

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное автономное образовательное учреждение  
высшего образования  
«Уральский федеральный университет  
имени первого Президента России Б.Н. Ельцина»

Институт радиоэлектроники и информационных технологий - РТФ

Школа профессионального и академического образования

ДОПУСТИТЬ К ЗАЩИТЕ ПЕРЕД ГЭК

РОП 09.04.03 Медведева М.А.

«01» июня 2024 г.

**МАГИСТЕРСКАЯ ДИССЕРТАЦИЯ**

«Совершенствование подхода к сегментации кровеносных сосудов сетчатки с  
применением нейронных сетей»

Научный руководитель:  
к.т.н., с.н.с., доцент

\_\_\_\_\_

подпись

Медведев А.Н.

Научный руководитель:  
ассистент

\_\_\_\_\_

подпись

Балунгу Д.М.

Студент группы  
РИМ-220981

\_\_\_\_\_

подпись

Мурас Д.К.

Екатеринбург  
2024

## РЕФЕРАТ

Тема магистерской диссертации:

«Совершенствование подхода к сегментации кровеносных сосудов сетчатки с применением нейронных сетей»

Магистерская диссертация выполнена на 122 страницах, содержит 16 таблиц, 29 рисунков, 81 использованный источник.

Автоматизация процесса сегментации кровеносных сосудов сетчатки с использованием нейронных сетей может существенно улучшить процесс определения существования глазных заболеваний и добиться получения наилучшей точности, а также уменьшить временную нагрузку на медицинских работников и сократить денежные расходы на возможное лечение.

Цель работы – усовершенствовать существующие методы и алгоритмы сегментации кровеносных сосудов сетчатки глаза на медицинских изображениях с использованием нейронных сетей.

Объектом исследования является процесс определения кровеносных сосудов сетчатки в изображениях глазного дна с применением нейронных сетей.

Предмет исследования – совершенствование методов и алгоритмов нейронных сетей для выделения кровеносных сосудов на изображениях глазного дна.

Научная новизна работы заключается в проведении совершенствования архитектуры нейронной сети CG-ResUnet для более эффективной обработки изображений глазного дна. В отличие от существующих методов сегментации, которые могут достигать высокой точности, но сталкиваются с проблемами в интеграции в медицинское оборудование, предложенный подход направлен на значительное снижение потребления ресурсов и оптимизацию производительности. Будут разработаны новые стратегии сегментации

кровеносных сосудов различной ширины и типов, что позволит получить более полное представление о состоянии сосудистой системы глазного дна.

Практическая реализация таких мер позволит значительно повысить эффективность работы медицинских учреждений за счет сокращения времени обработки изображений. Наш предложенный метод способен обрабатывать несколько десятков изображений всего за одну минуту, в то время как существующие методы могут занимать от 3 до 30 минут на одно изображение, в зависимости от квалификации медицинского специалиста и точности установления анализа. Подобное улучшение сократит время, затрачиваемое на обработку, и повысит производительность труда медицинских работников, за счет снижения количества повторяющихся задач.

Экономическая эффективность предложенных мер подтверждается значительным уменьшением времени обработки изображений, что приводит к повышению производительности труда медицинских работников и сокращению временных затрат при проведении медицинской диагностики. Автоматизируя процесс сегментации, наш метод позволяет оптимизировать рабочие процессы и уменьшить общие трудозатраты на лечение пациентов.

## **СОДЕРЖАНИЕ**

|                                                                                                                                           |           |
|-------------------------------------------------------------------------------------------------------------------------------------------|-----------|
| <b>ВВЕДЕНИЕ .....</b>                                                                                                                     | <b>6</b>  |
| <b>1 ОБЗОР НАУЧНОЙ ЛИТЕРАТУРЫ И СОВРЕМЕННЫХ МЕТОДОВ<br/>СЕГМЕНТАЦИИ КРОВЕНОСНЫХ СОСУДОВ СЕТЧАТКИ ГЛАЗНОГО ДНА</b>                         | <b>11</b> |
| 1.1 Актуальность разработки системы сегментации кровеносных сосудов<br>сетчатки .....                                                     | 11        |
| 1.2 Определение области исследования.....                                                                                                 | 13        |
| 1.3 Обзор литературных источников. Современные методы сегментации<br>кровеносных сосудов сетчатки в изображениях глазного дна .....       | 21        |
| 1.4 Обзор основных архитектур нейронных сетей для сегментации сосудов<br>сетчатки .....                                                   | 27        |
| 1.5 Постановка задачи управления организационной системы .....                                                                            | 33        |
| 1.6 Результаты обзора литературы. Обсуждение выбранных методов<br>сегментации кровеносных сосудов .....                                   | 43        |
| <b>2 РАЗРАБОТКА МЕТОДОЛОГИИ ДЛЯ СЕГМЕНТАЦИИ<br/>КРОВЕНОСНЫХ СОСУДОВ СЕТЧАТКИ ГЛАЗА.....</b>                                               | <b>46</b> |
| 2.1 Схематическое описание алгоритма сегментации кровеносных сосудов<br>сетчатки .....                                                    | 46        |
| 2.2 Описание используемых данных для задачи сегментации кровеносных<br>сосудов сетчатки в изображениях глазного дна .....                 | 49        |
| 2.3 Предобработка изображений. Применение фильтров Кирша с методом<br>адаптивного гистограммного выравнивания CLANE .....                 | 54        |
| 2.4 Архитектура модели свёрточной нейронной сети. Адаптация архитектуры<br>Unet под задачу сегментации кровеносных сосудов сетчатки ..... | 58        |
| 2.5 Модификация стандартного процесса сегментации кровеносных сосудов<br>сетчатки. Результаты разработки методологии .....                | 62        |

