

Министерство образования и науки Российской Федерации  
федеральное автономное образовательное учреждение высшего образования  
«САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ,  
МЕХАНИКИ И ОПТИКИ»  
Факультет среднего профессионального образования

«УТВЕРЖДАЮ»

Декан ФСПО Университета ИТМО

\_\_\_\_\_ /Д.М.Гриншпун/

«\_\_\_\_\_» \_\_\_\_\_ 20\_\_ г

**ОТЧЕТ**  
**ПО ПРАКТИКЕ ПО ПРОФИЛЮ СПЕЦИАЛЬНОСТИ**  
**«ПРОГРАММИРОВАНИЕ В КОМПЬЮТЕРНЫХ СИСТЕМАХ» (09.02.03)**  
на тему

«Разработка веб-сервиса  
с системой распознавания изображений  
основанной на работе нейронных сетей»

Руководитель от предприятия:

Антонов М.Б.

Консультант по специальной части:

Антонов М.Б.

Разработал

студент дневного отделения

гр. У2435

Макшеев И.Д.

# СОДЕРЖАНИЕ

ВВЕДЕНИЕ .....	3
1    АНАЛИЗ ПОСТАВЛЕННОЙ ЗАДАЧИ .....	4
1.1 Изучение предметной области .....	4
1.2 Обзор методов решения задачи.....	5
1.3 Обзор аналогичных решений .....	7
1.4 Функциональные требования .....	11
2    ПРОЕКТИРОВАНИЕ РАЗРАБОТКИ .....	13
2.1 Описание входных и выходных данных .....	13
2.2 Моделирование требований к разработке .....	13
2.3 Обоснование выбора технологий и средств разработки.....	14
ЗАКЛЮЧЕНИЕ.....	19
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ .....	20
ПРИЛОЖЕНИЕ 1 .....	21
ПРИЛОЖЕНИЕ 2 .....	21

## ВВЕДЕНИЕ

В текущих условиях для хранения большого количества изображений используются объёмные носители информации или облачные веб-сервисы. В большинстве программ группировка изображений происходит по метаданным, т.е. дате создания, объёму занимаемого места, геолокации или каталогизацию может произвести сам пользователь.

При небольшом количестве картин, для человека задача сортировки и поиска не составляет труда. Однако, когда библиотека фотографии начинает становиться большой, задача поиска и сортировки сильно затруднена.

Для решения поставленной задачи целесообразно создать систему, осуществляющую сортировку изображений автоматически, а также имеющая способ взаимодействия пользователя с этими изображениями.

Автоматизация подразумевает классификацию изображений. Изображение загружается пользователем на сервер, где происходит присваивание ярлыков изображению и сохранение на сервере.

# 1 АНАЛИЗ ПОСТАВЛЕННОЙ ЗАДАЧИ

## 1.1 Изучение предметной области

Одной из самых сложных задач в информационных технологиях является обработка и распознавание объектов на изображениях. О важности этой проблемы говорит тот факт, что исследования по распознаванию объектов, анализу изображений и речи включены в перечень приоритетных направлений развития науки и техники и критических технологий федерального уровня. [1]

Современные методы распознавания символов на изображениях используются для решения широкого круга задач, таких как распознавание текста, изображений маркировки на поверхностях различных объектов и др. Процесс распознавания образов на изображениях со сложным фоном можно разделить на 3 этапа:

1. Выделение области расположения образа на изображении.
2. Выделение отдельных частей образа.
3. Распознавание образа.

В настоящее время такие технологии реализуются тремя традиционными методами – структурным, признаковым и шаблонным. Каждый из этих методов ориентирован на свои условия применения, для которых они являются эффективными, но все они имеют недостатки. При регистрации изображений наибольшие искажения, влияющие на результат распознавания, вносят аффинные и проекционные искажения, возникающие в связи с изменением угла регистрации, изменением масштаба, погодными условиями. Наличие посторонних объектов на изображениях со сложным фоном существенно снижают надежность распознавания методами, используемыми в современных системах распознавания образов на изображениях и видеопоследовательностях. [2]

Эти недостатки особенно ярко проявились при масштабной эксплуатации программно-технологических систем, использующих в своей основе эти методы. Практически у всех систем распознавания образов на изображениях

точность идентификации резко падает при различных искажениях входного изображения. Вместе с тем, технологические условия получения информации не позволяют полностью устранить искажения. [3]

Анализ методов выделения и распознавания объектов на изображениях показал, что для решения задачи эффективно использовать искусственные нейронные сети в связи с тем, что они являются слабо чувствительными к искажениям входного сигнала, а также имеют высокую скорость и точность решения задачи по сравнению с остальными методами.

## 1.2 Обзор методов решения задачи

Наиболее часто в задачах распознавания и идентификации изображений используются классические нейросетевые архитектуры (многослойный персептрон, сети с радиально-базисной функцией и др.), но из анализа данных работ и экспериментальных исследований следует, что применение классических нейросетевых архитектур в задаче неэффективно по следующим причинам:

- изображения имеют большую размерность, соответственно возрастает размер нейронной сети; [4]
- большое количество параметров увеличивает вместимость системы и соответственно требует большей тренировочной выборки, увеличивает время и вычислительную сложность процесса обучения;
- для повышения эффективности работы системы желательно применять несколько нейронных сетей (обученные с различными начальными значениями синоптических коэффициентов и порядком предъявления образов), но это увеличивает вычислительную сложности решения задачи и время выполнения; [5, 6]
- отсутствует инвариантность к изменениям масштаба изображения, ракурсов съёмки камеры и других геометрических искажений входного сигнала. [1]

Поэтому, для решения задачи выделения области расположения образов были выбраны сверточные нейронные сети, т. к. они обеспечивают частичную устойчивость к изменениям масштаба, смещениям, поворотам, смене ракурса и прочим искажениям.

Каждый слой сверточной нейронной сети представляет собой набор плоскостей, состоящих из нейронов. Нейроны одной плоскости имеют одинаковые синоптические коэффициенты, ведущие ко всем локальным участкам предыдущего слоя. Каждый нейрон слоя получает входы от некоторой области предыдущего слоя (локальное рецептивное поле), т. е. входное изображение предыдущего слоя сканируется небольшим окном и пропускается сквозь набор синоптических коэффициентов, а результат отображается на соответствующий нейрон текущего слоя. Таким образом, набор плоскостей представляет собой карты характеристик, и каждая плоскость находит «свои» участки изображения в любом месте предыдущего слоя. Размер локального рецептивного поля выбирается самостоятельно в процессе разработки нейронной сети. [7]

Слои делятся на 2 типа: сверточные и подвыборочные. В сверточных слоях при сканировании рецептивные поля частично наслаиваются друг на друга по принципу черепицы, в подвыборочных слоях области соседних нейронов не перекрываются. Подвыборочный слой уменьшает масштаб плоскостей путём локального усреднения значений выходов нейронов, таким образом достигается иерархическая организация. Последующие слои извлекают более общие характеристики, меньше зависящие от искажений изображения. После прохождения нескольких слоев карта признаков выражается в вектор. [8, 9]

Постепенно нейронная сеть обучается выделять ключевые характеристики образов, в поступающих на вход изображениях.

