



Национальный исследовательский ядерный университет
«МИФИ»

Кафедра “Кибернетика”(22)

“Технология программирования кибернетических систем
(управление программными проектами)”

Техническое задание

«Модуль OCR (оптического распознавания текста)»

Руководитель:

Группа:

Иващенко Н.Г.

Шаров Кирилл

Белозёров Александр

Лоик Владимир

Аристова Ксения

Мясникова Елизавета

Андропов Андрей

Андреев Анатолий

Клипов Данил

Гольдварг Александр

Москва
2019

Содержание

1 Общие сведения	4
2 Назначение и цели создания проекта	5
2.1 Описание проекта	5
2.2 Цель проекта	5
3 Ключевые вехи проекта	6
4 Характеристика объекта автоматизации	8
5 Требования к библиотеке	9
5.1 Требования к продукту в целом	9
5.1.1 Требование к структуре и функционированию библиотеки	9
5.1.1.1 Перечень подсистем, их назначение и основные характеристики	9
5.1.1.2 Предобработка, обесцвечивание и бинаризация растровых изображений	10
5.1.1.3 Фильтрация монохромных изображений и морфологические операции	10
5.1.1.4 Выделение контуров на изображении	10
5.1.1.5 Выделение признаков символов	10
5.1.1.6 Сегментация символов в строке	11
5.1.1.7 Классификация на основе признаков, анализ профилей	11
5.1.2 Требования к надёжности	12
5.1.2.1 Состав показателей надёжности для системы в целом	12
5.1.2.2 Перечень аварийных ситуаций, по которым регламентируются требования к надёжности	12
5.1.2.3 Требования к надёжности технических средств и программного обеспечения	13
5.1.2.4 Требования к методам оценки и контроля показателей надёжности на разных стадиях создания системы в соответствии с действующими нормативно-техническими документами	13
5.2 Требование к функциям(задачам), выполняемые компонентом	13
5.2.1 Требования к функциям инструмента предобработки, обесцвечивания и бинаризации растровых изображений	13
5.2.2 Требования к функциям инструмента фильтрации монохромных изображений и морфологических операций	14
5.2.3 Требования к функциям инструмента выделения признаков в строке	14
5.2.4 Требования к функциям инструмента выделения символов в строке	14
5.2.5 Требования к функциям инструмента сегментация символов в строке	15

5.2.6 Требования к функциям инструмента классификации на основе признаков, анализ профилей	15
5.3 Требования к видам программного обеспечения	15
5.3.1 Программное обеспечение	15
5.3.2 Математическое обеспечение	16
5.3.3 Информационное обеспечение	16
6 Состав и содержание работ по созданию системы	18
6.1 Стадии проекта	18
7 Порядок контроля и приемки	23
7.1 Цель испытаний	23
7.2 Перечень характеристик библиотеки подлежащих проверке	23
7.3 Перечень проводимых проверок	23
7.3.1 Загрузка файлов изображений	23
7.3.2 Предобработка и сохранение изображений	24
7.3.3 Обесцвечивание изображения	25
7.3.4 Бинаризация изображения	26
7.3.5 Фильтрация изображения	26
7.3.6 Выделение контуров на изображении	28
7.3.7 Распознавание символов на изображении с текстом	29
7.4 Последовательность проведения испытаний	29
7.5 Перечень работ, проводимых после завершения испытаний, требования к ним, объем и порядок проведения	30
7.6 Порядок проведения испытаний	30
7.7 Условия начала и завершения отдельных этапов испытаний	30
7.8 Требования к персоналу, проводящему испытания, и порядок его допуска к испытаниям	30
7.9 Материально-техническое обеспечение испытаний	31
7.9.1 Рабочее место	31
7.10 Отчетность	31
8 Перечень сокращений	32

1 Общие сведения

В данном документе описывается техническое задание на разработку библиотеки языка программирования Python «Модуль OCR (оптического распознавания текста)».

Условное обозначение модуля: oavi_ocr.

Данный модуль создается объединением инициативных студентов групп М18-504, М18-514 и М18-Ш04 в рамках курса “Методология программной инженерии”.

При разработке данного модуля используются следующие документы:

- Стандарт на разработку требований к ПО [1]
- Стандарт на проектирование [2]
- Стандарт на кодирование [3]
- План верификации ПО [4]
- План управления конфигурацией ПО [5]
- План контроля качества [6]
- Технический регламент [7]
- План сертификации [8]

Плановый срок начала работ - 29.09.2019

Плановый срок окончания работ – 22.12.2019

Финансирование работ по разработке модуля не предусматривается.

Порядок оформления и предъявления заказчику результатов работ по созданию системы описан в плане сертификации [8] и техническом регламенте [7].

