2}$$

Подставив эти значения в (2.5.6), получим:

$$X'_н = \frac{X_н}{2}*(1+kx) + \frac{X_к}{2}*(1-kx)$$

$$Y'_н = \frac{Y_н}{2}*(1+ky) + \frac{Y_к}{2}*(1-ky)$$

$$X'_к = \frac{X_к}{2}*(1+kx) + \frac{X_н}{2}*(1-kx)$$

$$Y'_к = \frac{Y_к}{2}*(1+ky) + \frac{Y_н}{2}*(1-ky)$$

МАСШТАБИРОВАНИЕ ИЗОБРАЖЕНИЙ

Для масштабирования рисунка необходимо в соответствии с (2.5.5) вычислить новые координаты всех точек нового изображения, а затем полученные точки соединить линиями. При этом размеры рисунка равномерно увеличиваются или уменьшаются, если kx = ky.

Однако не всегда надо вычислять координаты всех точек нового рисунка. Например, при масштабировании окружности, достаточно вычислить новые координаты ее центра, а в качестве радиуса взять величину k*R (k - коэффициент масштабирования, R - радиус исходной окружности). Таким же образом можно поступить при масштабировании эллипсов, прямоугольников.

При вычислении координат точек нового рисунка следует иметь в виду, что коэффициент масштабирования, как правило, величина действительная, а координаты точек на экране должны быть целыми, поэтому в программе необходимо использовать в этом случае операцию округления.

С помощью масштабирования можно растянуть или сжать изображение вдоль одной координатной оси, оставив его без изменения вдоль другой оси. Например, масштабируя квадрат с коэффициентами масштабирования kx = 1, ky =2, получим прямоугольник, у которого большая сторона имеет вертикальное расположение.

Неравномерное масштабирование окружности приводит к тому, что будет изображен эллипс. Но в этом случае программист должен воспользоваться процедурой рисования эллипса, а не окружности. В такой ситуации в целях общности целесообразно далее окружности вычерчивать процедурой рисования эллипса (задавая равные значения полуосей).

2.5.4 ПОВОРОТ ИЗОБРАЖЕНИЯ

Наиболее часто поворот изображения используется при создании движущихся изображений в моделирующих, игровых программах. Однако, иногда бывает удобно повернуть на 90° график, гистограмму. В процессе проектирования также необходимо поворачивать изображение создаваемого объекта, чтобы рассмотреть его с разных сторон и избежать возможных ошибок.

ПОВОРОТ ТОЧКИ.

Для выполнения поворота надо указать величину угла, на который необходимо осуществить поворот, и координаты точки, которая берется за центр вращения. Пусть исходную точку А с координатами (X,Y) по дуге окружности с центром в точке С с координатами (X_c, Y_c) поворачивают на угол t. При этом точка занимает новое положение В с координатами X1,Y1. Имея все необходимые исходные данные, требуется определить значения X1,Y1.

Пользуясь рисунком 2.5.1, вычислим координаты X1,Y1:

$$X1 = X+R*cosf+R*cos(180^{\circ} -(f+t)) = X+(X_c-X)-R*cos(f+t) = X + (X_c -X) - R*cosf*cost + R*sinfsint = X_c - (X_c -X)*cost + (Y - Y_c)*sint = X_c + (X - X_c)*cost+(Y - Y_c)*sint; \qquad (2.5.7)$$

$$Y1 = Y - R*sinf+R*sin(180^{\circ} -(f+t)) = Y - (Y - Y_c) + R*sinfcost + R*cosfsint = Y_c+(Y - Y_c)*cost-(X - X_c)*sint;$$

Если центр поворота расположен в начале координат (X_c=0,Y_c=0), то матрица преобразования имеет вид:

$$M=\begin{pmatrix} \cos t & -\sin t & 0 \\ \sin t & \cos t & 0 \\ 0 & 0 & 1 \end{pmatrix} \qquad (2.5.8)$$

Если начало координат расположено в левой верхней точке экрана, то угол поворота измеряется в направлении против часовой стрелки. Если же начало координат лежит в левой нижней точке экрана, то угол поворота должен измеряться в направлении по часовой стрелке.

Центр поворота может быть расположен в любом месте экрана, а также за пределами его границ.

Угол поворота лежит обычно в пределах от 0° до 360°, другие углы поворота также допустимы, однако поворот при этих углах эквивалентен повороту при углах из указанного диапазона.