|                                                                                                                                     |            |
|-------------------------------------------------------------------------------------------------------------------------------------|------------|
| <b>3 РАЗРАБОТКА СВЕРТОЧНОЙ НЕЙРОННОЙ СЕТИ ДЛЯ СЕГМЕНТАЦИИ КРОВЕНОСНЫХ СОСУДОВ СЕТЧАТКИ ГЛАЗА НА ИЗОБРАЖЕНИЯХ ГЛАЗНОГО ДНА .....</b> | <b>64</b>  |
| 3.1 Программная реализация алгоритма предобработки изображений. Результаты применения фильтров Кирша с методом CLANE .....          | 64         |
| 3.2 Процесс обучения модели свёрточной нейронной сети. ....                                                                         | 67         |
| 3.3 Результаты обучения и анализ полученных результатов.....                                                                        | 71         |
| 3.4 Обсуждение и анализ результатов обучения модели нейронной сети. ....                                                            | 77         |
| <b>ЗАКЛЮЧЕНИЕ .....</b>                                                                                                             | <b>80</b>  |
| <b>СПИСОК ЛИТЕРАТУРЫ .....</b>                                                                                                      | <b>83</b>  |
| <b>ПРИЛОЖЕНИЕ А .....</b>                                                                                                           | <b>90</b>  |
| <b>ПРИЛОЖЕНИЕ Б .....</b>                                                                                                           | <b>92</b>  |
| <b>ПРИЛОЖЕНИЕ В .....</b>                                                                                                           | <b>95</b>  |
| <b>ПРИЛОЖЕНИЕ Г .....</b>                                                                                                           | <b>105</b> |
| <b>ПРИЛОЖЕНИЕ Е .....</b>                                                                                                           | <b>113</b> |
| <b>ПРИЛОЖЕНИЕ Ж .....</b>                                                                                                           | <b>118</b> |

## **ВВЕДЕНИЕ**

В наши дни медицинская индустрия является одной из самых передовых и инновационных сфер, где специалисты постоянно стремятся к созданию новых и эффективных методов диагностики различных заболеваний и патологий. Практически все методы обследования, используемые в медицине, теперь автоматизированы и основаны на компьютерных технологиях, включая диагностику заболеваний глаза.

Выделение кровеносных сосудов сетчатки глаза играет ключевую роль в диагностике и мониторинге различных глазных заболеваний и связанных с ними патологий. Правильная сегментация кровеносных сосудов позволяет установить более точный и надежный диагноз, что критически важно для эффективного лечения пациентов. Анализ изображений глазного дна и выделение кровеносных сосудов позволяют выявить на ранней стадии различные патологии, такие как диабетическая ретинопатия, нарушения пигментации глазного дна, разрывы сетчатки и тромбоз сосудов глаза.

Автоматизация процесса сегментации кровеносной системы глазного дна и его дальнейшее усовершенствование позволят существенно сократить временные и вычислительные затраты. Это может значительно ускорить и снизить стоимость процесса профилактики и диагностики заболеваний у пациентов.

С развитием искусственного интеллекта и машинного обучения становится возможным применение нейронных сетей для автоматизации процесса сегментации кровеносных сосудов сетчатки на изображениях глазного дна. Эти нейронные сети представляют собой эффективную замену ручному труду медицинских специалистов, позволяя сэкономить много времени до того, как заболевание достигнет активной стадии.

Исследования в области автоматической сегментации активно развиваются и привлекают все больше внимания. Количество научных публикаций, посвященных сегментации кровеносных сосудов сетчатки, стремительно возрастает. Согласно данным научного портала Elsevier, число таких работ увеличилось в 2,5 раза в период с 2018 по 2023 год. Этот значительный рост отражает не только увеличивающийся интерес ученых и медицинских специалистов к этой проблеме, но и подчеркивает нарастающую необходимость в разработке надежного и точного метода сегментации кровеносных сосудов сетчатки. Среди основных недостатков существующих методов сегментации следует отметить игнорирование этапа предварительной обработки медицинских изображений, что может привести к обучению моделей на необработанных данных.

Таким образом, цель данного исследования состоит в совершенствовании методов и алгоритмов сегментации кровеносных сосудов сетчатки в медицинских изображениях глазного дна с применением нейронных сетей.

Для достижения этой цели были сформулированы следующие задачи:

- 1) провести поиск научных публикаций и патентной документации, относящихся к теме проводимого исследования, и выполнить анализ;
- 2) определить наиболее эффективные и точные подходы к сегментации кровеносных сосудов;
- 3) постановить задачу управления для медицинских организаций;
- 4) выделить новаторские элементы предложенного метода и описать влияние на показатели эффективности предприятия;
- 5) создать методику внедрения предложенной инновации и описать её практическую значимость.

Объектом работы выступает процесс выделения кровеносных сосудов сетчатки в изображениях глазного дна с помощью нейронных сетей.

Предмет исследования – совершенствование методов и алгоритмов нейронных сетей для сегментации кровеносных сосудов сетчатки в изображениях глазного дна.

Для разработки более совершенных методов сегментации кровеносных сосудов сетчатки была проведена всесторонняя оценка 1554 научных работ. Из них было отобрано 60 наиболее релевантных научных публикаций, доступных на платформах eLibrary и Elsevier, которые были тщательно проанализированы. Дополнительно были изучены 10 существующих патентов, размещенных на платформах Роспатент и Google Patents. В результате этого анализа были выделены основные методы сегментации кровеносных сосудов и изучены архитектуры нейронных сетей, применяемые для автоматизации процесса сегментации.

В результате комплексного анализа медицинской литературы и научных материалов были выявлены два эффективных метода для предобработки изображений: адаптивная гистограммная корректировка с ограничением контрастности (CLANE) и использование серии фильтров Кирша.