### 1.3 Обзор аналогичных решений

Для сравнения были найдены следующие аналоги:

- «Google photo»;

Этот сервис предоставляет неограниченное пространство для хранения фотографий и видео. Загруженные фото сжимаются, чтобы занимать меньше пространства на серверах Google.

Есть возможность хранения контента в более высоком качестве, но при этом с ограниченным пространством для хранения. Для пользователей Mac и PC предусмотрена утилита автозагрузки, но просмотр фото производится через браузер. Google Фото предоставляет большие возможности по поиску и сортировке нужных фотографий и видео.

Загруженный в сервис контент сортируется по разным критериям: место съёмки, лица и объекты, изображенные на фото или видео. Присутствует возможность вручную сортировать фото или видео по альбомам.

- «Flickr»;

Пользователь системы может помещать на удалённый сервер свои фотографии. Сервис предполагает возможность загрузить 1 терабайт фотографий.

К каждой фотографии её владелец может добавить название, краткое описание и ключевые слова (тег) для дальнейшего поиска. Можно делать заметки и на самих фотографиях. Если на фотографии изображено несколько объектов (например, несколько зданий), то можно выделить любой из объектов и добавить к нему описание.

Фотография может иметь статус личной, семейной, групповой или общедоступной. Фотографию можно найти, по ключевым словам, указанным пользователем. Система позволяет вести поиск одновременно по нескольким ключевым словам. Дополнительный интерес представляет возможность использовать коллекции своих фотографий или отдельные фотографии на страницах своих сайтов или своего Живого Журнала.

Систему можно попросить сгенерировать html-код ссылки на отдельную фотографию. Дополнительные сервисы позволяют пользователям обмениваться фотографиями и метками на них. Для фотографий отображается список других объектов со схожими метками. Такая обратная связь приводит к коммуникации между пользователями посредством метаданных.

Система Flickr поддерживает возможность переписки между пользователями. Однако эти отношения по переписке не оказывают заметного влияния на формирование общей картины или карты меток, которыми пользуется всё сообщество. Сервис Flickr позволяет получить карту ключевых слов, наиболее часто употребляемых при классификации фотографий.

Пользователи системы могут образовывать группы по интересам с возможностью вести тематические обсуждения, приглашать в группу других пользователей и их фотографии, рассматривать расположение фотографий группы на карте.

– «ZZ Photo»;

Этот сервис может отсортировать фотографии на компьютере по папкам на основе даты и создать гибкую галерею из локальных и сетевых источников с упорядочиванием снимков по всем актуальным параметрам. Программа делает это практически без участия пользователя с помощью собственной технологии распознавания и сопоставления содержимого фотографий.

Существует только в качестве Windows-приложения. Может работать не только с локальной фото библиотекой, исключая нерелевантные изображения по отсутствию в них метаданных, но и с альбомами из Facebook, Instagram и Flickr. Есть возможность добавления фото из облачных хранилищ Dropbox и Google Drive.

Есть возможность физически перенести все изображения из приложения в единую коллекцию на компьютере. Разработчики ZZ Photo делают особый акцент на технологии распознавания объектов на изображениях. После того как фото были найдены приложением, алгоритмы найдут среди всех снимков похожие лица, и пользователя спросят: «Кто это?». Подписав имя человека на



нескольких фото, пользователь упростит дальнейшую работу алгоритма по распознаванию этого человека. Алгоритмы распространяются не только на людей, но и на животных. В совокупности с метаданными приложению удаётся логически связать весь обнаруженный массив фотографий набором тегов, по которым коллекцию можно отсортировать.

ZZ Photo может избавить получившуюся фотоколлекцию от дубликатов, дополнительно сгруппирует снимки, сделанные в рамках того или иного события, а также позволит скрыть от чужих глаз приватные фотографии, спрятав их в отдельном альбоме.

– «ThisLife»;

Есть возможность загружать неограниченное количество фотографий. Для того, чтобы загружать и видео, то нужно оплачивать дополнительное место.

Фото из основной библиотеки представлены в чётком хронологическом порядке. Можно посмотреть фото, нанесёнными на карту по геотегам. Так же есть импортированные и загруженные из библиотеки.

Фото могут быть организованы в альбомы, обрезаны и опубликованы онлайн. На ThisLife есть функция совместного аккаунта, которым можно делиться с кем угодно. Оба пользователя получают все функции работы с фотографиями.

Есть возможность импортировать изображения из Instagram, Facebook, Flickr, SmugMug и Picasa и загружает новые фото после того, как вы добавили их на один из этих сервисов.

Существует мобильные приложения для Android, iOS и Kindle Fire, каждое из которых автоматически загружает снятые на мобильное устройство фото. Есть специальный настольный загрузчик, который загружает фото непосредственно с ПК.

– «ACDSee»;

Программа для работы с цифровыми фотографиями, которая обладает расширенным набором функций, по сравнению с остальными аналогами. Поддерживает более 100 форматов графических файлов, позволяет работать с

RAW-изображениями, автоматически сортировать фотографии по параметрам, полученным из цифровых фотокамер, содержит эффективный визуальный метод добавления к изображениям метатегов и быструю пакетную обработку большого количества картинок, в том числе и в RAW формате.

Присутствуют инструменты для независимого регулирования цветовых каналов, исправления артефактов фотографий, появляющихся из-за погрешностей оптики, добавления "водяных знаков", работы с IPTC-метаданными, имеется возможность архивирования графических коллекций в ZIP-файлы, запись на CD или DVD, в последних версиях имеет два режима просмотра: быстрый, в котором доступны только инструменты поворота изображения и изменение масштаба, и полный, с загрузкой всех инструментов обработки и многое другое.

– «XnView»;

Кроссплатформенная программа для просмотра изображений, поддерживающая просмотр более 400 и сохранение (конвертирование) до 50 различных графических и мультимедийных форматов файлов. XnView работает со сканером и принтером, а также в полнокомплектной версии умеет проигрывать широко распространённые аудио-видео форматы файлов, но только при условии наличия нужных кодеков в системе.

