Физическая модель гладкой трансформации контуров

Денис Сергеевич Буренков
Физический факультет МГУ им. М.В.Ломоносова
Павел Васильевич Храпов
МГТУ им. Н.Э. Баумана
Москва, Россия

Аннотация

В настоящей работе предлагается реализация алгоритма гладкой трансформации, изложенного в работе [1]. Этот алгоритм был модифицирован нами в части нахождения углов между ребрами, исчезающими в процессе трансформации, что, по мнению авторов, должно давать более надежные результаты по сравнению с оригинальным алгоритмом. Также было сделано обобщение на случай замкнутых кубических сплайнов.

Алгоритм основан на физической модели многоугольника, то есть предполагается, что каждое ребро и угол многоугольника являются жесткими. Математически это означает введение понятия энергии контура и ее изменения при трансформации. После чего находится такое соответствие между вершинами начального и конечного контуров, при котором работа, совершаемая при трансформации, была бы минимальной. Решение этой задачи требует $O(mn)$ операций, где m и n – число вершин рассматриваемых многоугольников, при этом начальная вершина первого многоугольника переходит в начальную вершину второго многоугольника.

Ключевые слова: физическая модель, гладкая трансформация, анимация.

1. ВВЕДЕНИЕ

В последние годы задача непрерывных, гладких трансформаций одной фигуры (или целого изображения) в другую стала очень популярной в компьютерной графике и при построении моделей компьютерного зрения (см.[2], [3]).

Такие же задачи возникают в мультипликации, спецэффектах кинематографа, аппроксимации контуров и распознавании рукописного текста.

Постановка нашей задачи следующая: даны два плоских контура, требуется найти непрерывную последовательность фигур, переводящих один контур в другой. В качестве этих контуров мы брали многоугольники и замкнутые сплайны, порожденные этими многоугольниками.

Для построения промежуточных многоугольников используется стандартная модель, при которой положение вершин описывается формулой:

$P(t) = P^0 + t(P^1 – P^0)$,

где t меняется от 0 до 1, P^0 – точка начального многоугольника, P^1 - конечного многоугольника.

При такой простой модели возникает ряд трудностей, которых требуется избежать.

Во-первых, самопересечения. Эта проблема чаще всего возникает, когда в результате трансформации угол между двумя ребрами меняет знак, проходя через ноль.

Во-вторых, немонотонное изменение углов.

Ниже описано, как эти проблемы решаются в алгоритме, основанном на физической модели.

Для оптимизации работы алгоритма на многоугольниках с очень большим числом вершин (такие многоугольники обычно возникают при необходимости описать гладкую кривую) мы используем кубические сплайны, которые значительно уменьшают число необходимых операций.

2. МОДЕЛЬ ТРАНСФОРМАЦИИ МНОГОУГОЛЬНИКА

Пусть ребро AB переходит в A`B`, тогда свяжем с изменением длины ребра работу по формуле:

$$W_s = \frac{k_s \left| L_0 - L_1 \right|^{e_s}}{(1-c_s)\min(L_0, L_1) + c_s \max(L_0, L_1)},$$

где k_s, c_s, e_s – предопределенные константы; L_0, L_1, - начальная и конечная длины ребер.

Эта формула обобщает стандартную формулу для упругого стержня:

$$W = \frac{\delta^2 SE}{2L_0}$$

на случай начального или конечного ребра нулевой длины (не возникает деления на ноль). Здесь δ - абсолютное изменение длины стержня, S - площадь поперечного сечения, E - модуль Юнга, L_0 - длина стержня.

При изменении угла между сторонами многоугольника также введем работу определяемую формулой:

$$W_b = k_b(\Delta\theta + m_b \Delta\theta^*)^{e_b} + p_b,$$

где коэффициенты m_b, p_b вводят штраф на немонотонное изменение угла и на прохождение угла через ноль (в нашей программе коэффициент p_b брался равным бесконечности, если угол проходил через ноль, и нулю в противном случае); $\Delta\theta$ изменение угла при трансформации; $\Delta\theta^*$ область значений угла, где он меняется немонотонно.

