Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Ярославский государственный технический университет»

Кафедра «Информационные системы и технологии»

УДК 004.932 ДОПУСКАЕТСЯ К ЗАЩИТЕ

Заведующий кафедрой

канд. техн. наук

\_\_\_\_\_\_\_\_С.Ю. Бойков

«\_\_»\_\_\_\_\_\_\_\_\_\_\_2022

**ИССЛЕДОВАНИЕ ВОЗМОЖНОСТИ АВТОМАТИЧЕСКОГО РАСПОЗНАВАНИЯ ГРУПП РИСКА ПОСРЕДСТВОМ ЦВЕТОВОГО АНАЛИЗА ЦИФРОВЫХ ФОТОГРАФИЙ КОЖНЫХ ОБРАЗОВАНИЙ**

Пояснительная записка к выпускной квалификационной работе по направлению подготовки «Информационные системы и технологии»

ЯГТУ 09.04.02 – 010 ВКР

СОГЛАСОВАНО

|  |  |
| --- | --- |
| Руководитель  канд. физ.-мат. наук, доцент  \_\_\_\_\_\_\_\_А.Б. Раухваргер «\_\_»\_\_\_\_\_\_\_\_\_\_\_2022 | Нормоконтролер  канд. физ.-мат. наук, доцент  \_\_\_\_\_\_\_\_ А.Б. Раухваргер «\_\_»\_\_\_\_\_\_\_\_\_\_\_2022 |
|  | Работу выполнил  студент группы ЭМИС-24м  \_\_\_\_\_\_\_\_\_\_\_ В.А. Матвеев  «\_\_»\_\_\_\_\_\_\_\_\_\_\_2022 |

2022

Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«ЯРОСЛАВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

1. Кафедра «Информационные системы и технологии»

УТВЕРЖДАЮ

зав.кафедрой

*канд. техн. наук*

(уч. степень, звание)

\_\_\_\_\_\_\_*Бойков С.Ю.*

(Ф.И.О., подпись)

**ЗАДАНИЕ № 10**

**ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА**

1.Выдано студенту *Матвееву Виталию Александровичу*

2.Тема: *Исследование возможности автоматического распознавания групп риска посредством цветового анализа цифровых фотографий кожных образований*

утверждена приказом по университету от *24.11.21* № *1660/3*

3. Исходные данные

1. *Исходные сведения о предметной области, выраженные в виде документации организации, других информационных источников о структуре технологических, производственных и управленческих процессов*
2. *Справочная и научно-техническая литература*

4. Содержание расчетно-пояснительной записки (перечень подлежащих разработке вопросов)

1. *Общая характеристика предметной области.*
2. *Обоснование необходимости и целей автоматизации рассматриваемых задач.*
3. *Обоснование решений по информационному и программному обеспечению.*
4. *Реализация проектных решений в выбранной инструментальной среде.*

5. Перечень графического материала (с точным указанием обязательных чертежей)

1. *Иллюстративные материалы представляются студентом на защиту в виде компьютерной презентации в формате MS PowerPoint*

6. Консультанты (с указанием относящихся к ним разделов выпускной квалификационной работы)

7. Нормоконтролер *доцент кафедры ИСТ, канд. физ.-мат.. наук, доцент Раухваргер А. Б.*

8. Срок сдачи законченной выпускной квалификационной работы *25.05.2022*

9. Дата выдачи задания *24.11.2021*

**Руководитель** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись)

**Задание принял к исполнению**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись студента)

РЕФЕРАТ

60 с., 16 рис., 21 табл., 8 источников

ИССЛЕДОВАНИЕ, ЦИФРОВОЕ ИЗОБРАЖЕНИЕ, ПИКСЕЛЬ, ОНКОЛОГИЯ, ОБРАБОТКА ЦИФРОВЫХ ИЗОБРАЖЕНИЙ.

Целью выпускной квалификационной работы является исследование возможности автоматического распознавания групп риска посредством цветового анализа цифровых фотографий кожных образований.

Чтобы выполнить поставленную цель, необходимо решить следующие задачи:

* произвести анализ предметной области;
* выбрать требуемые средства для разработки программы;
* разработать базу данных для хранения выделенных точек изображения;
* разработать программу для вычисления характеристик изображения;
* сделать контекстные модели представления данных и описать их подробную спецификацию;
* изобразить работу алгоритмов в виде блок-схем;
* исследовать изображения кожных заболеваний на наличие закономерностей.

Содержание

[1 Аналитическая часть 6](#_Toc104566201)

[1.1 Характеристика объекта исследования 6](#_Toc104566202)

[1.2 Стек используемых технологий 7](#_Toc104566203)

[1.2.1 Плюсы и минусы языка Java 7](#_Toc104566204)

[1.2.2 Плюсы и минусы языка C# 7](#_Toc104566205)

[1.2.3 Плюсы и минусы языка Python 8](#_Toc104566206)

[1.3 Анализ существующих решений 9](#_Toc104566207)

[1.3.1 Платформа «Про родинки» 9](#_Toc104566208)

[1.3.2 Платформа «Celsus» 11](#_Toc104566209)

[1.4 Постановка цели и задачи 12](#_Toc104566210)

[1.5 Вывод по аналитической части 12](#_Toc104566211)

[2 Проектная часть 14](#_Toc104566212)

[2.1 Описание предметной области 14](#_Toc104566213)

[2.2 Концептуальное проектирование 14](#_Toc104566214)

[2.2.1 Варианты использования 16](#_Toc104566215)

[2.2.2 Диаграмма классов 20](#_Toc104566216)

[2.3 Блок-схема работы программы 25](#_Toc104566217)

[2.4 Выбор системы управлению базой данных 30](#_Toc104566218)

[2.5 Физическая модель базы данных 32](#_Toc104566219)

[2.6 Вывод по проектной части 36](#_Toc104566220)

[3 Технологическая часть 37](#_Toc104566221)

[3.1 Взаимодействие базы данных и программы 37](#_Toc104566222)

[3.2 Используемые классы объектов 38](#_Toc104566223)

[3.3 Описание работы программы 40](#_Toc104566224)

[3.4 Вывод по технологической части 45](#_Toc104566225)

[4 Исследовательская часть 46](#_Toc104566226)

[4.1 Входные данные 46](#_Toc104566227)

[4.2 Данные и инструменты анализа 49](#_Toc104566228)

[4.3 Вывод по исследовательской части 56](#_Toc104566229)

[Заключение 57](#_Toc104566230)

[Список используемой литературы 58](#_Toc104566231)

Введение

Выбранная тема «Исследование возможности автоматического распознавания групп риска посредством цветового анализа цифровых фотографий кожных образований» актуальна в наше время, так как важно обнаружить и диагностировать онкологию на ранних стадиях.

По ходу обработки используется алгоритм приведения изображения к средней яркости. Это необходимо для исследования всех экземпляров заболеваний в равных условиях, на которые не будет влиять освещенность, на момент создания фотографии.

Проводя обработку изображения, на гистограммах яркости (0-255) можно экспертным способом наблюдать какой из параметров частоты цветовой модели R, G, B возрастает или убывает, а также где именно это происходит. Из дополнительных, доступных данных для исследования, следует обращать внимание на значения средней яркости, среднеквадратичное отклонение и медиану каждой шкалы R, G, B.

Предполагается, что, владея данными, описанными ранее, можно будет определить предположительное заболевание онкологического характера.

# Аналитическая часть

## Характеристика объекта исследования

Главный объект исследования представляет собой экземпляры цифровых фотографий, которые необходимо проанализировать для установления возможных закономерностей на основе цветовой модели RGB.

Необходимость в диагностике онкологических заболеваний с помощью автоматизированных средств подтвердили между собой кафедра информационных технологий, онкологическая больница и медицинский институт.

На данный момент, предполагается исследование для следующих групп онкологических заболеваний [1-3]:

* Базальноклеточный рак – это наиболее часто встречающаяся y человека раковая опухоль. Состоит из клеток, подобных клеткам базального слоя эпидермиса. От других раков кожи отличается чрезвычайно редким метастазированием, однако способна к обширному местному росту, который приводит к существенным косметическим и функциональным нарушениям.
* Меланома – это злокачественная опухоль, развивающаяся из меланоцитов - пигментных клеток, продуцирующих меланины. Наряду с плоскоклеточным и базальноклеточным раком кожи относится к злокачественным опухолям кожи. Преимущественно локализуется в коже, реже - сетчатке глаза, слизистых оболочках. Одна из наиболее опасных злокачественных опухолей человека, часто рецидивирующая и метастазирующая лимфогенным и гематогенным путём почти во все органы.

Онкология – это раздел медицины, изучающий доброкачественные и злокачественные опухоли, механизмы и закономерности их возникновения и развития, методы их профилактики, диагностики и лечения.

Опухоль – патологическое образование, самостоятельно развивающееся в органах и тканях, отличающееся автономным ростом, полиморфизмом и атипией клеток.

Автономный рост – опухолевый рост не подлежит воздействию регуляторных механизмов (нервная и эндокринная регуляция, иммунная система и др.), т.е. не контролируется организмом.

Полиморфизм и атипия – трансформировавшись, клетки начинают размножаться быстрее, при этом выходит повреждение дифференцировки клеток, что приводит к атипии (морфологическому отличию с клеток ткани, из которой они произошли) и полиморфизму (наличию в опухоли в различных клетках). Чем менее дифференцированные клетки в опухоли, тем быстрее и агрессивнее ее рост.

Основные исторические теории происхождения опухолей:

* Теория раздражения Р. Вирхова – злокачественные опухоли чаще возникают в тех органах, где ткани чаще подвергаются травматизации.
* Теория зародышевых зачатков Д. Конгейма – под влиянием определенных факторов зачатки, находящиеся в латентном состоянии, начинают расти, приобретая опухолевые свойства.
* Регенерационно-мутационная теория Фишер-Вазельса – воздействие патологических факторов на регенерирующие ткани.
* Вирусная теория Л.А.Зильбера – вирус, внедряясь в клетку, на генном уровне нарушает процессы регуляции деления.
* Иммунологическая теория – нарушение идентификации и уничтожения трансформированных клеток иммунной системой.

Часто причиной возникновения злокачественных опухолей являются канцерогенные факторы внешней среды и образа жизни:

* курение - 25-40%;
* особенности питания - 35%;
* инфекционные агенты - 16%;
* низкая физическая активность - 4-5%
* профессиональные канцерогены - 4-5%;
* ионизирующее излучение - 4-5%;
* наследственные факторы - 4-5%;
* ультрафиолетовое излучение - 2-3%;
* загрязнение атмосферного воздуха - 1-2%.

Само по себе воздействие токсичных факторов не всегда знаменует появление новообразования. Для зарождения опухоли необходимо наличие генетической предрасположенности и определенного состояния иммунной и нейрогуморальной систем.

Диагностика онкологических заболеваний:

* Ранняя – установление анамнеза на стадии «in situ» и I клинической стадии заболевания – адекватное излечение приводит к полнейшему выздоровлению.
* Своевременная – диагноз поставлен на II и в некоторых случаях на III стадии процесса – полное выздоровление возможно только у некоторых больных, у других наблюдается развитие процесса.
* Поздняя – диагноз поставлен на III-IV стадии – низкая вероятность или невозможность излечения пациента.

Среди всех видов диагностических исследований, начиная с визуального досмотра пациента врачом любой специальности и завершая такими современными методами как компьютерная томография (КТ), магнитно-резонансная томография (МРТ), позитронно-эмиссионная томография (ПЭТ), лабораторная диагностика занимает особенное место.

Рекомендованные исследования к выполнению:

* анализ крови клинический с определением СОЭ (код 416, код 419);
* анализ мочи общий (код 458);
* исследование кала на скрытую кровь (код 462).

Именно лабораторные тесты, описанные выше, выявят первые признаки скрытого внутреннего кровотечения, которым не редко сопровождается развитие злокачественного новообразования. Микрогематурия (следы крови в моче) есть первый и надежный признаком рака мочевого пузыря. Скрытая (прошедшая ферментативную обработку) кровь в кале означает наличие кровоточащей язвы, полипа или опухоли в желудочно-кишечном тракте. Анализ крови покажет первые признаки заболеваний кроветворной и лимфатической систем (лейкоз, лимфогранулематоз, лимфому и др.).

Онкомаркеры – это вещества, чаще белковой природы, которые возникают в результате жизнедеятельности раковых клеток и появляются в крови онкологических больных. В онкологии онкомаркеры часто используются для отслеживания течения установленного заболевания и анализа эффективности проводимого лечения, но нельзя приуменьшать роль онкомаркеров при первой диагностике рака. Помимо популярных и поместно применяемых онкомаркеров за последнее время были исследованы и рекомендованы к использованию новые и очень удобные в работе тесты.

Наиболее известные и часто применяемые онкомаркеры:

* РЭА - маркер рака ЖКТ и других систем организма.
* ПСА - маркер рака предстательной железы.
* СА 15-3 - маркер рака молочной железы.
* СА-125 - маркер рака яичников и матки.
* SCC - маркер рака шейки матки, прямой кишки, кожи
* CYFRA- маркер рака мочевого пузыря
* АФП в гинекологии - один из основных маркеров состояния плода при беременности. В онкологии - маркер первичного рака печени и яичек.

## Стек используемых технологий

Для реализации исследования будут рассмотрены несколько языков программирования. Основными критериями будут выступать задачи исследования, описанные выше. Языками программирования, среди которых было необходимо выбирать, были следующие: Java, C#, Python.

