

УДК 681.3

ИСПОЛЬЗОВАНИЕ ЖИРНЫХ ЛИНИЙ ДЛЯ МОДЕЛИРОВАНИЯ РУКОПИСНЫХ ШРИФТОВ

Семёнов А.Б.* , Смирнов С.А.**

* Тверской государственный университет, кафедра информационных технологий

** Московский университет природы, общества и человека “Дубна”, магистратура
кафедры системного анализа и управления

Поступила в редакцию 15.04.2009, после переработки 20.05.2009.

В данной работе рассматривается задача представления рукописного шрифта в виде жирных линий - однопараметрических семейств кругов переменного радиуса с центрами на гладких кривых. Показана возможность моделирования «живого» почерка путем небольших изменений формы символов. Продемонстрирован метод автоматического кернинга – визуального выравнивания межсимвольных расстояний. Используемые и предложенные в статье идеи были реализованы при создании шрифтового редактора рукописных символов.

This work proposed task of represent human hand-written font as set of fat lines - one-parameter family of circles variable radius with centers on smooth curves. Achieved possibility modeling “live” hand-written by the way small modification of symbols forms. Showed automatic kerning method – visual levelling of intersymbol distances. Ideas that used and proposed in article was realized in hand-written font editor.

Ключевые слова: жирные B-сплайновая кривые, векторный рукописный шрифт.

Keywords: fat B-spline curves, vectorial hand-written font.

Введение

Существует три основных способа описания шрифта, применяемые для их компьютерного моделирования [1]: растровый, векторный и граничный. При растровом подходе каждая буква представляет собой бинарный рисунок – матрицу из черных и белых пикселей. Недостатки в использовании растрового шрифта – большой размер получаемого шрифта, зачастую более медленная перерисовка на экране, сложность редактирования. Также, растровый рисунок, в отличие от векторного, имеет фиксированное разрешение. Из-за этого, при печати в большем разрешении, автоматически теряется качество: появляется нечеткость изображения, угловатость форм. Разрешение же векторного рисунка зависит только от разрешения выводного устройства. Поэтому векторный рисунок всегда выводится в том максимальном качестве, доступном этому устройству.

Большинство используемых векторных способов описания шрифта являются граничными, при которых контур каждого символа моделируется кривыми. Наиболее часто встречаемый метод – задание контура каждого символа в виде семейства кривых Безье. На основе этого подхода созданы популярные шрифтовые форматы – TrueType и PostScript, где используются кривые Безье второго и третьего порядка [1]. Отличительной особенностью человеческого почерка является небольшое различие в написании одних и тех же букв в тексте. Поэтому важной задачей при выводе текста является «оживление» почерка – введение небольших вариаций ширины и формы штрихов для отдельных символов. Невозможность подобного оживления – недостаток граничного представления для создания рукописных шрифтов. Во-первых, при эмуляции вариации ширины пера требуется решить нетривиальную задачу синхронизации изменения обеих границ контура. Во-вторых, при подобной вариации необходимо решить еще более сложную задачу сохранения гладкости составных кривых Безье. На рис. 2 видно, что при простом редактировании опорных вершин происходит излом кривой, гладкость теряется.



Рис. 1: Основные типы шрифтов: растровый, векторный, контурный.

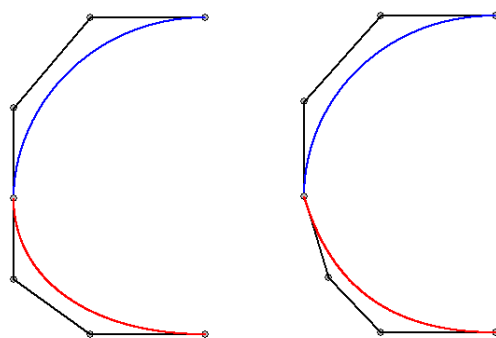


Рис. 2: Излом кривой Безье при редактировании.

Для поставленной задачи больше подходит модель представления шрифта в виде пера с шириной.

1. Использование аппарата жирных линий для рукописного шрифта

Представление символов через модель пера с шириной дает большие возможности для построения рукописного шрифта. Для описания рукописного шрифта более подходит его представление в виде семейства жирных линий. Жирная линия определяется как след от перемещения круга переменного радиуса вдоль гладкой кривой.