2.1 Анализ изменения угла

Для анализа поведения угла при трансформациях используется следующие представление.

Пусть у нас происходит трансформация как показано на рисунке 1.

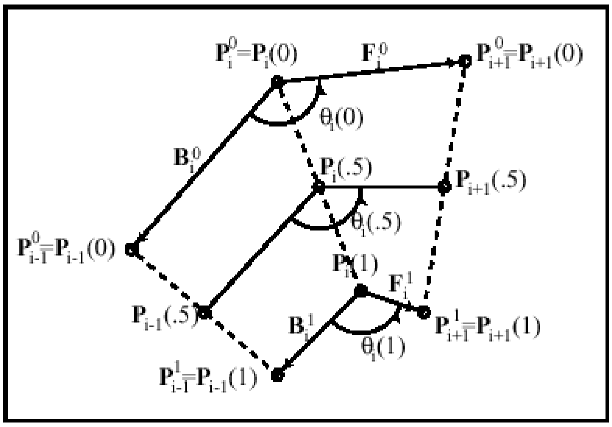

Рисунок 1: Трансформация угла.

Введем величины:

$$x_0 = \vec{F}_i^0 \cdot \vec{B}_i^0;$$

$$x_1 = \frac{\vec{F}_i^1 \cdot \vec{B}_i^0 + \vec{F}_i^0 \cdot \vec{B}_i^1}{2};$$

$$x_2 = \vec{F}_i^1 \cdot \vec{B}_i^1;$$

$$y_0 = \vec{F}_i^0 \times \vec{B}_i^0;$$

$$y_1 = \frac{\vec{F}_i^1 \times \vec{B}_i^0 + \vec{F}_i^0 \times \vec{B}_i^1}{2};$$

$$y_2 = \vec{F}_i^1 \times \vec{B}_i^1;$$

$$\vec{Q}(t) = (x_0, y_0)(1-t)^2 + (x_1, y_1)2t(1-t) + (x_2, y_2)t^2.$$

Где $\vec{P}_i \times \vec{P}_j \equiv (x_i, y_i) \times (x_j, y_j) \equiv x_i y_j - x_j y_i$.

Легко убедится, что угол между положительным направлением оси OX и $\vec{Q}$ равен углу между $\vec{F}$ и $\vec{B}$ в процессе трансформации. Угол проходит через ноль, тогда и только тогда, когда кривая $\vec{Q}(t)$ пересекает положительную полуось OX. Условие на монотонность - отсутствие экстремумов функции зависимости величины угла от t, т.е. отсутствие решений уравнения $\vec{Q}(t) \times \vec{Q}'(t) = 0$ на отрезке [0; 1].

Моментом, требующим наиболее пристального внимания, является случай ребер нулевой длины, то есть когда добавляются или убираются вершины многоугольника. В [1] предлагалось считать, что исчезающие вершины в бесконечно малой окрестности выстраиваются в цепочку и крайние углы равны $\frac{\pi + \alpha}{2}$, а средние π, см. рисунок 2 (пунктирные линии).

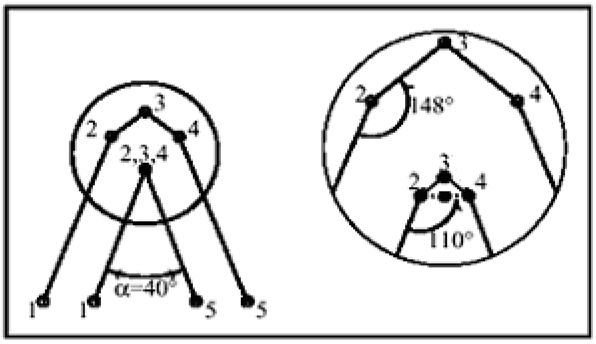

Рисунок 2: Пример слияния вершины.

