

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ ФЕДЕРАЛЬНОЕ
ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
Национальный исследовательский ядерный университет «МИФИ»
Институт Лазерных и Плазменных Технологий
Кафедра № 97 «Суперкомпьютерное моделирование
инженерно-физических процессов»

ОТЧЁТ
Учебная практика
(научно-исследовательская работа) на
тему:

Подготовка базы свойств материалов на основе открытых источников

Работу
выполнил:
А. Е. Диков
Группа: Б20-221
Научный
руководитель:
Д. П. Макаревич

Москва
2023

Содержание

1. Введение	3
1.1. Актуальность проблемы	3
1.2. Трудности реализации задачи	3
1.3. Используемые технологии	3
1.3.1. Обработка изображений с использованием OpenCV	3
1.3.2. Распознавание текста с помощью библиотеки Tesseract	4
1.3.3. Использование нейронных сетей	4
2. Постановка задачи	5
3. Алгоритмы, технологии и программный код	6
3.1. Обработка входных данных	6
4. Результаты	7
5. Заключение	8

1. Введение

1.1. Актуальность проблемы

В современном информационном обществе существует огромное количество данных, которые хранятся в различных форматах, включая оцифрованные изображения документов. Вместе с тем, большое количество информации представлено в виде таблиц, которые являются основным средством структурирования и организации данных. Однако, извлечение данных из изображений таблиц является нетривиальной задачей, которая требует автоматизации и использования специализированных технологий.

Проблема извлечения данных из изображений таблиц имеет высокую актуальность в различных сферах деятельности, таких как научные исследования, финансовый анализ, медицина, юридические и бухгалтерские отчеты и многие другие. Сфокусируем внимания на применении в научной сфере, а именно, для сбора данных о параметрах материалов, используемых в математическом моделировании. Автоматическое распознавание и извлечение данных из изображений таблиц позволяет значительно ускорить и упростить процесс обработки информации и повысить эффективность работы.

1.2. Трудности реализации задачи

Задача извлечения данных из изображений таблиц является сложной и требует решения нескольких проблем. Во-первых, требуется точное обнаружение и сегментация таблицы на изображении. Это включает в себя предобработку изображений, выделение контуров, определение границ и разделение на ячейки.

Во-вторых, необходимо правильно распознать текст и числа внутри каждой ячейки таблицы. Распознавание текста визуальным образом представленного на изображении является сложной задачей из-за различных шрифтов, размеров, стилей и размещения текста в ячейках таблицы.

В-третьих, требуется структурировать распознанные данные в виде таблицы с соответствующими заголовками столбцов и строками. Иными словами, сохранить общий вид исходной таблицы.

1.3. Используемые технологии

Для решения задачи извлечения данных из изображений таблиц используются различные технологии и методы. Ниже приведены некоторые из них:

1.3.1. Обработка изображений с использованием OpenCV

OpenCV (Open Source Computer Vision Library) является библиотекой с открытым исходным кодом на языке C++, предназначенной для обработки изображений и компьютерного зрения. Она предоставляет мощные инструменты для обработки и анализа изображений, включая выделение контуров, сегментацию, фильтрацию и морфологические операции над изображениями. На данный момент библиотека OpenCV — это множество модулей для разных целей:

1. математические функции и вычисления, алгебра и структуры данных;
2. модели для машинного обучения;
3. ввод и вывод изображений или видео, чтение и запись в файл;

4. обработка изображения;
5. распознавание примитивов;
6. детектирование объектов — лиц, предметов и других;
7. отслеживание и анализ движений на видео;
8. обработка трехмерной информации;
9. алгоритмы ускоряющие работу библиотеки;
10. устаревший и разрабатываемый код;

1.3.2. Распознавание текста с помощью библиотеки Tesseract

Tesseract - это библиотека с открытым исходным кодом для распознавания текста на изображениях. Она использует алгоритмы машинного обучения для распознавания текста и может работать с различными языками. Tesseract обладает хорошей точностью распознавания и широкими возможностями настройки.

Ядро Tesseract было разработано в Бристольской лаборатории Hewlett Packard в середине 1980-х и поддерживалось до середины 1990-х, а затем 10 лет никакой поддержки не было. С 2006 г. Google купил Tesseract и открыл исходные тексты под лицензией Apache 2.0 для продолжения разработки. В настоящий момент программа поддерживает кодировку UTF-8 и множество языков, включая русский.