### Плюсы и минусы языка Java

Java – это язык программирования общего назначения, объектно-ориентированный, является актуальным спустя 20 лет. Используется везде: мобильные, сетевые, настольные и корпоративные приложения.

Язык Java начал свой путь в начале 1990-х, компания Sun Microsystems начала разработку улучшенной версии C++, то есть независимую от одной платформы, автоматическим управлением памятью и достаточно простым в освоении.

Плюсы языка Java:

* Java включает в себя объектно-ориентированное программирование;
* Java – язык высокого уровня с простым синтаксисом;
* отсутствие указателей, что означает повышенную безопасность;
* Java дешевый язык и работать с ним можно с любого компьютера, он поддерживается в большом числе компаний;
* Java является кроссплатформенным языком;
* встроенная дистрибуция;
* автоматическое управление памятью;
* многопоточность;
* встроенная поддержка работы в сетях;
* мощные стандартные библиотеки;
* поддержка Oracle и большое сообщество.

Минусы языка Java:

* Низкая производительность;
* платное коммерческое использование;
* отсутствие нативного дизайна;
* повышенные требования к объему оперативной памяти;

### Плюсы и минусы языка C#

Язык программирования C# в наше время является актуальным и популярным. Так как C# принадлежит компании Microsoft, то он применяется почти во всех продуктах, которые были разработаны или куплены.

Основные положительные стороны языка программирования C#:

* Как и было сказано выше, C# поддерживается и успешно развивается Microsoft и по сегодняшний день;
* типы данных имеют фиксированный размер, что повышает мобильность;
* автоматическая очистка памяти
* синтаксис похож на C, C++ или Java, поэтому C# является перспективным языком для изучения;
* среда программирования Visual Studio, на которой, в основном, используют C#, распространяется бесплатно и с открытым исходным кодом для небольших компаний и частных лиц;
* благодаря приобретению Xamarin, на C# можно писать код под Android и iOS;
* большое сообщество программистов по всему миру;
* наличие синтаксического сахара;
* подсказки и подробное описание каждой функции;
* наличие объектно-ориентированного программирования;
* большое количество вакантных мест на должность C#-программиста.

Минусы языка программирования C#:

* Главная ориентированность - .Net (платформа Windows);
* сохранился оператор безусловного перехода go to.
* Бесплатно распространяется только на частные лица, небольшие компании и учащихся.

### Плюсы и минусы языка Python

Python – высокоуровневый объектно-ориентированный язык общего назначения, который используется в том числе и для разработки веб-приложений. Язык ориентирован на повышение производительности разработчика и читаемости кода.

Плюсы языка Python:

* Низкий порог вхождения;
* простой синтаксис;
* кроссплатформенный язык;
* большое сообщество и поддержка;
* широкое применение;
* востребованность на рынке труда;
* интеграция с другими языками;
* большое число библиотек.

Минусы языка Python:

* Низкая скорость выполнения программ;
* проблема работы с русскими символами;
* необходимость сторонних приложений для конвертации .py в .exe;
* при копировании кода не учитывается табуляция;
* большое потребление памяти;
* для разработки мобильных приложений нужна сторонняя программа;
* ограничения по дизайну.

## Анализ существующих решений

По результатам поиска в интернете, были найдены следующие программные аналоги:

* «Про родинки»;
* «Celsus».

### Платформа «Про родинки»

Методика получения рекомендаций платформы "Про родинки" была предложена экспертами дерматологами и онкологами "Приволжского исследовательского медицинского университета" Министерства здравоохранения Российской Федерации. Разработка велась в тесном сотрудничестве с врачами-экспертами. В процессе эксплуатации врачи осуществляют непрерывный контроль за работой платформы.

Основой платформы является нейросеть, специально построенная и обученная распознаванию рисков злокачественных заболеваний кожи по фотографиям подозрительных "родинок", сделанных смартфоном.

Нейросеть была обучена на более чем 4000 случаев с подтвержденными диагнозами.

Направлениями обучения было распознавание меланомы и базальноклеточного рака на ранних стадиях.

Работа платформы "Про родинки" была протестирована в ходе испытаний на 1000 реальных случаев.

В результате испытаний были получены следующие характеристики качества работы платформы "Про родинки":

Чувствительность - вероятность правильного заключения нейросети при предъявлении злокачественного заболевания (базальноклеточного рака или меланомы) составила 88%. Соответственно вероятность ложно отрицательной ошибки - 12%

Специфичность - вероятность правильного заключения нейросети при отсутствии злокачественного заболевания (базальноклеточного рака или меланомы) составила 78%. Соответственно вероятность ложно положительной ошибки - 22%.

Результат испытаний подтвержден специальным Отчетом Федерального государственного бюджетного образовательного учреждения высшего образования "Приволжский исследовательский медицинский университет" Министерства здравоохранения Российской Федерации.

По заключению Росздравнадзора РФ № КП-20-006 от 14.04.2020г. Программный комплекс "Про родинки" является немедицинским программным обеспечением для неограниченного круга пользователей в образовательных, научно-популярных, справочно-информационных целях, в том числе для выбора медицинского специалиста.

Принцип пользования приложением «Про родинки»:

* Устанавливаете приложение;
* указываете особенности кожи и другие характеристики, влияющие на риск заболевания;
* фотографируете кожное образование камерой телефона по особым правилам:
  + на фотографии должен быть только один объект – «родинка»;
  + фотография должна быть резкой, так как чем меньше резкость – тем больше вероятность ошибки;
  + фотографируемый объект должен занимать наибольшую площадь кадра со всеми контурами объекта.
* система искусственного интеллекта на основе нейросетей глубокого обучения анализирует данные;
* по результатам анализа Приложение выдает рекомендацию о выборе врача;

### Платформа «Celsus»

ООО «Медицинские скрининг системы» основана в 2018 году и является разработчиком высокотехнологичных решений в области радиологии.

Решение «Цельс» - сложноорганизованная нейронная сеть, искусственный интеллект, система поддержки принятия врачебных решений на основе нейронных сетей и математического анализа путем анализа цифровых медицинских изображений, детектирования объектов и интерпретации результата.

«Цельс» является первым в РФ медицинским изделием с применением ИИ, зарегистрированным по 3 классу риска, также зарегистрированным в Евросоюзе.

«Celsus» может применяться по различным сценариям: начиная с использования на рабочем места врача при приеме пациента заканчивая автоматической потоковой обработкой поступающих данных.

Сейчас один из важных сценариев применения — массовый скрининг, который проводится в регионах, например, на передвижных маммографах или при проведении диспансеризации.

Анализируя изображения, «Celsus» находит и классифицирует патологические объекты (новообразования), кальцинаты, плотность ткани и иные параметры. А дальше присваивает класс по шкале Bi-RADS (европейский стандарт чтения и интерпретации снимков молочной железы), ранжирует итоговый список пациентов от наибольшего к меньшему по риску наличия патологии. Врач получает информацию о том, снимки какого пациента необходимо исследовать в первую очередь или отправить в онкоцентр в приоритетном порядке.

На данном этапе, Celsus готов к промышленной эксплуатации. Вышел первый релиз блока по выявлению патологий при проведении флюорографии. Команда работает над релизом по выявлению рака легких при рентгенографических исследованиях. В планах — анализировать данные компьютерной томографии (современный метод рентгеновского послойного исследования, более точный, чем рентгенография) и морфологии (исследования тканей организма, полученных путем пункции, биопсии или хирургически, на предмет наличия раковых клеток; самый точный метод подтверждения диагноза).

## Постановка цели и задачи

Чтобы произвести исследование результатов применения алгоритма приведения к средней яркости и выявить возможные закономерности, необходимо разработать программу, которая должна решать следующие задачи:

* загрузка экземпляров изображений в базу данных;
* открытие изображений;
* преобразование яркости каждого пикселя изображения к средней яркости 127.5;
* выводить гистограммы яркостей изображения R, G, B;
* выводить значения средней яркости по гистограммам R, G, B;
* выводить значения медианы по гистограммам R, G, B;
* выводить значения среднеквадратичного отклонения по гистограммам R, G, B;
* сохранение данных исследования;
* загрузка данных исследования.

## Вывод по аналитической части

На данном этапе был проведен анализ предметной области, существующих на данный момент проблем и их последствий, а также были выполнены постановка цели и задач, решение которых необходимо для достижения поставленной цели.

Изучив существующие аналоги, нужно отметить, что данное исследование, относительно продукта «Про родинки», направлено на увеличение процента правильных результатов, путем поиска закономерностей в цветовой модели RGB. А относительно продукта «Celsus» - программа, созданная для осуществления исследования, не нуждается в интеграции в томограф и направлена на автоматизацию всего процесса анализа по кожным заболеваниям.

Проанализировав положительные и отрицательные стороны языков программирования [4-6], можно сделать вывод, что C# является оптимальным языком программирования для реализации программы. Что касается среды разработки, выбор был между Eclipse и Visual Studio. Сравнение сред разработки и представлены в Таблица 1.

Таблица 1 - Сравнение сред Visual Studio и Eclipse

|  |  |  |  |
| --- | --- | --- | --- |
| Название | Условия предоставления | Кроссплатформенность | Функциональность |
| Visual Studio | Бесплатна для малых предприятий, учащихся | Нацелено на .Net (Windows) | Необходимый функционал встроен в коробку, достаточно скачать необходимые плагины |
| Eclipse | Бесплатна | Поддерживает Windows, Linux, macOS | Необходимо скачать и настроить плагины для корректной работы |

Для разработки программы был выбран язык программирования C# из-за простоты написания программы и наличия библиотеки Marshal, которая позволила многократно ускорить обработку изображений. В качестве среды разработки была выбрана Visual Studio. С точки зрения кроссплатформенности нужно было выбрать Eclipse, но исходя из требований задания на ВКР, платформа Windows подходит наиболее лучше.

Используя Xamarin, можно выпустить программу и для Android, iOS, Windows Phone устройств.

# Проектная часть

## Описание предметной области

В соответствие с предметной областью, система строится с учетом следующих особенностей обращения с пользователем:

1. Можно подключаться к базе данных.
2. Можно загружать новые экземпляры изображений.
3. Возможно фильтровать и выбирать варианты из списка доступных изображений.
4. Проводить исследование доступными средствами.
5. Сохранять и загружать результаты исследований.

## Концептуальное проектирование

Концептуальное проектирование технических систем – начальная стадия проектирования, на которой принимаются решения, определяющие последующий облик, и проводится исследование и согласование параметров созданных технических решений с возможной их организацией. Термин «концепция» применяется для описания принципа действия не только в технических системах, но и в научных, художественных и прочих видах деятельности. «Концепт» - содержание понятия, смысл. Таким образом, проектирование на концептуальном уровне – на уровне смысла или содержания понятия систем.

Основной объем задач концептуального проектирования относится к ранним стадиям разработки технических систем (ТС): при постановке задачи на проектирование, выработке массива вариантов технических и оформительских решений.

Технология проектирования, разработки и сопровождения ИС ‑ общие требования:

* поддержка полного жизненного цикла;
* достижение целей разработки ИС с заданным качеством и в установленное время;
* возможность выполнения крупных проектов в виде подсистем (т.е. возможность декомпозиции проекта на составные части, разрабатываемые группами исполнителей ограниченной численности с последующей интеграцией составных);
* минимальное время получения работоспособной ИС. Речь идет не о сроках готовности всей ИС, а о сроках реализации отдельных подсистем;
* возможность управления конфигурацией проекта, ведения версий проекта и его составляющих, возможность автоматического выпуска проектной документации и синхронизацию ее версий с версиями проекта;

Применение любой технологии проектирования, разработки и сопровождения ИС невозможно без выработки ряда стандартов (правил, соглашений), которые должны соблюдаться всеми участниками проекта. К таким стандартам относятся:

* стандарт проектирования;
* стандарт оформления проектной документации;
* стандарт пользовательского интерфейса.

Стандарт проектирования должен устанавливать:

* набор необходимых моделей на каждой стадии проектирования и степень их детализации;
* правила фиксации проектных решений в документации, в том числе: правила именования объектов, набор атрибутов для всех объектов и правила их заполнения на каждой стадии;
* требования к конфигурации рабочих мест разработчиков, включая настройки операционной системы, настройки CASE-средств и т. д.;
* механизм обеспечения совместной работы над проектом, в том числе: правила интеграции подсистем проекта, правила поддержания проекта в одинаковом для всех разработчиков состоянии (регламент обмена проектной информацией, механизм фиксации общих объектов и т.д.)
* правила проверки проектных решений на непротиворечивость и т. д.

Стандарт оформления проектной документации должен устанавливать:

* комплектность, состав и структуру документации на каждой стадии проектирования;
* требования к ее оформлению (включая требования к содержанию разделов, подразделов, пунктов, таблиц и т.д.);
* правила подготовки, рассмотрения, согласования и утверждения документации с указанием предельных сроков для каждой стадии;
* требования к настройке средств для подготовки документации в соответствии с установленными требованиями.

Стандарт интерфейса пользователя должен устанавливать:

* правила оформления экранов (шрифты и цветовая палитра), состав и расположение окон и элементов управления;
* правила использования клавиатуры и мыши;
* правила оформления текстов помощи;
* перечень стандартных сообщений;
* правила обработки реакции пользователя.

Неотъемлемой частью концептуального проектирования является построение UML-диаграмм. Далее будут приведены некоторые UML- диаграммы, которые используются в данной работе.