Внедрение нашей инновации в практику существенно увеличит эффективность работы предприятия за счет сокращения времени, затрачиваемого на обработку изображений данных. В настоящее время система в ее текущем состоянии требует от 3 до 30 минут на загрузку, обработку и анализ данных в зависимости от квалификации специалиста и сложности анализируемых данных. Наш метод позволяет обрабатывать десятки фотографий всего за одну минуту. Ожидается, что такая оптимизация значительно повысит производительность труда врачей за счет сокращения времени, затрачиваемого на повторяющиеся монотонные операции.

В данной работе используется стандартная структура, включающая введение, три основные главы, заключение, список использованных источников и приложения.

Во введении исследования обосновывается его актуальность, формулируются цели и задачи исследования, а также описывается методика, применяемая в проведении исследования, и практическая значимость результатов.

В первой главе исследования был проведен поиск и углубленный анализ эмпирической базы научных трудов, статей и патентов, опубликованных за последние пять лет по теме исследования. Для более точного выбора публикаций к анализу была определена предметная область, а также был автоматизирован процесс сбора информации о доступности работ в открытом доступе с помощью разработанного Python скрипта для парсинга данных. На основе тщательно отобранных исследований были выделены ключевые методы обработки изображений и детально проанализированы архитектуры нейронных сетей, используемые для сегментации кровеносных сосудов сетчатки.

Во второй главе подробно рассмотрены методы, которые были применены для улучшения алгоритма сегментации и достижения наиболее точных результатов при использовании нейронной сети. Особое внимание уделено методу адаптивной гистограммной коррекции с ограничением контрастности (CLAHE) и применению фильтров Кирша. Эти методы были выбраны из-за их эффективности в предобработке изображений и выделении кровеносных сосудов на изображениях сетчатки глаза. В главе представлено описание принципов работы этих методов и их влияние на качество сегментации.

В третьей главе данного исследования представлено подробное описание разработки программного кода, специально созданного для процесса сегментации кровеносных сосудов сетчатки глаза. В этой главе приведены основные этапы разработки кода, используемые алгоритмы и структура программы для обеспечения эффективной сегментации сосудов на медицинских изображениях.

В заключении исследования подведены основные результаты работы. Описаны способы усовершенствования предложенного метода сегментации кровеносных сосудов сетчатки. Также подчеркнута практическая значимость новаторского решения и его потенциал для применения в медицинской практике.

# **1 ОБЗОР НАУЧНОЙ ЛИТЕРАТУРЫ И СОВРЕМЕННЫХ МЕТОДОВ СЕГМЕНТАЦИИ КРОВЕНОСНЫХ СОСУДОВ СЕТЧАТКИ ГЛАЗНОГО ДНА**

## **1.1 Актуальность разработки системы сегментации кровеносных сосудов сетчатки**

Глазное дно является единственной частью человеческого тела, кровеносную систему которой можно наблюдать непосредственно. Артерии и вены кровеносной системы глазного дна играют большую роль при диагностике различных заболеваний. При анализе длины, ширины, извилистости, разветвленности кровеносных сосудов сетчатки специалисты могут диагностировать различные заболевания и патологии. Так, Радха К. и Каруна Ю., и Наир А. и др., в своей работе нашли значимые связи между структурными изменениями в сосудах сетчатки и диабетической ретинопатией [49, 53]. Благодаря систематическому анализу авторы объединили результаты 35 исследований с 1980 по 2008 год, из которых им удалось отобрать данные, в виде изображений кровеносных сосудов сетчатки глазного дна, у 22896 пациентов и провести диагностику данных изображений на наличие диабетической ретинопатией. В результате было выявлено, что общая распространенность данного заболевания составила 34,6%. Помимо этого, Федотов Е. А., Черноморец Д. А. и Михелев В. М. в своей работе, на основе исследований зарубежных авторов показывают, что при помощи анализа снимков глазного дна можно выявить и локализовать: отслойку сетчатки, разрывы сетчатки, новообразования и нарушения пигментации глазного дна, микрокровоизлияния в сетчатку и под пигментный эпителий сетчатки, тромбоз сосудов глаза [20].

Насонов А. В. и другие утверждают, что выделение сосудистой системы вручную представляет собой достаточно сложный и трудоемкий процесс,

занимающий существенное количество времени, и иногда невозможный из-за слишком сложной структуры сосудистого дерева или низкого PSNR (peak signal-to-noise ratio) на изображении [2]. Информация из снимков кровеносных сосудов глазного дна собирается с помощью различных методов обработки изображений. Прослеживание (трассировка) и сегментация сосудов являются двумя основными методами выделения сосудов на изображениях глазного дна. Первый метод позволяет выделить все дерево сосудов в одной итерации. Для дальнейшего поиска информации о сосудах можно использовать сегментацию изображений. Существует множество различных подходов к сегментации, поэтому необходимо сравнить их и определить наиболее подходящий.

Моокиах М. Р. К. и другие в своем исследовании произвели обзор основных методов сегментации и классификации кровеносных сосудов глазного дна [48]. Авторы провели обширный анализ литературы по данной теме, включая работы с 2012 по 2020 год, уделяя пристальное внимание методам классификации, которые базируются на применении машинного и глубинного обучения для автоматической сегментации изображений. В обзоре все методы разделены на классы по: задачам (сегментация или классификация артерий и вен), методам (контролируемые или неконтролируемые, глубокое и неглубокое обучение, ручные методы) и конкретным алгоритмам (например, мультимасштабность, морфология).

Кумар К. С. и Сингх Н. П. провели всеобъемлющий обзор современных автоматических методов классификации кровеносных сосудов в изображениях глазного дна и сравнили исследования их по таким показателям как: Accuracy, Sensitivity (Recall), Specificity и Precision [40]. Благодаря этому можно оценить эффективность того или иного метода и выбрать наиболее точные методы классификации для дальнейшей модернизации и достижения более высоких показателей точности.