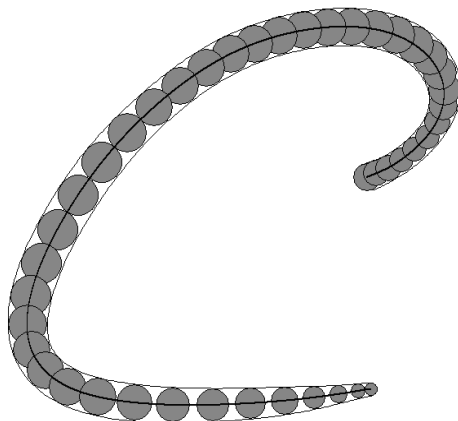


Рис. 3: Жирная линия.

Видно, что задачи сохранения гладкости и синхронизации границ контура решаются сами собой при выборе этой модели. Что характерно, вариация толщины пера и формы штрихов сводится к простому редактированию кругов.

В качестве математического аппарата для описания жирных линий используются В-сплайны третьего порядка.

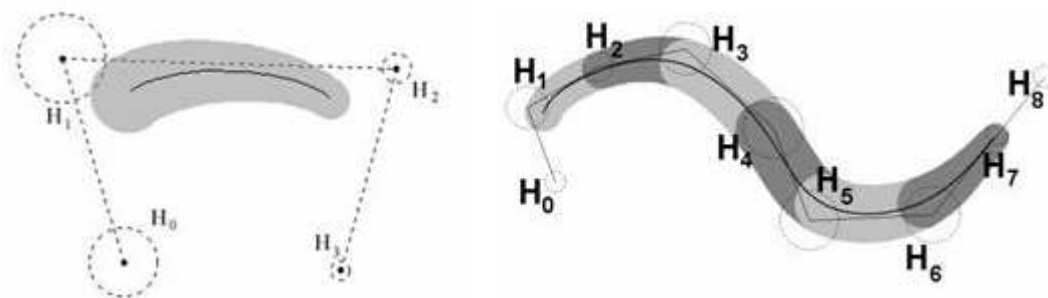


Рис. 4: Элементарная и составная жирные В-сплайновые кривые.

Элементарная жирная В-сплайновая кривая третьего порядка [2] задается следующим векторным уравнением

$$C(t) = (x(t), y(t), r(t)) = \frac{1}{6} \sum_{i=0}^3 B_i(t) H_i,$$

где $t \in [0, 1]$ – параметр кривой, $H_i = \{H_{ix}, H_{iy}, H_{ir}\}$ – семейство контрольных кругов с центрами в точках (H_{ix}, H_{iy}) и радиусами H_{ir} , $B_i(t)$ – базовые функции В-сплайна третьей степени, имеющие следующий вид:

$$\begin{aligned} B_0(t) &= (1-t)^3 \\ B_1(t) &= 3t^3 - 6t^2 + 4 \\ B_2(t) &= -3t^3 + 3t^2 + 3t + 1 \\ B_3(t) &= t^3 \end{aligned}$$

Пример элементарной жирной В-сплайновой кривой с контрольными кругами $\{H_0, \dots, H_3\}$ представлен на рис. 4, слева. Серым цветом показана жирная линия (след от перемещения круга), черным цветом показана ось жирной линии, пунктирной линией показаны контрольные круги.

Элементарные кривые могут объединяться в более сложные графические примитивы – составные жирные линии. Кривая γ , которая представляется в виде объединения элементарных кубических В-сплайновых кривых $\gamma^{(1)}, \dots, \gamma^{(m-2)}$, называется составной кубической В-сплайновой кривой с общим семейством контрольных кругов $H = (H_0, \dots, H_m)$, где кривая $\gamma^{(k)}$ имеет контрольные круги $H_{k-1}, H_k, H_{k+1}, H_{k+2}; k = 1, \dots, m - 2$. Пример составной жирной линии показан на рис. 4, справа.

2. Методы создания шрифтов

На данный момент разработано два подхода в создании шрифтов предлагаемым методом.

2.1 Создание шрифта «с нуля»

При таком способе шрифт формируется при непосредственном вводе данных. Оператор вводит каждый символ с помощью пера графического планшета. Планшет позволяет получать в каждый момент времени координаты положения пера (x, y) и относительную величину его нажатия f .