### Варианты использования

Диаграммы вариантов использования описывают функциональное назначение системы или то, что система должна делать. Разработка диаграммы преследует следующие цели:

1. определить общие границы и контекст моделируемой предметной области;
2. сформулировать общие требования к функциональному поведению проектируемой системы;
3. разработать исходную концептуальную модель системы для ее последующей детализации в форме логических и физических моделей;
4. подготовить исходную документацию для взаимодействия разработчиков системы с ее заказчиками и пользователями.

Прецедент – возможность моделируемой системы (часть её функциональности), благодаря которой пользователь может получить конкретный, измеримый и нужный ему результат. Прецедент соответствует отдельному сервису системы, определяет один из вариантов её использования и описывает типичный способ взаимодействия пользователя с системой. Варианты использования обычно применяются для спецификации внешних требований к системе.

Основное назначение диаграммы – описание функциональности и поведения, позволяющее заказчику, конечному пользователю и разработчику совместно обсуждать проектируемую или существующую систему.

При моделировании системы с помощью диаграммы прецедентов системный аналитик стремится:

* чётко отделить систему от её окружения;
* определить действующих лиц (акторов), их взаимодействие с системой и ожидаемую функциональность системы;
* определить в глоссарии предметной области понятия, относящиеся к детальному описанию функциональности системы (то есть прецедентов).

Работа над диаграммой может начаться с текстового описания, полученного при работе с заказчиком. При этом нефункциональные требования (например, конкретный язык или система программирования) при составлении модели прецедентов опускаются (для них составляется другой документ).

Для отражения модели прецедентов на диаграмме используются:

* рамки системы – прямоугольник с названием в верхней части и эллипсами (прецедентами) внутри. Часто может быть опущен без потери полезной информации;
* актор – стилизованный человечек, обозначающий набор ролей пользователя (понимается в широком смысле: человек, внешняя сущность, класс, другая система), взаимодействующего с некоторой сущностью (системой, подсистемой, классом). Акторы не могут быть связаны друг с другом (за исключением отношений обобщения/наследования);
* прецедент – эллипс с надписью, обозначающий выполняемые системой действия (могут включать возможные варианты), приводящие к наблюдаемым акторами результатам. Надпись может быть именем или описанием (с точки зрения актора) того, «что» делает система (а не «как»). Имя прецедента связано с непрерывным (атомарным) сценарием – конкретной последовательностью действий, иллюстрирующей поведение. В ходе сценария акторы обмениваются с системой сообщениями. Сценарий может быть приведён на диаграмме прецедентов в виде UML-комментария. С одним прецедентом может быть связано несколько различных сценариев.

Часть дублирующейся информации в модели прецедентов можно устранить указанием связей между прецедентами:

* обобщение прецедента – стрелка с не закрашенным треугольником (треугольник ставится у более общего прецедента);
* включение прецедента – пунктирная стрелка со стереотипом «include»;
* расширение прецедента – пунктирная стрелка со стереотипом «extend» (стрелка входит в расширяемый прецедент, в дополнительном разделе которого может быть указана точка расширения и, возможно в виде комментария, условие расширения).

При работе с вариантами использования важно помнить несколько простых правил:

* каждый прецедент относится как минимум к одному действующему лицу;
* каждый прецедент имеет инициатора;
* каждый прецедент приводит к соответствующему результату.

Проектируемая система представляется в виде множества сущностей или актеров, взаимодействующих с системой с помощью вариантов использования. При этом актером (actor) или действующим лицом называется любая сущность, взаимодействующая с системой извне. Это может быть человек, техническое устройство, программа или любая другая система, которая может служить источником воздействия на моделируемую систему так, как определит сам разработчик. Вариант использования служит для описания сервисов, которые система предоставляет актеру. Диаграмма вариантов использования может дополняться пояснительным текстом, который раскрывает смысл или семантику составляющих ее компонентов.

На диаграмме вариантов использования (Рисунок 1) показано как пользователь может взаимодействовать с программой и какие действия доступны. Основные функции – это открытие экземпляров изображений, обработка и сохранение результатов исследований, а также обратная связь в виде выдачи ошибок в ходе выполнения алгоритма.

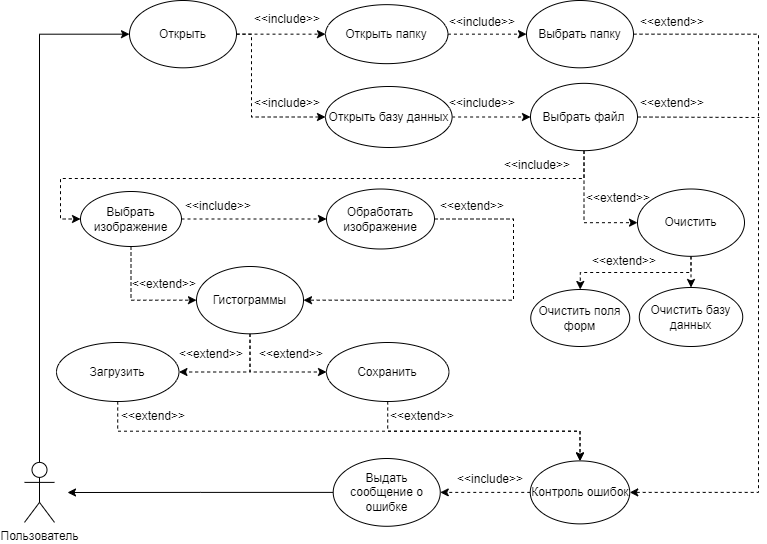


Рисунок 1 - Диаграмма вариантов использования пользователя

Спецификации диаграммы вариантов использования:

В роли актера выступает пользователь, который взаимодействует с программой.

Пользователю доступно действие открыть.

Для начала процесса обработки изображений, необходимо выполнить следующие шаги:

Пользователь – открыть базу данных – выбрать файл.

Чтобы выполнить загрузку экземпляров изображений в базу данных, необходимо выполнить следующие шаги:

Пользователь – открыть папку – выбрать папку.

Шаги, необходимые для исследования изображения:

Пользователь – открыть – открыть базу данных – выбрать файл – выбрать изображение – обработать изображение.

Шаги, необходимые для сохранения исследования:

Пользователь – открыть – открыть базу данных – выбрать файл – выбрать изображение – обработать изображение – гистограммы – сохранить.

Шаги, необходимые для загрузки исследования:

Пользователь – открыть – открыть базу данных – выбрать файл – выбрать изображение – обработать изображение – гистограммы – загрузить.

Шаги, необходимые для очистки данных из базы данных:

Пользователь – открыть – открыть базу данных – выбрать файл – очистить – очистить базу данных.

Шаги, необходимые для очистки рабочего пространства:

Пользователь – открыть – открыть базу данных – выбрать файл – очистить – очистить поля.

В программе также предусмотрена обратная связь, реализуемая выводом ошибок в ходе выполнения программы, чтобы вызвать сообщение достаточно смоделировать одну из следующих ситуаций:

* при загрузке исследования;
* при сохранении исследования;
* при открытии некорректного типа базы данных;
* при записи некорректного типа файла в базу данных.

### Диаграмма классов

Одно из важных мест в объектно-ориентированном программировании занимает разработка логической модели системы в виде диаграммы классов. Диаграмма классов необходима для представления статической структуры модели системы в терминологии классов объектно-ориентированного программирования. Диаграмма классов может показывать различные взаимосвязи между не связанными сущностями предметной области, такими как объекты и подсистемы, а также описывать их внутреннюю структуру и типы отношений.

Диаграмма классов представляет собой граф, вершинами которого являются элементы типа «классификатор», связанные различными типами структурных отношений. Диаграмма классов может также содержать интерфейсы, пакеты, отношения и даже отдельные экземпляры, такие как объекты и связи.

Взаимосвязь – это особый тип логических отношений между сущностями, показанных на диаграммах классов и объектов. В UML представлены следующие виды отношений:

Зависимость обозначает такое отношение между классами, что изменение спецификации класса-поставщика может повлиять на работу зависимого класса, но не наоборот.

Ассоциация показывает, что объекты одной сущности (класса) связаны с объектами другой сущности таким образом, что можно перемещаться от объектов одного класса к другому. Является общим случаем композиции и агрегации.

Двойные ассоциации представляются линией без стрелочек на концах, соединяющей два классовых блока. Ассоциации более высокой степени имеют более двух концов и представляются линиями, один конец которых идёт к классовому блоку, а другой к общему ромбику. В представлении однонаправленной ассоциации добавляется стрелка, указывающая на направление ассоциации.

Ассоциация может быть именованной, и на концах представляющей её линии могут быть подписаны роли, принадлежности, индикаторы, мультипликаторы, видимости или другие свойства.

Агрегация – это разновидность ассоциации при отношении между целым и его частями. Как тип ассоциации агрегация может быть именованной. Одно отношение агрегации не может включать более двух классов (контейнер и содержимое).

Агрегация встречается, когда один класс является коллекцией или контейнером других. Причём по умолчанию, агрегацией называют агрегацию по ссылке, то есть, когда время существования содержащихся классов не зависит от времени существования содержащего их класса. Если контейнер будет уничтожен, то его содержимое – нет.

Графически агрегация представляется пустым ромбом на блоке класса, и линией, идущей от этого ромба к содержащемуся классу.

Композиция – более строгий вариант агрегации. Известна также как агрегация по значению.

Композиция имеет жёсткую зависимость времени существования экземпляров класса контейнера и экземпляров содержащихся классов. Если контейнер будет уничтожен, то всё его содержимое будет также уничтожено.

Графически представляется, как и агрегация, но с закрашенным ромбиком.

Обобщение (Generalization) показывает, что один из двух связанных классов (подтип) является частной формой другого (надтипа), который называется обобщением первого. На практике это означает, что любой экземпляр подтипа является также экземпляром надтипа.

Графически обобщение представляется линией с пустым треугольником у супертипа.

Обобщение также известно как наследование или «is a» взаимосвязь (или отношение «является»).

Реализация – отношение между двумя элементами модели, в котором один элемент (клиент) реализует поведение, заданное другим (поставщиком). Реализация – отношение целое-часть. Графически реализация представляется так же, как и наследование, но с пунктирной линией.

Поставщик, как правило, является абстрактным классом или классом-интерфейсом.

Зависимость (dependency) – это слабая форма отношения использования, при которой изменение в спецификации одного влечёт за собой изменение другого, причём обратное не обязательно. Возникает, когда объект выступает, например, в форме параметра или локальной переменной.

Графически представляется штриховой стрелкой, идущей от зависимого элемента к тому, от которого он зависит.

Существует несколько именованных вариантов.

Зависимость может быть между экземплярами, классами или экземпляром и классом.

Уточнение имеет отношение к уровню детализации. Один пакет уточняет другой, если в нём содержатся те же самые элементы, но в более подробном представлении. Например, при написании книги вы наверняка начнете с формулировки предложения, в котором кратко будет представлено содержание каждой главы.

Мощность отношения (мультипликатор) означает число связей между каждым экземпляром класса (объектом) в начале линии с экземпляром класса в её конце.

На рисунке 2 представлено изображение классов данной системы, а также соответствующих им операций и атрибутов.

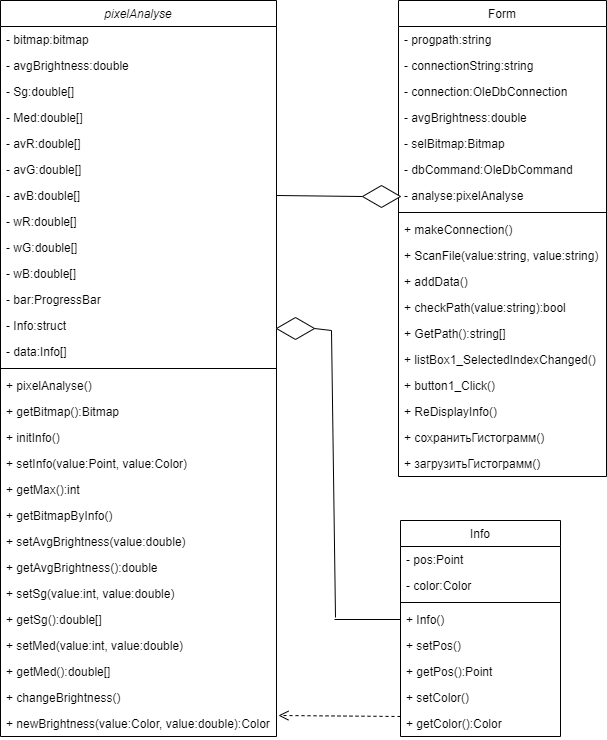


Рисунок 2 - Диаграмма классов

Класс Form имеет атрибуты:

1. progpath;
2. connectionString;
3. connection;
4. avgBrightness;
5. selBitmap;
6. dbCommand;
7. analyse.

Класс Form имеет операции:

1. makeConnection();
2. ScanFile();
3. addData();
4. checkPath();
5. GetPath();
6. listBox1\_SelectedIndexChanged();
7. button1\_Click();
8. ReDisplayInfo();
9. сохранитьГистограмм();
10. загрузитьГистограмм().

Класс pixelAnalyse имеет атрибуты:

1. bitmap;
2. avgBrightness;
3. Sg;
4. Med;
5. avR;
6. avG;
7. avB;
8. wR;
9. wG;
10. wB;
11. bar;
12. data

Класс pixelAnalyse имеет операции:

1. pixelAnalyse();
2. getBitmap();
3. initInfo();
4. setInfo();
5. getMax();
6. getBitmapByInfo();
7. setAvgBrightness();
8. getAvgBrightness()
9. setSg();
10. getSg();
11. setMed();
12. getMed();
13. changeBrightness();
14. newBrightness().

