Федеральное государственное автономное
образовательное учреждение высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт математики и фундаментальной информатики
Базовая кафедра вычислительных и информационных технологий

УТВЕРЖДАЮ
Заведующий кафедрой
_____ / В.В.Шайдуров
(подпись)
«___» _____2019 г.

БАКАЛАВРСКАЯ РАБОТА

Направление 02.03.01 Математика и компьютерные науки

ПРИМЕНЕНИЕ МЕТОДОВ РАСПОЗНАВАНИЯ ТЕКСТА НА ПРАКТИКЕ

Научный руководитель
кандидат физико-математических наук,
доцент базовой кафедры ВИиТ _____/ С.Н.Баранов

Выпускник _____/ Ю.А.Машуков

Красноярск 2019

Федеральное государственное автономное
образовательное учреждение высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт математики и фундаментальной информатики
Базовая кафедра вычислительных и информационных технологий

УТВЕРЖДАЮ
Заведующий кафедрой
_____ / В.В. Шайдуров

«___» _____2018 г.

ЗАДАНИЕ
НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ РАБОТУ
в форме бакалаврской работы

Студенту Машукову Юрию Александровичу.

Группа ИМ15-05б, направление 02.03.01 «Математика и компьютерные науки».

Тема выпускной квалификационной работы: " Применение методов распознавания текста на практике".

Утверждена приказом по университету №7950/с.

Руководитель ВКР: С.Н.Баранов, кандидат физико-математических наук, доцент базовой кафедры вычислительных и информационных технологий ИМФИ СФУ.

Исходные данные для ВКР – задание на бакалаврскую работу от базовой кафедры вычислительных и информационных технологий ИМФИ СФУ.

Перечень разделов ВКР:

- Сегментация текста на изображении;
- Подготовка изображений символов к распознаванию;
- Распознавание символов;

Перечень графического материала: презентация «Применение методов распознавания текста на практике».

Руководитель ВКР _____ С.Н.Баранов

Задание принял к исполнению _____ Ю.А.Машуков

«___ » _____ 2018 г.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

Сейчас имеется немалое количество печатных книг, которых не существует в электронном виде. Для того, чтобы печатную книгу привести в электронный вид, её необходимо оцифровать. То есть отсканированную страницу необходимо пропустить через специальные алгоритмы которые выделят каждый символ текста, распознают его, и представят в цифровом виде.

Цель бакалаврской работы – построить алгоритмы по распознаванию текста на языке программирования C#, и выявить тот, у которого наибольшая точность.

Для достижения цели поставлены следующие задачи:

- Построить алгоритм сегментации текста на изображении;
- Написать алгоритмы для подготовки символов к распознаванию;
- Построение алгоритмов по распознаванию;
- Анализ работы алгоритмов, относительно точности распознавания.

1. ОБРАБОТКА ИЗОБРАЖЕНИЯ

Исходное изображение изначально правильно ориентированно. Необходимо вычленить все символы из текста на изображении. Сделать это можно с помощью алгоритма сегментации.

1.1. Сегментация

Алгоритм, позволяющий выделить строки/слова/символы в изображении. Для его реализации необходимо знать такую характеристику каждого пикселя изображения, как его яркость.

Работа с цветом пикселя в C# осуществляется с помощью цветовой модели RGB, чтобы узнать яркость пикселя, необходимо умножить каждый цветовой канал на соответствующий коэффициент и сложить их:

$$Y = 0.3 * R + 0.59 * G + 0.11 * B \qquad (1)$$

Яркость варьируется от 0 до 255. Для определенности считаем, что значение 0 – черный, а 255 – белый.

1.1.1. Сегментация по строкам

Сегментацию по строкам реализует процедура Lines. Формируем список *yar*, в котором хранится яркость (1) каждого пикселя по строке. Знаем, что яркости пикселей в строках, где присутствуют символы текста, отличаются от строк, где символов нет («пустые»). Опытным путем была подобрана константа, для используемого условия по отделению нужных строк от «пустых»:

$$if\ ((yar.Max() \ / \ 1.5) \ > \ yar.Min())$$

Индексы нужных строк формируют список *index*. В этом списке находятся индексы всех строк, в которых есть символы текста, но нам нужно узнать индексы строк, которые находятся на границе с «пустыми» строками. Используем следующее условие:

$$if\ ((index[i] - index[i - 1]) > 2),$$

т.е. те индексы строк, хранящиеся в списке *index*, которые отличаются друг от друга более чем на 2, соответствуют строкам, граничащим с «пустыми» строками. Сохраним эти индексы в список *result*. Первый, и последний индексы списка *index* тоже добавляем в *result*. На основании списка *result* сохраняем выделенные строки.