2 Назначение и цели создания проекта

2.1 Описание проекта

Разрабатываемый продукт представляет собой библиотеку языка программирования Python, содержащую набор инструментов для подготовки изображения к распознаванию на нем текста и, соответственно, последующего распознавания текста. Проект разрабатывается на основе лекционного и практического материала по курсу «Обработка аудиовизуальной информации». Библиотека содержит функции, автоматизирующие следующие действия:

- Поэтапная подготовка изображения к распознаванию;
- Определение строк и символов на изображении;
- Непосредственное распознавание символов.

2.2 Цель проекта

Целью процесса разработки является получение модуля, способного осуществлять следующие процессы:

1. Приведение изображения из RGB формата к полутоновому;
2. Приведение полутонового изображения к монохромному методом пороговой обработки;
3. Фильтрация изображений (различными методами) и применение к нему морфологических операций;
4. Выделение контуров на изображении (различными методами);
5. Выделение признаков символов шрифта (вес, центр тяжести итд) с последующей выгрузкой в .csv файл;
6. Выявление строк из абзацев на изображении и сегментация символов в строке;
7. Классификация (распознавание) найденных символов.

Критерием достижимости цели проекта является распознавание символов разработанным модулем на тестовых изображениях с точностью **n%**.

3 Ключевые вехи проекта

В настоящем разделе представлены ключевые моменты разработки настоящего проекта, которые предстоит декомпозировать для включения в спринты в виде конкретных задач. Все нижеперечисленные пункты имеют ориентировочную дату выполнения. Кроме того, элементы данной таблицы отсортированы в порядке приоритета их выполнения в рамках даты.

Табл.1. Перечень решаемых задач и сроки их выполнения

№	Наименование задачи	Срок выполнения
1	Реализация набора функций, осуществляющих приведение изображения из RGB формата к полутоновому	16.10.2019
2	Реализация набора функций, осуществляющих приведение полутонового изображения к монохромному методом пороговой обработки. (различными методами)	24.10.2019
3	Реализация набора функций, осуществляющих фильтрацию изображений (различными методами) и применение к нему морфологических операций	30.10.2019
4	Реализация набора функций, осуществляющих выделение контуров на изображении (различными методами).	04.11.2019
5	Реализация набора функций, осуществляющих выделение признаков символов шрифта (вес, центр тяжести итд) с последующей выгрузкой в .csv файл	10.11.2019
6	Реализация набора функций, осуществляющих выявление строк из абзацев на изображении и сегментация символов в строке.	16.11.2019
7	Реализация набора функций, осуществляющих классификацию найденных символов на основе признаков, анализ профилей.	22.11.2019

8	Реализация объединенной библиотеки, готовой к использованию конечным пользователем.	20.12.2019
---	---	------------

4 Характеристика объекта автоматизации

В качестве объекта автоматизации рассматривается процесс распознавания символов.

Распознавание символов – это операция преобразования текстовой информации из графических форматов данных в текстовые. Применяется для ввода текстов в компьютер посредством сканирования печатного или рукописного материала.

Распознавание текста является одним из направлений распознавания образов. Распознавание образов представляет собой очень сложную задачу в теоретическом и практическом смыслах, несмотря на то, что с ней достаточно легко справляются многие живые организмы и человек. Крайне сложно создать искусственную систему и ее технически реализовать для того, чтобы эффективно выполнять данный процесс. В данном случае, под распознаванием понимается соотнесение изображения объекта, его образа, набора признаков самому объекту.

Чтобы реализовать автоматический или автоматизированный перевод бумажных документов в электронный вид, необходимо выполнить сканирование бумажных документов и распознать их содержимое с помощью разрабатываемой библиотеки.

5 Требования к библиотеке

5.1 Требования к продукту в целом

«Модуль OCR» должен выполнять следующие основные функции:

- Обеспечить возможность загрузки файлов изображений в формате bmp с дальнейшей возможностью их обработки с помощью функционала, описанного в пунктах ниже.
- Сохранение полученных после обработки входных файлов результатов на жёсткий диск или съёмный носитель в качестве изображений или же текстовых файлов.
- Предусмотреть возможность вывода диагностических сообщений, а также входных и выходных данных на экран.

Перечисленный функционал должен быть обеспечен в рамках работ первой очереди. Также он должен быть легко расширяемым и подготовлен для работы с большим количеством инструментов.

5.1.1 Требование к структуре и функционированию библиотеки

«Модуль OCR» должен быть реализован в виде базового класса, выполняющего функции, описанные в пункте 3.1 и независимых друг от друга подклассов, описанных в пунктах 3.3 и далее.

Взаимодействия между подклассами и классами должно осуществляться с помощью специфицированных программных интерфейсов.

5.1.1.1 Перечень подсистем, их назначение и основные характеристики

Для обеспечения выполнения всех функций библиотеки в **Project name**, инструментарий должен быть разбит на следующий части:

- Предобработка, обесцвечивание и бинаризация растровых изображений;
- Фильтрация монохромных изображений и морфологические операции;
- Выделение контуров на изображении;
- Выделение признаков символов;
- Сегментация символов в строке.
- Классификация на основе признаков, анализ профилей

Каждая инструмент независим и может разрабатываться отдельно, но на вход каждого из них ожидаются данные полученных с помощью других инструментов.