## Блок-схема работы программы

Блок-схема — это диаграмма, на которой обычно представлен процесс, система или компьютерный алгоритм и которая используется для документирования, планирования, уточнения или визуализации многоэтапного рабочего процесса. Благодаря блок-схемам вы можете определить цели и масштабы рабочего процесса, а также установить необходимые задачи в хронологическом порядке. Впервые эта концепция была задокументирована в 1921 году, когда инженеры-технологи Фрэнк и Лилиан Гилбреты представили Американскому обществу инженеров-механиков (ASME) графическую схему технологического процесса. Популярность блок-схем росла на протяжении 20-х и 30-х годов, как и известность Арта Спинангера и Бена С. Грэхема, ставших приверженцами этой системы. Опираясь на результаты работы Гилбретов, в 1947 году Американское общество инженеров-механиков приняло систему символов для блок-схем. К 1949 году блок-схемы начали использовать при планировании компьютерных программ.

Ниже приведены некоторые наиболее часто встречающиеся символы блок-схем:

* Символ «Процесс», или «Действие»: этот символ обозначается прямоугольником и представляет собой этап процесса. Это самый распространенный символ блок-схемы.
* Символ «Начало/конец», или «Терминатор»: этот символ представляет собой прямоугольник с закругленными краями, который изображается как в начальной, так и в конечной точках блок-схемы.
* Символ «Решение»: ромбовидный символ, обозначающий вопрос, на который необходимо ответить «да»/«нет» или «правда»/«ложь». Затем блок-схема разбивается на различные ветви, представляющие собой отдельное направление для каждого ответа.
* Символ «Данные», или «Ввод/вывод данных»: символ в форме параллелограмма, означающий данные, которые вводятся в процесс или выводятся из него, то есть задействованные или полученные впоследствии ресурсы.
* Символ «Документ»: символизирует ввод или вывод документа. Под вводом документа может подразумеваться поступление отчета, электронного письма или заказа. Примеры вывода документов: создание презентации, рабочего конспекта или письма.
* Символ «Межстрочный соединитель»: часто применяется в сложных схемах для соединения отдельных блоков, расположенных на разных страницах. Для удобства интерпретации внутри фигуры, как правило, указывается номер страницы.
* Символ «Комментарий»: в сочетании с другими материалами этот символ позволяет добавить необходимый контекст, разъяснение или комментарий к определенному диапазону данных. Комментарий также можно присоединить к необходимому разделу блок-схемы с помощью пунктирной линии.
* Символ «Предопределенный процесс» («Подпрограмма»): символизирует сложные процессы и операции, которые уже известны или охарактеризованы в другом месте.

Рассмотрим работу алгоритма на блок-схемах (Рисунок 3-6), далее будут представлены четыре блок-схемы:

* основной принцип работы;
* загрузка изображения в базу данных;
* запись изображения в базу данных;
* обработка изображения.

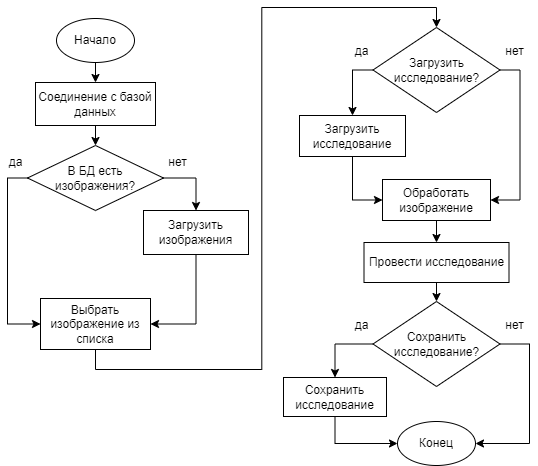


Рисунок 3 - Блок-схема основного процесса проведения исследования

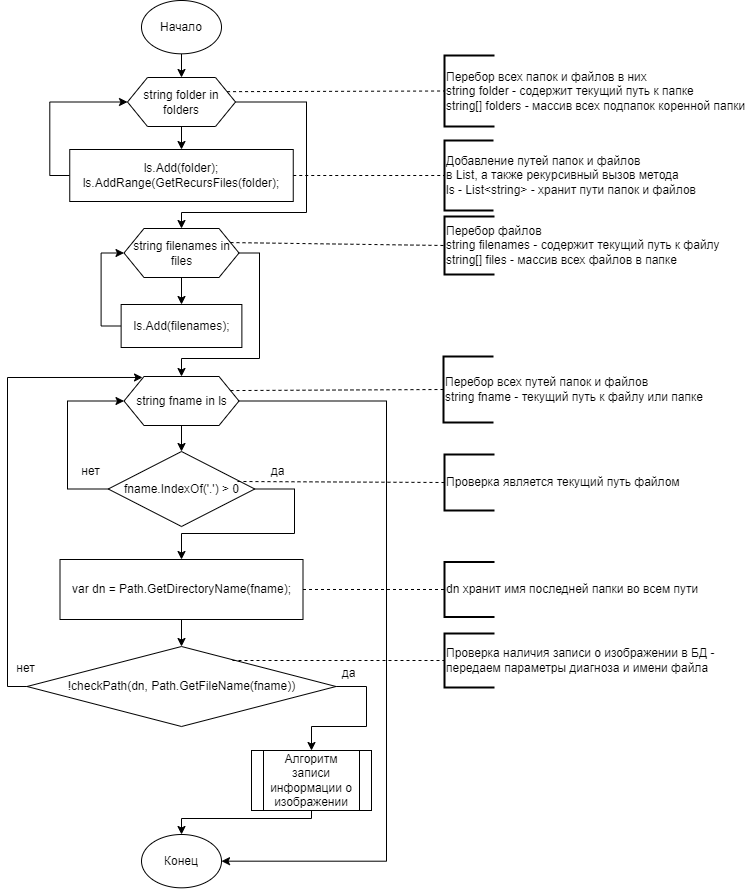


Рисунок 4 – Блок-схема загрузки информации о изображениях в базу данных

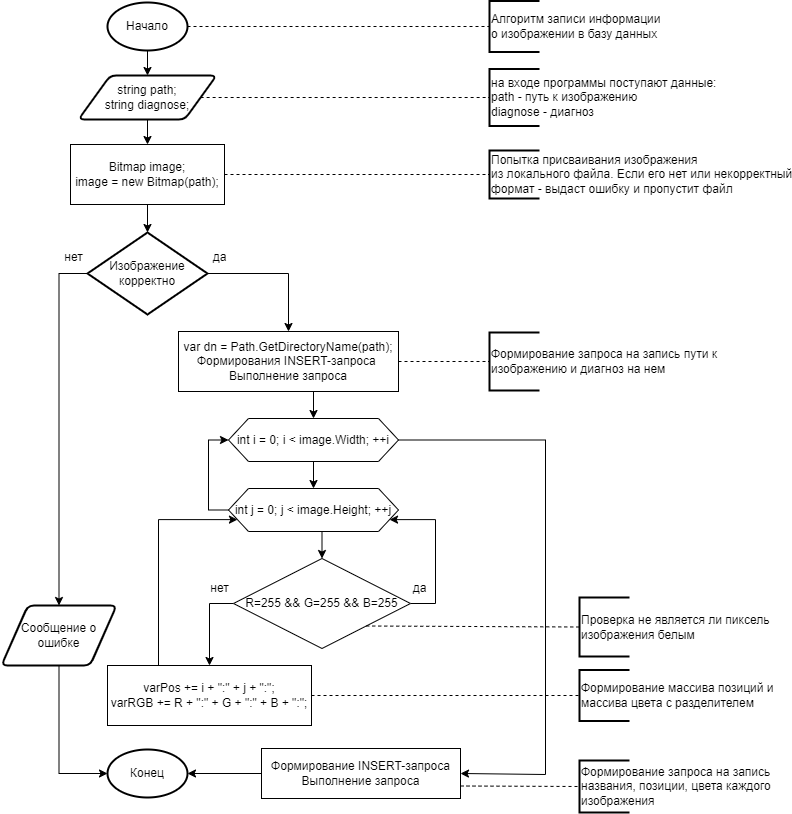


Рисунок 5 - Блок-схема алгоритма записи информации о изображении в базу данных

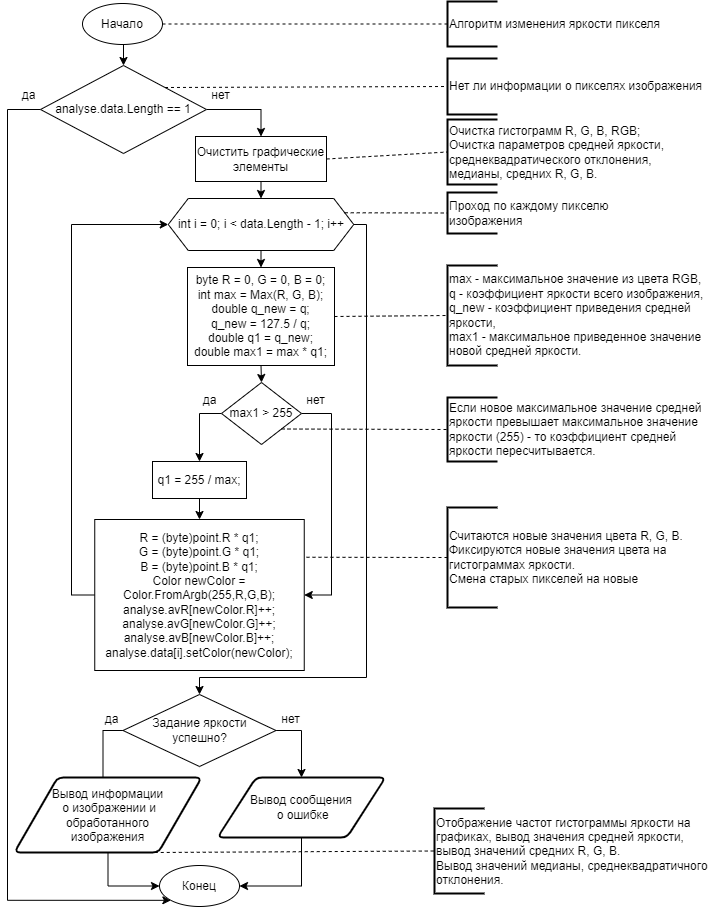


Рисунок 6 - Блок-схема расчета новой яркости изображения

## Выбор системы управлению базой данных

Для записи и выборки данных о изображениях и исследованиях была выбрана база данных Microsoft Access.

К основным плюсам MS Access можно отнести:

* MS Access распространен, так как является продуктом Microsoft, программное обеспечение и операционные системы которой использует большая часть пользователей персональных компьютеров. MS Access полностью совместим с операционной системой Windows, постоянно обновляется производителем, поддерживает множество языков;
* MS Access ориентирован на людей с всесторонней профессиональной подготовкой по средствам наличия многих и простых инструментов работы с базой данных;
* возможность создавать базы данных и работать с ними, не зная языка запросов SQL, через диалоговые средства.

К основным минусам MS Access можно отнести:

* Ограничены возможности по обеспечению многопользовательской среды;
* в ранних версиях (до Access 2003) отсутствуют такие средства как триггеры и хранимые процедуры, что заставляет разработчиков возлагать поддержание бизнес-логики БД на клиентскую программу или разрабатывать процедуры с помощью встроенного средства VBA;
* обладает несложными способами защиты с использованием пароля БД (возможно применения дополнительных мер по защите от несанкционированного доступа с использованием процедур VBA);
* в вопросах поддержки целостности данных отвечает только моделям БД небольшой и средней сложности;
* не распространяется бесплатно.

В отличие от других настольных СУБД, Access хранит все данные в одном файле, хотя и распределяет их по разным таблицам, как и положено реляционной СУБД. К этим данным относится не только информация в таблицах, но и другие объекты базы данных, которые будут описаны ниже.

Для выполнения почти всех основных операций Access предлагает большое количество Мастеров (Wizards), которые делают основную работу за пользователя при работе с данными и разработке приложений, помогают избежать рутинных действий и облегчают работу не обученному в программировании пользователю.

Создание многопользовательской БД Access и получение одновременного доступа нескольких пользователей к общей базе данных возможно в локальной одноранговой сети или в сети с файловым сервером. Сеть обеспечивает аппаратную и программную поддержку обмена данными между компьютерами. Access следит за разграничением доступа разных пользователей к БД и обеспечивает защиту данных. При одновременной работе. Так как Access не является клиент серверной СУБД, возможности его по обеспечению многопользовательской работы несколько ограничены. Обычно для доступа к данным по сети с нескольких рабочих станций, файл БД Access (с расширением \*.mdb) выкладывается на файловый сервер. При этом обработка данных ведется в основном на клиенте – там, где запущено приложение, в силу принципов организации файловых СУБД. Этот фактор ограничивает использование Access для обеспечения работы множества пользователей (более 15–20) и при большом количестве данных в таблицах, так как многократно возрастает нагрузка не сеть.

Основным структурным компонентом базы данных является таблица. В таблицах хранятся вводимые данные. Каждая таблица состоит из столбцов, называемых полями, и строк, называемых записями. Каждая запись таблицы содержит всю необходимую информацию об отдельном элементе базы данных.

При разработке структуры таблицы, прежде всего, необходимо задать поля, определив их свойства.

