

# Использование патча Безье для аппроксимации искажения изображений текстовых документов

Антон Масалович, Леонид Местецкий  
Московский государственный университет,  
Москва, Россия  
anton\_m@abbyy.com, l.mest@ru.net

## Аннотация

Данная работа описывает метод аппроксимации деформации текстового документа кубическим патчем Безье, основанный на непрерывном гранично-скелетном представлении текста.

В статье описаны основные шаги алгоритма: построение непрерывного гранично-скелетного представления изображения, предобработка скелета, выделение в скелете ветвей лежащих между строками текста, итерационный процесс построения патча Безье на основе аппроксимаций межстрочных ветвей, распрямление искривленного изображения.

В статье приведены экспериментальные результаты, показывающие эффективность предложенного механизма для систем распознавания изображений.

Проведен краткий анализ существующих решений проблемы распрямления текстовых строк в документах.

В конце статьи описаны недостатки предложенного подхода и определены направления дальнейшей работы.

Данная статья является развитием работы [1].

**Ключевые слова:** распознавание текста, предобработка изображений, распрямление строк, непрерывное гранично-скелетное представление, внешний скелет, патч Безье.

## 1. ВВЕДЕНИЕ

Все современные коммерческие системы распознавания текстов предназначены для работы с прямыми горизонтальными строками текста. К сожалению, для большинства реальных документов это требование не выполняется. Изображение может быть деформировано перед распознаванием множеством различных способов. Таким образом, появляется задача аппроксимации искажения документа.

Подход к распрямлению строк, описанный в этой статье, базируется на построении внешнего скелета текстового изображения. Основная идея описанного подхода основана на предположении, что во внешнем скелете документа легко выделить длинные непрерывные ветви, которые определяют межстрочные просветы в тексте [1]. После выделения межстрочных ветвей мы аппроксимируем их кривыми Безье. На основе набора таких кривых Безье мы строим модель аппроксимации всей страницы в виде двумерного патча Безье. В статье приведен итерационный алгоритм построения двумерного патча Безье на основе набора одномерных кривых. В статье также приведен механизм, который позволяет использовать построенный патч для распрямления изображения.

Работа выполнена при поддержке РФФИ, грант 05-01-00542.

## 2. СУЩЕСТВУЮЩИЕ РЕШЕНИЯ

В последние несколько лет появилось большое количество алгоритмов распрямления строк текста [4-8]. Это связано с тем, что решение данной задачи востребовано для всех профессиональных систем распознавания текстов.

Существующие решения проблемы можно отнести к трем различным подходам:

- Первый подход – это выделение текстовых строк на изображении путем объединения близко лежащих объектов на изображении в одну строку и аппроксимация формы строки по некоторым характерным точкам объектов входящих в строку. Основной недостаток этого метода – это то, что очень трудно выделять такие характеристические точки на черных объектах, которые могли бы дать стабильную аппроксимацию формы строки.

- Второй подход – идентификация математической модели искажения для конкретной страницы. Основной недостаток подобных методов в ограниченности их применения только для какого специфического типа искажения.

- И, наконец, третий подход – это итерационные методы исправления искажения страницы. Изображение последовательно деформируется пока некоторая оценка прямоты его строчек не станет максимальной. Основной недостаток этого подхода то, что в нем используются численные методы минимизации функционала очень сложного вида, из-за за чего выполнение таких алгоритмов занимает очень много времени и может породить непредсказуемые результаты.

Как видно из описания подходов, большинство этих алгоритмов имеют серьезные недостатки, которые затрудняют их использование в коммерческих системах.

## 3. ОПИСАНИЕ ПРОБЛЕМЫ

Предположим, что имеется изображение  $I(x, y)$ , где  $I$  - цвет пикселя с координатами  $(x, y)$  (подробные требования к обрабатываемым изображениям приводятся в работе [1]). Данное изображение содержит блок с деформированными строками. Предположим также, что возможно перераспределить пиксели в этом изображении без изменения их цвета так, чтобы получить изображение документа, где текстовые строки будут прямыми и горизонтальными. Итак, мы хотим найти непрерывную векторную функцию

$\bar{D}(x, y)$ , чтобы получить распрямленное изображение в виде:  $\bar{I}(x, y) = I(\bar{D}_x(x, y), \bar{D}_y(x, y))$ . Эта функция  $\bar{D}(x, y)$  будет аппроксимацией искажения всего изображения.

