



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА
К ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЕ
НА ТЕМУ:

**«Классификация документов, удостоверяющих
личность, по фотографии»**

Студент ИУ7-83Б
(Группа)

(Подпись, дата)

А.Г. Сиденко
(И.О.Фамилия)

Руководитель ВКР

(Подпись, дата)

Д.С. Бабарыкин
(И.О.Фамилия)

Консультант

(Подпись, дата)

(И.О.Фамилия)

Нормоконтролер

(Подпись, дата)

(И.О.Фамилия)

2021 г.

РЕФЕРАТ

Отчёт содержит 29 страниц, 5 рисунка, 3 таблицы, 37 источников,
1 прил.

СОДЕРЖАНИЕ

Введение	4
1 Аналитический раздел	6
1.1 Постановка задачи	6
1.2 Существующие решения	6
1.3 Визуальное представление документа	9
1.3.1 Методы классификации	9
1.3.2 Выбор архитектуры нейронной сети	13
1.4 Обзор и анализ методов распознавания и классификации текстовой информации	15
1.4.1 Оптическое распознавание символов	16
1.5 Вывод	19
2 Конструкторский раздел	20
2.1 Функциональная модель	21
3 Технологический раздел	22
4 Исследовательский раздел	23
Заключение	24
Список использованных источников	25
Приложение А Картинки	29

ВВЕДЕНИЕ

Правильная организация работы с документами в наши дни имеет большое значение, так как от эффективности реализации документооборота напрямую зависит эффективность работы любой организации. Например, в рамках кредитных конвейеров юридических лиц банки запрашивают у компаний оригиналы различных документов. Для удобства использования их нужно классифицировать на отдельные документы, в том числе одностраничные или многостраничные [1].

Во многих интернет-сервисах, от платежных систем до схем восстановления доступа к учетным записям в социальных сетях, активно используются инструменты распознавания документов.

Например, в таких системах, как портал «Госуслуги» [2] и платежная система «Webmoney» [3], целью обработки документа является получение всей персональной информации о клиенте, например, в случае паспорта РФ это фамилия, имя, серия и номер паспорта, место и дата выдачи, код подразделения, машиночитаемая зона, и установление ее подлинности.

В социальных сетях, таких как «ВКонтакте» [4], для восстановления доступа к учетной записи пользователя, не требуется получать всю персональную информацию, указанную на странице документа, удостоверяющего личность пользователя. В большинстве случаев достаточно проверить изображение определенной части документа, содержащей некоторую окрестность фотографии пользователя. Данная окрестность должна полностью содержать фотографию пользователя, имя, фамилию.

Тип документа определяется как текстом, так и визуальной информацией. Например, паспорт или трудовую книжку легко различить визуально без анализа текста внутри. Более того, качество распознавания текста в таких документах достаточно низкое, если используются неспециализированные решения. Поэтому визуальная составляющая несет намного больше релевантной информации для классификации. Однако различные типы виз могут быть визуально похожи, однако текстовая информация, которую они содержат, отличается.

В результате задача классификации документов сводится к модели, которая должна объединить два источника неструктурированных данных: визуальное представление документа (устойчивые признаки для проверки изображения, удостоверяющего личность человека) и результаты распознавания текстовой информации.

Целью работы является разработка и исследование метода классификации документов по фотографии. Имеется множество документов написанных на естественном языке и множество заранее известных категорий. Требуется для каждого документа выбрать категорию, к которой он, в силу своего смыслового (семантического) содержания, относится с наибольшей долей уверенности.

Для достижения поставленной цели необходимо решить следующие задачи.

- Анализ существующих решений.
- Разработка метода классификации.
- Спроектировать и реализовать ПО, демонстрирующее работу метода.
- Провести исследование на применимость данного метода, его соответствие цели работы.

1 Аналитический раздел

В данном разделе рассматривается задача анализа существующих алгоритмов классификации.

1.1 Постановка задачи

Классификация документов – одна из задач информационного поиска, заключающаяся в определении документа к одной из нескольких категорий на основании его содержания.

Задача распознавания типа удостоверения личности на изображениях, рассматриваемая в данной работе, может быть сформулирована следующим образом.

Имеется множество документов написанных на естественном языке, и множество заранее известных классов документов. Изображение удостоверения личности Q должно быть отнесено к одному из классов $C = C_i, i \in [0, N]$ с определенной вероятностью.

C_0 – класс прочих изображений, соответствующий значению: «не является фотографией удостоверения личности, пригодной для идентификации пользователя».