СУБД MS Access позволяет создавать и использовать объекты семи различных типов:

* таблицы. Это основные объекты любой базы данных. Именно в них хранятся, во-первых, все данные, имеющиеся в базе, а, во-вторых, структура самой базы (поля, их типы и свойства);
* запросы. Это объекты предназначены для извлечения данных из таблиц и предоставления их пользователю в удобном виде. Особенность запросов состоит в том, что берут информацию из базовых таблиц и создают на их основе временную результирующую таблицу, которая не имеет аналога на жестком диске, это только образ отобранных полей и записей;
* формы. Это средства для ввода и просмотра данных. С помощью форм можно закрыть некоторые поля для несанкционированного ввода, можно разместить специальные элементы управления (счетчики, раскрывающиеся списки, переключатели, флажки и пр.) для автоматизации ввода. Можно представить форму с помощью графических средств, в виде бланка, если ввод производится со специальных бланков;
* отчеты. Предназначены для вывода данных на печатающее устройство. В них приняты специальные меры для группирования выводимых данных и для вывода специальных элементов оформления, характерных для печатных документов (верхний и нижний колонтитулы, номера страниц и т.п.);
* страницы. Этот объект позволяет создать Web-страницы, компоненты которых осуществляют связь с базой данных. Web-страницу можно передать клиенту, сама же база будет располагаться на сервере;
* макросы и модули. Эти объекты предназначены как для автоматизации повторяющихся операций при работе с СУБД, так и для создания новых функций путем программирования. Макросы состоят из последовательности внутренних команд СУБД Access и являются одним из средств автоматизации работы с базой. Модули создаются средствами внешнего языка программирования Visual Basic for Applications.

СУБД представляет несколько средств создания каждого из основных объектов базы. Их можно классифицировать как:

* ручные (разработка объектов в режиме Конструктора);
* автоматизированные (разработка с помощью программ-мастеров);
* автоматические – средства ускоренной разработки простейших объектов.

При разработке таблиц и запросов лучше использовать ручные средства – работать в режиме Конструктора.

При разработке учебных форм, отчетов, лучше пользоваться автоматизированными средствами, предоставляемыми мастерами.

## Физическая модель базы данных

Физическое проектирование базы данных – процесс подготовки описания реализации базы данных на вторичных запоминающих устройствах; на этом этапе рассматриваются основные отношения, организация файлов и индексов, предназначенных для обеспечения эффективного доступа к данным, а также все связанные с этим ограничения целостности и средства защиты.

Физическое проектирование является третьим и последним этапом создания проекта базы данных, при выполнении которого проектировщик принимает решения о способах реализации разрабатываемой базы данных. Во время предыдущего этапа проектирования была определена логическая структура базы данных (которая описывает отношения и ограничения в рассматриваемой прикладной области). Хотя эта структура не зависит от конкретной целевой СУБД, она создается с учетом выбранной модели хранения данных, например реляционной, сетевой или иерархической. Однако, приступая к физическому проектированию базы данных, прежде всего необходимо выбрать конкретную целевую СУБД. Поэтому физическое проектирование неразрывно связано с конкретной СУБД. Между логическим и физическим проектированием существует постоянная обратная связь, так как решения, принимаемые на этапе физического проектирования с целью повышения производительности системы, способны повлиять на структуру логической модели данных (Таблицы 2-4).

Можно выделить два этапа создания физической модели данных. Основными целями первого этапа являются:

* удовлетворение потребности в хранении данных предметной области в рамках реляционной модели данных, т.е. должны быть созданы базовые таблицы для хранения информации обо всех сущностях предметной области;
* удовлетворение требования целостности данных, т.е. должны быть определены типы колонок и наложены ограничения на значения колонок базовых таблиц, которые бы следовали из бизнес-правил предметной области;
* удовлетворение требования ссылочной целостности (referential integrity, RI), т.е. в случае принятия решения о поддержки ссылочной целостности встроенными средствами СУБД должны быть наложены ограничения ссылочной целостности на таблицы, исходя из бизнес-правил ссылочной целостности предметной области;
* удовлетворение (частично) требования независимости представления данных для конечного пользователя от характера физического хранения данных.

На первом этапе в рамках требований реляционной модели создаются объекты хранения данных, соответствующие сущностям и взаимосвязям логической модели данных — таблицы, индексы, представления и т.д.

Главной целью второго этапа является обеспечение требуемого уровня производительности. Для достижения этой цели необходимо учитывать как особенности реализации СУБД, для которой создается физическая модель, так и особенности функционирования будущей информационной системы в целом. Обычно производительность БД измеряется в терминах производительности транзакций (transaction performance).

Для повышения производительности транзакций могут быть модифицированы объекты, созданные на первом этапе, или созданы новые объекты БД. Другими словами, разработка физической модели представляет собой итерационный процесс, причем итераций (создание объектов – анализ транзакций – модификация объектов – анализ транзакций) может быть несколько.

Таблица 2 - Физическая структура таблицы Exsamples (Экземпляры)

|  |  |  |  |
| --- | --- | --- | --- |
| Имя поля | Тип данных | Ограничение целостности | Комментарии |
| ID | Int | Primary key | Первичный ключ |
| Path | Short Text |  | Путь к файлу |
| Diagnose | Short Text |  | Диагноз |

Таблица 3 - Физическая структура таблицы Spot\_info (Информация о пикселе)

|  |  |  |  |
| --- | --- | --- | --- |
| Имя поля | Тип данных | Ограничение целостности | Комментарии |
| ID\_spot | Int | Primary key | Первичный ключ |
| ID\_photo | Int | Foreign key | ID фотографии, к которой относится массив пикселей, внешний ключ, берется из таблицы Exsamples |

Окончание Таблица 3

|  |  |  |  |
| --- | --- | --- | --- |
| Имя поля | Тип данных | Ограничение целостности | Комментарии |
| PosXY | Long Text |  | Массив позиций пикселей в формате X:Y |
| RGB | Long Text |  | Массив позиций пикселей в формате R:G:B |

Таблица 4 - Физическая структура таблицы Gist\_info (Информация о исследованиях)

|  |  |  |  |
| --- | --- | --- | --- |
| Имя поля | Тип данных | Ограничение целостности | Комментарии |
| ID | Int | Primary key | Первичный ключ |
| ID\_photo | Int | Foreign key | ID фотографии, к которой относится массив пикселей, внешний ключ, берется из таблицы Exsamples |
| AllBright | Short Text |  | Значение средней яркости изображения |
| Pos | Long Text |  | Массив позиций пикселей в формате X:Y |
| RGB | Long Text |  | Массив позиций пикселей в формате R:G:B |
| avR | Short Text |  | Среднее значение гистограммы яркости R |
| avG | Short Text |  | Среднее значение гистограммы яркости G |

Окончание Таблица 4

|  |  |  |  |
| --- | --- | --- | --- |
| Имя поля | Тип данных | Ограничение целостности | Комментарии |
| avB | Short Text |  | Среднее значение гистограммы яркости B |
| MedR | Short Text |  | Среднее медианы R |
| MedG | Short Text |  | Среднее медианы G |
| MedB | Short Text |  | Среднее медианы B |
| SgR | Short Text |  | Среднее значение среднеквадратичного отклонения R |
| SgG | Short Text |  | Среднее значение среднеквадратичного отклонения R |
| SgB | Short Text |  | Среднее значение среднеквадратичного отклонения R |

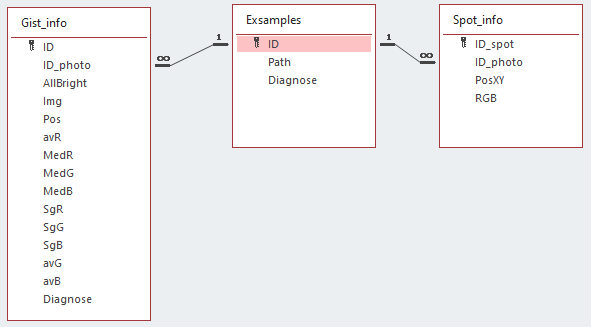


Рисунок 7 - Изображение структуры таблиц базы данных

## Вывод по проектной части

На данном этапе был проведен системный анализ предметной области, концептуальное проектирование базы данных, произведен выбор системы управления базами данных, пошагово выполнено логическое проектирование базы данных и физическое проектирование базы данных.

# Технологическая часть

## Взаимодействие базы данных и программы

Взаимодействие базы данных и программы организовано при помощи технологии OleDB в C#.

OLE DB – набор COM-интерфейсов, которые позволяют приложениям унифицировано работать с данными разных источников и хранилищ информации. Разработана Microsoft в качестве дальнейшего развития технологии доступа к данным и должен был прийти на замену и в качестве преемника ODBC, расширяя набор функций для поддержки более широкого круга нереляционных источников данных, таких как объектно-ориентированные базы данных или электронные таблицы, и для которых не обязательно использовать SQL.

OLE DB отделяет хранилище данных от приложения, которое должно иметь доступ к нему через набор абстракций, состоящий из источника данных (DataSource), сессии (Session), команды (Command) и набора строк (Rowset). Это было сделано для предоставления унифицированного доступа к различным видам и источникам данных и изоляцию специфики взаимодействия с конкретным хранилищем. OLE DB концептуально разделена на потребителей (клиентов) и поставщиков (провайдеров). Потребителем является приложение, которому необходим доступ к данным, а поставщик реализует интерфейс доступа к данным и, следовательно, обеспечивает информацией потребителя.

Спецификация OLE DB является частью Microsoft Data Access Components (MDAC), представляющей собой группу технологий Microsoft, формирующих основу для единого и всеобщего способа разработки приложений для доступа к данным практически любого хранилища. В состав MDAC входят, к примеру, сервисы OLE DB (пул подключений и прочее) и компоненты ADODB.

Поскольку различные хранилища данных могут иметь разные возможности, поставщики OLE DB, как правило, не поддерживают все интерфейсы, описанные в спецификации OLE DB. Доступные возможности поставщика данных определяются через запрос указателей на COM интерфейсы его объектов или через чтение информационных свойств источника данных (DataSource). Поставщик данных также может реализовывать и предоставлять свои собственные COM-интерфейсы и структуры данных, не описанные в спецификации OLE DB.

OLE DB определяет полный набор интерфейсов доступа к широкому спектру типов данных, содержащихся в различных хранилищах. OLE DB Programmer's Reference может испугать своим объемом, но ни одно из хранилищ данных не может обеспечить поддержку всего набора возможностей, описанных в этом справочнике. OLE DB-провайдеры данных всего лишь предоставляют интерфейсы, отражающие исходные свойства и возможности хранилищ данных. Базовый уровень функциональности поддерживается всеми провайдерами. Взаимодействие над этим базовым уровнем посредством реализаций общих дополнений – таких, как прокрутка (scrolling) или позиционирование, базирующееся на утверждении – обеспечивают общие обслуживающие компоненты, если провайдер не предоставляет такой поддержки.

Под минимальным уровнем поддержки, предоставляемым всеми хранилищами данных с общим набором дополнительных сервисов, понимается:

* OLE DB-провайдеры данных легче писать. Им нужно только предоставить набор интерфейсов, отражающих исходный набор возможностей, что приведет к возможности получать через OLE DB расширенный диапазон данных.
* Приложения, работающие с OLE DB напрямую или через ADO, могут стать функциональнее, поскольку обслуживающие компоненты предоставляют больше возможностей, нежели хранилище данных.

Чтобы предоставить доступ к данным всех типов в среде СОМ (Component Object Model – Объектно-компонентная модель), OLE DB спроектирована как компонентно-ориентированная технология. В OLE DB хранилища данных предоставляют интерфейсы, отражающие их изначальный набор возможностей. На основе этих интерфейсов для обеспечения более устойчивых моделей данных могут строиться общие компоненты. Компонентную архитектуру можно определить так: OLE DB определяет общие характеристики различных провайдеров данных и сервисов, а также определяет общие интерфейсы, которые отражали бы эти характеристики. Например, хотя набор строк (rowset) и может быть получен при помощи множества различных механизмов, результатом всегда будет набор строк с точно определенными интерфейсами, методами и характеристиками. Работа OLE DB с результатами составной таблицы не отличается от работы с результирующим текстовым файлом, содержащим табличные данные. Такое определение общих интерфейсов позволяет компонентам более эффективно расширять индивидуальную функциональность провайдера данных.

После определения базового набора возможностей всю дополнительную функциональность следует рассматривать как дополнения к базовой. Таким образом, более сложные провайдеры могут предоставлять свои расширенные возможности в дополнение к интерфейсам базового уровня. Более того, отдельные обслуживающие компоненты могут реализовывать эти возможности, строясь на базе более простых провайдеров.

## Используемые классы объектов

Класс OleDb – содержит классы, используемые для подключения к поставщику OLE DB, включая OleDbCommand, OleDbConnection и OleDbDataAdapter. Эти классы поддерживают большинство поставщиков OLE DB, но не те, что требуют интерфейсов OLE DB версии 2.5.

Класс DbConnection – позволяет подключаться к хранилищу данных и отключаться от него. Кроме того, объекты подключения обеспечивают доступ к соответствующим объектам транзакций.

Класс DbCommand – представляет SQL-запрос или хранимую процедуру. Кроме того, объекты команд предоставляют доступ к объекту чтения данных конкретного поставщика данных.

Класс DbDataReader – представляет SQL-запрос или хранимую процедуру. Кроме того, объекты команд предоставляют доступ к объекту чтения данных конкретного поставщика данных.

Класс Windows Forms – представляет окно или диалоговое окно, которое составляет пользовательский интерфейс приложения.