В приведенных ранее литературных обзорах по предметной области авторы отмечают, что существующие методы всё еще не могут достигнуть наивысших показателей точности, которые бы были надежны для применения в реальных условиях. А также не были затронуты новые архитектуры нейронных сетей, которые можно применять для анализа медицинских изображений.

## **1.2 Определение области исследования**

При проведении литературного обзора был использован системный подход, ориентированный на всестороннее изучение методов сегментации кровеносных сосудов сетчатки на изображениях глазного дна, с особым упором на архитектуру методов машинного обучения. Критерии включения публикаций в обзор определялись научными статьями, материалами конференций и книгами, содержащими детальное описание методов сегментации кровеносных сосудов сетчатки, а также обзор используемых методологий машинного обучения. В рамках исследования также рассматривались патентные документы, содержащие подробные описания указанных методов и их архитектур.

Для оптимизации процесса отбора литературы применялись критерии исключения, направленные на исключение научных статей, материалов конференций, патентов и книг, не содержащих полных описаний методов сегментации кровеносных сосудов сетчатки на изображениях глазного дна. Документы, недоступные для широкой публики или недоступные по корпоративной подписке УрФУ, также были исключены из анализа.

Формулировка вопросов обзора играла ключевую роль в проведении исследования. Первый вопрос направлен на выявление эффективных методов сегментации кровеносных сосудов сетчатки с применением нейронных сетей и оценку их характеристик. Второй вопрос затрагивает потенциальные улучшения существующих методов для повышения качества модели. Формулировки вопросов представлены в таблице 1.

Таблица 1 – Формулировка вопросов исследования

| Вопрос   | Формулировка вопроса                                                                                                                                                                                                                                                                 |
|----------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1 вопрос | Какие методы, основанные на применении нейронных сетей, являются наиболее эффективными для сегментации кровеносных сосудов сетчатки в изображениях глазного дна, обладая высокими показателями точности (accuracy), чувствительности (recall), специфичности и точности (precision)? |
| 2 вопрос | Как можно улучшить существующие методы сегментации кровеносных сосудов сетчатки в изображениях глазного дна для достижения более высокого качества модели?                                                                                                                           |

Предполагаемые пользователи результатов - администраторы ИТ процессов в медицинских центрах и врачи. Исследование посвящено применению нейронных сетей для сегментации кровеносных сосудов сетчатки в контексте автоматизации диагностики глазных заболеваний. Ожидаемый практический результат - разработка системы, улучшающей процесс диагностики и мониторинга заболеваний глазного дна, что может повысить качество жизни пациентов и оптимизировать затраты на лечение.

Логический процесс включал использование различных библиотек, в том числе Elsevier, eLibrary, Papers with Code и других. Анализировались публикации за период с 2019 по 2024 год, соответствующие критериям качества индексации в РИНЦ, ВАК и Scopus. Запрос на поиск литературы был тщательно составлен для выявления релевантных публикаций по теме сегментации кровеносных сосудов сетчатки на изображениях глазного дна с акцентом на применение методологий машинного обучения, нейронных сетей и глубокого обучения. Точные формулировки поисковых запросов для библиотек eLibrary и Elsevier представлены в таблице 2.

Таблица 2 – Формулировка поисковых запросов к научным библиотекам

| Библиотека    | Формулировка запроса                                                                                                                                                                                                |
|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| eLibrary [77] | Сегментация изображений кровеносных сосудов глазного дна                                                                                                                                                            |
| Elsevier [78] | ((retinal AND blood AND vessel) AND (classification OR segmentation) AND (machine OR learning OR neural OR network OR deep OR learning) AND (methods OR algorithms OR analysis OR review OR application OR system)) |

Процесс анализа публикаций для литературного обзора был разделен на 5 шагов. На каждом из шагов, кроме первого исключались неподходящие под критерии отбора публикации. Количество публикаций на каждом из шагов приведено в таблице 3.

Первый шаг анализа публикаций заключался в формировании правильного поискового запроса. Запросы были сформулированы исходя из основной темы исследования и были приведены в таблице 2.

На втором шаге анализа происходила проверка возможности получения полного текста научной публикации. Для библиотек eLibrary и Elsevier процесс анализа проходил по-разному. На портале eLibrary не удалось найти автоматическую фильтрацию публикаций, находящихся в полном доступе. Для оптимизации был разработан скрипт для поиска полного текста публикации [79] й. В результате работы скрипта был получен файл формата .csv, содержащий все статьи из поискового запроса, полный текст которых полностью открыт или доступен на сайте издателя. В итоге удалось получить 75 научных работ. Получение полнотекстовых работ из библиотеки Elsevier устроено более просто, благодаря встроенному функционалу на выгрузку необходимых работ. В системе SciVal присутствует 4 уровня публикаций в открытом доступе:

1. Gold - опубликованная версия с лицензией Creative Commons, доступная на платформе издателя. Документы находятся в журналах, которые публикуются только в открытом доступе.
2. Hybrid Gold - опубликованная версия с лицензией Creative Commons, доступная на платформе издательства. Документы публикуются в журналах, которые предоставляют авторам выбор публикации в открытом доступе.
3. Bronze - опубликованная версия записи или труда, принятого к публикации, для которой издатель решил предоставить временный или постоянный свободный доступ. Бронзовый статус присваивается документу при наличии другой (специфической для издательства) лицензии, отличной от лицензии Creative Commons (например, издательской лицензии Elsevier для Open Archive), или при отсутствии лицензии вообще.
4. Green - опубликованная версия или рукопись, принятая к публикации, доступна в репозитории. Документы также могут быть доступны в золотом или ином виде для свободного чтения на платформе издательства.

В выборку были включены публикации с любым уровнем открытого доступа. После применения фильтров и выгрузки данных с портала Elsevier осталось 463 публикации для дальнейшего анализа.