Чтобы оценить качество нашей аппроксимации, мы распознаем изображение до и после распрямления, используя одну из современных систем распознавания изображений. Качество распознавания в современных системах распознавания напрямую зависит от прямоты строк в изображении. Таким образом, улучшение в качестве распознавания после распрямления изображения может служить хорошей оценкой алгоритма аппроксимации искажения.

#### 4. НЕПРЕРЫВНОЕ ГРАНИЧНО-СКЕЛЕТНОЕ ПРЕДСТАВЛЕНИЕ ИЗОБРАЖЕНИЯ

Для решения поставленной задачи воспользуемся непрерывным гранично-скелетным представлением изображения [2,3].

**Скелетом многоугольной фигуры** на плоскости называется такое множество точек, что для каждой точки имеется не менее двух ближайших точек на границе фигуры. Скелет изображения можно представить в виде плоского графа (в [1] приведено более подробное определение). На рис. 1 приведен пример непрерывного гранично-скелетного представления изображения

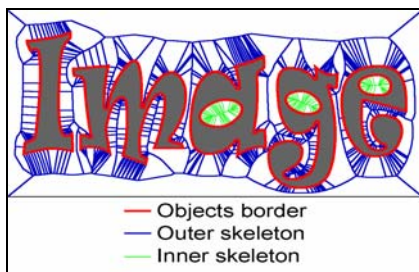


Рис. 1. Непрерывное гранично-скелетное представление изображения.

#### 5. ОСНОВНАЯ ИДЕЯ АЛГОРИТМА

Основная идея предлагаемого алгоритма заключается в том, что во внешнем скелете документа легко можно выделить ветви, которые лежат между соседними текстовыми строками. Эти ветви можно использовать, чтобы аппроксимировать деформацию межстрочных пробелов на изображении. Затем по аппроксимациям отдельных межстрочных просветов можно построить аппроксимацию искажения всего документа. Вкратце, предложенный алгоритм состоит из следующих шагов:

- Построение внешнего непрерывного скелета изображения.
- Фильтрация лишних ребер скелета.
- Выделение в скелете межстрочных ветвей.
- Аппроксимации каждой выделенной ветви кубическими кривыми Безье.

- Построение общей модели искажения страницы в виде патча Безье на основе построенных аппроксимаций ветвей скелета.
- Итерационный процесс подгонки и уточнения полученного патча.

#### 6. ПРЕОБРАБОТКА ИЗОБРАЖЕНИЯ И СКЕЛЕТА

При построении внешнего скелета необработанного изображения в скелете появляется очень много мелких ребер, обработка которых разрабатываемым алгоритмом требует значительного времени. В действительности, значимыми для задачи являются только те ребра, которые лежат между объектами в соседних строках. Из-за этого возникает задача преобработки скелета.

Перед построением скелета выполняется заполнение черным белых горизонтальных промежутков. Другими словами, если расстояние между двумя черными точками в одной строке изображения меньше некоего порога ( $N_{\min}$ ), все точки в данном интервале делаются черными. Таким образом, объекты, лежащие в одной строке текста, сливаются вместе.

Порог  $N_{\min}$  выставлялся вручную на основе разрешения исходного изображения.

Также мы удаляем из скелета все ребра и узлы, которые порождены различными гранями одного и того же объекта и те ребра скелета, которые лежат между объектами изображения и границей картинки (в [1] приведено более подробное описание данной фильтрации).

Описанная выше процедура преобработки позволяет в несколько раз увеличить скорость алгоритма и значительно повышает его надежность.

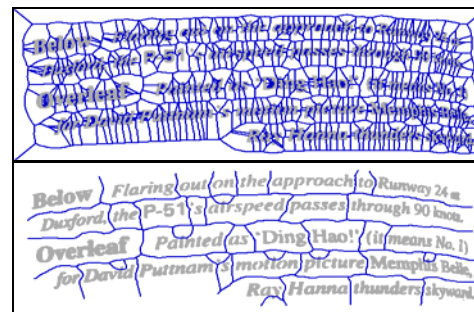
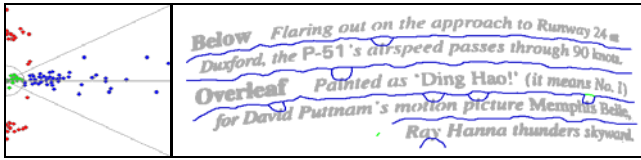


Рис. 2. Скелет изображения до и после преобработки

#### 7. КЛАСТЕРИЗАЦИЯ РЕБЕР СКЕЛЕТА

Для выделения межстрочных ветвей в скелете текста мы кластеризуем все ребра скелета на три группы: горизонтальные ребра, вертикальные ребра и короткие соединительные ребра. Для кластеризации мы используем два параметра: минимальную длину горизонтального ребра ( $L_{\min}$ ) и максимальный угол горизонтального ребра ( $A_{\max}$ ).