Form – это представление любого окна, отображаемого в приложении. Класс Form можно использовать для создания стандартных, инструментов, безграничных и плавающих окон. Класс также можно использовать для создания модальных окон, таких как диалоговое окно. Форма с несколькими документами (MDI) может содержать другие формы, называемые дочерними формами MDI. Форма MDI создается путем присвоения свойству IsMdiContainer значения true. Дочерние формы MDI создаются путем установки MdiParent свойства родительской формы MDI, которая будет содержать дочернюю форму.

Используя свойства, доступные Form в классе, можно определить внешний вид, размер, цвет и функции управления окнами создаваемого окна или диалогового окна. Свойство Text позволяет указать заголовок окна в строке заголовка. Свойства Size позволяют DesktopLocation определить размер и положение окна при его отображении. Свойство цвета можно использовать для ForeColor изменения цвета переднего плана по умолчанию для всех элементов управления, размещенных в форме. MinimizeBox, FormBorderStyle, MaximizeBox свойства позволяют управлять тем, может ли форма быть свернута, развернута или изменена во время выполнения.

Помимо свойств, можно использовать методы класса для управления формой. Например, можно использовать ShowDialog метод для отображения формы в виде модального диалогового окна. Метод можно использовать для SetDesktopLocation размещения формы на рабочем столе.

События Form класса позволяют реагировать на действия, выполняемые в форме. Событие можно использовать для Activated выполнения таких операций, как обновление данных, отображаемых в элементах управления формы при активации формы.

Форму можно использовать в качестве начального класса в приложении, поместив метод, вызываемый Main в классе. В методе Main добавьте код для создания и отображения формы. Кроме того, для запуска формы необходимо добавить STAThread атрибут Main в метод. При закрытии начальной формы приложение также закрывается.

Класс OpenFileDialog – отображает диалоговое окно, позволяющее пользователю открыть файл. Этот класс позволяет проверить, существует ли файл и открыть его. Свойство ShowReadOnly определяет, отображается ли в диалоговом окне флажок только для чтения. Свойство ReadOnlyChecked указывает, установлен ли флажок только для чтения.

Класс SaveFileDialog – запрашивает у пользователя местоположение для сохранения файла. Этот класс может открывать и перезаписывать существующий файл или создавать новый файл.

Класс Drawing – абстрактный класс, описывающий двумерный рисунок. Этот класс не может наследоваться кодом.

Drawing объекты — это легковесные объекты, позволяющие добавлять геометрические фигуры, изображения, текст и мультимедиа в приложение. Drawing объекты считаются легкими, так как они не поддерживают макет, обзор ввода и фокус. Благодаря преимуществам производительности рисунки идеально подходят для фона и картинки. При программировании на Visual уровне также используются чертежи.

Поскольку они наследуют от Freezable класса, Drawing объекты предоставляют дополнительные функции, которые делают их полезными для описания картинок и фона: их можно объявить, как ресурсы, совместно используемые среди нескольких объектов, сделать доступными только для чтения и сделать потокобезопасными. Дополнительные сведения о различных функциях, предоставляемых Freezable объектами.

Класс IO – cодержит типы, позволяющие осуществлять чтение и запись в файлы и потоки данных, а также типы для базовой поддержки файлов и папок.

Класс Path – выполняет операции для экземпляров класса String, содержащих сведения о пути к файлу или каталогу. Эти операции выполняются межплатформенным способом.

Класс Collections.Generic – содержит интерфейсы и классы, определяющие универсальные коллекции, которые позволяют пользователям создавать строго типизированные коллекции, обеспечивающие повышенную производительность и безопасность типов по сравнению с не универсальными строго типизированными коллекциями.

Класс pixelAnalyse – собственный класс, необходимый для хранения и обработки информации для текущего изображения в исследовании. К основным функциям этого класса относится следующее:

* Хранение информации о пикселях изображения (позиция на полотне и цвет RGB);
* хранение и расчет количества пикселей каждой яркости (0-255) всех трех шкал: R, G, B;
* хранение и расчет частот яркости (0-255) всех трех шкал: R, G, B;
* хранение и расчет значения средней яркости всего изображения;
* хранение и расчет значений среднеквадратичного отклонения и медианы;
* хранение и обработка экземпляра изображения Bitmap;
* обработка новой яркости.

## Описание работы программы

При запуске программы (Рисунок 8) необходимо соединиться с базой данных для доступа к экземплярам изображений (Рисунок 9).

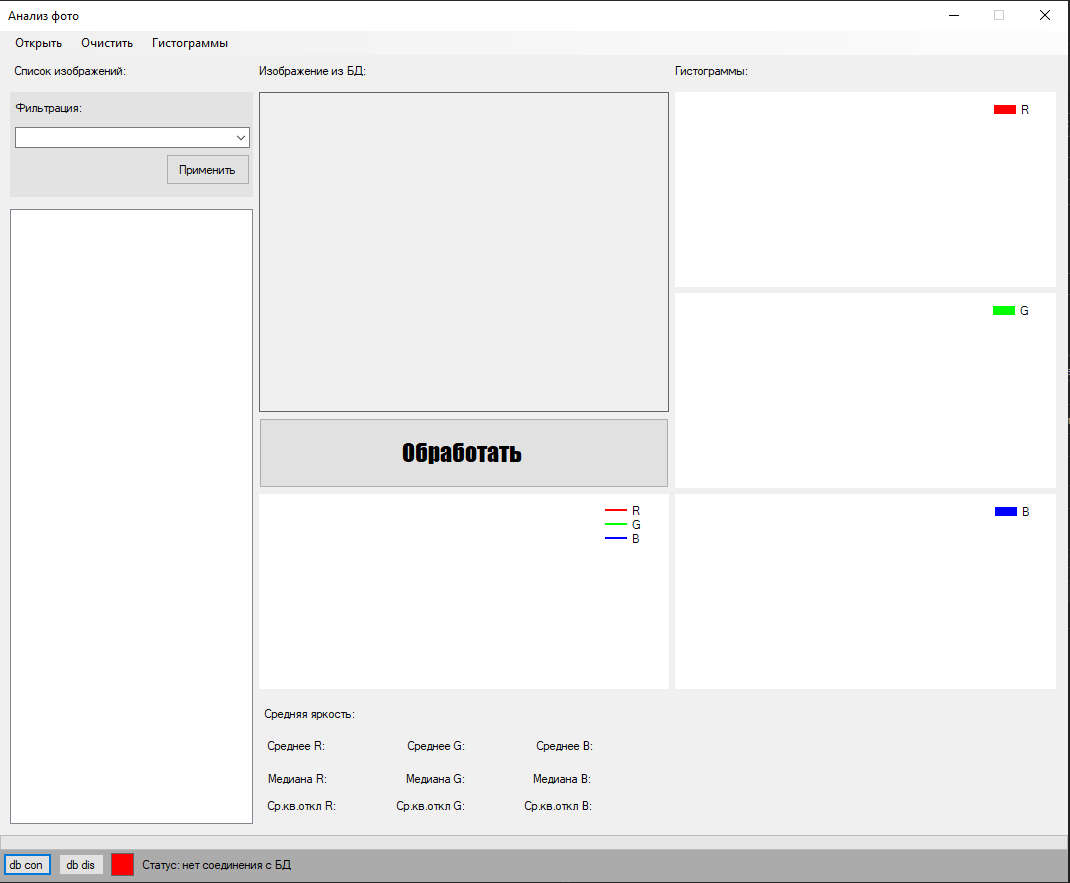


Рисунок 8 – Интерфейс программы

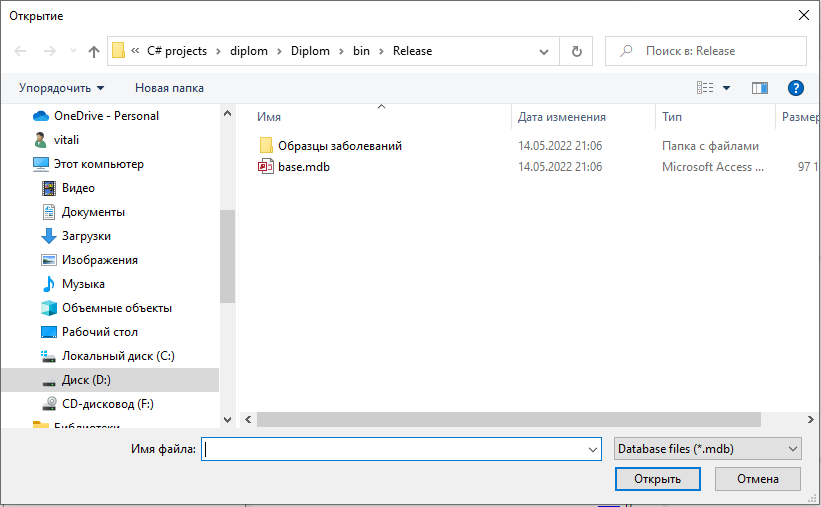


Рисунок 9 - Подключение к базе данных

Подключившись к базе данных, выводится список всех доступных изображений для исследования. Для более удобной навигации по списку, была реализована фильтрация по группе заболеваний (Рисунок 10).

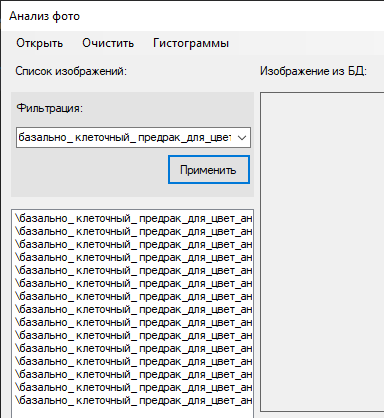


Рисунок 10 - Список изображений с фильтрацией

Нажав на имя изображения из списка – в область исследования картинок загружаются пиксели из базы данных (Рисунок 11).

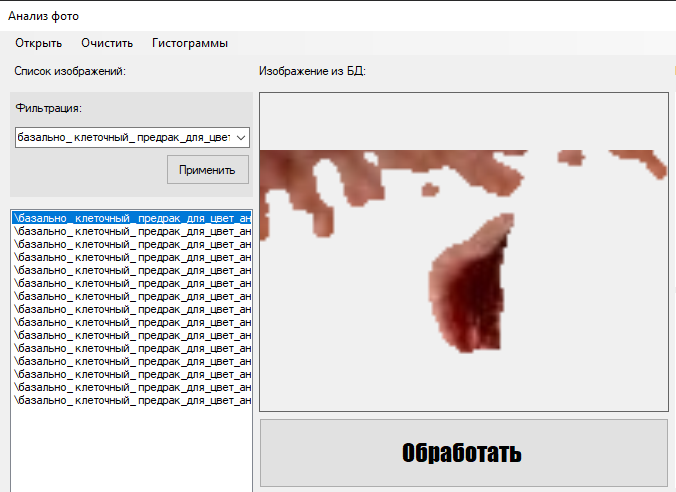


Рисунок 11 - Загруженное и базы данных изображение

После нажатия на кнопку обработать – происходит приведение всех пикселей к средней яркости 127,5 (Рисунок 12).

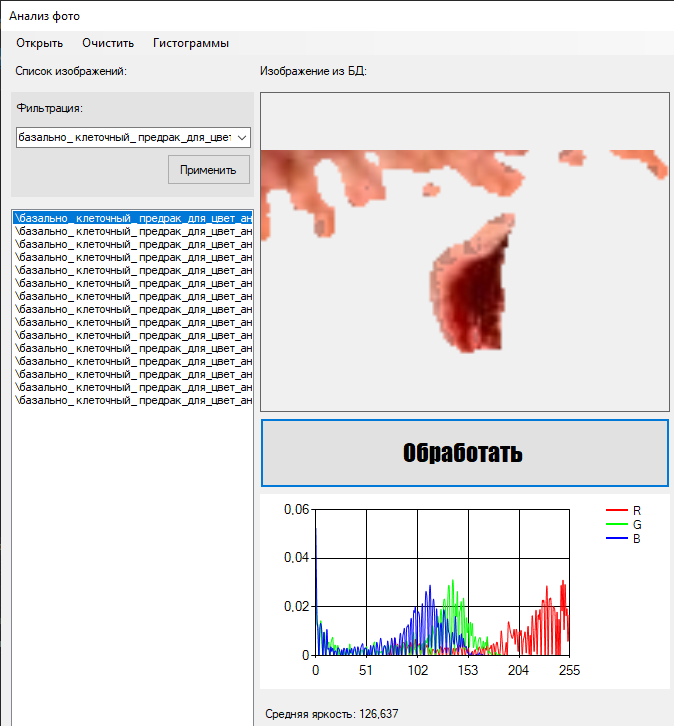


Рисунок 12 - Обработанное изображение

Загружая и проводя исследование над изображением – у него меняется яркость, а соответственно и частоты гистограмм R, G, B. Чтобы наглядно изучить данные – внизу есть информационно-графический блок (Рисунок 13), содержащий в себе следующие элементы:

* значение средней яркости всего изображения;
* значения средней яркости R, G, B;
* значение параметра среднеквадратичного отклонения;
* значение параметра медианы;
* графики частот гистограмм R, G, B и совместный RGB.