Третий шаг анализа представляет собой анализ метаданных и заголовков статей. На этом шаге можно исключить большинство статей, которые не подходят под критерии отбора. В результате анализа заголовков и метаданных число научных работ, подходящих для литературного обзора, сократилось с 75 до 42 и с 463 до 126 для статей с порталов eLibrary и Elsevier соответственно.

За счет обзора аннотаций к научным работам удалось сократить выборку с 42 до 21 и с 126 до 65 работ с порталов eLibrary и Elsevier соответственно. В рамках анализа аннотаций исключались статьи, в которых не содержится описание архитектур применяемых для сегментации моделей, а также статей, в

которых отсутствует описание современных методов по предобработке и анализу изображений глазного дна.

На последнем шаге анализировались результаты исследований, за счет тщательного обзора результатов удалось незначительно сузить выборку научных трудов. Для eLibrary выборка сократилась с 21 до 15, а для Elsevier с 65 до 45.

Таблица 3 – Результаты отбора научной литературы

|       | Результаты eLibrary [77] | Результаты Elsevier [78] |
|-------|--------------------------|--------------------------|
| Шаг 1 | Найдено: 390             | Найдено: 1,164           |
| Шаг 2 | Осталось: 75             | Осталось: 463            |
| Шаг 3 | Осталось: 42             | Осталось: 126            |
| Шаг 4 | Осталось: 21             | Осталось: 65             |
| Шаг 5 | Осталось: 15             | Осталось: 45             |

После работы по отбору публикаций были определены основные ценности для извлечения. Исходя из основной цели исследования было решено извлечь из отобранных научных работ следующее:

- 1) методы и алгоритмы сегментации кровеносных сосудов сетчатки в изображениях глазного дна;
- 2) визуализированные схемы архитектур нейронных сетей, которые применяются для сегментации кровеносных сосудов сетчатки в изображениях глазного дна.

В заключении обзора научной литературы была поставлена задача синтезировать извлеченные данные, систематизировать их и найти наиболее эффективные методы сегментации с возможностями их совершенствования.

Помимо научной литературы в работе были проанализирована патентная документация. Для анализа были выбраны такие патентные базы как Google Патент [80] и РосПатент [81]. Для эффективного патентного поиска следует определить к какой области относится проводимая работа. Проанализировав

международную патентную классификацию, был сделан вывод, что разработку можно отнести к следующим классификациям:

- 1) A61B 3/18 устройства для испытания остроты зрения, приборы для исследования глаз, устройство в виде нескольких приборов для испытания или исследования глаза ;
- 2) G06F 18/10 предварительная обработка, очистка данных;
- 3) G06F 18/213 извлечение признаков, например, путем преобразования пространства признаков, подведение итогов отображения, например, подпространственные методы;
- 4) G06F 18/2413 методы классификации на основе расстояний до обучающих или эталонных шаблонов;
- 5) G06F 18/40 программные средства, специально адаптированные для распознавания образов, например пользовательские интерфейсы или наборы инструментов для них;
- 6) G06T 7/00 анализ изображений;
- 7) G06T 7/10 сегментация, обнаружение краев.

После определения классификаций необходимо определить основные ценности для извлечения. Исходя из цели исследования из отобранных патентов было извлечено следующее:

- 1) методы и алгоритмы сегментации кровеносных сосудов сетчатки в изображениях глазного дна;
- 2) визуализированные схемы архитектур нейронных сетей, которые применяются для сегментации кровеносных сосудов сетчатки в изображениях глазного дна.

Следующий этап патентного поиска – формирование поискового запроса, представлен в таблице 4.

Таблица 4 – Формулировка поисковых запросов к патентным библиотекам

| Название библиотеки | Формулировка запроса                                                                                                                                                                                                                                                                                                  |
|---------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Роспатент [81]      | ((система OR метод OR алгоритм) AND (сегментации OR классификации) (кровеносных сосудов OR сосудов OR глазных сосудов) AND (сетчатки глазного дна OR глазного дна OR изображений OR глаза OR сетчатки) AND (машинное обучение OR машинного обучения OR нейронных сетей OR нейронные сети OR искусственный интеллект)) |
| Google Патент [80]  | (retinal blood) (vessel OR vessels) (classification OR segmentation) (machine learning OR neural network OR deep learning) (methods OR algorithms OR analysis OR review OR application OR system) 2018-01-01 - 2024-01-01                                                                                             |

После формулировки запроса было оценено количество найденных публикаций, проведена проверка возможности обращения к полному тексту патента и анализ информации, предоставленной в патентном документе. В таблице 5 представлены количественные показатели поиска, выполненного по следующим шагам:

1. Поиск патентов по сформулированному ранее поисковому запросу в Google Патент и Роспатент.
2. Проверка полнотекстовой доступности. Обращение к полному тексту документа, необходимо отобразить только доступные для нас патенты.
3. Анализ документации и извлечение данных. Проверка текста каждого патента на соответствие стандарту извлечения данных, сформулированному ранее.

В рамках патентного поиска на российском портале Роспатент удалось найти всего 15 работ. Все документы были предоставлены в полном тексте, однако большинство из них не соответствовало целям извлечения данных. Все патенты кроме одного оказались отличными от тематики работы. Однако единственный подходящий патент всё еще находится лишь в состоянии заявки.

Поиск по Google Патентам дал намного больше результатов. Изначально были рассмотрены все патентные офисы, но результаты поиска превысили 10000, поэтому был принято решение сузить поисковый запрос только до RU офиса. В результате были получены 62 документа, из которых лишь 9 оказались релевантными в контексте нашей работы.