Алгоритм кластеризации ребер скелета подробно описан в работе [1]. На рисунке 3 приведен пример скелета текста после удаления в нем вертикальных ребер.



**Рис. 3.** Ребра скелета с рис. 2.б отмеченные на одной плоскости и скелет изображения с рис. 2.б после удаления в нем вертикальных ребер.

## 8. ПОСТРОЕНИЕ МЕЖСТРОЧНЫХ ВЕТВЕЙ

Оставшиеся горизонтальные и короткие ребра объединяются в ветви по следующим правилам:

- Если две длинные горизонтальные ветви соединяются в одном узле на краю изображения, то мы разделяем этот узел на два узла, к каждому из которых примыкает по одной ветви. Такая ситуация происходит, когда две межстрочных ветви сливаются друг с другом за пределами текста (см. рис. 4.а).
- Если в каком-то узле скелета соединяются только два ребра, то эти два ребра принадлежат одной ветви.
- Если между двумя узлами в скелете проходит две непересекающиеся ветви, то мы удаляем более искривленную из этих ветвей. Такая ситуация возникает, если в тексте есть символы с диакритикой и одна из ветвей скелета проходит между символами в разных строках, а вторая ветвь проходит между символом и его диакритическим знаком (см. рис. 4.б).
- Если три ветви соединяются в одном узле, то мы удаляем наиболее короткую из этих ветвей. Такая ситуация происходит, когда у нас остаются неправильно определенные короткие ребра, которые на самом деле лежат между объектами одной строки (см. рис. 4.в).
- Если появляется ветвь скелета, в которую входят только короткие ребра и которую невозможно продолжить не одним горизонтальным ребром, то мы удаляем такую ветвь.



**Рис. 4.** Примеры основных проблем при соединении межстрочных ветвей скелета

Описанные выше правила последовательно в цикле применяются к скелету, пока в нем не останутся только длинные горизонтальные ветви, лежащие между соседними строками текста. Затем каждая такая ветвь аппроксимируется кубической кривой Безье.

## 9. ПОСТРОЕНИЕ ПАТЧА БЕЗЬЕ ПО НАБОРУ КРИВЫХ

После того как мы получили аппроксимацию большинства межстрочных просветов в тексте нам нужно построить аппроксимацию всего изображения.

Обозначим контрольные точки межстрочных кривых Безье как  $I_{ki}$ , где  $k$  – номер кривой и  $i$  – порядок контрольной точки при построении кривой.

Для каждого набора точек  $\{I_{ki}\}_{i=0}^{k=0}$  (контрольные точки всех кривых с одинаковым порядковым номером) мы построим аппроксимацию кривой Безье. Контрольные точки получившейся кривой мы обозначим как  $P_{ij}$ , где  $i$  – порядок контрольных точек, по которым эта аппроксимация построена, а  $j$  – порядковый номер этой контрольной точки в этой кривой (см. рис. 5).



**Рис. 5.** Пример построения контрольных точек общего патча Безье по набору горизонтальных кривых.

После того, как мы получили набор точек  $P_{ij}$ , мы можем построить по ним модель деформации всей страницы в виде патча Безье. Другими словами, наша аппроксимация искажения страницы может быть описана следующей формулой:

$$\bar{D}(x, y) = \sum_{i=0}^3 \sum_{j=0}^3 P_{ij} * b_{i,3}(x) * b_{j,3}(y)$$

$b_{r,3}(t)$  - кубический полином Бернштейна.

## 10. ИТЕРАЦИОННЫЙ ПРОЦЕСС ПОСТРОЕНИЯ ПАТЧА БЕЗЬЕ

К сожалению, при построении межстрочных кривых невозможно точно указать, где у нас начинается каждая строка. Соответственно, каждая кривая у нас может начинаться и кончаться в достаточно произвольном месте и вертикальные границы патча могут изгибаться достаточно причудливым образом. Чтобы избежать этого, мы запускаем итерационный процесс подгонки патча.

Для каждой межстрочной кривой  $C_i(x) = \sum_{j=0}^3 A_j^i * b_{j,3}(x)$  мы

фиксируем параметр  $Y_i$  - параметр контрольных точек этой кривой при построении вертикальных аппроксимаций контрольных точек при создании патча.