Таким образом, в каждый момент времени получаем некоторый круг с центром в (x, y) и радиусом $C * \frac{f}{f_{\max}}$, где C – константа, задающая максимальный радиус круга, f_{\max} – максимальная величина нажатия пера. След от перемещения пера при вводе символа задает последовательность кругов переменного радиуса. Эта последовательность представляет собой множество опорных кругов искомой кривой.

Чтобы построить жирную линию по опорным кругам, необходимо решить задачу ее восстановления (интерполяции). Этих кругов может быть избыточно много, так что может потребоваться их аппроксимация – выбор достаточного подмножества кругов с сохранением заданной точности построения. Решения этих задач рассмотрены далее.

2.2 Интерполяция жирной В-сплайновой кривой

Задачу можно поставить следующим образом. Дана последовательность кругов G_1, G_2, \dots, G_{n-1} , где каждый круг задается в виде $G_i = (x_i, y_i, r_i)$. Необходимо

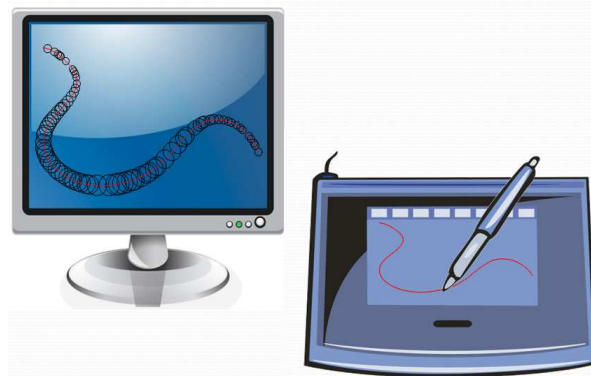


Рис. 5: Ввод жирных линий с помощью графического планшета.

построить жирную В-сплайновую кривую $C(t) = (u(t), v(t), r(t))$ такую, чтобы она в точности проходила через все круги G_i .

Задача нахождения требуемых кругов решена в [2] и сводится к решению трех систем уравнений, отдельно для положений центров и радиусов.

2.3 Аппроксимация жирной В-сплайновой кривой

Задача аппроксимации сводится к уменьшению количества опорных кругов, путем их удаления из исходного множества, с достижением требуемой точности.

Вначале в искомое множество вносятся первый, последний и два круга, примерно делящих исходное множество на три равные части. Решение задачи сводится к последовательному включению в искомое множество наиболее «удаленных» от уже включенных в это множество кругов.

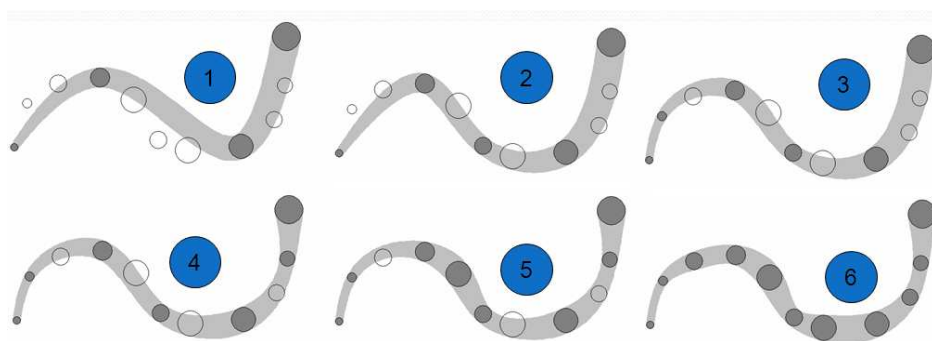


Рис. 6: Последовательное выполнение алгоритма аппроксимации.

На рис. 6 видны последовательные шаги аппроксимации. Алгоритм останавливается, когда достигается требуемая точность ε , то есть все невключенные круги находятся на расстоянии $r \leq \varepsilon$ от кривой, задаваемой включенными кругами.

2.4 Создание шрифта из отсканированного образца почерка

При таком подходе оператор пишет отдельные символы шрифта на листе бумаги. Далее эти символы заносятся в компьютер с помощью сканера. Для каждого из них необходимо построить представление в виде жирных линий. Оператор выделяет каждый символ, и для него строится множество срединных осей – скелет. Каждая ось скелетного графа аппроксимируется составной жирной В-сплайновой кривой, с заданной точностью. Задачу составления скелетного графа можно решить, например, методом из [5].

После построения, разрабатываемый шрифт необходимо откорректировать, так как при аппроксимации возможны дефекты изображения. Это можно делать вручную, с помощью изменения положения центров и радиусов кругов. Также это можно сделать автоматически.