Таблица 5 - Описание процесса анализа патентной документации

|       | Google Патенты [80] | Роспатент [81] |
|-------|---------------------|----------------|
| Шаг 1 | Найдено: 62         | Найдено: 15    |
| Шаг 2 | Осталось: 62        | Осталось: 15   |
| Шаг 3 | Осталось: 9         | Осталось: 1    |

Библиографические показатели заданной области исследования представлены на рисунке 1. За последние 5 лет прослеживается положительная тенденция в отношении количества публикаций, число научных работ в данной области увеличилось с 133 до 338 работ в год. Показатель цитирования не высок и составляет всего 0.25, это может говорить о том, что исследования в этой области довольно узко специализированы и многие авторы представляют качественно новые подходы, которые ранее не были достаточно исследованы. Также стоит отметить довольно высокий показатель международных коллабораций в публикациях внутри этой области, можно предположить, что тема актуальна для многих стран и ученые открыты к взаимодействию по развитию данного направления.

**1,164**  
Scholarly Output ⓘ

**0.25**  
Field-Weighted Citation Impact ⓘ

**269**  
International Collaboration ⓘ

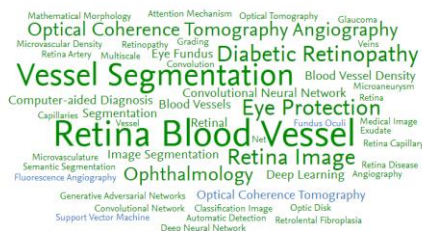
**19,807**  
Views Count ⓘ

**22,203**  
Citation Count ⓘ

Рисунок 1 - Библиометрические показатели области исследования [78]

+ Add to Reporting

Top 50 keywords by relevance, based on 1,164 publications



A A A relevance of keyphrase | declining A A A growing (2018-2022)

Рисунок 2 - Облако ключевых фраз области исследования [78]

### 1.3 Обзор литературных источников. Современные методы сегментации кровеносных сосудов сетчатки в изображениях глазного дна

Для извлечения информации из снимков кровеносных сосудов глазного дна применяют различные методы обработки изображений. Е.А. Федотов, Д.А. Черноморец и В.М. Михеев выделяют основные подходы в задачах выделения сосудов на изображениях глазного как сегментацию сосудов, позволяющую

выделить всё дерево сосудов в одну итерацию, и прослеживание сосудов (трассировка) [20]. Сегментация изображений может использоваться для последующего извлечения информации о сосудах.

Методы, основанные на кластеризации пикселей. Наиболее популярный в наше время метод представляет собой процесс разбиения  $N$ -мерной совокупности на  $k$  множеств на основе выборки. Этот процесс называется "к-средние", позволяет получить разделения, которые являются достаточно эффективными в смысле внутриклассовой дисперсии [74].

Основные шаги процедуры сегментации сосудов глазного дна можно сформулировать следующим образом:

- 1) выполнить предварительную обработку изображения с целью удаления шумов;
- 2) осуществить кластеризацию множества пикселей обработанного изображения на классы;
- 3) выполнить сегментацию толстых сосудов на изображении глазного дна (сформировать бинарное изображение, используя пиксели старшего класса);
- 4) сформировать изображение, содержащее пиксели класса;
- 5) выполнить многомасштабную обработку, позволяющую выделить тонкие сосуды (бинарное изображение) на изображении;
- 6) добавить результат сегментации тонких сосудов к изображению, которое является результатом сегментации толстых сосудов.

В обзоре современных методов сегментации медицинских изображений К.К.Д. Рамеш и др., выделяет подвиды метода K-means [54]. Основной алгоритм кластеризации, K-means, относит каждую точку данных только к одному кластеру. В отличие от основного, мягкая кластеризация, например Fuzzy C-Means (FCM), представленный в работе М. Ахмеда и др., [4] позволяет точкам данных принадлежать к нескольким кластерам с нечетким значением

принадлежности между 0 и 1. FCM стремится минимизировать квадратичное евклидово расстояние между точками данных и центрами кластеров, известное как объективная функция. Этот алгоритм универсален и используется для разбиения как серых, так и цветных изображений, предварительно определив количество кластеров. FCM можно адаптировать к различным типам изображений, изменяя его объективную функцию.

Для повышения эффективности FCM были разработаны такие варианты, как Kernelized fuzzy C-means (KFCM) и Fast generalized fuzzy C-means (FGFCM) [12, 71]. Эти адаптации включают в себя пространственные и серые данные для улучшения подавления шума и сохранения деталей на изображениях. Тестирование этих алгоритмов на КТ-изображениях мозга и изображениях бактерий показало их эффективность в сегментации аномальных областей и отделении объектов от фона. Примечательно, что алгоритм Type-II Fuzzy C-Means (T2FCM) преуспел в удалении шума, но увеличил размер объектов, а Intuitionist Fuzzy C-Means (IFCM) продемонстрировал эффективность в сегментации изображений по сравнению с другими методами [7, 16].

Следующий метод, который можно выделить – это метод сегментации водоразделом. Водораздельный подход в обработке изображений предполагает рассмотрение изображения как топографической поверхности, где низкая интенсивность пикселей представляет долины, а высокая – пики [39]. Этот метод начинается с заливки долин из локальных минимумов, рассматривая их как источники воды, которые затапливают окружающие территории. Когда встречаются "цветные воды", происходящие из разных источников, для предотвращения их слияния очерчиваются линии водоразделов. Процесс затопления продолжается до тех пор, пока не будут затоплены все вершины, определяя границы сегментации [29].

Сегментация водоразделом объединяет информацию о градиенте и интенсивности пикселей, чтобы определить водосборные бассейны как регионы,

где вода скапливается вокруг локальных минимумов. Для сегментации изображений водоразделов используются два основных алгоритма - дождевой и паводковый. В то время как стандартные методы водоразделов могут привести к чрезмерной сегментации из-за шума, такие методы, как мощные водоразделы и классификация с помощью нейронных сетей без контроля, повышают точность.