ПОВОРОТ РИСУНКА

Для поворота всего рисунка прежде всего необходимо выбрать центр поворота и определить угол, на который необходимо повернуть исходное изображение. После этого в соответствии с (2.5.8) вычислить координаты всех точек рисунка и соединить вновь полученные точки линиями.

При вычислении координат следует иметь в виду, что для хранения новых координат точек нельзя использовать ту же область памяти, где хранятся исходные координаты. Это объясняется тем, что при вычислении координаты Y1 в выражении (2.5.7) используется значение исходной координаты X. Таким образом, используя одну область памяти, мы будем затирать нужную нам старую координату X и неправильно вычислять новое значение координаты Y1.

Следует обратить внимание на то, что результат вычисления значений функций синуса и косинуса - действительный, а координаты точек должны быть целыми, поэтому в программе необходимо использовать операцию округления.

Преобразованное изображение (повернутое, промасштабированное или перенесенное) может высвечиваться вместе с исходным. Если же это нежелательно, то перед рисованием нового изображения необходимо произвести очистку экрана.

При выполнении преобразований желательно заранее предвидеть расположение рисунка, т.к. точки, выходящие за границы экрана, не высвечиваются. Если в программе введен контроль значений координат точек, то он только может обеспечить поиск ошибки, но программу в этом случае все равно придется корректировать.

При выполнении поворота могут возникать искажения рисунков, если разрешающая способность графической системы неодинакова вдоль координатных осей.

С целью исключения искажений в расчетные формулы (2.5.7) надо ввести поправочные множители, равные отношению разрешающих способностей. Тогда (2.5.7) могут быть записаны в следующем виде:

$$X1 = X_c + (X-X_c)*\cos t+ (Y-Y_c)*\sin t*rx/ry$$

$$Y1 = Y_c + (Y-Y_c)*\cos t+ (X-X_c)*\sin t*ry/rx, \qquad (2.5.9)$$

где rx, ry - разрешающие способности вдоль оси X и Y соответственно.

Отношение rx/ry может быть получено при использовании графической библиотеки Graph с помощью процедуры GetAspectRatio, оно является обратной величиной к отношению коэффициентов xa, ya, возвращаемых процедурой: rx/ry =ya/xa.

Поясним введение коэффициентов. Пусть имеется вертикальный отрезок длиной L (расстояние задано в точках растра). Его длина в единицах измерения длины составит L/ry. При повороте на 90° этот отрезок также будет иметь длину L (в точках растра), но в единицах измерения длины это составит L/rx. Чтобы линейные размеры не изменялись при повороте, надо, чтобы отрезок сохранял свою длину L/ry. Эта величина, выраженная в точках растра, составит L*rx/ry. Аналогично рассуждая для случая поворота горизонтального отрезка, получим коэффициент ry/rx.

Еще раз отметим, что при вращении прямоугольников, эллипсов нельзя использовать соответствующие процедуры библиотеки Graph, т.к. они позволяют рисовать эти фигуры частного положения.

Нижеприведенная программа иллюстрирует поворот эллипса. Для этого, во-первых, необходимо определить координаты точек, принадлежащих исходному эллипсу, затем определить координаты точек уже повернутого эллипса и соединить их отрезками прямых (рис.2.5.2).

Однако не всегда требуется расчет новых координат всех точек изображения: для прямоугольника это только четыре точки - его вершины, для окружности необходимо рассчитать только координаты центра и вычертить ее на новом месте.

Программа Elip строит исходный эллипс с осями, параллельными координатным осям, а затем строит повернутый на некоторый угол против часовой стрелки новый эллипс. Центр нового эллипса располагается ниже, в качестве центра вращения взят центр эллипса.

2.5.5 КОМПОЗИЦИЯ ПРЕОБРАЗОВАНИЙ.

Композиция преобразований представляет собой совокупность последовательного выполнения базовых преобразований: переноса, масштабирования, поворота. Зная матрицы преобразований, можно заранее вычислить результирующую матрицу преобразований как произведение отдельных матриц, и использовать уже полученную матрицу для определения итоговых координат точек рисунка, не вычисляя промежуточных координат.

Однако операция умножения матриц достаточно трудоемка, поэтому желательно знать такие варианты преобразований, когда можно не использовать перемножение матриц.