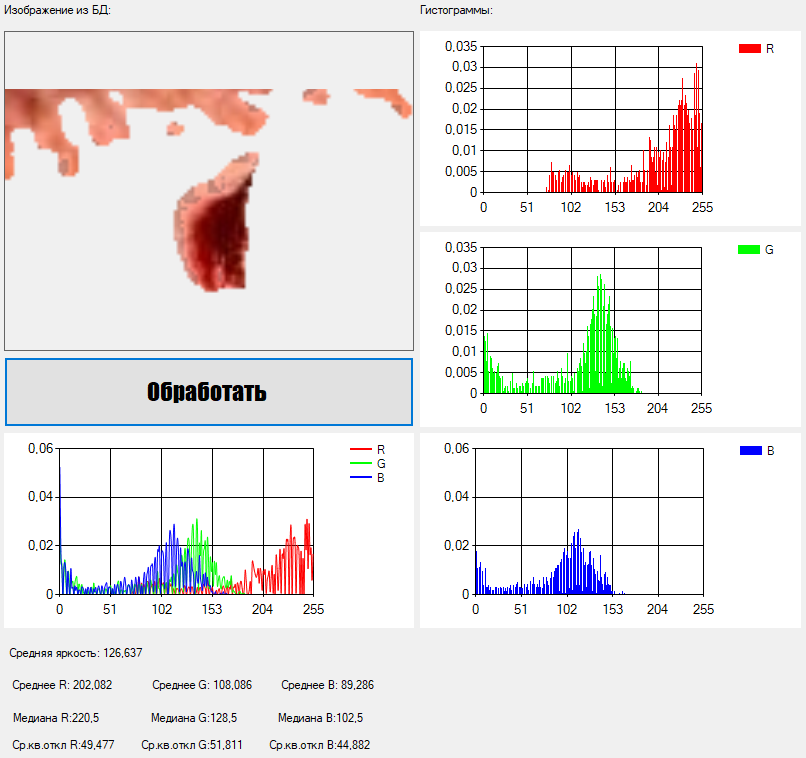


Рисунок 13 - Данные исследования

Добившись необходимых результатов по ходу изучения – существует возможность сохранить его для последующего открытия и просмотра (Рисунок 14).

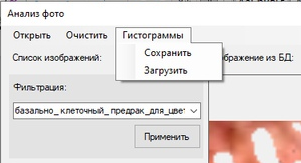


Рисунок 14 - Сохранение и загрузка исследования

Если возникла ситуация, когда необходимо использовать другие экземпляры изображений – это возможно сделать, если выбрать в контекстном меню пункт Очистить – Базу данных (Рисунок 15).

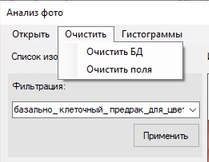


Рисунок 15 - Очистка базы данных и пользовательских полей ввода-вывода

Для обновления экземпляров изображений нужно выбрать в контекстном меню пункт Открыть – Папку (Рисунок 16). После некоторого времени, зависящего от количества и размера файлов, в базе данных будут записаны данные о названии изображения, диагнозе, информации о пикселях.

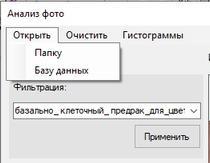


Рисунок 16 - Открытие новых экземпляров изображений

## Вывод по технологической части

На данном этапе были рассмотрены взаимодействие базы данных и программы, используемые классы объектов и их поведение в рамках разрабатываемой информационной системы.

# Исследовательская часть

## Входные данные

Для проведения исследования была создана таблица «Gist\_info» в базе данных, включающая в себя следующие поля (Таблица 4):

1. ID (номер записи).
2. ID\_photo (номер хранимой фотографии).
3. AllBright (общая средняя яркость всего изображения).
4. Img (массив яркостей обработанного изображения).
5. Pos (массив позиций каждого пикселя).
6. avR (среднее значение яркости гистограммы R).
7. avG (среднее значение яркости гистограммы G).
8. avB (среднее значение яркости гистограммы B).
9. MedR (значение медианы гистограммы яркости R).
10. MedG (значение медианы гистограммы яркости G).
11. MedB (значение медианы гистограммы яркости B).
12. SgR (значение среднеквадратического отклонения гистограммы яркости R).
13. SgG (значение среднеквадратического отклонения гистограммы яркости G).
14. SgB (значение среднеквадратического отклонения гистограммы яркости B).
15. Diagnose (заболевание).

В ходе анализа данных, были сформированы несколько таблиц с исходными данными (Таблица 5-8), на основании которых будет сроится поиск зависимостей в цветовой модели

Для выявления зависимостей, необходимо проанализировать соотношение между доступными величинами набора данных пяти фотографий с разными диагнозами:

* базальноклеточный рак;
* базальноклетчатый предрак;
* рак меланома;
* предрак меланома.

На Таблица 5 представлены данные пяти изображений для базальноклетчатого рака.

Таблица 5 - Исходные данные для пяти изображений с диагнозом базальноклеточный рак

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Среднее R | Среднее G | Среднее B | Медиана R | Медиана G | Медиана B | Ср. кв. откл. R | Ср. кв. откл. G | Ср. кв. откл. B |
| 183,739 | 113,473 | 92,176 | 177,5 | 104,5 | 85,5 | 25,226 | 29,724 | 30,407 |
| 193,992 | 110,93 | 101,703 | 190,5 | 109,5 | 102,5 | 23,178 | 26,373 | 26,34 |
| 212,388 | 106,58 | 79,18 | 214,5 | 109,5 | 80,5 | 19,857 | 24,688 | 21,144 |
| 185,315 | 113,285 | 92,028 | 183,5 | 114,5 | 92,5 | 17,496 | 18,468 | 18,933 |
| 170,634 | 113,672 | 130,278 | 162,5 | 111,5 | 126,5 | 28,968 | 22,408 | 22,576 |

На Таблица 6 представлены данные пяти изображений для базальноклетчатого предрака.

Таблица 6 - Исходные данные для пяти изображений с диагнозом базальноклетчатый предрак

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Среднее R | Среднее G | Среднее B | Медиана R | Медиана G | Медиана B | Ср. кв. откл. R | Ср. кв. откл. G | Ср. кв. откл. B |
| 215,875 | 104,555 | 86,033 | 215,5 | 116,5 | 94,5 | 22,14 | 44,727 | 40,07 |
| 175,184 | 115,457 | 98,862 | 174,5 | 115,5 | 98,5 | 13,917 | 13,47 | 13,296 |
| 180,597 | 113,952 | 112,298 | 181,5 | 114,5 | 111,5 | 14,781 | 12,842 | 11,193 |
| 221,727 | 103,034 | 80,103 | 223,5 | 104,5 | 83,5 | 21,065 | 36,813 | 33,391 |
| 198,492 | 108,816 | 96,209 | 198,5 | 108,5 | 94,5 | 17,074 | 17,487 | 17,419 |

На Таблица 7 представлены данные пяти изображений для рака меланома.

Таблица 7 - Исходные данные для пяти изображений с диагнозом рака меланома

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Среднее R | Среднее G | Среднее B | Медиана R | Медиана G | Медиана B | Ср. кв. откл. R | Ср. кв. откл. G | Ср. кв. откл. B |
| 138,332 | 122,612 | 137,829 | 134,5 | 120,5 | 141,5 | 39,425 | 26,981 | 40,065 |
| 140,181 | 119,629 | 161,822 | 144,5 | 121,5 | 177,5 | 73,925 | 51,204 | 66,318 |
| 161,277 | 118,942 | 105,922 | 153,5 | 112,5 | 99,5 | 38,262 | 40,983 | 34,144 |
| 140,672 | 122,641 | 130,538 | 145,5 | 122,5 | 126,5 | 41,81 | 29,658 | 44,341 |
| 141,016 | 122,919 | 124,848 | 142,5 | 123,5 | 120,5 | 39,764 | 30,746 | 42,463 |

На Таблица 8 представлены данные пяти изображений для предрак меланомы.

Таблица 8 - Исходные данные для пяти изображений с диагнозом предрак меланома

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Среднее R | Среднее G | Среднее B | Медиана R | Медиана G | Медиана B | Ср. кв. откл. R | Ср. кв. откл. G | Ср. кв. откл. B |
| 199,635 | 110,071 | 89,265 | 199,5 | 109,5 | 88,5 | 13,833 | 18,721 | 17,432 |
| 185,444 | 119,194 | 80,694 | 185,5 | 119,5 | 78,5 | 14,531 | 14,832 | 15,175 |
| 176,362 | 114,404 | 105,146 | 167,5 | 109,5 | 101,5 | 30,369 | 25,091 | 21,847 |
| 171,73 | 114,07 | 121,673 | 167,5 | 110,5 | 119,5 | 23,439 | 20,841 | 18,683 |
| 168,192 | 117,718 | 99,664 | 164,5 | 120,5 | 100,5 | 17,902 | 15,301 | 15,017 |

## Данные и инструменты анализа

Для проведения исследования, по нахождению зависимостей цветовой модели RGB, были высчитаны новые величины, такие как медиана и среднеквадратическое отклонение.

Расчет среднеквадратического отклонения представлена на формуле (1):

(1)

где – среднее значение яркости, i – индекс значения яркости 0≤i≤255, – частота яркости каждого цвета i.

Расчет медианы представлен на формуле (2):

(2)

где – частота яркости i-го цвета, m – конечное значение яркости, – последнее значение яркости, когда частота близка к 0,5.

Таблица 9 - Высчитывание зависимостей с средними значениями и медианами к диагнозу базальноклеточный рак

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Отношение среднего G и R | Отношение среднего B и R | Отношение медианы R и среднего R | Отношение медианы G и среднего G | Отношение медианы B среднего B |
| 0,617577107 | 0,501668127 | 0,966044226 | 0,92092392 | 0,927573338 |
| 0,571827704 | 0,524263887 | 0,981999258 | 0,987108988 | 1,007836544 |
| 0,501817428 | 0,372808257 | 1,009944065 | 1,02739726 | 1,016670876 |
| 0,611310471 | 0,496603081 | 0,990205866 | 1,010725162 | 1,005128874 |
| 0,666174385 | 0,763493794 | 0,95233072 | 0,980892392 | 0,971000476 |

Таблица 10 - Высчитывание зависимостей с среднеквадратического отклонения и средней яркости к диагнозу базальноклеточный рак

|  |  |  |
| --- | --- | --- |
| Отношение среднеквадратического отклонения R и среднего R | Отношение среднеквадратического отклонения G и среднего G | Отношение среднеквадратического отклонения B и среднего B |
| 0,137293 | 0,261948 | 0,32988 |
| 0,119479 | 0,237745 | 0,258989 |
| 0,093494 | 0,231638 | 0,267037 |

Окончание Таблица 10

|  |  |  |
| --- | --- | --- |
| Отношение среднеквадратического отклонения R и среднего R | Отношение среднеквадратического отклонения G и среднего G | Отношение среднеквадратического отклонения B и среднего B |
| 0,094412 | 0,163022 | 0,205731 |
| 0,169767 | 0,197129 | 0,173291 |

Таблица 11 - Высчитывание зависимостей с средними значениями и медианами к диагнозу базальноклетчатый предрак

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Отношение среднего G и R | Отношение среднего B и R | Отношение медианы R и среднего R | Отношение медианы G и среднего G | Отношение медианы B среднего B |
| 0,484331 | 0,398532 | 0,998263 | 1,114246 | 1,098416 |
| 0,659061 | 0,564332 | 0,996096 | 1,000372 | 0,996338 |
| 0,630974 | 0,621815 | 1,005 | 1,004809 | 0,992894 |
| 0,464689 | 0,361269 | 1,007996 | 1,014228 | 1,042408 |
| 0,548214 | 0,4847 | 1,00004 | 0,997096 | 0,982237 |

Таблица 12 - Высчитывание зависимостей с среднеквадратического отклонения и средней яркости к диагнозу базальноклеточный предрак

|  |  |  |
| --- | --- | --- |
| Отношение среднеквадратического отклонения R и среднего R | Отношение среднеквадратического отклонения G и среднего G | Отношение среднеквадратического отклонения B и среднего B |
| 0,102559 | 0,427784 | 0,465752 |
| 0,079442 | 0,116667 | 0,134491 |
| 0,081845 | 0,112697 | 0,099672 |
| 0,095004 | 0,35729 | 0,416851 |
| 0,086019 | 0,160702 | 0,181054 |

Таблица 13 - Высчитывание зависимостей с средними значениями и медианами к диагнозу рака меланома

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Отношение среднего G и R | Отношение среднего B и R | Отношение медианы R и среднего R | Отношение медианы G и среднего G | Отношение медианы B среднего B |
| 0,88636 | 0,996364 | 0,972299 | 0,982775 | 1,026634 |
| 0,85339 | 1,154379 | 1,03081 | 1,01564 | 1,096884 |
| 0,737501 | 0,656771 | 0,951779 | 0,945839 | 0,93937 |

Окончание Таблица 13

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Отношение среднего G и R | Отношение среднего B и R | Отношение медианы R и среднего R | Отношение медианы G и среднего G | Отношение медианы B среднего B |
| 0,871822 | 0,92796 | 1,034321 | 0,99885 | 0,969066 |
| 0,871667 | 0,885346 | 1,010524 | 1,004727 | 0,965174 |

Таблица 14 - Высчитывание зависимостей с среднеквадратического отклонения и средней яркости к диагнозу рака меланома

|  |  |  |
| --- | --- | --- |
| Отношение среднеквадратического отклонения R и среднего R | Отношение среднеквадратического отклонения G и среднего G | Отношение среднеквадратического отклонения B и среднего B |
| 0,285002747 | 0,220051871 | 0,290686285 |
| 0,527353921 | 0,428023305 | 0,409820667 |
| 0,237243996 | 0,344562896 | 0,322350409 |
| 0,297216219 | 0,241827774 | 0,339678867 |
| 0,281982186 | 0,250132201 | 0,340117583 |

Таблица 15 - Высчитывание зависимостей с средними значениями и медианами к диагнозу предрак меланома

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Отношение среднего G и R | Отношение среднего B и R | Отношение медианы R и среднего R | Отношение медианы G и среднего G | Отношение медианы B среднего B |
| 0,551361 | 0,447141 | 0,999324 | 0,994812 | 0,99143 |
| 0,642749 | 0,435139 | 1,000302 | 1,002567 | 0,972811 |
| 0,648688 | 0,596194 | 0,949751 | 0,957134 | 0,965324 |
| 0,66424 | 0,708513 | 0,975368 | 0,968703 | 0,982141 |
| 0,699902 | 0,592561 | 0,978049 | 1,023633 | 1,008388 |

Таблица 16 - Высчитывание зависимостей с среднеквадратического отклонения и средней яркости к диагнозу предрак меланома

|  |  |  |
| --- | --- | --- |
| Отношение среднеквадратического отклонения R и среднего R | Отношение среднеквадратического отклонения G и среднего G | Отношение среднеквадратического отклонения B и среднего B |
| 0,069291 | 0,170081 | 0,195284 |
| 0,078358 | 0,124436 | 0,188056 |
| 0,172197 | 0,219319 | 0,207778 |

Окончание Таблица 16

|  |  |  |
| --- | --- | --- |
| Отношение среднеквадратического отклонения R и среднего R | Отношение среднеквадратического отклонения G и среднего G | Отношение среднеквадратического отклонения B и среднего B |
| 0,136488 | 0,182704 | 0,153551 |
| 0,106438 | 0,12998 | 0,150676 |

По результатам исследования, были определены диапазоны возможных значений самих характеристик гистограмм и некоторых их комбинаций. Результат представлен в Таблица 17-19.