В медицинских приложениях, таких как обнаружение опухолей на ультразвуковых или магнитно-резонансных изображениях, вариации метода водоразделов успешно повышают точность сегментации и автоматизируют процессы. При рассмотрении данного алгоритма в контексте сегментации кровеносных сосудов мы рассматриваем изображение как топографическую поверхность, математическая морфология усиливает контраст на малоконтрастных цветных изображениях сетчатки, отделяя сосудистое дерево от фона. Это позволяет автоматически и точно анализировать диагностически значимые признаки на изображениях сетчатки.

Эффективность применения водораздельных алгоритмов для сегментации кровеносных сосудов сетчатки подтверждается научной литературой. К. Кумар и Н. Сингх в своей работе, "Анализ методов сегментации кровеносных сосудов сетчатки: систематический обзор", представляет всеобъемлющий обзор методик сегментации сосудов сетчатки, предлагающий ключевые идеи [40]. Дж. Лиу и др., описывают алгоритм биометрической идентификации с использованием водораздельной сегментации сосудов сетчатки [41]. Эта работа демонстрирует практическое применение метода водоразделов для сегментации кровеносных сосудов сетчатки. В совокупности эти работы подчеркивают ценность алгоритма водораздела для сегментации сосудов сетчатки, что способствует развитию анализа и диагностики медицинских изображений.

Методы пороговой обработки изображений также могут быть применимы в задачах сегментации. Метод Оцу - один из наиболее популярных методов, используемый для автоматической пороговой обработки изображений, относится

к большой группе методов пороговой обработки [50]. Был разработан в 1979 году, используется в области компьютерного распознавания и обработки изображений для получения чёрно-белых изображений. Суть метода заключается в делении пикселей на 2 класса «полезные» и «фоновые».

В работе К. Кхан и др., используется улучшенная версия данного метода для выделения кровеносных сосудов сетчатки глазного дна [37]. Авторы применяют данный метод для классификации пикселей с сосудами и без сосудов на обоих улучшенных изображениях. Пороговый метод Оцу в статье используется для классификации пикселей с сосудами и пикселей без сосудов на улучшенных изображениях, которые являются результатом предварительной обработки изображений из баз данных DRIVE, STARE и HRF, с наложенными на них морфологическими фильтрами. Такой подход позволяет более точно классифицировать кровеносные сосуды, что очень важно для выявления заболеваний глаз. Помимо метода Оцу, к методам пороговой обработки можно отнести метод Киттлера-Иллингворта [38], алгоритмы на основе энтропии [34] и др.

Методы с выделением краёв являются одними из первых методов выделения извлечения признаков на изображениях. В работе С.Ц. Жу и др., описан метод полученный с помощью минимизации обобщенного критерия Байеса/MDL, который включает в себя окно выборки (размер которого зависит от соотношения сигнал/шум) [75]. Алгоритм объединяет привлекательные геометрические особенности моделей и статистические методы. Данный подход напрямую обобщается на многополосную сегментацию. Сначала происходит распространение его на цвет, используя новую модель, которая для определенных типов материалов позволяет сегментировать изображение на основе альбедо материала. Это позволяет избежать основного недостатка многих подходов к сегментации изображений, которые ищут только разрывы в интенсивности, что может дать неверные результаты. Например, одежда обычно

имеет области с однородным альбедо или текстурой, но часто имеет резкие тени из-за складок. Одним из новых методов, основанных на выделении краев является метод RBVSLE, представленный в работе Л. Сюй и С. Ло [67]. RBVSLE был разработан для решения проблемы разного контраста между крупными и тонкими кровеносными сосудами на изображениях сетчатки. Этот метод использует локальное адаптивное пороговое выделение для создания бинарного изображения, а затем выделяет крупные связные компоненты, такие как крупные сосуды и тонкие сосуды по отдельности. RBVSLE состоит из четырех основных этапов:

1. Предварительная обработка. Изображение сетчатки подвергается предварительной обработке, чтобы повысить его качество и подготовить к сегментации сосудов. Это может включать в себя подавление шума, повышение контрастности и нормализацию.
2. Выделение фрагментов сосудов. Адаптивный локальный порог применяется к предварительно обработанному изображению для создания бинарного изображения. Этот шаг помогает отделить сосуды от фона, устанавливая соответствующие пороговые значения на основе локальных характеристик изображения.
3. Классификация фрагментов с помощью метода опорных векторов. Классификатор метода опорных векторов используется для деления выделенных фрагментов сосудов на категории, такие как крупные сосуды и тонкие сосуды, на основе признаков, полученных на предыдущем этапе.
4. Выделения тонких сосудов путем отслеживания. После классификации для уточнения сегментации тонких сосудов используется процесс роста тонких сосудов, основанный на методе отслеживания. Этот шаг помогает соединить фрагментированные тонкие сосуды и повысить общую точность обнаружения сосудов.

Интегрируя эти этапы, метод RBVSLE стремится решить проблемы, возникающие из-за контрастных различий между различными кровеносными сосудами на изображениях сетчатки, с целью достижения более точных и надежных результатов сегментации.

С. Чаттерджи и др., в своем исследовании сравнили различные методы с выделением краев для выбора лучшего метода в контексте сегментации кровеносных сосудов в изображениях глазного дна [14]. Авторы применяют 5 методов среди которых фильтрации Sobel, Prewitt, Kirch, Fuzzy C mean, Canny на изображениях из базы данных DRIVE. Благодаря сравнению различных методов удалось выяснить, что Фильтр Кирша (Kirch filter) является наиболее эффективным методом выделения краёв в изображениях глазного дна. Благодаря применению данного метода удалось добиться показателей Accuracy, Specificity и Sensitivity на уровнях 94%, 100%, и 52% соответственно. Однако метод Fuzzy C mean показал себя лучше других в рамках показателя Sensitivity с результатом в 84%.