1.2 Существующие решения

Необходимо рассмотреть существующие решения и целесообразность создания нового. На рынке представлено множество разнообразных систем, занимающихся распознаванием документов, как платных, так и бесплатных.

1. Smart ID Engine [5].
2. ABBYY [6].
3. IRIS [7].
4. Jumio [8].
5. Idmatch [9].

Проведем сравнение данных решений, представлено в таблице 1.1.

Таблица 1.1 — Сравнение существующих решений.

	Распознаваемые документы	Распознаваемые языки	Платформы	Доступность	Дополнительно
Smart ID Engine	Международные документы (паспорта, визы, водительские права)	Все основные	Android iOS Windows MacOSX	Платное	Сильно ориентируется на машинно-читаемую строку
ABBYY	Международные документы (паспорта, визы, водительские права)	Все основные	Android iOS	Платное	
IRIS	Документы США, международные паспорта и ID-карты	Нет поддержки русского	Windows MacOSX	Платное	
Jumio	Документы Европы и США	Нет поддержки русского	Android iOS	Платное	
Idmatch	Id-карты Кыргызской Республики	Киргизский и русский языки	Нет готового ПО	Бесплатное	

Вывод следует такой, что IRIS, Jumio, Idmatch – не идентифицируют необходимые документы. ABBYY – решение только для мобильных устройств. Smart ID Engine – идеальная система..

1.3 Визуальное представление документа

Далее необходимо рассмотреть этапы обработки изображения, а именно методы классификации по визуальным и текстовым признакам.

Для начала, рассмотрим способы классификации документов по визуальным признакам.

1.3.1 Методы классификации

Существует множество методов классификации, которые используют различный математический аппарат и различные подходы при реализации. Однако эффективность этих методов зависит от конкретной решаемой задачи. Несмотря на то, что последнее десятилетие коммерческие компании занимаются проблемой повышения качества машинного обучения, на сегодняшний день не существует методов, которые могли бы однозначно эффективно решить задачу классификации.

Можно выделить следующие типы методов классификации: вероятностные, метрические, логические, линейные, логическая регрессия. Обобщенно опишем некоторые из них, указывая преимущества и недостатки каждого из них.

1. Метод Байеса.

Метод Байеса (Naive Bayes, NB) относится к вероятностным методам классификации [10].

Пусть имеется множество классов $C = \{c_0, c_1, \dots, c_N\}$ документов. Согласно теореме Байеса вероятность того, что документ принадлежит классу c , имеет вид 1.1.

$$P(c|d) = \frac{P(c)P(d|c)}{P(d)} \quad (1.1)$$

Вероятность $P(d)$ не требует вычислений ввиду того, что его значение не зависит от класса c , а значит, не влияет на нахождение наибольшей вероятности.

Преимущества метода состоит в следующем:

- высокая скорость работы;
- поддержка инкрементного обучения;
- простая реализация алгоритма в виде программы,;
- легкая интерпретируемость результатов работы алгоритма.

Несмотря на приведенные достоинства, метод Байеса имеет так же и минусы в своей реализации.

- Относительно низкое качество классификации.
- Неспособность учитывать зависимость результата классификации от сочетания признаков являются главными недостатками этого метода.

2. Метод k ближайших соседей.

Метод k ближайших соседей (k Nearest Neighbors, KNN) относится к метрическим методам и считается простейшим классификатором [11]. Суть метода заключается в том, что документу d присваивается тот класс s , к которому принадлежит большинство из k ближайших соседей документа, вычисленных с помощью какой-либо метрики расстояния.

Достоинства данного метода:

- простая реализация;
- проработанная теоретическая база;
- адаптация под нужную задачу выбором метрики;
- интерпретируемость.

К недостаткам относятся:

- недостаточная производительность в реальных задачах, так как число соседей, используемых для классификации, будет достаточно большим;
- трудность в наборе подходящих весов и определением, какие признаки необходимы для классификации;

— зависимость от выбранной метрики расстояния между примерами.

3. Метод деревьев решений.

Метод деревьев решений (Decision Trees, DT) относится к логическим методам классификации [12]. Дерево решений представляет собой конечный связный граф с множеством вершин, по которому производится классификация документов, описанных набором признаков. В узлах (вершинах) дерева прописаны условия, после проверки которых выполнение алгоритма классификации продолжается по правому или левому поддереву рассматриваемого узла. Процедура повторяется в каждом посещенном узле до тех пор, пока очередной узел не окажется листом. У каждого узла столько ветвлений, сколько значений имеет выбранный признак.