5.1.1.2 Предобработка, обесцвечивание и бинаризация растровых изображений

Инструмент предобработки, обесцвечивания и бинаризации растровых изображений предназначен для работы с одним изображением, подающегося на вход с жёсткого диска или съёмного носителя.

Изображение опционально может подвергнуться предварительной обработке при помощи таких преобразований, как растяжение (интерполяция) и сжатия (децимация).

После предобработки изображение может подвергнуться двум операциям:

- Перевод изображения в режим полутона (только для полноцветных изображений), в результате которого создаётся новое изображение в оттенках серого.
- Приведение изображения к монохромному (как для полутоновых, так и для полноцветных) с использованием указанного метода.

После выполнения вышеперечисленных операций изображение сохраняется в файл с названием указанным пользователем.

5.1.1.3 Фильтрация монохромных изображений и морфологические операции

Инструмент фильтрации монохромных изображений и морфологических операций предназначен для работы с набором монохромных изображений и фильтрации его по определённому признаку. В качестве входных данных берётся монохромное изображение (несколько штук).

Дальше изображения фильтруются с использованием указанного метода и полученное отфильтрованное монохромное изображение и разностное изображение сохраняются в место, указанное пользователем или же демонстрируется на экране.

5.1.1.4 Выделение контуров на изображении

Инструмент выделения контуров предназначен для выделения контуров на монохромном изображении. В качестве входных данных берётся одно цветное или полутоновое изображение. В случае цветного изображения — сводится к полутоновому.

Затем изображение проходит обработку с использованием указанного метода. В качестве результата сохраняются несколько файлов: Градиентная матрица и бинаризованная градиентная матрица в виде таблицы (текстового файла), а также полученное изображение

5.1.1.5 Выделение признаков символов

Инструмент выделения признаков символов предназначен для работы с набором монохромных изображений и выделения у них определённого набора признаков. В

качестве входных данных берётся набор(папка) монохромных изображений со значимой информацией (к примеру, символов какого-нибудь алфавита).

Затем для всех изображений рассчитывается следующий набор признаков:

1. Вес (масса чёрного);
2. Удельный вес (вес, нормированный к площади);
3. Координаты центра тяжести;
4. Нормированные координаты центра тяжести;
5. Осевые моменты инерции по горизонтали и вертикали;
6. Нормированные осевые моменты инерции.

Полученный результат записывается в единую таблицу и сохраняется в файл.

5.1.1.6 Сегментация символов в строке

Инструмент сегментации символов в строке предназначен для выделения в монохромном тексте символов (сегментация) и создания профилей для каждого символа. В качестве входных данных берётся монохромное изображение, содержащее произвольный текст без белого фона вокруг текста.

На основе полученного изображения рассчитывается горизонтальный и вертикальный профиль изображения и производится сегментация текста на отдельные символы с использованием алгоритма на основе профилей с прореживанием.

Для каждого символа строится отдельный профиль с информацией о его местоположении на изображении.

В результате создаётся текстовый файл, содержащий профили всех символов из исходного текста, упорядоченные в порядке чтения текста слева направо, сверху вниз.

5.1.1.7 Классификация на основе признаков, анализ профилей

Инструмент классификации на основе признаков, анализ профилей предназначен для анализа файлов, содержащих набор определённых признаков и профилей некоторых символов. На вход подаются результаты, полученные с использованием инструментов из пунктов 3.3.4 и 3.3.5.

Производятся расчёт мер близости между профилями символов текста из 3.3.5 и набором признаков для каждого изображения из 3.3.4 с использованием евклидова расстояния. Нулевое расстояние между изображениями будет соответствовать единичной мере близости.

Каждый расчёт формирует множество гипотез. Гипотезой называется пара, состоящая из символа и его меры близости. Гипотезы должны быть отсортированы в порядке убывания мер близости.

Полученные гипотезы должны быть сохранены в качестве результата в файл.

На экране должны быть выведены символы наиболее достоверных гипотез из каждого расчёта.

5.1.2 Требования к надёжности

5.1.2.1 Состав показателей надёжности для системы в целом

Надёжное (устойчивое) функционирование программы должно быть обеспечено выполнением Заказчиком совокупности организационно-технических мероприятий, перечень которых приведен ниже:

- организацией бесперебойного питания технических средств;
- использованием лицензионного программного обеспечения;
- предварительного обучения пользователей;
- регулярным выполнением рекомендаций Министерства труда и социального развития РФ, изложенных в Постановлении от 23 июля 1998г. Об утверждении межотраслевых типовых норм времени на работы по сервисному обслуживанию ПЭВМ и оргтехники и сопровождению программных средств»;
- регулярным выполнением требований ГОСТ 51188-98. Защита информации. Испытания программных средств на наличие компьютерных вирусов

Время восстановления после отказа, вызванного сбоем электропитания технических средств (иными внешними факторами), не фатальным сбоем (не крахом) операционной системы, не должно превышать 30-ти минут при условии соблюдения условий эксплуатации технических и программных средств.