1.3.3. Использование нейронных сетей

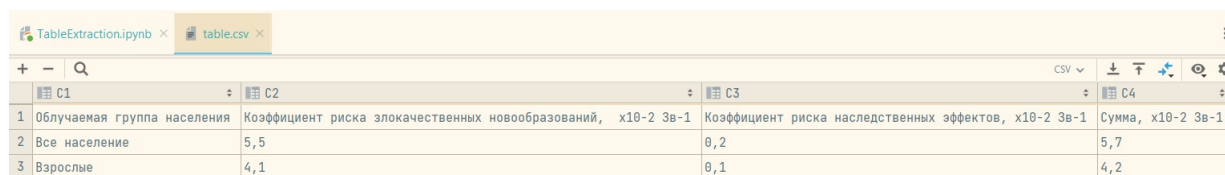
Нейронные сети, такие как Faster R-CNN, SSD, YOLO и RetinaNet, являются мощными инструментами для обнаружения объектов и сегментации на изображениях. Они используют алгоритмы глубокого обучения для автоматического обнаружения и распознавания объектов, включая таблицы и их компоненты. [пока не функционируют, из использовние в планах]

2. Постановка задачи

В качестве набора данных были предоставлены ГОСТ, технические условия и справочники в различных форматах - rtr, djvu, pdf. В дополнению к этому был сгенерирован еще один набор данных, имеющий меньшую вариативность наполнения, но имеющий метки - файлы в более удобном формате (был выбран csv) для оценки качества работы алгоритма (размеченные данные). Задача состояла в выделении таблицы из корпуса текста в виде изображений. Затем нужно было распознать таблицу с помощью алгоритмов компьютерного зрения так, чтобы сохранилась общая структура таблицы. Необходимо было создать универсальную и масштабируемую систему, точно работающую для всех входных файлов.

Облучаемая группа населения	Коэффициент риска злокачественных новообразований, $\times 10^{-2}$ Зв-1	Коэффициент риска наследственных эффектов, $\times 10^{-2}$ Зв-1	Сумма, $\times 10^{-2}$ Зв-1
Все население	5,5	0,2	5,7
Взрослые	4,1	0,1	4,2

Рисунок 2.1. Пример таблицы (взят из НРБ 99/2009)



C1	C2	C3	C4
1	Облучаемая группа населения	Коэффициент риска злокачественных новообразований, $\times 10^{-2}$ Зв-1	Коэффициент риска наследственных эффектов, $\times 10^{-2}$ Зв-1
2	Все население	5,5	0,2
3	Взрослые	4,1	0,1

Рисунок 2.2. Результат работы алгоритма

3. Алгоритмы, технологии и программный код

3.1. Обработка входных данных

Для решения задачи напомним класс на языке Python, полями которого будут подаваемый на вход файл, из которого нужно извлечь таблицы, и базы данных (`pandas.DataFrame`) в которых будет храниться результат работы.

Поскольку входной файл может иметь разные форматы, обработаем различные варианты.

Отсканированные изображения страниц документов в общем случае могут быть цветными, что затормозит работу алгоритмов поиска контуров и распознавания букв в дальнейшем, поэтому опустим информацию о цвете. Более того обросим и оттенки серого, тем самым бинаризовав изображение

Затем будем рассматривать документ как набор черно-белых изображений и все сказанное ниже будем применять для каждой страницы документа. Для нахождения таблицы на странице сделаем копию изображения и применим к ней морфологическое сверточное преобразование. Оно размоет границы, замкнет линии и уменьшит возможные шумы, возникающие в результате сканирования.

Применим к предобработанному изображению метод `findContours()`. Внутри метода `findContours()` в OpenCV используется алгоритм обхода границы объектов, известный как алгоритм Сазерленда-Ходжмана (Suzuki and Abe's Algorithm).

Алгоритм Сазерленда-Ходжмана работает следующим образом:

Алгоритм работает с бинаризованным изображением

Поиск начальной точки границы: Начальная точка границы объекта выбирается на белом пикселе, который ещё не был посещен. Она может быть найдена путём сканирования изображения слева направо и сверху вниз.

Следование по границе объекта: Алгоритм следует по границе объекта, перемещаясь от текущей точки к следующей, пока не вернётся в исходную точку. Каждая точка границы добавляется в список контура.

Маркировка посещенных точек: Каждая посещенная точка на границе объекта маркируется или удаляется из изображения, чтобы не повторно посещать её при обходе других контуров.

Повторение для всех объектов: Процесс повторяется для оставшихся объектов на изображении, пока не будут найдены все контуры.

Алгоритм Сазерленда-Ходжмана обеспечивает эффективное выделение контуров объектов на изображении. Он поддерживает различные режимы поиска контуров и методы аппроксимации, что позволяет гибко настраивать процесс обнаружения контуров в зависимости от конкретной задачи.

4. Результаты

5. Заключение