– «Zoner photo studio»;

Приложение, созданное для обработки цифровых снимков. Одной из функций программы является извлечение из снимков информации о местонахождении фотографа во время совершения съемки и соотношение этих данных с картографическими программами. PhotoStudio содержит средства для редактирования снимков, позволяет добавлять к фотографиям текстовые описания, создавать каталоги фотографий на различных носителях.

В процессе обзора аналогичных решений были выявлены следующие функциональные параметры, по которым будет произведено их сравнение в таблице 1:

– поиск фото, расположенных в разных директориях на ПК;

- поддержка RAW формата;
- поиск и удаление дубликатов фотографий;
- скрывание снимков от обзора;
- сбор всех снимков в одном месте;
- Windows-клиент;
- онлайн-клиент;
- внешнее хранилище (облако).

Таблица 1

Сравнительная таблица аналогов

	«Google photo»	«Flickr»	«ThisLife»	«ZZ Photo»	«ACDSee»	«XnView»	«Zoner»
Поиск фото	+	-	-	+	-	-	-
Поддержка Raw	+	+	+	-	+	+	+
Поиск и удаление дубликатов	+	-	+	+	-	-	-
Скрытие фотографий	-	-	-	+	-	-	-
Сбор всех изображений в одном месте	+	+	+	+	-	-	-
Windows-клиент	Picasa	Только импорт	Только импорт	+	+	+	+
Онлайн-клиент	+	+	+	-	+	-	+
Внешнее хранилище	+	+	+	-	+	-	+

Функциональные особенности, которые будут использованы в разработке, представлены ниже.

#### 1.4 Функциональные требования

Исходя из требований дипломного проекта были выделены функциональные возможности, которые должны быть реализованы в текущей разработке.

Система должна:

- предоставлять возможность регистрации пользователю;

- предоставлять интерфейс для работы с изображениями;
- автоматически раздавать ярлыки к изображениям;
- предоставлять возможность поиска изображений по ярлыкам.

Каждый пользователь должен иметь возможность:

- создавать библиотеки изображений разных градаций;
- добавлять изображения в библиотеку;
- редактировать или добавлять новые ярлыки для изображений.

Исходя из обзора аналогичных решений, были выделены функциональные требования:

- поддержка наиболее распространённых форматов изображений (TIFF, PNG, GIF, RAW, BMP, JPEG, JPEG 2000);

В текущее время очень много различных форматов изображений. Если ограничить пользователя в выборе формата изображения – это может повлиять на выбор разрабатываемого продукта в худшую сторону.

- поиск и удаление дубликатов изображений;
- скрывание фотографий;

В разработке все фотографии будут изначально доступны только пользователю, который их загрузил. Однако, если пользователь захочет поделиться изображением, он сможет запросить у программы создать ссылку на изображение.

- онлайн клиент;

Так же, возможность, с минимальными потерями функционала, работать с мобильных устройств и планшетных компьютеров.

- техническое задание на разработку представлено в приложении 1.

## 2 ПРОЕКТИРОВАНИЕ РАЗРАБОТКИ

### 2.1 Описание входных и выходных данных

Входными данными являются запросы пользователя к веб-ресурсу. Структура входных и выходных данных для системы представлена в таблице П1.1.

### 2.2 Моделирование требований к разработке

При построении функциональной модели программы клиента использовалась IDEF0-методология. Эта методология позволила создать функциональные модели, отображающие структуру и функции системы, а также потоки информации, связывающие эти функции. На диаграмме изображен процесс работы с системой.

Для наглядного описания функциональных возможностей и поведения системы была использована диаграмма вариантов использования (UML Use-Case). На диаграмме изображены все возможные варианты действий, которые пользователь может осуществить во время использования системы.

Система должна удовлетворять функциональной модели, представленной на рисунках П2.1 – П2.5.

Для описания структуры нейронной сети используется синтаксис создания моделей фреймворка Caffe. [10], [11] Структура сети представлена на рисунке П2.6.

Caffe определяет нейронную сеть в виде слоёв (Layers) и больших бинарных объектов (BLOB). Слои занимаются вычислениями, а BLOB являются стандартными массивами и унифицированными интерфейсами памяти для фреймворка.

Изначально изображения подаётся на входной слой (Input). Изображение проходит через всю сеть взаимодействуя с каждым слоем.

Слой свёртки (Convolutional layer) — это основной блок свёрточной нейронной сети. Слой свёртки включает в себя для каждого канала свой

фильтр, ядро свёртки которого обрабатывает предыдущий слой по фрагментам (суммируя результаты матричного произведения для каждого фрагмента).

Блок линейной ректификации (ReLU - rectified linear unit) - это функция активации после свёрточного слоя, которая отвечает за отсечение ненужных деталей в канале (при отрицательном выходе).

Слой субдискретизации (Pooling) представляет собой нелинейное уплотнение карты признаков, при этом группа пикселей уплотняется до одного пикселя, проходя нелинейное преобразование. Наиболее употребительна при этом функция максимума. Преобразования затрагивают непересекающиеся прямоугольники или квадраты, каждый из которых ужимается в один пиксель, при этом выбирается пиксель, имеющий максимальное значение. Операция пулинга позволяет существенно уменьшить пространственный объём изображения. Пулинг интерпретируется так. Если на предыдущей операции свёртки уже были выявлены некоторые признаки, то для дальнейшей обработки настолько подробное изображение уже не нужно, и оно уплотняется до менее подробного. К тому же фильтрация уже ненужных деталей помогает не переобучаться. Слой пулинга вставляется после слоя свёртки перед слоем следующей свёртки.

Уровень нормализации локального ответа (LRN - Local Response Normalization) выполняет «боковое торможение» путем нормализации локальных входных областей.

Полно-связанный слой (InnerProduct) рассматривает входные данные как простой вектор и выводит результат в виде одного вектора.

## 2.3 Обоснование выбора технологий и средств разработки

В ходе обзора средств разработки нейросети были рассмотрены следующие языки программирования (ЯП):

– C++;

C++ – язык разработан в начале 1980-х годов, Бьёрном Страуструпом. Разрабатывая C++ Страуструп решил дополнить язык C возможностью работы

объектами и классами. В результате полученный язык сочетал в себе такие плюсы С, как быстрота, многофункциональность и переносимость, так и объектно-ориентированную парадигму.

С++ сочетает свойства как высокоуровневых, так и низкоуровневых языков, благодаря чему широко используется для разработки программного обеспечения, являясь одним из самых популярных языков программирования. Область его применения включает создание операционных систем, разнообразных прикладных программ, драйверов устройств, приложений для встраиваемых систем и высокопроизводительных серверов.

Основным преимуществом С++, при создании нейросети, является быстрота его исполнения, так как часто требуется, чтобы приложения выполняли тяжелые вычисления.