Время восстановления после отказа, вызванного неисправностью технических средств, фатальным сбоем (крахом) операционной системы, не должно превышать времени, требуемого на устранение неисправностей технических средств и переустановки программных средств.

5.1.2.2 Перечень аварийных ситуаций, по которым регламентируются требования к надёжности

Под аварийной ситуацией понимается аварийное завершение процесса исполнения программного модуля, а также «зависание» этого процесса.

При работе с разрабатываемым «модулем OCR» возможны следующие аварийные ситуации, которые влияют на надёжность работы системы:

- сбой в электроснабжении рабочей станции пользователя;

- ошибки «модуля OCR», не выявленные при отладке и испытании системы;
- сбои программного обеспечения рабочей станции пользователя.

5.1.2.3 Требования к надежности технических средств и программного обеспечения

К надежности оборудования предъявляются следующие требования:

- применение технических средств, соответствующих классу решаемых задач;
- аппаратно-программный комплекс рабочей станции должен иметь возможность восстановления в случаях сбоев.

Надежность аппаратных и программных средств должна обеспечиваться за счёт следующих организационных мероприятий:

- предварительного обучения пользователей и обслуживающего персонала;
- своевременного выполнения процессов администрирования;
- соблюдения правил эксплуатации и технического обслуживания программно-аппаратных средств.

Надежность программного обеспечения подсистем должна обеспечиваться за счёт:

- надежности общесистемного ПО и ПО, разрабатываемого Разработчиком;
- проведением комплекса мероприятий отладки, поиска и исключения ошибок;
- ведением журналов системных сообщений и ошибок по подсистемам для последующего анализа и изменения конфигурации.

5.1.2.4 Требования к методам оценки и контроля показателей надежности на разных стадиях создания системы в соответствии с действующими нормативно-техническими документами

Оценка надежности осуществляется на стадии проектирования за счет анализа технических решений по построению системы и их соответствия требованиям данного ТЗ.

Текущий контроль показателей надежности должен быть организован в процессе эксплуатации системы за счет анализа работоспособности разрабатываемого модуля.

5.2 Требование к функциям(задачам), выполняемые компонентом

5.2.1 Требования к функциям инструмента предобработки, обесцвечивания и бинаризации растровых изображений

Инструмент предобработки, обесцвечивания и бинаризации растровых изображений должен обеспечивать выполнение следующих функций:

- Растяжение(интерполяция) входного изображения в М раз.
- Сжатие(децимация)входного изображения в N раз.
- Перевод полноцветного изображения (3 цветовых канала) к изображению в режиме полутона (1 яркостный канал) путём усреднения по каналам.
- Перевод полутонового изображения к монохромному с использованием одного из доступных алгоритмов.

5.2.2 Требования к функциям инструмента фильтрации монохромных изображений и морфологических операций

Инструмент фильтрации монохромных изображений и морфологических операций должен обеспечивать выполнение следующих функций:

- Получение отфильтрованного изображение из входных монохромных изображений с использованием указанного алгоритма фильтрации.
- Получение разностного изображения путём применения модуля разности или же попиксельного исключаящего ИЛИ.

5.2.3 Требования к функциям инструмента выделения признаков в строке

Инструмент выделения признаков в строке должен обеспечивать выполнение следующих функций:

- Автоматический перевод входного цветного изображения к полутоновому (в случае если на вход пришло цветное изображение)
- Получение градиентной матрицы из полутонового изображения с помощью указанного оператора.
- Бинаризация градиентной матрицы

5.2.4 Требования к функциям инструмента выделения символов в строке

Инструмент выделения символов в строке должен обеспечивать выполнение следующих функций:

- Расчёт для каждого входного изображения набора следующих признаков
 - Вес (масса чёрного);
 - Удельный вес (вес, нормированный к площади);
 - Координаты центра тяжести;
 - Нормированные координаты центра тяжести;
 - Осевые моменты инерции по горизонтали и вертикали;
 - Нормированные осевые моменты инерции.

- Сохранение признаков всех изображений в текстовых файл формата *.csv (таблица, где значения в строках разделены точкой с запятой)

5.2.5 Требования к функциям инструмента сегментация символов в строке

Инструмент сегментации символов в строке должен обеспечивать выполнение следующих функций:

- Расчёт горизонтального профиля монохромного изображения;
- Расчёт вертикального профиля монохромного изображения;
- Сегментация символов в строке на основе профилей с прореживанием и создание профилей всех символов в строке.