1.1.2. Сегментация строк по словам

Сегментацию строк по словам реализует процедура Words. Подгружаем поочередно уже выделенные строки. Формируем список *yar*, в котором хранится яркость (1) каждого пикселя по столбцу. Знаем, что яркости пикселей в столбцах, где присутствуют символы текста, отличаются от столбцов, где символов нет («пустые»). Пользуемся тем же условием, для отделения нужных столбцов от «пустых», которым пользовались в сегментации по строкам для выделения нужных строк:

$$if\ ((yar.Max() / 1.5) > yar.Min())$$

Индексы нужных столбцов формируют список *index*. В этом списке находятся индексы всех столбцов, в которых есть символы текста, но нам нужно узнать индексы столбцов, которые находятся на границе с «пустыми» столбцами. Используем следующее условие:

$$if\ ((index[i] - index[i - 1]) > 4),$$

т.е. те индексы столбцов, хранящиеся в списке *index*, которые отличаются более чем на 4, соответствуют столбцам, граничащим с «пустыми» столбцами.

Сохраним эти индексы в список *result*. Первый, и последний индексы списка *index* тоже добавляем в *result*. На основании списка *result* сохраняем выделенные слова.

1.1.3. Сегментация слов по символам

Сегментацию слов по символам реализует процедура Simbols. Подгружаем уже выделенные слова. Формируем список *yar*, в котором хранится яркость (1) каждого пикселя по столбцу. Знаем, что яркости пикселей в столбцах, где присутствуют символы текста, отличаются от столбцов, где символов нет («пустые»). Пользуемся тем же условием, для отделения нужных столбцов от «пустых», которым пользовались в сегментации строк по словам:

$$if\ ((yar.Max() \ / \ 1.5)\ >\ yar.Min())$$

Индексы нужных столбцов формируют список *index*. В этом списке находятся индексы всех столбцов, в которых есть символы текста, но нам нужно узнать индексы столбцов, которые находятся на границе с «пустыми» столбцами. Используем следующее условие:

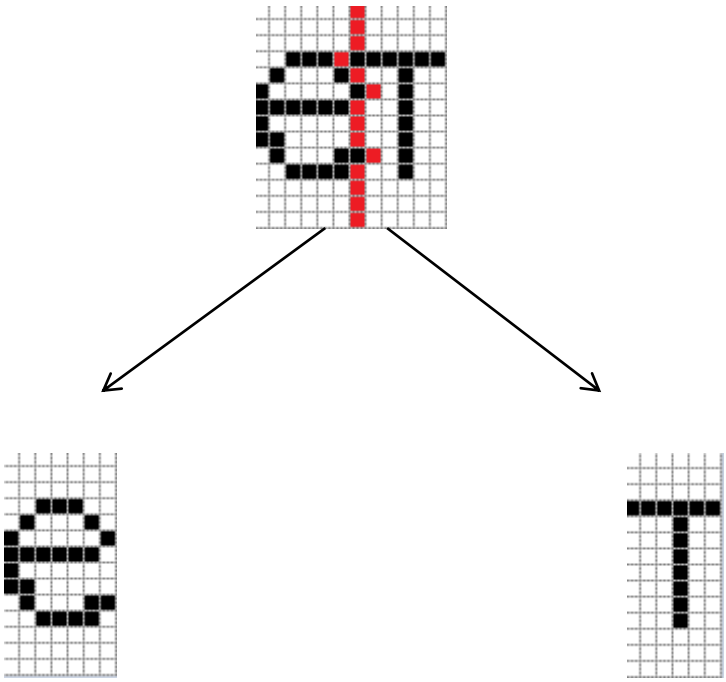
$$if\ ((index[i]\ -\ index[i\ -\ 1])\ >\ 1),$$

т.е. те индексы столбцов, хранящиеся в списке *index*, которые отличаются более чем на 1, соответствуют столбцам, граничащим с «пустыми» столбцами. Сохраним эти индексы в список *result*. Первый, и последний индексы списка *index* также добавляем в *result*.

Теперь выделенные символы нужно привести к черно-белому виду. За это отвечает процедура Monochrom, которая основывается на рассмотрении условия: если яркость пикселя (1) больше порога (взята константа 170), то пиксель будет белым, иначе черным.