Метод обладает рядом преимуществ, таких как:

- простота восприятия алгоритма;
- быстрота обучения;
- высокая точность прогнозирования.

Однако также присущи и недостатки, такие как:

- неустойчивость алгоритмов к выбросам;
- с увеличением размера обучающего множества увеличение затрачиваемого на построение дерева времени.

4. Метод опорных векторов.

Метод опорных векторов (Support Vector Machine, SVM) является линейным методом классификации, в настоящее время призван одним из лучших [13]. Главным принципом SVM является определение разделителя в искомом пространстве, который разделяет классы наилучшим образом.

Одно из преимуществ SVM-метода в том, что так как он пытается определить оптимальное направление разделения признакового пространства, рассматривая комбинации признаков, он достаточно устойчив к большим размерностям.

Потенциальные недостатки метода опорных векторов заключается в следующем: невозможность калибровки вероятности попадания в определенный класс, подходит только для решения задач с 2 классами, параметры модели сложно интерпретировать.

5. Линейный метод наименьших квадратов.

Одним из ранних применений регрессии к классификации текстов является линейный метод наименьших квадратов (ЛМНК) [14].

Как и в случае классификатора SVM, обучение регрессионной модели так же использует затратный оптимизационный процесс

Линейный метод наименьших квадратов показал себя устойчивым на практике.

6. Нейронные сети.

Нейронные сети состоят из набора «нейронов», которые являются преобразователями входных сигналов в выходные [15]. Преобразование задается весами сети, которые являются параметрами и могут изменяться. Выходные сигналы вычисляются как функция от входных сигналов. В общем случае нейронная сеть строится как соединение множества нейронов, объединенных в уровни, при этом выходы одного уровня являются входами следующего. Самая простая нейронная сеть состоит из одного слоя, однако круг решаемых такими сетями задач ограничен. Поэтому на практике часто используют нейронные сети, содержащие большее число слоев.

Преимуществом нейронных сетей является их способность аппроксимировать любую непрерывную функцию.

Однако, как показали многочисленные исследования, нейронные сети не лишены и недостатков. Существенным недостатком нейронных сетей является то, что результат классификации полностью зависит от начальных установок сети. Поэтому качество обучения сети тем выше, чем больше тестовых примеров участвует в обучении.

Оптимальным методом классификации являются **нейронные сети**.

Существует множество архитектур нейронных сетей, необходимо проанализировать и сравнить их.

1.3.2 Выбор архитектуры нейронной сети.

Задача классификации изображений является популярной и хорошо изученной. Выбор алгоритма классификации будет производиться на базе результатов международного соревнования ILSVRC [16] (ImageNet Large Scale Visual Recognition Challenge). Соревнование оценивает алгоритмы обнаружения объектов и классификации изображений. Его целью является выявление самых лучших с точки зрения точности и скорости алгоритмов обнаружения и классификации. Оценка алгоритмов производится с помощью выборки данных ImageNet [17]. Эта выборка представляет из себя большую визуальную базу данных, содержащую более 14 миллионов изображений, которые в свою очередь разбиваются на 21841 категорию.

Начиная с 2012 года данное соревнованию выигрывают сверточные нейронные сети (Convolutional Neural Networks), а именно AlexNet [18], ZFNet [19], VGGNet [20], GoogLeNet [21], ResNet [22], TrimpsNet (Не было никакого нового научного вклада, который оправдывал бы подготовку статьи, и по этой причине авторы TrimpsNet только поделились результатами) и SENet [23]. С 2015 года CNN превзошли человеческие показатели – 5.1% [24].

1. Архитектура AlexNet.

AlexNet была первой свёрточной нейросетью, выигравшей соревнование по классификации ImageNet в 2012 году. Архитектура AlexNet состоит из пяти свёрточных слоёв, между которыми располагаются pooling-слои и слои нормализации, а завершают нейросеть три полносвязных слоя.

На схеме архитектуры все выходные изображения делятся на два одинаковых участка – это связано с тем, что нейросеть обучалась на старых GPU GTX580, у которых было всего 3 Гб видеопамяти. Для обработки использовались две видеокарты, чтобы параллельно выполнять операции над двумя частями изображения.

2. Архитектура ZFnet.

В 2013 году нейросеть ZFnet смогла достичь результата 11.7% – архитектура AlexNet использовалась в качестве основы, но с изменёнными параметрами и слоями.

3. Архитектура VGGnet.

Появилась в 2014 году. Основная идея – использовать вместо больших сверток (11x11 и 5x5) маленькие свертки (3x3). С маленькими фильтрами мы получим не так много параметров, но при этом сможем гораздо эффективнее обрабатывать их.