Таблица 17 - Диапазоны значений по каждому заболеванию

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Среднее R | Среднее G | Среднее B | Медиана R | Медиана G | Медиана B | Ср. кв. откл. R | Ср. кв. откл. G | Ср. кв. откл. B |
| Базальноклетчатый рак | | | | | | | | |
| Мин | 145,445 | 62,735 | 25,76 | 119,5 | 56,5 | 19,5 | 7,568 | 8,458 | 9,45 |
| Макс | 232,189 | 122,894 | 149,705 | 235,5 | 121,5 | 150,5 | 43,518 | 56,939 | 51,481 |
|  | Базальноклетчатый предрак | | | | | | | | |
| Мин | 153,442 | 103,034 | 80,103 | 139,5 | 104,5 | 83,5 | 11,284 | 10,396 | 10,277 |
| Макс | 221,727 | 121,237 | 121,649 | 223,5 | 116,5 | 121,5 | 41,305 | 44,727 | 40,07 |
|  | Рак меланома | | | | | | | | |
| Мин | 101,675 | 82,812 | 44,624 | 103,5 | 64,5 | 38,5 | 13,427 | 15,475 | 16,575 |
| Макс | 232,684 | 134,011 | 163,852 | 233,5 | 139,5 | 177,5 | 73,925 | 51,204 | 66,318 |
|  | Предрак меланома | | | | | | | | |
| Мин | 168,192 | 106,133 | 80,694 | 164,5 | 105,5 | 78,5 | 13,833 | 14,832 | 15,017 |
| Макс | 210,704 | 119,194 | 121,742 | 210,5 | 120,5 | 119,5 | 30,369 | 26,454 | 29,133 |

Таблица 18 - Диапазоны отношений параметров яркости G и B к R

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Отношение avG и avR | Отношение avB и avR | Отношение MedG и MedR | Отношение MedB и MedR | Отношение SgG и SgR | Отношение SgB и SgR |
| Базальноклетчатый рак | | | | | |
| Мин | 0,303197993 | 0,124497973 | 0,964394086 | 0,730100398 | 0,345132743 | 0,064961373 |
| Макс | 0,838495651 | 0,990459619 | 7,744565217 | 2,193322981 | 1,303030303 | 0,417274336 |
|  | Базальноклетчатый предрак | | | | | |
| Мин | 0,464688558 | 0,361268587 | 1,171402971 | 0,900130704 | 0,786046512 | 0,103050228 |
| Макс | 0,790116135 | 0,741057287 | 2,790157672 | 1,354131554 | 1,089686099 | 0,377214612 |
|  | Рак меланома | | | | | |
| Мин | 0,355898987 | 0,191779409 | 0,77749399 | 0,723213632 | 0,483253589 | 0,11427234 |
| Макс | 1,318032948 | 1,309269732 | 5,232610255 | 1,888565594 | 1,46090535 | 0,608436214 |
|  | Предрак меланома | | | | | |
| Мин | 0,503706622 | 0,40857791 | 1,375860426 | 0,907657177 | 0,656903766 | 0,121598326 |
| Макс | 0,699902492 | 0,711055819 | 2,445143979 | 1,480903165 | 1,081447964 | 0,277342466 |

Таблица 19 - Диапазон отношений медиан и среднеквадратических отклонений к средней яркости

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Отношение MedR и avR | Отношение MedG и avG | Отношение MedB и avB | Отношение SgR и avR | Отношение SgG и avG | Отношение SgB и avB |
| Базальноклетчатый рак | | | | | |
| Мин | 0,821616419 | 0,873272929 | 0,756987578 | 0,043615574 | 0,072387114 | 0,088034114 |
| Макс | 1,03651485 | 1,12455018 | 1,079060154 | 0,269992093 | 0,506272951 | 0,779619565 |
|  | Базальноклетчатый предрак | | | | | |
| Мин | 0,868076708 | 0,90318962 | 0,913364799 | 0,058525453 | 0,093792008 | 0,0996723 |
| Макс | 1,00799632 | 1,114246091 | 1,098415724 | 0,25162041 | 0,42778442 | 0,465751514 |
|  | Рак меланома | | | | | |
| Мин | 0,822126776 | 0,778872627 | 0,810469054 | 0,05911021 | 0,12629457 | 0,153266399 |
| Макс | 1,093382196 | 1,068821287 | 1,096884231 | 0,527353921 | 0,4462276 | 0,622086769 |
|  | Предрак меланома | | | | | |
| Мин | 0,94975108 | 0,957134366 | 0,965324406 | 0,069291457 | 0,124435794 | 0,144313384 |
| Макс | 1,000301978 | 1,023632749 | 1,008388184 | 0,172196959 | 0,239810719 | 0,303168739 |

Проверим получившиеся данные и проверим вхождение в диапазоны данных исходных пяти изображений на соответствующие им диагнозы.

Таблица 20 - Проверка исходных значений на вхождение в соответствующий диапазон

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Попадание в диапазон | Среднее R | Среднее G | Среднее B | Медиана R | Медиана G | Медиана B | Ср. кв. откл. R | Ср. кв. откл. G | Ср. кв. откл. B |
| Базальноклетчатый рак | | | | | | | | |
| + | + | + | + | + | + | + | + | + |
| + | + | + | + | + | + | + | + | + |
| + | + | + | + | + | + | + | + | + |
| + | + | + | + | + | + | + | + | + |
| + | + | + | + | + | + | + | + | + |
| Базальноклетчатый предрак | | | | | | | | |
| + | + | + | + | + | + | + | + | + |
| + | + | + | + | + | + | + | + | + |
| + | + | + | + | + | + | + | + | + |
| + | + | + | + | + | + | + | + | + |
| + | + | + | + | + | + | + | + | + |
| Рак меланома | | | | | | | | |
| + | + | + | + | + | + | + | + | + |
| + | + | + | + | + | + | + | + | + |
| + | + | + | + | + | + | + | + | + |
| + | + | + | + | + | + | + | + | + |
| + | + | + | + | + | + | + | + | + |
| Предрак меланома | | | | | | | | |
| + | + | + | + | + | + | + | + | + |
| + | + | + | + | + | + | + | + | + |
| + | + | + | + | + | + | + | + | + |
| + | + | + | + | + | + | + | + | + |
| + | + | + | + | + | + | + | + | + |

Проверим попадут ли исходные данные пяти изображений из диагноза базальноклетчатого предрака под диагнозы предрака меланома, рак меланома и законченную стадию базальноклетчатого рака (Таблица 21).

Таблица 21 - Сопостовление базальтоклетчатого предрака с другими диагнозапи

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Попадание в диапазон | Среднее R | Среднее G | Среднее B | Медиана R | Медиана G | Медиана B | Ср. кв. откл. R | Ср. кв. откл. G | Ср. кв. откл. B |
| Базальноклетчатый рак | | | | | | | | |
| + | + | + | + | + | + | + | + | + |
| + | + | + | + | + | + | + | + | + |
| + | + | + | + | + | + | + | + | + |
| + | + | + | + | + | + | + | + | + |
| + | + | + | + | + | + | + | + | + |
| Базальноклетчатый предрак (Исходный) | | | | | | | | |
| + | + | + | + | + | + | + | + | + |
| + | + | + | + | + | + | + | + | + |
| + | + | + | + | + | + | + | + | + |
| + | + | + | + | + | + | + | + | + |
| + | + | + | + | + | + | + | + | + |
| Рак меланома | | | | | | | | |
| + | + | + | + | + | + | + | + | + |
| + | + | + | + | + | + | + | - | - |
| + | + | + | + | + | + | + | - | - |
| + | + | + | + | + | + | + | + | + |
| + | + | + | + | + | + | + | + | + |
| Предрак меланома | | | | | | | | |
| - | - | + | - | + | + | + | - | - |
| + | + | + | + | + | + | + | - | - |
| + | + | + | + | + | + | + | - | - |
| - | - | - | - | - | + | + | - | - |
| + | + | + | + | + | + | + | + | + |

По результатам сравнения базальноклетчатый предрака с другими диагнозами, базальноклетчатый предрак полностью совпал с его конечной стадией, но не подошел по некоторым параметрам к другим диагнозам.

## Вывод по исследовательской части

Основываясь на исходных данных (Таблица 5-8) и полученных путем анализа данных о диапазонах (Таблица 17-19), а также сравнением одного типа диагноза с другими (Таблица 21), можно наблюдать, что диапазоны для некоторых параметров и комбинаций получаются разные для каждого вида диагноза.

То есть, имея обработанные данные и вычисленные диапазоны, можно, в принципе, определить тип диагноза, но не его стадию. Тип предположительного диагноза можно получить, подставляя значения обработки его параметров в заранее высчитанные диапазоны этих же значений.

# Заключение

В результате выпускной квалификационной работы была выполнена цель, а именно исследование возможности автоматического распознавания групп риска посредством цветового анализа цифровых фотографий кожных заболеваний. Чтобы достичь цели, были также решены следующие задачи:

* произведен анализ предметной области;
* выбраны требуемые средства для разработки программы;
* разработана базу данных для хранения выделенных точек изображения;
* разработана программу для вычисления характеристик изображения;
* сделаны контекстные модели представления данных и описаны их спецификации;
* изображена работа алгоритмов в виде блок-схем;
* исследованы изображения кожных заболеваний на наличие закономерностей.

В ходе исследования было установлено, что допустимо возможно распознавание конкретно взятого изображения. Диапазоны для некоторых параметров и комбинаций существенно разные, но иногда они накладываются или пересекаются. Так что возможно определить тип диагноза, но не его стадию.

По материалам выпускной квалификационной работы были опубликованы две статьи [7-8].

Список используемой литературы

1. Елькин В. Д., Митрюковский Л. С., Лысов А. Ю. Практическая дерматоонкология: иллюстрированное справочное руководство по опухолям кожи, опухолеподобным заболеваниям и связанным с ними синдромам. – М.: Практическая медицина, 2014. – 480 с.

2. Атлас по классификации стадий злокачественных опухолей: приложение к 7-му изданию «Руководства по (TNM) классификации стадий злокачественных опухолей» и «Справочника» AJCC: пер. с англ. – 2-е изд. / под ред. А. Д. Каприна, А. Х. Трахтенберга. – М.: Практическая медицина, 2014. – 649 с.

3. МЕДСИ // Базальноклеточный рак (базалиома). [Электронный ресурс] <https://spb.medsi.ru/articles/bazalnokletochnyy-rak-bazalioma/> (дата обращения 10.03.2022).

4. Приоров, А.Л. Цифровая обработка изображений: учебное пособие / А.Л. Приоров, И.В. Апальков, В.В. Хрящев; Яросл. ос. ун-т. – Ярославль: ЯрГУ, 2007. – 235 с.

5. Фисенко В.Т., Фисенко Т.Ю. Компьютерная обработка и распознавание

изображений. СПб: СПбГУ ИТМО. 2008. 192 с.

6. А. О. Боровкова, О. В. Рвачева, А. М. Чмутин. Управление яркостным

контрастом: от телевидения к компьютерной технике // Журнал

радиоэлектроники No2, 2012.

7. Матвеев В.А., Раухваргер А.Б. Исследование возможности использования цветной модели RGB для автоматизации спектрального анализа. / Сб. материалов конф. В 2 ч. Ч. 1 [Электронный ресурс]. – Ярославль: Издательство ЯГТУ, 2021. с 659-661.

8. Матвеев В.А., Раухваргер А.Б. Исследование возможности автоматического распознавания групп риска посредством цветового анализа цифровых фотографий кожных заболеваний. / Сб. материалов конф. В 3 ч. Ч. 3 [Электронный ресурс]. – Ярославль. Издательство ЯГТУ, 2022. с 18-21.