5.2.6 Требования к функциям инструмента классификации на основе признаков, анализ профилей

Инструмент классификации на основе признаков должен обеспечивать выполнение следующих функций:

- Расчёт меры близости изображения символов на основе признаков с использованием евклидова расстояния в n-мерном пространстве нормализованных признаков;
- Расчёт мер близости для каждого обнаруженного символа со всеми символами выбранного алфавита и создание упорядоченного массива с гипотезами;
- Сохранения всех гипотез в один единый текстовый файл;
- Вывод лучшей гипотезы.

5.3 Требования к видам программного обеспечения

5.3.1 Программное обеспечение

Разрабатываемый модуль оптического распознавания текста должен быть написан на языке Python версии 3.5 или выше. Для создания модуля в качестве дополнительного инструментария могут быть использованы следующие библиотеки обработки изображений и алгоритмов компьютерного зрения:

- OpenCV – версии 4 и выше;
- Pillow - версии 6 и выше;
- SciPy - версии 1.3.1 и выше;
- NumPy - версии 1.17 и выше;
- SimpleITK- версии 1.2.0 и выше.

Для написания программного кода могут быть использованы следующие программные среды разработки:

- Visual Studio;
- Pycharm;
- Eclipse.

Для управления процессом командного написания исходного кода модуля оптического распознавания текста может быть использована распределенная система управления версиями GIT в совокупности с веб-сервисами GitHub и GitLab (на выбор разработчиков).

Конечный продукт будет распространяться в виде пакета либо модуля с документированным функционалом.

5.3.2 Математическое обеспечение

Любые математические методы и алгоритмы, используемые для шифрования/дешифрования данных используемые в данном ПО или реализующее их, должны быть сертифицированы.

Для работы с файлами необходимо использовать стандартные методы, представляемые установленным языком программирования или уже дополнительных библиотек, оговоренных в требованиях к техническому обеспечению.

Операции для работы с изображениями, такие как получение цветовых компонент заданного пикселя и их изменение, считывание из файла и сохранение в файл должны использоваться посредством дополнительной библиотеки для работы с изображениями и не реализовываться командой разработки в рамках данного проекта.

То же относится к алгоритмам и методам для работы с большими массивами данных.

Любые другие алгоритмы, отсутствующие в стандарте выбранного языка программирования и в вспомогательных библиотеках должны быть реализованы самостоятельно командой разработчиков без привлечения дополнительного функционала.

5.3.3 Информационное обеспечение

Состав, структура и способы организации данных в системе должны быть определены на этапе технического проектирования.

Уровень хранения данных в системе должен быть построен на уровне работы программы, в оперативной памяти. При закрытии программы, использующей библиотеку распознавания изображений все несохраненные данные должны удаляться из памяти.

Средства библиотеки, а также средства используемых операционных систем должны обеспечивать документирование и протоколирование обрабатываемой в системе информации.

Структура внутренних компонентов должна быть организована рациональным способом.

Технические средства, обеспечивающие хранение информации, должны использовать современные технологии, позволяющие обеспечить повышенную надежность хранения данных и оперативную замену

6 Состав и содержание работ по созданию системы

Работы по разработке библиотеки языка программирования Python «Модуль OCR (оптического распознавания текста)» производятся объединением инициативных студентов групп М18-504, М18-514 и М18-Ш04. Разработка должна включать следующие работы со стороны команды разработки:

- Разработка технического задания, содержащего требования по разработке библиотеки,
- Разработка самой библиотеки языка программирования Python

Согласно ГОСТ 34.601-90 «Автоматизированные системы. Стадии создания» выделяют следующие основные стадии создания и этапы разработки автоматизированной системы (АС):

- Формирование требований к АС;
- Разработка концепции АС;
- Техническое задание;
- Эскизный проект;
- Рабочая документация;
- Ввод в действие.

Ниже представлены стадии создания библиотеки языка программирования Python «Модуль OCR. Также дана оценка длительности каждого из этапов внедрения системы, которая может не совпадать с реальными сроками.

6.1 Стадии проекта

Стадия I. Организация проекта

Этапы работ	Результат	Дней
Согласование процедур управления	Процедуры управления проектом и устав проекта согласованы	3
Сбор команды проекта	Команда проекта сформирована	3
Обучение членов проектной команды	Компетенция участников проекта соответствует требованиям.	5

Стадия II. Формирование требований и разработка технического задания

Этапы работ	Результат	Дней
Системно-аналитическое обследование области распознавания текстов	Изучены лекционные и практические материалы по курсу «Обработка аудиовизуальной информации»	20
Разработка технического задания	Техническое задание и приложения к нему	10
Согласование и утверждение	Согласованное и утвержденное техническое задание	5

Стадия III. Эскизный проект

Этапы работ	Результат	Дней
Определение общей функциональной и технической архитектур	Общее описание функциональной и технической архитектур	10
Обоснование выбора программного обеспечения и технической инфраструктуры	ПО выбрано. Утвержден план формирования необходимой технической инфраструктуры	10