4. Архитектура GoogleNet.

GoogleNet или Inception-v1 – ещё более глубокая архитектура с 22 слоями. Целью Google было разработать нейросеть с наибольшей вычислительной эффективностью. Для этого они придумали так называемый модуль Inception – вся архитектура состоит из множества модулей, следующих друг за другом.

5. Архитектура ResNet.

ResNet – это сокращенное название для Residual Network (дословно – «остаточная сеть»). Решает проблему деградации (простая сеть увеличивает частоту ошибок по мере того, как сеть углубляется), и после использования остатка, когда сеть углубляется, частота ошибок все еще может уменьшаться.

6. Архитектура SENet.

Squeeze-and-Excitation Networks (SENets) представляют собой специальный блок для свёрточной нейронной сети, который улучшает взаимозависимости каналов практически без дополнительных вычислений. Основная идея – добавить параметры к каждому каналу свёрточного блока, чтобы сеть могла адаптивно регулировать вес каждой карты признаков.

Результаты сравнения архитектур представлены в таблице 1.2.

Таблица 1.2 — Сравнение архитектур сверточных нейронных сетей.

Название сети [25]	Год	Ошибка Top-5	Количество обучаемых параметров [26]	Требуемое оборудование [27]
AlexNet	2012	16.4 %	60 миллионов	2 GPU
ZFNet	2013	11.7 %	60 миллионов	1 GPU
VGGNet	2014	7.3 %	138 миллионов	4 GPU
GoogleNet	2014	6.7 %	4 миллиона	CPU
ResNet	2015	3.5 %	25.6 или 1.7 миллиона	2 GPU
TrimpsNet	2016	2.99 %		
SENet	2017	2.25 %	27.5 миллионов	8 GPU

В соответствии с результатами ILSVRC последних лет, можно отметить, что наилучшей точностью классификации объектов на сегодняшний день обладает свёрточная нейронная сеть SENet. Однако этот алгоритм является очень требовательным к вычислительным ресурсам.

Принято компромиссное решение – использовать архитектуру нейронной сети GoogleNet, так как она дает хорошие показатели точности и не требует мощные вычислительные ресурсы.

1.4 Обзор и анализ методов распознавания и классификации текстовой информации

Второй составляющей метода является распознавание и классификация текстовой информации в документе.

Выше уже были рассмотрены методы классификации.

Таким образом, процесс делится на два связанных между собой этапа:

— этап преобразования изображения в вектор признаков, в процессе которого определяется наиболее полное и информативное представление изображения в виде числового вектора;

— этап классификации, в процессе которого проверяется гипотеза принадлежности изображения классу изображений объекта на основании наблюдения (вектор признаков).

1.4.1 Оптическое распознавание символов

Преобразованием графического изображения в текст занимаются специальные программы распознавания текста (Optical Character Recognition - OCR).

Основной принцип автоматического распознавания образов – это обучение машины определению всевозможных эталонных образцов, с которыми будет сравниваться распознаваемый объект. В OCR-системах это буквы, цифры, знаки препинания. Обучение осуществляется за счет того, что машине показываются образцы символов различных классов. На основании этих образцов машина формирует прототип описания каждого класса объектов. Затем в процессе распознавания неизвестные символы сравниваются с заранее полученными образцами и определяется класс, с которым обнаружено больше всего совпадений.

Системы оптического распознавания текста – OCR-системы (optical character recognition) технология, которая позволяет преобразовывать различные типы документов, такие как отсканированные документы, PDF-файлы или фото с цифровой камеры, в редактируемые форматы с возможностью поиска [28].

OCR – это одно из направлений компьютерного зрения.

Современные системы оптического распознавания можно разделить на коммерческие и свободно распространяемые системы с открытыми исходными кодами.

В контексте задач массовой оцифровки интерес представляют как коммерческие системы по причине высокого качества, так и открытые системы по причине своей доступности и гибкости в настройке. Поскольку целью данной работы является обработка русскоязычных документов, то интерес представляют системы с поддержкой распознавания русского языка.

ABBYY FineReader [29] – это программное обеспечение оптического распознавания символов (OCR). Для распознавания текста поддерживается до 190 языков. OCR или распознавание текста, использует интеллектуальные алгоритмы, которые преобразуют изображения в редактируемый текст, сохраняя исходный макет и форматирование исходного документа. Является признанным лидером на рынке. Распространяется на коммерческой основе.