2.5 Автоматическое сглаживание жирных линий

Задачу сглаживания можно решить следующим образом [6]. Для каждого контрольного круга H_i , $i = 1, \dots, n - 2$, за исключением первого и последнего, можно пересчитать значение его радиуса по следующей формуле: $H'_{ir} = k_1 H_{i-1r} + k_2 H_{ir} + k_3 H_{i+1r}$, где H'_{ir} – новое значение радиуса круга H_i , k_i – весовые коэффициенты, удовлетворяющие условию $\sum_{i=1}^3 k_i = 1$. Данная процедура представляет собой операцию дискретной свертки для одномерного сигнала с ядром (k_1, k_2, k_3) . Выбор ядра свертки (k_1, k_2, k_3) будет характеризовать степень сглаживания радиусов контрольных кругов в жирной линии. На рис. 7 продемонстрирована операция сглаживания радиусов контрольных кругов с ядром свертки $(1/3, 1/3, 1/3)$, слева исходная жирная линия, справа – сглаженная. Ломаными линиями показаны соответствующие круги до и после операции сглаживания.

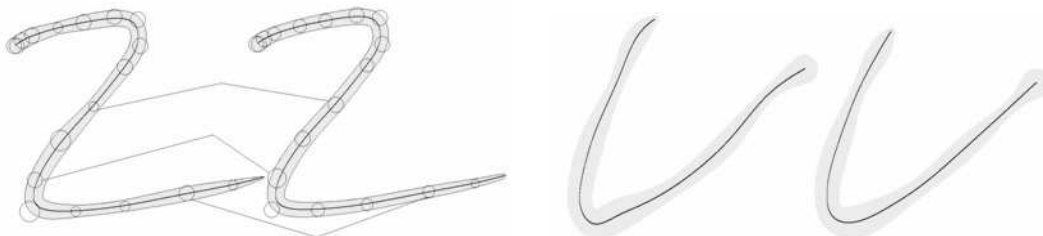


Рис. 7: Сглаживание радиусов (слева) и осей (справа) жирной линии.

Аналогично, подобный шаг можно проделать и для оси жирной линии. Формулы для изменения x и y идентичны формуле изменения радиуса.

Результат сглаживания оси жирной линии с ядром свертки $(1/3, 1/3, 1/3)$ показан на том же рис. 7 справа. Слева – исходная кривая, справа – сглаженная.

3. Распечатка текста

Предложенный метод моделирования рукописного символов был реализован в виде шрифтового редактора рукописного почерка. Ключевой результат работы программы – распечатка текстов созданным шрифтом.

Основная задача при выводе рукописного текста на печать – проблема вычисления кернинга. Кернинг – визуальное выравнивание между различными парами символов шрифта. На рис. 8 показана распечатка слов без кернинга (вверху) и с ним (внизу).

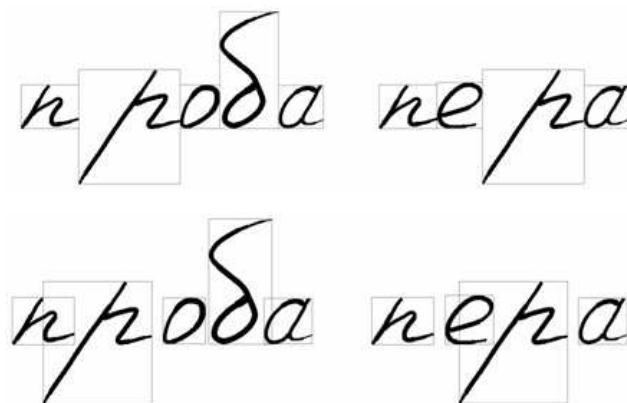


Рис. 8: Распечатка слов без кернинга (вверху) и с ним (внизу).

Нетрудно понять, что если в шрифте n символов, то полное описание кернинга для него требует n^2 единиц информации. Конечно, этой информации зачастую избыточно много для обычных текстов, где различных сочетаний букв как минимум на порядок меньше. Задача автоматического вычисления кернинга очень актуальна, поскольку ручная обработка такого количества информации достаточно трудоемка. Кернинг можно задавать как вручную, так и программно. Можно выделить несколько способов его программного вычисления.