Эскизное проектирование процессов и алгоритмов, используемых в библиотеке	Общее описание процессов загрузки, извлечения данных, алгоритмов преобразования изображений	10
Эскизное проектирование интерфейсов пользователя	Общее описание интерфейсов ввода и предоставления данных	5
Согласование и утверждение	Оформленная, согласованная и утвержденная пояснительная записка к эскизному проекту	10

Стадия IV. Технический проект

Этапы работ	Результат	Дней
Разработка набора функций, осуществляющих приведение изображения из RGB формата к полутоновому	Набор реализованных целевых функций и их описание	7
Разработка набора функций, осуществляющих приведение полутонового изображения к монохромному методом пороговой обработке. (различными методами)	Набор реализованных целевых функций и их описание	7
Реализация набора функций, осуществляющих фильтрацию изображений (различными методами) и применение к нему морфологических операций	Набор реализованных целевых функций и их описание	7

Реализация набора функций, осуществляющих выделение контуров на изображении (различными методами).	Набор реализованных целевых функций и их описание	7
Реализация набора функций, осуществляющих выделение признаков символов шрифта (вес, центр тяжести) с последующей выгрузкой в .csv файл	Набор реализованных целевых функций и их описание	7
Реализация набора функций, осуществляющих выявление строк из абзацев на изображении и сегментация символов в строке.	Набор реализованных целевых функций и их описание	7
Реализация набора функций, осуществляющих классификацию найденных символов на основе признаков, анализ профилей.	Набор реализованных целевых функций и их описание	7
Реализация объединенной библиотеки, готовой к использованию конечным пользователем.	Набор реализованных целевых функций и их описание	7

Стадия V. Рабочая документация

Этапы работ	Результат	Дней
-------------	-----------	------

Разработка рабочей документации на систему и на её части	<p>Разработаны следующие документы:</p> <ul style="list-style-type: none"> · Общее описание системы · Описание технологического процесса обработки данных · Состав выходных данных · Программа и методика испытаний (ПИМ) 	40
Согласование и утверждение	Согласованная и утвержденная рабочая документация	15

Стадия VI. Ввод в действие

Этапы работ	Результат	Дней
Подготовка персонала	Обучение пользователей системы	1
Проведение предварительных испытаний	<p>Испытания системы на работоспособность и соответствие техническому заданию проведены.</p> <p>Устранение неисправностей и внесение изменений в документацию.</p> <p>Акт приема со стороны сертифицирующего органа получен.</p>	5
Проведение опытной эксплуатации	<p>Опытная эксплуатация проведена.</p> <p>Устранение неисправностей, доработка ПО и дополнительная наладка технических средств проведены</p>	5
Завершение работ	Акт завершения работ подписан	1

7 Порядок контроля и приемки

7.1 Цель испытаний

Целью проводимых испытаний библиотеки является определение функциональной работоспособности библиотеки на этапе проведения испытаний.

Программа испытаний должна удостоверить работоспособность библиотеки в соответствии с функциональным назначением, ее соответствие функциональным и техническим требованиям.

7.2 Перечень характеристик библиотеки подлежащих проверке

Испытания проводятся в один этап, в ходе которого предполагается проверить следующий перечень функциональных возможностей библиотеки:

- Библиотека должна осуществлять загрузку файлов изображений в формате bmp с дальнейшей возможностью их обработки, а также обеспечивать вывод выходных и входных изображений на экран;
- Библиотека должна предоставлять функции предобработки изображения (сжатие и растяжение), обесцвечивания, бинаризации и фильтрации растровых изображений, а также выделения контуров на изображении;
- Библиотека должна предоставлять инструмент распознавания символов на изображении с текстом;
- Библиотека должна сохранять полученные после обработки входные файлы результатов на жёсткий диск или съёмный носитель в виде изображений или же текстовых файлов.

7.3 Перечень проводимых проверок

7.3.1 Загрузка файлов изображений

Номер:	1.1	
Название:	Загрузка файлов изображений	
Действие	Ожидаемый результат	
Предусловие:		
1. Сохранить изображение для проведения теста	Вывод на экран сообщения об успешном создании экземпляра класса обработчика изображений	
2. Скопировать путь до выбранного изображения		
3. Подключить модуль библиотеки		

4. Создать экземпляр класса обработчика изображений	
Шаги теста:	
5. Вызвать функцию загрузки изображения экземпляра класса обработчика, аргумент – путь до выбранного изображения	Вывод на экран сообщения об успешной загрузке изображения
6. Вызвать функцию вывода изображения на экран экземпляра класса обработчика	Вывод на экран загруженного изображения