IRIS Readiris [30] – это программа для преобразования документов в различные форматы. Распознает более 130 языков, включая русский, принимает файлы и сохраняет результаты во всех возможных форматах. Программное обеспечение платное.

Cuneiform [31] – свободно распространяемая открытая система оптического распознавания текстов российской компании Cognitive Technologies. CuneiForm позиционируется как система преобразования электронных копий бумажных документов и графических файлов в редактируемый вид с возможностью сохранения структуры и гарнитуры шрифтов оригинального документа в автоматическом или полуавтоматическом режиме. Система включает в себя две программы для одиночной и пакетной обработки электронных документов. Поддерживает все необходимые нам языки: английский, русский, испанский, итальянский, французский, немецкий. Кроме того, поддерживается смесь русского и английского языка.

Tesseract [32] – свободная компьютерная программа для распознавания текстов, разрабатывавшаяся Hewlett-Packard, а затем Google купил её и открыл исходные тексты под лицензией Apache 2.0 для продолжения разработки. Сегодня Tesseract считается одним из самых мощных решений с открытым исходным кодом (open-source) для распознавания данных со сканированных документов. Tesseract поддерживает более 100 различных языков, что делает его универсальным и широко распространённым решением во всём мире. Многие технологические компании используют Tesseract в основе для построения комплексных решений по интеллектуальной обработке данных.

OCRFeeder [33] – программа, предоставляющая графический интерфейс пользователя для систем оптического распознавания символов CuneiForm,

Tesseract, GOCR и Ocrad. OCRFeeder является свободно распространяемой программой для операционной системы Linux.

Выбор OCR систем основывался на сравнительном анализе их результатов по трем типам данных (среднее качество, высокое качество и очень высокое качество) [34]. Примеры изображений приведены на рисунках 1.1, 1.2, 1.3.

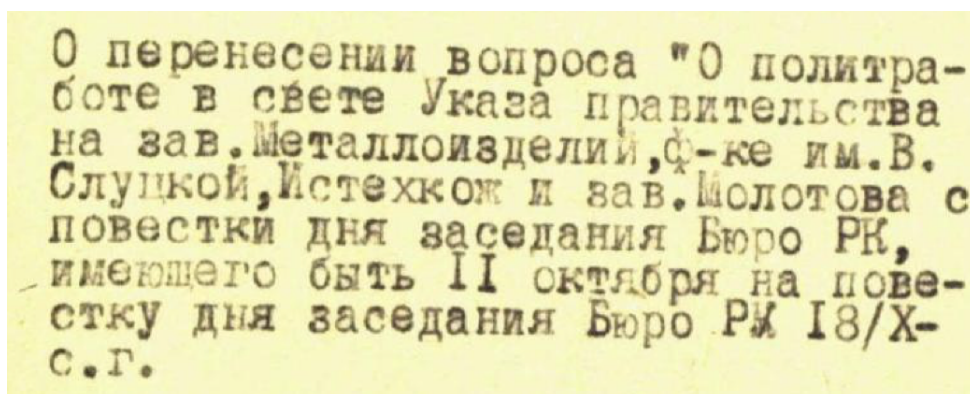


Рисунок 1.1 — Среднее качество.

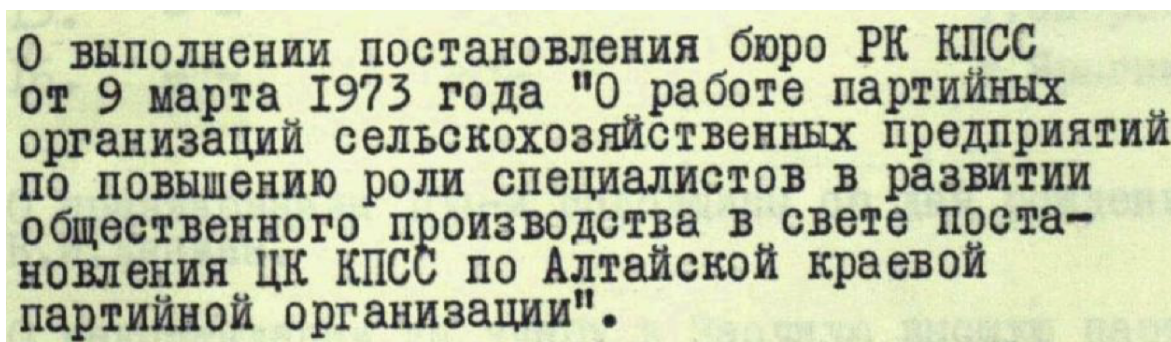


Рисунок 1.2 — Высокое качество.