- Метод описан в [3]. Задаем постоянное минимальное горизонтальное расстояние между всеми парами символов. Если его обозначить через c , то, при минимальном расстоянии между парой символов $r(i, j) = m_{ij}$, сдвигаем символы на величину $k_{ij} = c - m_{ij}$. Минимальное расстояние вычисляется, например, горизонтальным сканированием выстроенных на одной линии символов.
- Задаем постоянную пропорциональную площадь между всеми парами символов. Задаем кернинг следующим образом: при базовой высоте общей части h_c и базовой площади s_c значение базового кернинга обозначим k_c . Теперь, если имеем для пары символов (i, j) высоту общей части h и площадь s , то сдвиг символов относительно друг друга равен $k_{ij} = \frac{s_c}{h_c} - \frac{s}{h}$.

Рассмотрим часто встречающиеся задачи создания нового шрифта, подобного некоему созданному, имеющее другое начертание.

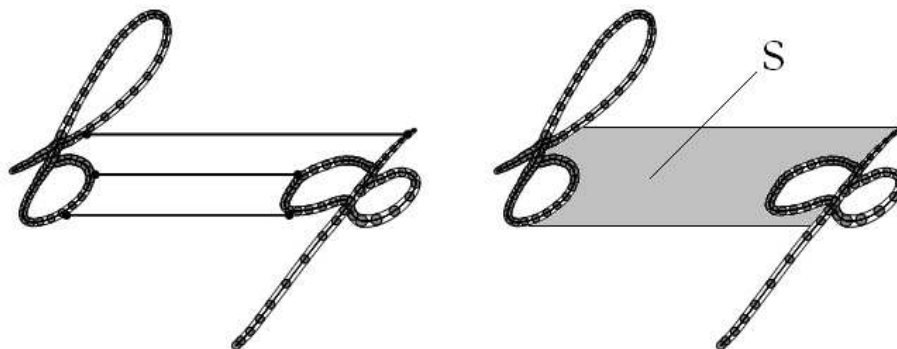


Рис. 9: Методы вычисления кернинга, моноширинный и пропорциональный.

4. Создание тонкого и жирного начертания символа

На рис. 10 видно, как путем пропорционального изменения радиусов всех кругов можно добиться изменения толщины шрифта. Слева – исходный символ, в середине – с радиусами, уменьшенными на 30%, справа – с радиусами, увеличенными на 50%.

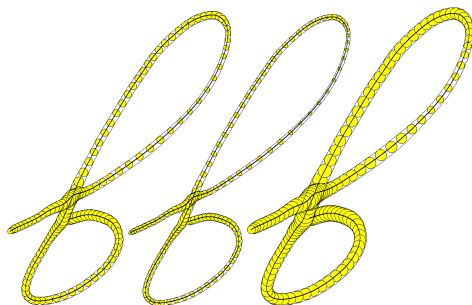


Рис. 10: Вариации ширины пера.

5. Модификация символов методом рамки. Создание курсивного начертания символа

Можно предложить еще один способ изменения формы символов. Находим минимальный прямоугольник, содержащий символ. Теперь, элементарно модифицируем этот прямоугольник, меняем положение вершин, и соответственно меняем положения центров опорных кругов. Таким образом, получаем другое начертание символов. Например, для создания курсивного почерка, выполняем задачу элементарного преобразования прямоугольника в параллелограмм. Для этого элементарно трансформируем центр каждого круга.

Трансформация точки (x, y) описывается в виде [7]

$$x_{new} = a \cdot x \cdot y + b \cdot x + c \cdot y + d,$$

$$y_{new} = e \cdot x \cdot y + f \cdot x + g \cdot y + h.$$

Подставив значения x и y для четырех точек, определяющих модифицируемый прямоугольник, получим восемь уравнений с восемью неизвестными. Система является разрешимой.

При эмуляции курсивного почерка масштабирование по (x, y) и изменение положений центров по y не происходит, в результате решения системы получим $b = g = 1$ и $a = k$, где константа k задает степень наклона символа. Остальные коэффициенты нулевые.