7.3.2 Предобработка и сохранение изображений

Номер:	1.2
Название:	Предобработка изображений
Действие	Ожидаемый результат
Предусловие:	
1. Сохранить изображение для проведения теста	Вывод на экран сообщений об успешном создании экземпляра класса обработчика изображений и успешной загрузке изображения
2. Скопировать путь до выбранного изображения	
3. Подключить модуль библиотеки	
4. Создать 2 экземпляра класса обработчика изображений	
5. Вызвать функцию загрузки изображения для первого и второго экземпляра класса обработчика, аргумент – путь до выбранного изображения	
Шаги теста:	
6. Вызвать функцию сжатия изображения первого экземпляра класса обработчика, аргумент n – коэффициент сжатия	Сохранение уменьшенного в n раз изображения по указанному пути
7. Вызвать функцию сохранения изображения второго экземпляра класса обработчика, аргумент – путь до нового изображения	

8. Вызвать функцию растяжения изображения второго экземпляра класса обработчика, аргумент n – коэффициент сжатия	Сохранение увеличенного в n раз изображения по указанному пути
9. Вызвать функцию сохранения изображения второго экземпляра класса обработчика, аргумент – путь до нового изображения	

7.3.3 Обесцвечивание изображения

Номер:	1.3
Название:	Обесцвечивание изображения
Действие	Ожидаемый результат
Предусловие:	
1. Сохранить изображение для проведения теста	Вывод на экран сообщений об успешном создании экземпляра класса обработчика изображений и успешной загрузке изображения
2. Скопировать путь до выбранного изображения	
3. Подключить модуль библиотеки	
4. Создать экземпляр класса обработчика изображений	
5. Вызвать функцию загрузки изображения экземпляра класса обработчика, аргумент – путь до выбранного изображения	
Шаги теста:	
6. Вызвать функцию обесцвечивания изображения экземпляра класса обработчика	Сохранение обесцвеченного изображения по указанному пути
7. Вызвать функцию сохранения изображения, аргумент – путь до нового изображения	

--	--

7.3.4 Бинаризация изображения

Номер:	1.4
Название:	Бинаризация изображения
Действие	Ожидаемый результат
Предусловие:	
1. Сохранить изображение для проведения теста	Вывод на экран сообщений об успешном создании экземпляра класса обработчика изображений и успешной загрузке изображения
2. Скопировать путь до выбранного изображения	
3. Подключить модуль библиотеки	
4. Создать экземпляр класса обработчика изображений	
5. Вызвать функцию загрузки изображения экземпляра класса обработчика, аргумент – путь до выбранного изображения	
Шаги теста:	
6. Вызвать функцию бинаризации изображения экземпляра класса обработчика, аргумент – название метода бинаризации	Сохранение бинаризованного изображения по указанному пути
7. Вызвать функцию сохранения изображения, аргумент – путь до нового изображения	
7. Повторить шаги 5, 6, 7 для всех методов бинаризации (на каждой итерации менять аргумент функции на шаге 6 – название метода бинаризации)	Сохранение бинаризованных изображений по указанным путям

7.3.5 Фильтрация изображения

Номер:	1.5
Название:	Фильтрация изображения

Действие	Ожидаемый результат
Предусловие:	
1. Сохранить изображение для проведения теста	Вывод на экран сообщений об успешном создании экземпляра класса обработчика изображений и успешной загрузке изображения
2. Скопировать путь до выбранного изображения	
3. Подключить модуль библиотеки	
4. Создать экземпляр класса обработчика изображений	
5. Вызвать функцию загрузки изображения экземпляра класса обработчика, аргумент – путь до выбранного изображения	
6. Вызвать функцию бинаризации изображения	Вывод на экран бинаризованного изображения
7. Вызвать функцию вывода изображения на экран экземпляра класса обработчика	
Шаги теста:	
8. Вызвать функцию фильтрации изображения экземпляра класса обработчика, аргумент – название метода фильтрации	Сохранение отфильтрованного изображения по указанному пути
8. Вызвать функцию сохранения изображения, аргумент – путь до нового изображения	
9. Повторить шаги 5, 6, 7 для всех методов фильтрации (на каждой итерации менять аргумент функции на шаге 6 – название метода фильтрации)	Сохранение отфильтрованных изображений по указанным путям

7.3.6 Выделение контуров на изображении

Номер:	1.6
Название:	Выделение контуров на изображении
Действие	Ожидаемый результат
Предусловие:	
1. Сохранить изображение для проведения теста	Вывод на экран сообщений об успешном создании экземпляра класса обработчика изображений и успешной загрузке изображения
2. Скопировать путь до выбранного изображения	
3. Подключить модуль библиотеки	
4. Создать экземпляр класса обработчика изображений	
5. Вызвать функцию загрузки изображения экземпляра класса обработчика, аргумент – путь до выбранного изображения	
Шаги теста:	
6. Вызвать функцию выделения контуров на изображении экземпляра класса обработчика, аргумент – название метода выделения контуров	Сохранение файла с таблицей градиентной матрицы
7. Вызвать функцию сохранения градиентной матрицы, аргумент – название файла	
8. Вызвать функцию бинаризации изображения	Сохранение изображение с выделенными контурами исходного изображения
9. Сохранить изображение	
10. Повторить шаги 5, 6, 7, 8, 9 для всех методов выделения контуров на изображении (на каждой итерации менять аргумент функции на шаге 6 – название метода выделения контуров)	Сохранение файла с таблицей градиентной матрицы и изображения с выделенными контурами исходного изображения