Рисунок 1.3 — Очень высокое качество.

Качество результатов распознавания представлено в таблице 1.3.

Таблица 1.3 — Сравнение качества распознавания текста.

	Качество распознавания документов среднего качества	Качество распознавания документов высокого качества	Качество распознавания документов очень высокого качества	Доступность
ABBYY FineReader	76,8%	90,55%	99,25%	коммерческое
IRIS Readiris	22,01%	68,34%	94,40%	коммерческое
Cuneiform	0%	30,28%	90,20%	бюджетное
Tesseract	36,06%	74,68%	94,13%	бюджетное

Полученные результаты сравнительного анализа свидетельствуют о том, что коммерческая система «Abby Finereader» достигает максимального уровня качества. Среди свободно распространяемых систем наилучшей является Tesseract.

1.5 Вывод

В данном разделе описаны способы визуального сравнения изображений, распознавания и классификации текстовой информации.

2 Конструкторский раздел

Задача этого проекта – удостовериться, что пользователь отправляет валидный документ и дать ответ, какой тип документа он загрузил.

Данное программное обеспечение предоставляет возможность классификации документов следующих типов.

1. Паспорт гражданина Российской Федерации.
 - 1) Первая страница – когда и кем выдан.
 - 2) Вторая страница – фотография и личные данные.
 - 3) Третья страница – прописка.
2. Водительское удостоверение.
 - 1) Образец №1 – ламинированное бумажное с 1995 по 2011.
 - 2) Образец №2 – пластиковое с 1995 по 2011.
 - 3) Образец №3 – удостоверение нового образца.
3. Загранпаспорт гражданина Российской Федерации.
4. Шенгенская виза.
 - 1) Франция.
 - 2) Германия.
 - 3) Италия.
 - 4) Испания.

Классификатор получает входные данные: фотографию документа. На выходе классификатор сообщает, что он получил (паспорт, водительское удостоверение и так далее) и насколько он уверен в правильности ответа.

2.1 Функциональная модель

На рисунке 2.1 изображена функциональная модель, отображающая структуру и функции программного обеспечения.

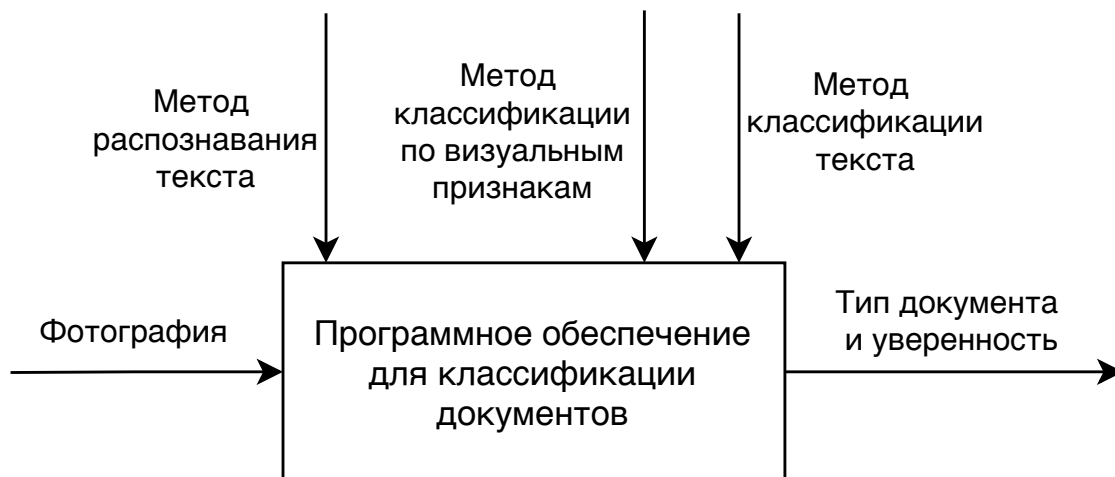


Рисунок 2.1 — Функциональная модель разрабатываемого ПО.

На рисунке 2.2 изображена функциональная модель первого уровня.