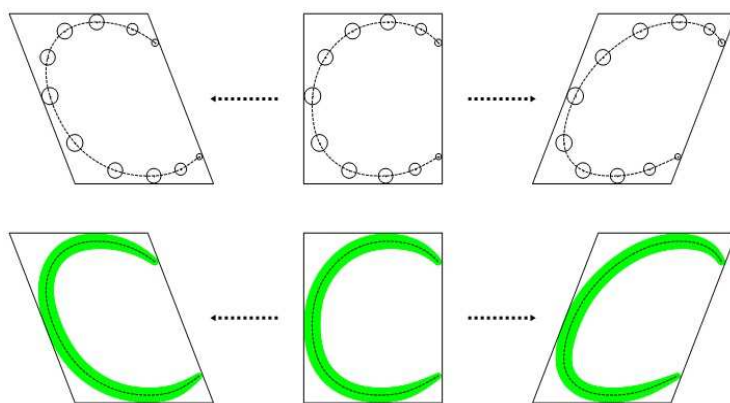


Рис. 11: Модификация символа методом рамки.

На рис. 11 показан результат применения метода рамки к символу.

6. Введение небольших вариаций (оживление) почерка

Интересная задача – оживление почерка, введение небольших вариаций, может быть решена введением в шрифт различных начертаний одних и тех же букв. В дальнейшей распечатке текста, случайным образом используя различные начертания одной буквы можно добиться желаемого эффекта.

Также эта задача может быть решена программными методами. Простой и эффективный способ – создание по символу нескольких его вариаций.

Например, при выводе символа на печать изменим радиусы кругов на некоторую случайную величину: $r_i = r_i + \xi_i$, где r_i – радиус контрольного круга H_i , ξ_i – случайная величина, равномерно распределенная в заданном интервале $[-a, a]$. Чем больше величина a , тем более различное начертание примут символы. Наглядно суть метода можно увидеть на рис. 12, где пунктиром показана новая вариация символа. Различность начертания становится очевидной при наложении исходной и новой кривой.

Аналогичную процедуру можно провести с осями жирной линии, введя вариацию положения центров опорных кругов. Для большей непохожести возможно одновременное создание «шума» радиусов и центров. Также большего эффекта можно добиться, совместив случайность и последующее сглаживание.

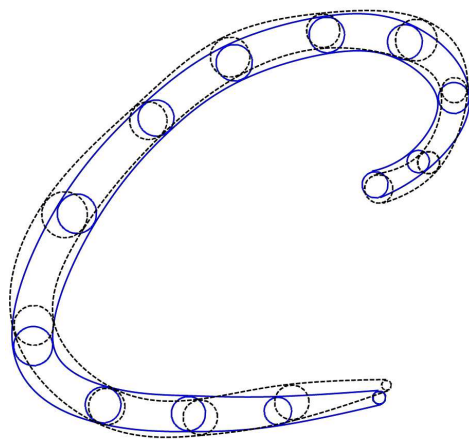


Рис. 12: Вариация символа.

В В

В статье рассматривается задача представления рукописного шрифта в виде жирных линий - однопараметрических семейств кругов переменного радиуса с центрами на гладких кривых.

Рис. 13: Печать текста.

В созданном редакторе была реализована печать текста. При увеличении можно наблюдать различность начертания отдельных символов.

Заключение

Показанные в статье методы показывают преимущество использования аппарата жирных линий при моделировании рукописного почерка. Предложенный подход может найти применение в шрифтовых редакторах.

Список литературы

- [1] Барышников Г.М., Бизяев А.Ю., Ефимов В.В., Моисеев А.А., Почтарь Э.И., Ярмола Ю.А. Шрифты. Разработка и использование. – М.: Издательство ЭКОМ, 1997.
- [2] Местецкий Л.М., Семенов А.Б. Жирные линии на основе B-сплайнов// Сложные системы: моделирование и оптимизация. Сборник научных трудов. – Тверь: Тверской государственный университет, 2001.
- [3] Семенов А.Б. Метод автоматического кернинга рукописных символов// Труды международной конференции по компьютерной графике «Графикон-2007» (Новосибирск, 2007). – Новосибирск, 2007.
- [4] Местецкий Л.М. Скелет многосвязной многоугольной фигуры// Труды международной конференции по компьютерной графике «Графикон-2005» (Новосибирск, 2005). – Новосибирск, 2005.
- [5] Клименко С.В., Местецкий Л.М., Семенов А.Б. Моделирование рукописного шрифта с помощью жирных линий// Труды международной конференции по компьютерной графике «Графикон-2006» (Новосибирск, 2006). – Новосибирск, 2006.
- [6] Роджерс Д., Адамс Дж. Математические основы машинной графики: Пер. с англ. – М.: Мир, 2001. – 604 с.