7.3.7 Распознавание символов на изображении с текстом

Номер:	1.7	
Название:	Распознавание символов на изображении с текстом	
Действие	Ожидаемый результат	
Предусловие:		
1. Сохранить изображение с текстом для проведения теста	Вывод на экран сообщений об успешном создании экземпляра класса обработчика изображений и успешной загрузке изображения	
2. Скопировать путь до выбранного изображения		
3. Подключить модуль библиотеки		
4. Создать экземпляр класса обработчика изображений		
5. Вызвать функцию загрузки изображения экземпляра класса обработчика, аргумент – путь до выбранного изображения		
Шаги теста:		
6. Вызвать функцию сегментации символов на изображении экземпляра класса обработчика	Сохранение файла с текстом на выбранном изображении	
9. Вызвать функцию классификации сегментированных символов		
10. Вызвать функцию сохранения, полученного текста		

7.4 Последовательность проведения испытаний

Испытания проводятся путем выполнения проверок функций указанных в п. 4.2, в соответствии со сценариями п. 4.3. По результатам проведения проверок заполняется протокол испытаний.

На испытательном стенде, предварительно, устанавливается программное обеспечение, в соответствии с п. 3.5, подготавливаются изображения в заданном формате, структура которых позволяет оценивать степень соответствия реализуемых библиотекой функций заявленным функциональным требованиям.

7.5 Перечень работ, проводимых после завершения испытаний, требования к ним, объем и порядок проведения

После завершения испытаний и подписания Протокола испытаний оформляется Акт о готовности библиотеки к вводу в эксплуатацию.

Акт о готовности библиотеки к вводу в эксплуатацию содержит вывод о степени работоспособности библиотеки.

Если при проведении испытаний выявлены замечания, Исполнителем должны быть выполнены работы по устранению замечаний и выполнению рекомендаций в объеме, указанном в Протоколе испытаний.

Работы по устранению замечаний и выполнению рекомендаций проводятся в произвольном порядке, если иное не указано в Акте о готовности библиотеки к вводу в эксплуатацию.

7.6 Порядок проведения испытаний

Испытания библиотеки должны проводиться на оборудовании соответствующем техническим требованиям. Оборудование должно быть предоставлено в той конфигурации, которая достаточна для развёртывания библиотеки, и указана в п. 3.5.

Во время испытаний проводится полное функциональное тестирование, согласно заявленным функциональным требованиям.

В ходе проведения испытаний специалисты работают с библиотекой, в соответствии с п. 4.4, подвергая тем самым библиотеку полнофункциональному тестированию в течение установленного срока.

7.7 Условия начала и завершения отдельных этапов испытаний

Условием начала проведения испытаний является наличие работоспособной библиотеки.

Условием окончания испытаний является выполнение всех проверок, определенных в п. 4.3 и занесение результатов их выполнения в Протокол испытаний.

Для обеспечения безаварийности при проведении испытаний на вычислительной технике, используемой для проведения испытаний, не должны выполняться другие задачи, требующие значительных ресурсов программного и технического обеспечения.

7.8 Требования к персоналу, проводящему испытания, и порядок его допуска к испытаниям

Специалисты, проводящие испытания, должны иметь навыки разработки на языке программирования Python, работы с операционной библиотекой Windows.

7.9 Материально-техническое обеспечение испытаний

Для проведения испытаний используются следующие виды аппаратно-технического обеспечения:

- АРМ пользователя – обеспечивает возможность пользователя осуществлять взаимодействие с библиотекой.

7.9.1 Рабочее место

- ПК в составе АРМ пользователя;
- ОС Microsoft Windows 10 или новее;
- программное обеспечение предусмотренное пунктом 3.5;

Программа испытаний не требует использования специализированного измерительного оборудования.

7.10 Отчетность

Результаты испытаний библиотеки, предусмотренные настоящей программой, фиксируются в протоколе, содержащем следующие разделы:

- Назначение испытаний и номер сценария, по которому проводят испытание;
- Состав технических и программных средств, используемых при испытаниях;
- Указание методик, в соответствии с которыми проводились испытания, обработка и оценка результатов;
- Условия проведения испытаний и характеристики исходных данных;
- Обобщённые результаты испытаний;
- Выводы о результатах испытаний и соответствии созданной библиотеки определённому разделу требований.

В протоколах могут быть занесены замечания персонала по удобству эксплуатации библиотеки.

Этап проведения предварительных испытаний завершается оформлением «Акта предварительных испытаний библиотеки».

8 Перечень сокращений

Сокращение	Значение
ТЗ	Техническое задание
АРМ	Автоматизированное рабочее место
ПК	Персональный компьютер
ОС	Операционная библиотека
ПС	Программное средство