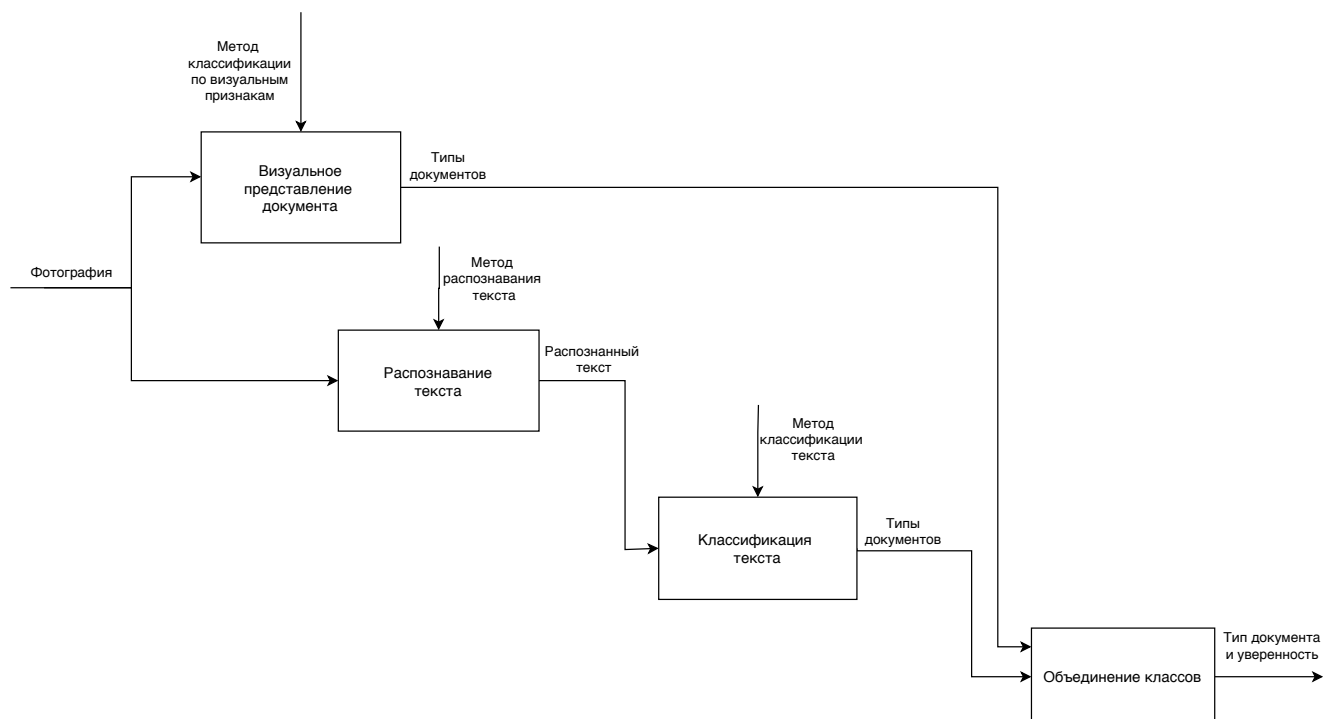


Рисунок 2.2 — Функциональная модель первого уровня.

3 Технологический раздел

4 Исследовательский раздел

ЗАКЛЮЧЕНИЕ

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Использование алгоритмов ML для классификации многостраничных документов: опыт ВТБ. — Режим доступа: <https://habr.com/ru/company/vtb/blog/497484/> (дата обращения: 28.12.2020).
2. Госуслуги. — Режим доступа: <https://www.gosuslugi.ru> (дата обращения: 28.12.2020).
3. Webmoney. — Режим доступа: <https://www.webmoney.ru> (дата обращения: 28.12.2020).
4. ВКонтакте. — Режим доступа: <https://vk.com/feed> (дата обращения: 28.12.2020).
5. Smart ID Engine – Безопасное распознавание документов удостоверяющих личность 193 стран мира. — Режим доступа: <https://smartengines.ru/smart-idreader/> (дата обращения: 15.04.2021).
6. Abbyy – Извлечение данных из документов, удостоверяющих личность. — Режим доступа: <https://www.abbyy.com/ru/solutions/forms-processing/> (дата обращения: 15.04.2021).
7. IRIS – Manage and process easily official Identity documents. — Режим доступа: <https://www.irislink.com/PT/c1842/IRIS---Manage---process-easily-official-Identity-documents.aspx> (дата обращения: 15.04.2021).
8. Jumio Uses AI for Automatic Recognition of ID Documents. — Режим доступа: <https://www.jumio.com/ai-id-documents/> (дата обращения: 15.04.2021).
9. Idmatch – Распознавание документов. — Режим доступа: <https://idmatch.co> (дата обращения: 15.04.2021).
10. Наивный байесовский классификатор. — Режим доступа: <https://www.machinelearningmastery.ru/the-naive-bayes-classifier-e92ea9f47523/> (дата обращения: 28.12.2020).