– R;

R – язык программирования для статистической обработки данных и работы с графикой, а также свободная программная среда с открытым исходным кодом, развиваемая в рамках проекта GNU. Этот язык широко используется как статистическое программное обеспечение для анализа данных и фактически стал стандартом для статистических программ.

R поддерживает широкий спектр статистических и численных методов и обладает хорошей расширяемостью с помощью пакетов. Пакеты представляют собой библиотеки для работы специфических функций или специальных областей применения. В базовую поставку R включен основной набор пакетов, а всего по состоянию на 2013 год доступно более 4000 пакетов.

Ещё одной особенностью R являются графические возможности, заключающиеся в возможности создания качественной графики, которая может включать математические символы.

Отличительной особенностью использования языка R, при создании нейросети, является наличие большого количества пакетов, которые могут значительно сократить время проектировки сети.

– Python.

Python – высокоуровневый язык программирования общего назначения, ориентированный на повышение производительности разработчика и читаемости кода. Синтаксис ядра Python минималистичен. В то же время стандартная библиотека включает большой объём полезных функций.

Python поддерживает несколько парадигм программирования, в том числе структурное, объектно-ориентированное, функциональное, императивное и аспектно-ориентированное. Основные архитектурные черты – динамическая типизация, автоматическое управление памятью, полная интроспекция, механизм обработки исключений, поддержка многопоточных вычислений и удобные высокоуровневые структуры данных. Код в Python организовывается в функции и классы, которые могут объединяться в модули (они в свою очередь могут быть объединены в пакеты).

Python очень широко используется для создания искусственного интеллекта. Существует большое количество пакетов для создания самых разных видов АИ:

- общий ИИ;
- машинное обучение;
- обработчики натурального языка;
- нейронные сети.

Как инструмент разработки выбран язык Python, так как помимо реализации нейросети, нужно ещё реализовать веб-приложение. Существует большое количество свободно распространяемых библиотек для этого ЯП, позволяющих облегчить процесс реализации как нейросети, так и веб-приложения.

Клиентская часть будет реализована при помощи языка гипертекстовой разметки HTML с применением каскадных таблиц стилей (CSS) и скриптового языка JavaScript. Для увеличения скорости разработки пользовательского интерфейса необходимо использовать фреймворк. Был выполнен обзор наиболее популярных JavaScript-фреймворков:



– «Bootstrap»;

Фреймворк для создания современных, кросс-браузерных и стандартизованных интерфейсов. Продуманная структура кода HTML, JavaScript и CSS дает возможность создавать множество самых разнообразных элементов интерфейса и сетку сайта. Свидетельством того, что Bootstrap это очень удачная разработка, является его непрерывно растущая популярность. Нынешний уровень развития этого фреймворка уже позволяет полностью создать любой веб-интерфейс. Bootstrap использует новейшие возможности HTML и CSS. Начиная с версии Bootstrap 2 все макеты, создаваемые с его помощью стали адаптивными. Основным преимуществом использования Bootstrap является Less - динамический язык стилей, существенно расширяющий возможности CSS. С его помощью разработчики могут создавать переменные, вложенные колонки, управлять цветами и т.д.

– «jQuery»;

Популярная и распространенная библиотека на JavaScript, фокусирующаяся на взаимодействии JavaScript и HTML. jQuery предоставляет удобный API по работе с технологией Ajax. Имеется возможность достаточно просто расширять функциональность базовой библиотеки за счет включения дополнительных плагинов. Помимо этого, jQuery весьма производителен благодаря простому содержанию и коду.

– «MooTools»;

Является модульным, объектно-ориентированным фреймворком. Данная JavaScript-библиотека разбита на несколько модулей и хорошо документирована. Кроме того, этот фреймворк содержит плагин MooTools.More.js, который позволяет наращивать возможности сторонними библиотеками, и имеет практичные инструменты для манипуляций с массивами, датами, строками и т.д. Дополнительным достоинством является возможность локализовать англоязычный синтаксис языка самого MooTools на любой национальный язык, что позволяет программировать на нем максимально комфортно, например, на русском языке.

– «Yahoo! UI Library»;

Набор утилит и так называемых контроллов, написанных на JavaScript и CSS, для создания интерактивных WEB-приложений, использующих техники DOM-скриптинга и Ajax. В этом фреймворке доступны два типа компонентов: утилиты и контролы. Утилиты упрощают браузерную разработку, связанную с DOM и Ajax. Контролы – это набор готовых, высоко интерактивных визуальных элементов для проектирования веб-страниц. Все эти элементы создаются и работают только на клиентской стороне и не требуют обновления страницы для их изменения.

– «ExtJS».

Главная отличительная черта этого фреймворка в том, что он предоставляет собой наиболее приближенную среду к классическому программированию, реализуя развитые графические средства отображения и взаимодействия с пользователем.

Таким образом, исходя из вышеприведённого обзора, в реализации приложения будет использоваться фреймворк Bootstrap, как наиболее производительная и популярная библиотека, которая позволит реализовать интерактивное и динамичное приложение.

## ЗАКЛЮЧЕНИЕ

В процессе прохождения практики по профилю специальности были решены следующие задачи:

1. Изучена предметная область.
2. Рассмотрены аналогичные решения.
3. Обоснована целесообразность разработки.
4. Сформулированы функциональные требования к разработке.
5. Проведено функциональное моделирование.
6. Обоснован выбор средств реализации.
7. Разработано техническое задание.

Задачи, поставленные на практику, были выполнены. Цель практики достигнута.

## СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Макаренко А.А., Калайда В.Т. Методика локализации изображения лица для систем видеоконтроля на основе нейронной сети // Известия Томского политехнического университета. – 2006. – Т.309. – №8. – С. 113 – 118.
2. Le Cun Y., Bengio Y. Convolutional networks for images, speech and time series // The handbook of brain theory and neural networks. – 1998. – V.7. – №1. – P.255 – 258.
3. Гонсалес Р., Вудс Р. Цифровая обработка изображений. – М.: Техносфера, 2005. – 1072 с.
4. Болотова, Ю.А., Спицын В.Г., Кермани А.К. Распознавание символов на цветном фоне на основе иерархической временной модели с предобработкой фильтрами Габора // Электромагнитные волны и электронные системы. – 2012. – Т.16. – №1. – С.14 – 19.
5. Bundzel M., Hashimoto S. Object identification in dynamic images based on the memory prediction theory of brain function // Journal of Intelligent Learning Systems and Applications. – 2010. – V.2. – №4. – P.212 – 220.
6. Буй Тхи Тху Чанг, Фан Нгок Хоанг, Спицын В.Г. Алгоритмическое и программное обеспечение для классификации цифровых изображений с помощью вейвлет-преобразования Хаара и нейронных сетей // Известия Томского политехнического университета. – 2011. – Т.319. – №5. – С.103 – 106.
7. Болотова Ю.А., Спицын В.Г., Фомин А.Э. Применение модели иерархической временной памяти в распознавания изображений // Известия Томского политехнического университета. – 2011. – Т.318. – №5. – С.60 – 63.
8. Кермани К.А., Спицын В.Г., Хамкер Ф. Нахождение параметров и удаление постоянной составляющей фильтра Габора для обработки

