

АВТОМАТИЗАЦИЯ УЧЕТА ПОСЕЩАЕМОСТИ ЗАНЯТИЙ

И.Л.Ковалева, А.С. Навоева, О.Э. Ошуковская

Белорусский национальный технический университет, г.Минск, Беларусь

Практика преподавания в вузах показывает, что ритмичность работы студентов является одним из важнейших условий достижения ими качественного освоения учебного материала. Кроме того, согласно Кодекса об образовании РБ, студент может быть отчислен из вуза в случае пропусков учебных занятий без уважительной причины. Поэтому анализ посещаемости является одним из направлений работы преподавателей, старост и работников деканатов.

Большинство современных систем управления обучением (LMS-систем) и систем дистанционного обучения (СДО-систем) реализуют функционал, позволяющий отслеживать посещаемость студентов. Однако во многих вузах республики до настоящего времени сохраняется практика обязательного использования журналов групп, заполняемых старостами групп, и журналов преподавателей, с помощью которых осуществляется контроль проведения и посещения занятий.

В БНТУ, наряду с журналами старост и журналами преподавателей, внедрена система электронного журнала. Однако в электронном журнале не предусмотрена возможность автоматического ввода данных о пропусках из журналов групп или журналов преподавателей. Ручной же перенос информации требует временных затрат, может приводить к потере информации. К тому же старосты и преподаватели не всегда вовремя заполняют электронный журнал.

Поэтому возникает проблема актуального и качественного заполнения данных в электронном журнале. Чтобы сделать работу старост и преподавателей менее рутинной, предлагается автоматизировать ввод информации о пропусках занятий в электронный журнал. В предлагаемом подходе старостам и преподавателям нужно только отсканировать требуемую страницу журнала группы и загрузить полученное изображение в программу. В программе реализуется ряд операций по предварительной обработке, сегментации и анализу загруженного изображения.

На этапе предварительной обработки изображений страниц журналов выполняются бинаризация и фильтрация сканированных изображений. Для этих операций в процессе настройки и тестирования программы были подобраны локальные параметры.

Данные о пропусках в журналах размещены в ячейках таблиц. Поэтому на первом этапе сегментации выделяются линии, образующие ячейки таблиц, и вычисляются координаты их пересечений (рисунок 1).

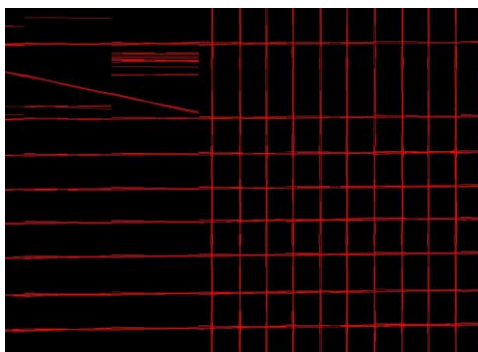


Рис. 1 – Визуализация этапа «Сегментация линий»

Для сегментации линий используется алгоритм Hough [1]. Была выполнена настройка этого алгоритма, в результате которой удалось определить параметры, позволяющие выделять требуемые фрагменты таблицы. На основании полученных данных далее последовательно сегментируются ячейки таблицы.

Для обоснования и разработки алгоритма распознавания данных в ячейках была проанализирована информация, которая заносится в ячейки старостами и преподавателями. Оказалось, что в ячейки могут быть либо пустыми, либо в ячейках могут храниться цифры «1» или «2». Сложность автоматического распознавания этих цифр состоит в том, что цифры – рукописные символы, написанные разными почерками. Поэтому для распознавания цифр были разработаны два алгоритма, параллельное использование которых повышает качество распознавания.

В первом алгоритме при распознавании учитывается площадь символа в ячейке. Как видно из рисунка 2, для изображения цифры «2», как правило, требуется больше пикселей, чем для изображения цифры «1». Поэтому площадь цифры «2», как правило, больше, чем площадь цифры «1». Был подобран порог по площади для разделения «1» и «2».

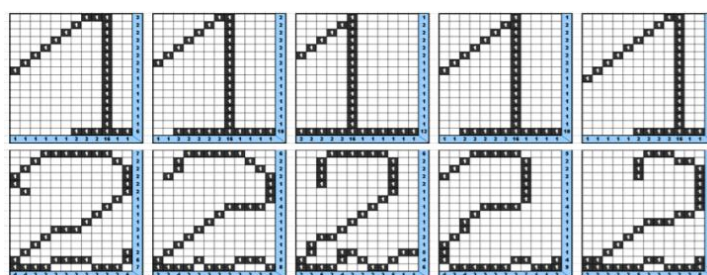


Рис.2 – Различные способы написания цифр

Второй алгоритм распознавания построен с использованием так называемого «метода зондов», в котором распознавание ведется по числу пересечений контура цифр с прямыми (зондами), расположенными специальным образом. На основании тестовых данных был сформирован набор зондов для распознавания рукописных символов «1» и «2» (рисунок 3).

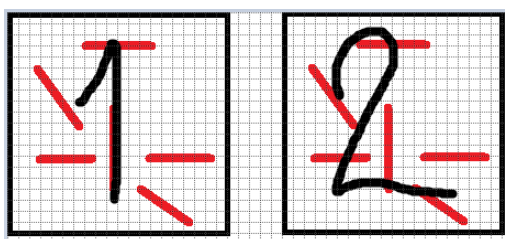


Рис.3 - Схема расположения зондов для распознавания цифр

Программа реализована на языке Python, используются библиотеки OpenCV и Numpy.

Литература

1. Преобразование Хафа [Электронный ресурс]: Википедия. Свободная энциклопедия. – Режим доступа: https://ru.wikipedia.org/wiki/Преобразования_Хафа (дата обращения: 05.05.2021).