В построенном по набору кривых патче кривой  $C_i$  будет соответствовать кривая:

$$\bar{C}_i = \bar{D}(x, y_i) = \sum_{i=0}^3 \sum_{j=0}^3 P_{ij} * b_{i,3}(x) * b_{j,3}(y_i)$$

Определим параметры  $\alpha$  и  $\beta$ , как параметры точек на кривой  $\bar{C}_i$ , ближайших к началу и концу кривой  $C_i$ .

$$\begin{cases} \alpha = \arg \min_t \rho(\bar{C}_i(t), C_i(0)) \\ \beta = \arg \min_t \rho(\bar{C}_i(t), C_i(1)) \end{cases}$$

Затем построим кривую  $C'_i$ , совпадающую с  $C_i$  по точечно, но со смещенными параметрами, так что  $C'_i(\alpha) = C_i(0)$ ,  $C'_i(\beta) = C_i(1)$ . То есть,

$$C'_i(t) = C_i\left(t - \frac{\alpha}{\beta - \alpha}\right) = \sum_{j=0}^3 A_j^i * b_{j,3}\left(t - \frac{\alpha}{\beta - \alpha}\right) = \sum_{j=0}^3 B_j^i * b_{j,3}(t)$$

Посчитаем среднее отклонение  $d$  построенной кривой  $C'_i$  от кривой из патча  $\bar{C}_i$ . Если это отклонение выше некоего заданного порога, то удалим начальную кривую  $C_i$  из набора для построения патча, иначе просто заменим кривую  $C_i$  в наборе на  $C'_i$ .

После обработки всех кривых из набора по описанному выше алгоритму, мы снова строим патч Безье по новым кривым. Процесс продолжается до тех пор, пока на очередном шаге параметры  $\alpha$  и  $\beta$  для всех кривых не будут близки к 0 и 1, соответственно, а максимальное отклонение  $d$  кривых из набора от кривых из построенного патча не будет превышать заданного порога.

Описанный итерационный процесс позволяет подогнать границы межстрочных кривых под линию, характеризующую вертикальное искажение страницы (см. рис. 6).

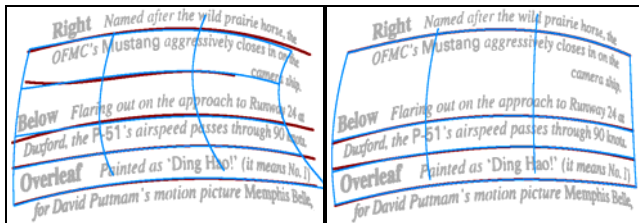


Рис. 6. Построение патча на изображении без подгонки и после подгонки.

## 11. РЕЗУЛЬТАТЫ ЭКСПЕРИМЕНТОВ

Для тестирования алгоритма был проведен следующий эксперимент. Был взят набор из 31 черно-белых искривленных изображений текста. Каждое изображение было распрямлено по описанному в статье алгоритму.

Результаты распознавания исходного и распрямленного изображений приведены в таблице 1.

Число изображений	31
Число строк на изображениях	1020
Число символов на изображениях	47363
Количество неправильно распознанных символов на исходных изображениях	2721 (4,92%)
Количество неправильно распознанных символов на распрямленных изображениях	830 (1,50%)
Количество исправленных ошибок	1891 (69,5%)

Таблица 1. Сравнение результатов распознавания изображений до и после распрямления строчек.

Полученные результаты показывают высокую эффективность предложенного алгоритма. Но, к сожалению, качество распознавания распрямленных документов все еще не идеальное. На некоторых страницах, где искривление строк было не очень существенно, распрямление изображения привело к ухудшению качества. Это произошло из-за того, что при распрямлении текстовых строк немного повреждаются текстовые символы. Однако на изображениях с большим искривлением строк предложенный механизм дает очень существенное улучшение качества.

## 12. БУДУЩИЕ РАБОТЫ

Существующий метод распрямления изображения повреждает символы при распрямлении строчек – это основная причина ошибок при распознавании текстов после применения предложенного алгоритма. Соответственно, основным направлением дальнейшей работы будет совершенствование метода распрямления изображения на основе имеющейся модели искажения страницы.

Использование итерационных процессов в коммерческих профессиональных системах распознавания и обработки изображений может привести к проблемам из-за непредсказуемости результатов такого механизма. Разработка методов нахождения вертикальной деформации страницы без использования итерационных методов или, по меньшей мере, повышение надежности разработанного метода будет еще одним существенным направлением дальнейшей работы.

Еще одним направлением дальнейших работ является разработка алгоритмов автоматического определения параметров предобработки и кластеризации скелета.