- изображений // Известия Томского политехнического университета. – 2011. – Т.318. – №5. – С.57 – 59.
9. Hansen D.W., Hansen J.P., Nielsen M. Eye typing using Markov and active appearance models // Applications of computer vision. – 2002. – V.12. – P.132 – 136.
  10. Yangqing Jia, Evan Shelhamer. Caffe Deep learning framework by the BVLC. Layers. [Электронный ресурс]. – Электрон. дан. – Режим доступа: <http://caffe.berkeleyvision.org/tutorial/layers.html>.
  11. Yangqing Jia, Evan Shelhamer. Caffe Deep learning framework by the BVLC. Blobs, Layers and Nets: anatomy of a Caffe model. [Электронный ресурс]. – Электрон. дан. – Режим доступа: [http://caffe.berkeleyvision.org/tutorial/net\\_layer\\_blob.html](http://caffe.berkeleyvision.org/tutorial/net_layer_blob.html).

# ПРИЛОЖЕНИЕ 1

## **Техническое задание**

# 1 НАЗНАЧЕНИЕ РАЗРАБОТКИ

Назначением разработки является создание веб-сервиса с системой распознавания изображений основанной на работе нейронных сетей.

Целевой аудиторией разработки являются люди, которые работают с большим количеством фотографий.

Автоматическая система позволит давать ярлыки изображениям, которые позволят производить быстрый поиск и группировку картин по выбранному ярлыку.

Идентификация предметов позволит автоматически закреплять ярлыки, характеризующие свойства и признаки изображений. Это позволит производить поиск и группировку по закреплённому на изображениях ярлыку.

## 2 ЦЕЛИ И ЗАДАЧИ

Целью является разработка веб-ресурса для хранения изображений с системой распознавания основанной на работе нейронных сетей.

Для достижения поставленной цели необходимо выполнить следующие задачи:

1. Проанализировать требования к разработке.
2. Проанализировать модель работы программного продукта.
3. Спроектировать нейронную сеть, веб-приложение и интерфейс для взаимодействия.
4. Разработать нейронную сеть, веб-приложение и интерфейс для взаимодействия.
5. Провести тестирование программного продукта.



### 3 ТРЕБОВАНИЯ К РАЗРАБОТКЕ

#### 3.1 Функциональные требования к разработке

Исходя из требований дипломного проекта были выделены функциональные возможности, которые должны быть реализованы в текущей разработке:

Система должна:

- предоставлять возможность регистрации и авторизации пользователю;
- предоставлять интерфейс для работы с изображениями;
- автоматически раздавать ярлыки к изображениям;
- предоставлять возможность поиска изображений по ярлыкам, закреплённым на изображениях.

Каждый пользователь должен иметь возможность:

- создавать или удалять библиотеки изображений разных градаций;
- добавлять или убирать изображения из библиотеки;
- редактировать или добавлять новые ярлыки для изображений.

Исходя из обзора аналогичных решений, были выделены функциональные требования:

- поддержка наиболее распространённых форматов изображений (TIFF, PNG, GIF, RAW, BMP, JPEG, JPEG 2000);

В текущее время очень много различных форматов изображений. Если ограничить пользователя в выборе формата изображения – это может повлиять на выбор разрабатываемого продукта в худшую сторону.

- поиск и удаление дубликатов изображений;
- скрытие фотографий;

В разработке все фотографии будут изначально доступны только пользователю, который их загрузил. Однако, если пользователь захочет поделиться изображением, он сможет создать запрос к программе для создания ссылки на изображение.

– онлайн клиент.

### 3.2 Описание входных и выходных данных

На основе функциональных требований определены входные и выходные данные, представленные в таблице П1.1.

Таблица П1.1

#### Входные и выходные данные

Функция	Входные данные	Выходные данные
Регистрация на ресурсе	Логин, адрес электронной почты, пароль для доступа к ресурсу	Не активированный аккаунт пользователя, изменение информации в базе данных
Авторизация на ресурсе	Логин и пароль	Права доступа к функциям ресурса
Загрузка изображения	Изображение	Изображение с ярлыками в библиотеке, изменение информации в базе данных.
Удаление изображения	Идентификатор изображения	Статус операции, изменение информации в базе данных
Добавление ярлыка	Текст	Ярлык у изображения, изменение информации в базе данных
Удаления ярлыка	Идентификатор ярлыка	Статус операции, изменение информации в базе данных.
Редактирование ярлыка	Текст	Изменённый ярлык, изменение информации в базе данных.
Создание библиотеки	Название и описание	Созданная библиотека, изменение информации в базе данных
Удаление библиотеки	Идентификатор библиотеки	Статус операции, изменение информации в базе данных
Редактирование названия библиотеки	Текст	Измененная библиотека, изменение информации в базе данных
Добавление изображения в библиотеку	Идентификатор библиотеки, идентификатор изображения	Добавленное изображения в библиотеку, изменение информации в базе данных
Удаление изображения из библиотеки	Идентификатор библиотеки, идентификатор изображения	Удалённое изображение из библиотеки, изменение информации в базе данных

### 3.3 Модель работы продукта разработки

Система должна удовлетворять функциональной модели, представленной на рисунках П2.1 – П2.5.

Модель нейронной сети представлена на рисунке П2.6.

### 3.4 Требования к программным средствам разработки

Разрабатываемое ПО должно быть реализовано как веб–ресурс в сети интернет. Для разработки информационного обеспечения должна использоваться технология баз данных.

Для веб-разработки должны использоваться следующие программные средства:

- язык программирования Python не ниже версии 3.4 с использованием библиотеки flask;
- СУБД MySQL не ниже версии 5.5;
- веб-сервер Apache версии не ниже 2.0.

При разработке нейронной сети в качестве программных средств требуется использовать язык программирования Python с использованием фреймворка caffe.

В качестве используемой технологии разработки должна использоваться парадигма объектно-ориентированного программирования.

### 3.5 Требования к составу и параметрам технических средств, применяемых при использовании системы

При использовании системы пользователь должен иметь доступ к следующим техническим средствам:

- любое электронное устройство с возможностью выхода в сеть Интернет;
- программа-браузер;
- наличие доступа к сети Интернет.