Такой подход, по нашему мнению, не отражает реальной динамики изменения угла, и может приводить к самопересечениям. Вместо этого в программе использовалось следующее приближение. Ребро нулевой длины заменялось очень малым (0,1% от максимального значения), на рисунке 2 этот случай обозначен жирной линией. Это приближение дает ряд преимуществ: убирается вырождение, анализ исчезающих вершин не приходится выделять в отдельный программный блок, можно пользоваться уже описанными выше формулами, также такой подход дает наиболее правильную картину о поведении угла при трансформации многоугольника.

2.2 Нахождение наилучшей трансформации

Для поиска наилучшего соответствия между многоугольниками используется представление трансформаций с помощью графа.

Пусть у нас есть два многоугольника с пронумерованными вершинами, первая и последняя вершины совпадают. Построим решетку m на n, где пересечение i-того столбца с j-той строкой означает, что i-тая вершина первого многоугольника переходит в j-тую вершину второго. Тогда путь в этом графе будет соответствовать некоторому варианту трансформации одного многоугольника в другой. Наша задача найти такой путь, при котором совершаемая работа минимальна. Реализованный нами алгоритм позволяет сделать это за *O(mn)* операций, при этом считается, что начальная вершина многоугольника переходит в начальную вершину второго многоугольника.

3. ОБОБЩЕНИЕ НА ГЛАДКИЕ КРИВЫЕ

Описанный выше алгоритм был обобщен на трансформацию циклических кубических сплайнов.

Как известно, кубический сплайн задается набором точек, через которые проходит кривая, и в этих точках выполняется условие непрерывности кривой, а также ее первой и второй производной.

В программе использовалось параметрическое задание сплайна:

$$x_i(t) = a_i^1 + b_i^1 t + c_i^1 t^2 + d_i^1 t^3$$

$$y_i(t) = a_i^2 + b_i^2 t + c_i^2 t^2 + d_i^2 t^3,$$

где верхние индексы нумеруют компоненты x, y; $t \in [0; t_i]$; t_i - расстояние между (i-1)-ой и i-ой вершинами.

Для нахождения параметров этого представления необходимо решить две линейные системы из n уравнений. Специальный вид этих систем позволяет решить их за *O(n)* операций.

Алгоритм трансформации заключатся в следующем.

Основой трансформации являются опорные точки (те через которые проходит сплайн), считая их вершинами многоугольника, мы пользуемся описанным выше алгоритмом, но вместо длины ребра берем длину сплайна между этими двумя точками, в работе связанную с изменением величины угла берем только штрафы за не монотонность и самопересечения. Также добавляется новый вид работы, связанный с изменением кривизны сплайна. Алгоритм поиска пути остается без изменений, модернизации подвергаются только веса ребер графа.

Кривизна сплайна вычисляется по стандартной формуле:

$$k(t) = \frac{\left| x''y' - y''x' \right|}{\left[(x')^2 + (y')^2 \right]^{3/2}} .$$

Энергия, связанная с кривизной сплайна вычисляется по формуле:

$$U = \int_{t_{i-1}}^{t_i} k^2(t)dt .$$

Данный интеграл можно взять аналитически. Работа по изменению кривизны равна U_1 - U_2.

4. ПРИМЕРЫ И ОБСУЖДЕНИЕ

В программе использовались следующие значения констант: $k_s = 2$, $c_s = 0{,}01$, $e_s = 2$, $e_b = 1$, $k_b = 50$, $m_b = 50000$.

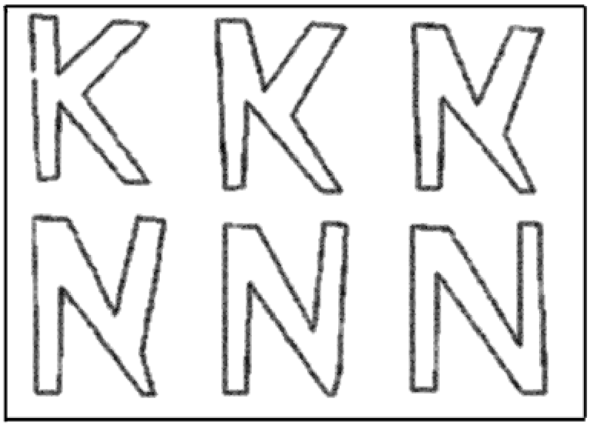

Рисунок 3: Пример гладкой трансформации многоугольника.