Параметр заливки изображения ( $n_{\min}$ ) в текущей реализации выставляется исходя из разрешения картинки, однако для изображений с цифровых камер разрешение – очень нечеткий параметр. Для надежной кластеризации ребер скелета нужно разработать механизм автоматического определения параметров кластеризации ( $L_{\min}$ ,  $A_{\max}$ ). Возможно, из-за очень простой структуры кластеризуемых данных подойдет любой из простейших существующих механизмов автоматической кластеризации, однако эксперименты в этом направлении еще не проводились.

### 13. ЗАКЛЮЧЕНИЕ

В данной статье описан алгоритм построения аппроксимации искажения страницы в виде патча Безье на основе непрерывного гранично-скелетного представления изображения. Метод также позволяет распрямить искаженное изображение на основе построенной аппроксимации. Также описан итерационный метод, позволяющий аппроксимировать не только искажение горизонтальных строк текста, но и искажение вертикальных границ страницы.

Проведенные эксперименты показали достаточно высокую эффективность предложенного метода. Однако алгоритм требует ряд усовершенствований, прежде чем его можно будет применять в коммерческих системах распознавания.

### 14. БИБЛИОГРАФИЯ

[1] Местецкий Л.М., Масалович А.А.; *Распрямление текстовых строк на основе непрерывного гранично-скелетного представления изображений; Труды международной конференции "Графикон-2006", Новосибирск, 2006*

[2] Местецкий Л.М.; *Скелет многосвязной многоугольной фигуры; Труды международной конференции "Графикон-2005", Новосибирск, 2005*

[3] Местецкий Л.М.; *Непрерывный скелет бинарного растрового изображения; Труды международной конференции "Графикон-98", Москва, 1998*

[4] Hironori Ezaki, Seiichi Uchida, Akira Asano, Hiroaki Sakoe, "Dewarping of document images by global optimization", Proceedings of international conference ICDAR, 2005.

[5] Udrian Ulges, Christoph H. Lampert, Thomas M. Breuel, "A Fast and Stable Approach for Restoration of Warped Document Images", Proceedings of international conference ICDAR, 2005.

[6] Li Zhang, Chew Lim Tan, "Warped Image Restoration with Application to Digital Libraries", Proceedings of international conference ICDAR, 2005.

[7] Yamashita, A. Kawarago, T. Kaneko, K.T. Muira, "Shape reconstruction and image restoration for non-flat surfaces of documents with a stereo vision system", Proceedings of international conference ICPR, 2004

[8] M.S. Brown, W.B. Seales, "Document restoration Using 3D shape: A General Deskewing Algorithm for Arbitrary Warped Documents". Proceedings of international conference ICCV, 2001.

Об авторах

Антон Андреевич Масалович – аспирант факультета вычислительной математики и кибернетики Московского Государственного Университета им. М.В. Ломоносова

Адрес: Москва, 119899, Воробьевы горы, МГУ, 2-й учебный корпус, факультет ВМиК, кафедра математических методов прогнозирования.

E-mail: [anton\\_m@abbyy.com](mailto:anton_m@abbyy.com)

Леонид Моисеевич Местецкий – доктор технических наук, профессор факультета вычислительной математики и

кибернетики Московского Государственного Университета им. М.В. Ломоносова.

Адрес: Москва, 119899, Воробьевы горы, МГУ, 2-й учебный корпус, факультет ВМиК, кафедра математических методов прогнозирования.

E-mail: [l.mest@ru.net](mailto:l.mest@ru.net)

## Text document images deformation approximation using Bezier patches

### Abstract

In this work a novel method of text document image deformation approximation is described. Proposed algorithm based on approximation of continuous border-skeletal representation of the image with Bezier patch.

In this article all main steps of approximation algorithm is described: preprocessing of the image, building of continuous border-skeletal representation of the image, extracting and approximation of long interlinear branches from skeleton, building of whole image approximation with Bezier patch, iteration process of adjusting Bezier patch.

Algorithm of image de-warping using deformation approximation is also described in this paper.

Experimental results are given, which prove efficiency of proposed algorithm for OCR systems.

Disadvantages of method and directions of future works are also mentioned.

**Keywords:** text recognition, OCR, image preprocessing, warped image restoration, continuous border-skeletal representation, outer skeleton, Bezier patch.

### About the authors

Anton Masalovitch is a Ph.D. student at Moscow State University, Faculty of Computational Mathematics and Cybernetics. His contact email is [anton\\_m@abbyy.com](mailto:anton_m@abbyy.com).

Leonid Mestetskiy is a professor at Moscow State University, Department of Mathematical methods of Forecast. His contact email is [l.mest@ru.net](mailto:l.mest@ru.net).