### 3.6 Методы тестирования соответствия разработки предъявленным требованиям

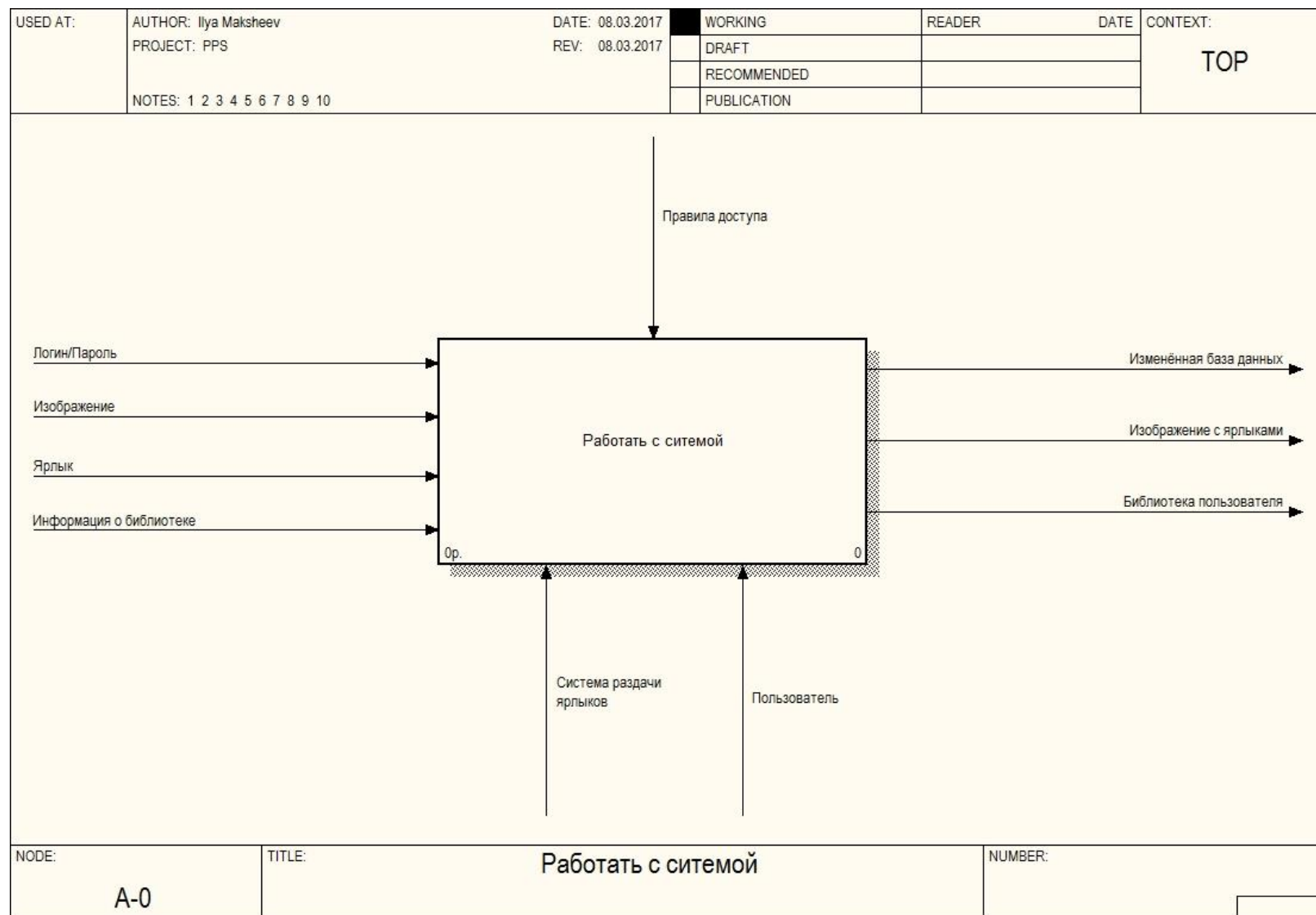
Тестирование системы должно проводиться на модульном и интеграционном уровне с использованием технологий «черного» и «белого» ящиков с применением тестирования переходов между состояниями и функциональной эквивалентности.