11. Метод k ближайших соседей. — Режим доступа: <https://intuit.ru/studies/courses/6/6/lecture/176?page=3> (дата обращения: 28.12.2020).
12. Метод деревьев решений. — Режим доступа: <https://loginom.ru/blog/decision-tree-p1> (дата обращения: 28.12.2020).
13. Н.О. Баев. Использование метода опорных векторов в задачах классификации. — Международный журнал информационных технологий и энергоэффективности, 2017.
14. Линейные модели классификации и регрессии. — Режим доступа: <https://habr.com/ru/company/ods/blog/323890/> (дата обращения: 28.12.2020).
15. Применение нейронных сетей для задач классификации. — Режим доступа: <https://basegroup.ru/community/articles/classification> (дата обращения: 28.12.2020).
16. ImageNet Large Scale Visual Recognition Challenge. — Режим доступа: <https://www.image-net.org/challenges/LSVRC/indexo> (дата обращения: 15.04.2021).
17. Data for ILSVRC. — Режим доступа: <https://www.kaggle.com/c/imagenet-object-localization-challenge/data> (дата обращения: 15.04.2021).
18. Sutskever Alex Krizhevsky Ilya, Hinton Geoffrey E. Imagenet classification with deep convolutional neural networks. — Springer, 2012.
19. Zeiler Matthew D, Fergus Rob. Visualizing and understanding convolutional networks. In European conference on computer vision. — Springer, 2014.
20. Simonyan Karen, Zisserman Andrew. Very deep convolutional networks for large-scale image recognition. — 2014.
21. Vanhoucke Christian Szegedy Wei Liu Yangqing Jia Pierre Sermanet Scott Reed Dragomir Anguelov Dumitru Erhan Vincent, Rabinovich Andrew.

Going deeper with convolutions. — 2015.

22. Ren Kaiming He Xiangyu Zhang Shaoqing, Sun Jian. Deep residual learning for image recognition. — 2016.

23. Shen Jie Hu Li, Sun Gang. Squeeze-and-excitation networks. — 2017.

24. What I learned from competing against a ConvNet on ImageNet. — Режим доступа: <http://karpathy.github.io/2014/09/02/what-i-learned-from-competing-against-a-convnet-on-imagenet/> (дата обращения: 17.04.2021).

25. Adversarial Machine Learning in Image Classification: A Survey Towards the Defender's Perspective. — Режим доступа: <https://arxiv.org/pdf/2009.03728.pdf> (дата обращения: 15.04.2021).

26. A Survey of the Recent Architectures of Deep Convolutional Neural Networks. — Режим доступа: <https://arxiv.org/pdf/1901.06032.pdf> (дата обращения: 15.04.2021).

27. Transfer Learning Using Convolutional Neural Network Architectures for Brain Tumor Classification from MRI Images. — Режим доступа: <https://europepmc.org/article/pmc/pmc7256397> (дата обращения: 15.04.2021).

28. Компьютерное зрение. — Режим доступа: <https://www.jetinfo.ru/computer-vision-technology-review/> (дата обращения: 28.12.2020).

29. ABBYY FineReader. — Режим доступа: <https://finereaderonline.com/ru-ru> (дата обращения: 17.04.2021).

30. Readiris 17. — Режим доступа: <https://www.irislink.com/EN-AU/c1729/Readiris-17--the-PDF-and-OCR-solution-for-Windows-.aspx> (дата обращения: 17.04.2021).

31. Cuneiform for Linux. — Режим доступа: <https://launchpad.net/cuneiform-linux> (дата обращения: 17.04.2021).

32. Tesseract Open Source OCR Engine. — Режим доступа: <https://github.com/tesseract-ocr/tesseract> (дата обращения: 17.04.2021).

33. OCRFeeder - A Complete OCR Suite. — Режим доступа: <https://gitlab.gnome.org/GNOME/ocrfeeder> (дата обращения: 17.04.2021).

34. Владимирович Смирнов Сергей. Технология и система автоматической корректировки результатов при распознавании архивных документов. — СПИИРАН, 2015.

35. OpenCV. — Режим доступа: <https://opencv.org> (дата обращения: 28.12.2020).

36. SIFT (Scale-Invariant Feature Transform). — Режим доступа: https://docs.opencv.org/master/da/df5/tutorial_py_sift_intro.html (дата обращения: 28.12.2020).

37. Распознавание документов в go-idmatch (препроцессинг). — Режим доступа: <https://blog.maddevs.io/goidmatch-internals-e75378cee35a> (дата обращения: 28.12.2020).

ПРИЛОЖЕНИЕ А

КАРТИНКИ