На рисунках 4 и 5 представлен пример трансформации сплайна. Количество точек на фигуру порядка 10, это дает существенный выигрыш во времени, по сравнению с поточечным заданием кривой (аппроксимация гладкой кривой с помощью многоугольника), так как алгоритм занимает *O(mn)* операций, то выигрыш может составлять, в зависимости от качества задаваемой кривой, до четырех порядков.

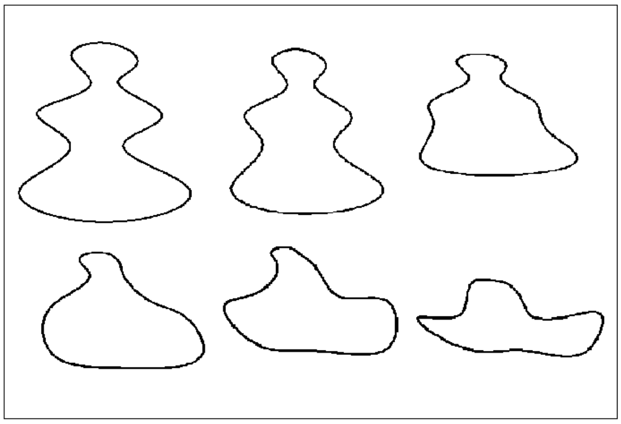

Рисунок 4: Пример гладкой трансформации сплайна. Снеговик преобразуется в катер.

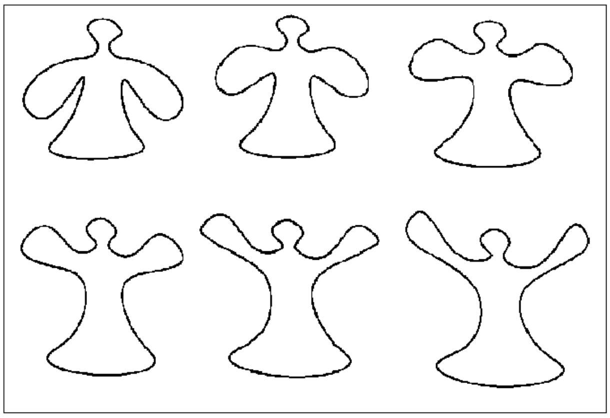

Рисунок 5: Пример гладкой трансформации сплайна. Привидение.

Благодарности

Работа выполнена при поддержке компании Intel в рамках сотрудничества Intel с МГУ.

Авторы считают приятным долгом поблагодарить Л.В. Нестеренко и В.Л. Ерухимова за внимание к работе, а также Лагно Дениса и Соболева Андрея за полезные обсуждения и замечания.

5. ЛИТЕРАТУРА

[1] Thomas W. Sederberg and Eugence Grenwood. *A Physically Based Approach to 2-D Shape Blending. In Computer Graphics (Siggraph '92 Proceedings), volume 26, pages 25-34, 1992.*

[2] A. Efrat, S.Har-Peled, L.Guibas and T.M.Murali. *Morphing between Curves. Proc. 12th Annual ACM-SIAM Symposium on Discrete Algorithms, 2001, 680-689.*

[3] Helmut Alt and Leonidas Guibas. *Discrete Geometric Shapes: Matching, Interpolation, and Approximation.In J.R.Sack,J. Urrutia, editors, Handbook of Computational Geometry, pages 121 - 153. Elsevier Science Publishers B.V. North-Holland, Amsterdam, 1999.*

Об авторах

Буренков Денис Сергеевич – студент 6-ого курса физического факультета МГУ им. М. В. Ломоносова, e-mail: denis@msu.nstl.nnov.ru

Храпов Павел Васильевич – доцент МГТУ им. Н.Э.Баумана, факультет ФН, e-mail: pkhrapov@msu.nstl.nnov.ru