## ПРИЛОЖЕНИЕ 2

### **Модель разработки**



Рисунок П2.1 Use-Case диаграмма пользователя



*Рисунок П2.2 Модель IDEF0. Главный процесс*

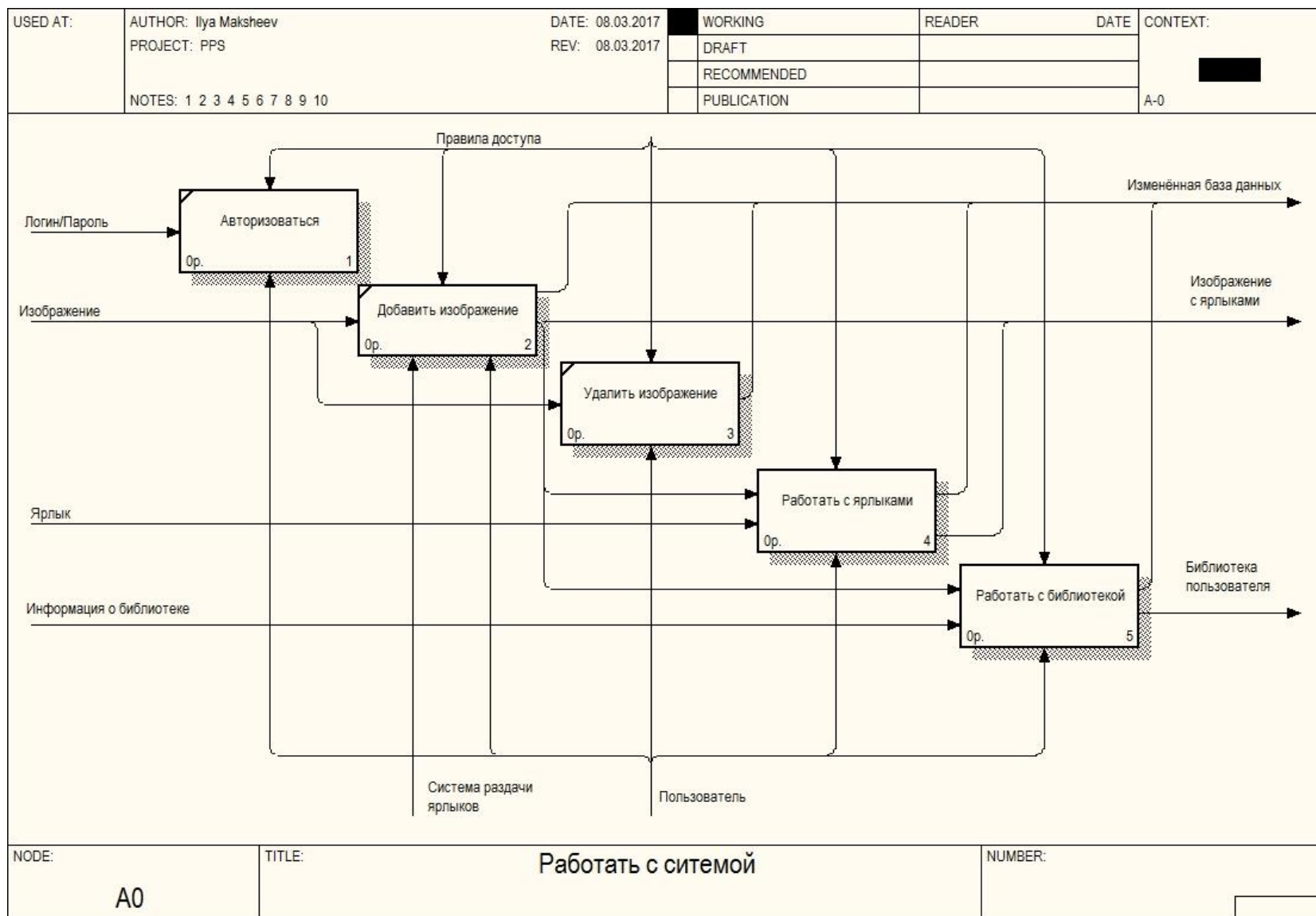


Рисунок П2.3 Модель IDEF0. Детализация главного процесса



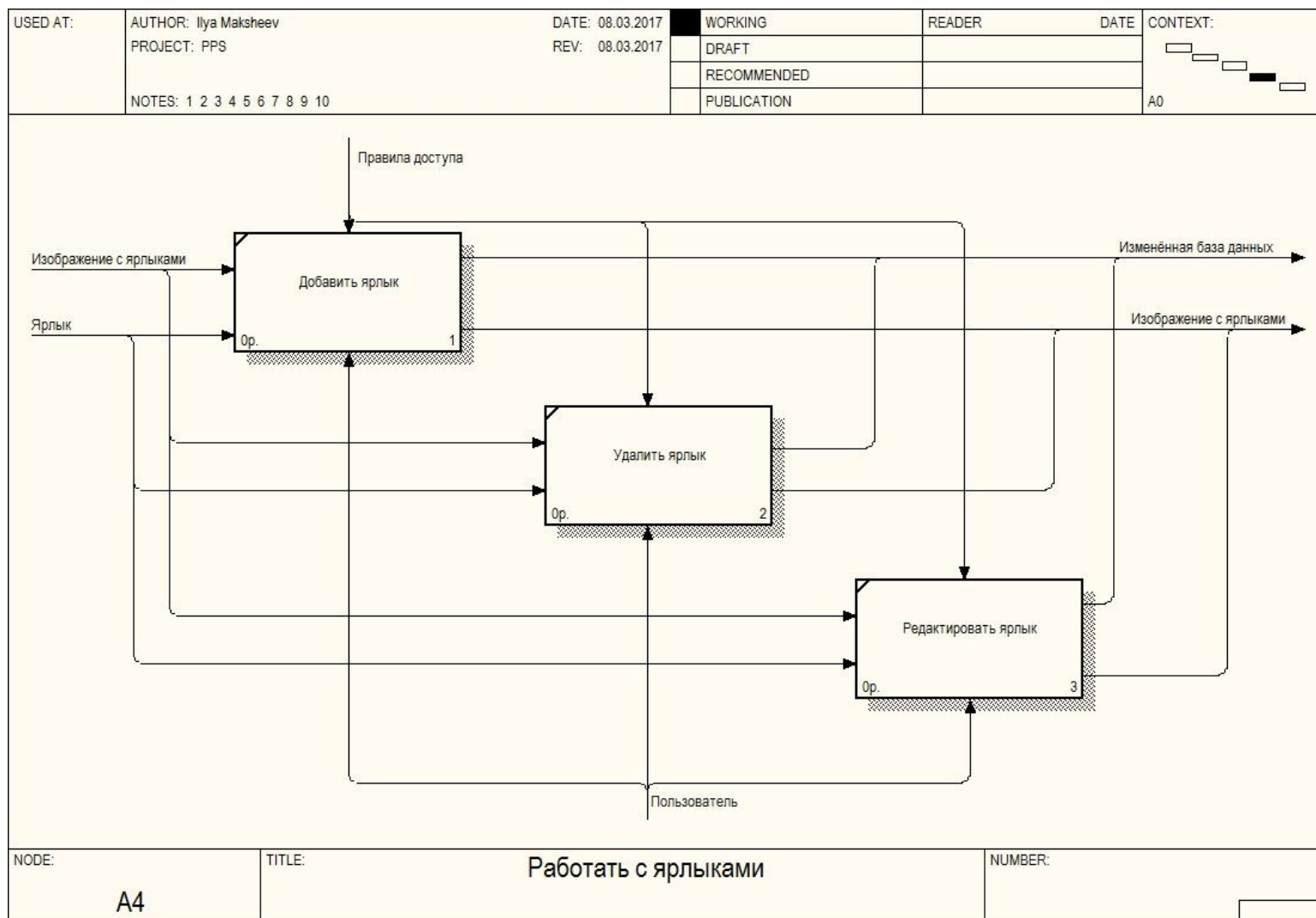


Рисунок П2.1 Модель IDEF0. Детализация процесса «Работать с ярлыками»