Покажем, что последовательные переносы являются аддитивными:

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ DX1 & DY1 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ DX2 & DY2 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ DX1+DX2 & DY1+DY2 & 1 \end{pmatrix} \qquad (2.5.10)$$

Последовательные масштабирования являются мультипликативными:

$$\begin{pmatrix} KX1 & 0 & 0 \\ 0 & KY1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} KX2 & 0 & 0 \\ 0 & KY2 & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} KX1*KX2 & 0 & 0 \\ 0 & KY1*KY2 & 0 \\ 0 & 0 & 1 \end{pmatrix} \qquad (2.5.11)$$

Последовательные повороты являются аддитивными:

$$\begin{pmatrix} \cos t & -\sin t & 0 \\ \sin t & \cos t & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos f & -\sin f & 0 \\ \sin f & \cos f & 0 \\ 0 & 0 & 1 \end{pmatrix} =$$

$$= \begin{pmatrix} \cos t*\cos f - \sin t*\sin f & -\cos t*\sin f - \sin t*\cos f & 0 \\ \sin t*\cos f + \cos t*\sin f & -\sin t*\sin f + \cos t*\cos f & 0 \\ 0 & 0 & 1 \end{pmatrix} =$$

$$= \begin{pmatrix} \cos(t+f) & -\sin(t+f) & 0 \\ \sin(t+f) & \cos(t+f) & 0 \\ 0 & 0 & 1 \end{pmatrix} \qquad (2.5.12)$$

Как уже отмечалось, в общем случае ряд последовательно выполняемых преобразований не обладает свойством коммутативности. Однако в ряде частных случаев наблюдается коммутативность. В таблице 2.5.1 приведены варианты таких преобразований:

Таблица 2.5.1

M1	M2

Перенос	Перенос
Масштабирование	Масштабирование
Поворот	Поворот
Масштабирование (однородное)	Поворот

Чтобы убедиться в коммутативности первых трех вариантов, достаточно выполнить перемножение матриц в выражениях (2.5.10), (2.5.11) и (2.5.12) справа налево и результат будет тот же. Рассмотрим четвертый вариант:

$$\begin{pmatrix} KX & 0 & 0 \\ 0 & KY & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos t & -\sin t & 0 \\ \sin t & \cos t & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} KX * \cos t & -KX * \sin t & 0 \\ KY * \sin t & KY * \cos t & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$\begin{pmatrix} \cos t & -\sin t & 0 \\ \sin t & \cos t & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} KX & 0 & 0 \\ 0 & KY & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} KX * \cos t & -KY * \sin t & 0 \\ KX * \sin t & KY * \cos t & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Если же KX≠KY, то коммутативность нарушается.

В заключение остается ответить на вопрос: что эффективнее при выполнении композиции преобразований: иметь результирующую матрицу преобразований и на нее умножать вектор-строку для каждой точки рисунка или не вычислять результирующую матрицу, а для каждой точки выполнять перемножение матриц.

Пусть рисунок состоит из M точек, выполняется всего L преобразований. Матрица преобразований имеет размерность NxN (для преобразований на плоскости N=3).

Трудоемкость операции оценим количеством операций умножения, поскольку операция сложения на порядок менее трудоемка.

Для перемножения двух матриц потребуется N^3 умножений. Для нахождения результирующей матрицы преобразований потребуется $(L-1)*N^3$ умножений. Затем вектор-строка для каждой точки умножается на результирующую матрицу, это потребует $M*N^2$ вычислений, всего получим:

$N1 = M*N^2 + (L-1)*N^3$.

Во втором случае надо для каждой точки всего выполнить $L*N^2$ умножений, общее количество вычислений составит $N2 = M*L*N^2$. Подставляя N=3, получим N1=9*M+27*(L-1), N2=9*M*L.

Определим в зависимости от L наиболее целесообразный вариант:

$9*M+27-27*L \le 9*M*L \quad M*L \ge M+3*L-3$

$$L \ge \frac{M-3}{M-3} = 1$$

Таким образом, всегда целесообразно заранее получить результирующую матрицу преобразований, если число точек рисунка больше трех.

Если учесть структуру матрицы преобразований, то можно заметить, что перемножение двух матриц требует 12 умножений и 8 сложений, а умножение вектора-строки на матрицу - 4 умножений и 4 сложений. В этом случае

N1 = (4у + 4сл)*M + (L-1)*(12у + 8сл)

N2 = M*L*(4у + 4сл)

Однако результат от этого не изменяется.