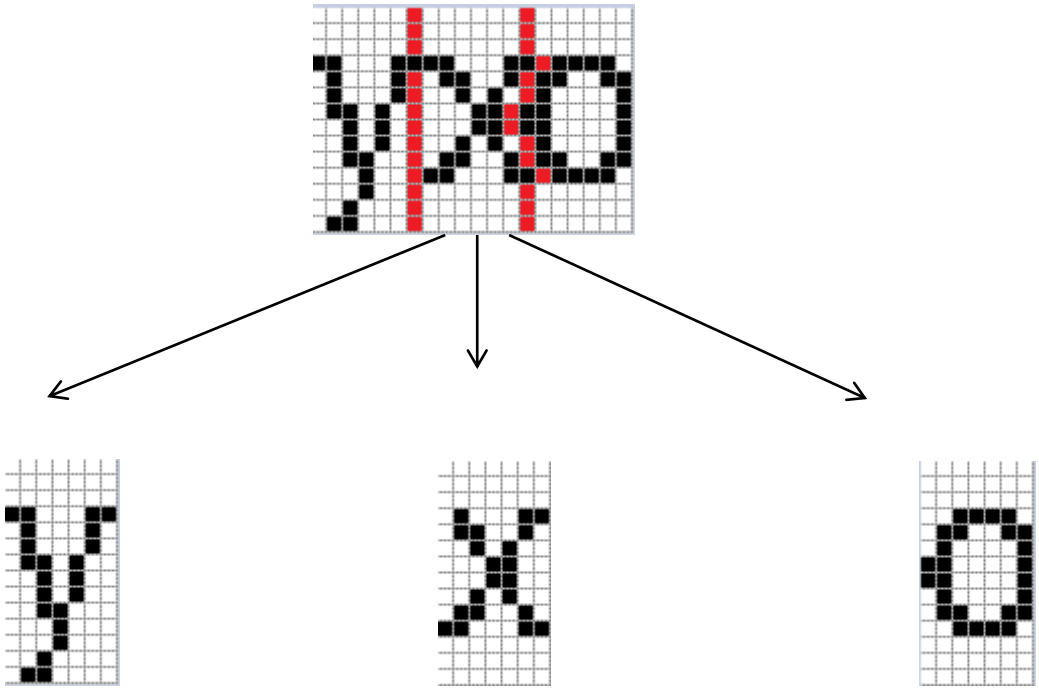
К сожалению не всегда между символами есть «пустой» столбец, необходимо обработать такие случаи. При обработке изображения было выявлено, что возможны следующие случаи:

- слиплись два символа



Начальным положением красной границы, по которой нужно отделить символы, берется середина верхней строки пикселей, которой соответствует индекс $(j; 0)$. Далее при спуске сравниваем значения яркости трёх пикселей с индексами $(j-1; 0), (j, 0), (j+1; 0)$, берется тот, у которого наибольшая яркость. Если более одного из рассматриваемых пикселей имеют белый цвет, то берется тот, что ближе к $(j; 0)$, это необходимо, чтобы граница не ушла в сторону.

- слиплись три символа



Начальными положениями красных границ, по которым отделяем символы, берутся треть и две трети верхней строки пикселей, которым соответствуют индексы $(j_1; 0)$ и $(j_2; 0)$. Далее при спуске сравниваем значения яркости следующих пикселей с индексами $(j_1 - 1; 0)$, $(j_1; 0)$, $(j_1 + 1; 0)$, и $(j_2 - 1; 0)$, $(j_2; 0)$, $(j_2 + 1; 0)$. Берутся те, у которых наибольшая яркость для соответствующих границ. Если более одного из рассматриваемых для 1-ой либо 2-ой границ пикселей имеют белый цвет, то берется тот, что ближе к $(j_1; 0)$ или $(j_2; 0)$ соответственно. Это необходимо, чтобы границы не ушли в сторону. На этом сегментация исходного изображения завершена, у нас есть черно-белые изображения каждого символа.

СПИСОК ЛИТЕРАТУРЫ

1.Гонсалес Р. Цифровая обработка изображений / Р. Гонсалес, Р. Вудс. – М. : Техносфера. – 2005. – С. 1007.

2. Журавлев Ю.И. Об алгебраическом подходе к решению задач распознавания или классификации // Проблемы кибернетики. М.: Наука, 2005. - Вып. 33.

3. Алгоритмы выделения контуров изображений [Electronic resourse] / Интернет-ресурс. – Режим доступа: http://habrahabr.ru/post/114452/. – Загл. с экрана.

4. Распознавание текстовых изображений [Electronic resourse] / Интернет-ресурс. – Режим доступа: https://www.graphicon.ru/html/2013/papers/250-253.pdf/. – Загл. с экрана.

Федеральное государственное автономное
образовательное учреждение высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт математики и фундаментальной информатики
Базовая кафедра вычислительных и информационных технологий

УТВЕРЖДАЮ
Заведующий кафедрой
_____ / В.В.Шайдуров

«17» июня 2019 г.

БАКАЛАВРСКАЯ РАБОТА

Направление 02.03.01 Математика и компьютерные науки

ПРИМЕНЕНИЕ МЕТОДОВ РАСПОЗНАВАНИЯ ТЕКСТА НА ПРАКТИКЕ

Научный руководитель
кандидат физико-математических наук,
доцент _____ / С.Н.Баранов
17.06.19

Выпускник _____ / Ю.А.Машуков
17.06.19

Красноярск 2019