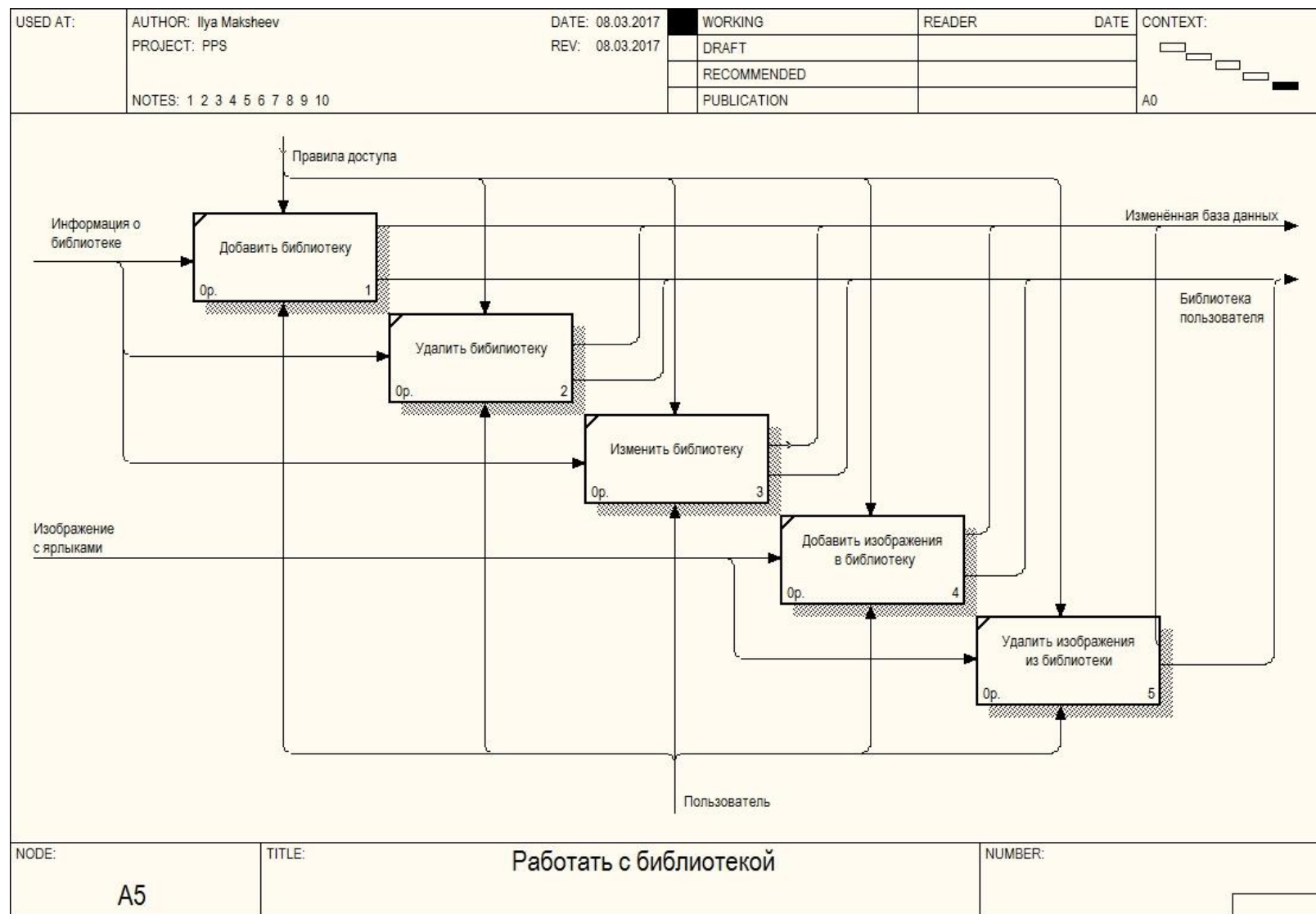


Рисунок П2.2 Модель IDEF0. Детализация процесса «Работать с библиотекой»

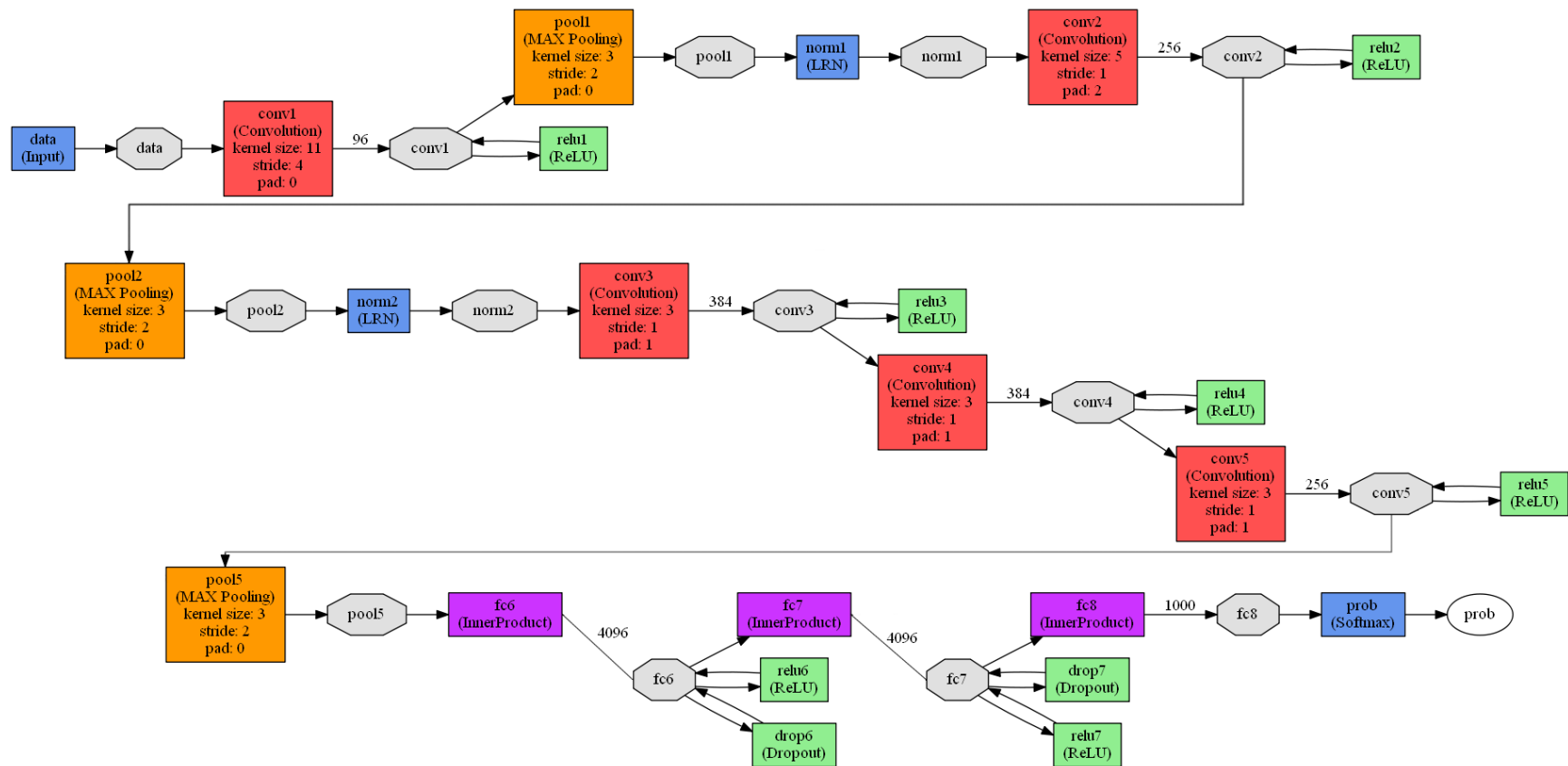


Рисунок П2.6 Модель нейронной сети