#### **1.4 Обзор основных архитектур нейронных сетей для сегментации сосудов сетчатки**

М. Э. Хок и К. Кипли в своей работе сравнили различные подходы к сегментации кровеносных сосудов сетчатки в изображениях глазного дна и пришли к выводу, что наиболее эффективными являются работы с применением свёрточных нейронных сетей, однако применение таких технологий наиболее сложно в техническом плане из-за сложности устройства моделей [26]. За последние 5 лет было проведено множество различных исследований по теме сегментации кровеносных сосудов сетчатки с использованием свёрточных нейронных сетей, результаты работ приведены в таблице 1 из приложения А.

В работе Л. Пан и др., была представлена модель MSC-Net [51]. MSC-Net использует подход мультитаскового обучения, который позволяет ему

одновременно изучать несколько задач (сегментацию сосудов и извлечение центральных линий). MSC-Net продемонстрировала высокую точность как в задачах сегментации сосудов, так и в извлечении центральных линий, что делает ее надежным инструментом для медицинских специалистов. Модель может быть адаптирована к различным модальностям изображений и наборам данных, что делает ее универсальной и применимой в различных клинических условиях. Как и многие глубокие модели обучения, производительность MSC-Net сильно зависит от качества и количества обучающих данных. Недостаточные или предвзятые обучающие данные могут привести к неоптимальным результатам.

Модель Anam-Net, авторами которой являются К. Аурангзеб и др., представляет собой инновационный подход к сегментации сосудов сетчатки, основанный на глубоком обучении [9]. Anam-Net может обрабатывать большие объемы данных, что делает ее идеальным инструментом для автоматизированных систем диагностики. Глубокие модели, такие как Anam-Net, могут быть сложными в настройке и использовании, требуя высоких вычислительных ресурсов. А интерпретация решений, принятых моделью, может быть сложной из-за ее сложной архитектуры и методов обучения.

Дж. Вэй и др., предлагают новый автоматизированный метод проектирования, называемый Генетическим U-Net, для создания U-образной сверточной нейронной сети, которая может достигать лучшей сегментации сосудов сетчатки, но с меньшим количеством параметров, основанных на архитектуре [65]. Экспериментальные результаты показывают, что архитектура, полученная с использованием предложенного метода, обеспечивает превосходную производительность с менее чем 1% от количества параметров оригинальной U-Net, в частности, и со значительно меньшим количеством параметров, чем у других передовых моделей. Более того, путем глубокого анализа экспериментальных результатов были выявлены несколько

эффективных операций и шаблонов сетей для генерации превосходной сегментации сосудов сетчатки.

Модель авторов Р. Лиу и др., под названием LWS-Net проводила многочисленные эксперименты на наборе данных BraTS2020, которые показывают, что модель LSW-Net улучшила коэффициент Дайса, точность и чувствительность классических моделей FCN, SegNet и At-Unet как минимум на 3,51%, 2,11% и 0,46% соответственно [42]. Кроме того, LSW-Net имеет преимущество в среднем значении коэффициентов Дайса по сравнению с некоторыми передовыми моделями сегментации.

Ю. Сюй и др., представили новую сетевую модель под названием Bi-directional ConvLSTM Residual U-Net (BCR-UNet), которая полностью использует U-Net, Dropblock, Residual convolution и Bi-directional ConvLSTM (BConvLSTM) [68]. После проведения экспериментов на четырех общедоступных наборах данных по сосудам сетчатки, результаты показывают, что предложенная модель BCR-UNet может сохранять больше мелких сосудов в областях низкого контраста на периферии, даже превосходя предыдущие передовые методы.

С. А. Камран и др., предложили RV-GAN - новую многошкальную генеративную архитектуру для точной сегментации сосудов сетчатки [35]. Предложенная архитектура использует два генератора и два многошкальных автоэнкодирующих дискриминатора для лучшей локализации и сегментации микрососудов. Путем комбинирования потерь восстановления и взвешенного сопоставления признаков предложенная архитектура достигает площади под кривой (AUC) 0.9887, 0.9914 и 0.9887 в пиксельной сегментации сосудов сетчатки из трех общедоступных наборов данных, а именно DRIVE, CHASE-DB1 и STARE, соответственно. Кроме того, RV-GAN превосходит другие архитектуры по двум дополнительным релевантным метрикам, среднему пересечению-по-объединению (Mean-IOU) и мере структурной схожести (SSIM).

Авторами Дж. Жанг и др., была предложена Pyramid-Net для точной сегментации сосудов сетчатки, которая включает в себя блоки агрегации пирамидального масштаба внутри слоя (IPABs) [73]. На каждом уровне IPABs генерируют две связанные ветви на более высоком и более низком масштабах, соответственно, и две с основной ветвью на текущем масштабе работают в пирамидальном масштабе. Предлагаются три дополнительных улучшения, включая улучшение пирамидальных входов, глубокое пирамидальное наблюдение и пирамидальные пропускающие соединения, чтобы улучшить производительность. Экспериментальные результаты показывают, что Pyramid-Net может эффективно улучшить производительность сегментации, особенно на тонких сосудах, и превосходит текущие передовые методы на всех трех принятых наборах данных.

С. Мишра и др., разработали новую архитектуру на основе CNN, VTG-Net (сеть графовой топологии сосудов), для классификации артерий и вен на сетчатке путем внедрения информации о топологии сосудов [47]. VTG-Net использует топологию сосудов сетчатки вместе с признаками CNN для улучшения точности классификации артерий и вен. Используя общедоступный набор данных AV-DRIVE и внутренний набор данных, подтверждается высокая производительность и результативность сети VTG-Net для классификации артерий и вен на сетчатке по сравнению с передовыми методами (с приблизительным улучшением точности на наборе данных AV-DRIVE на ~2%).

С. А. Рамми и др., предложили технику на основе генеративно-сопоставительных сетей (GAN), основанная на патчах, которая итеративно обучает как толстые, так и тонкие сосуды на фундусных изображениях [55]. Она вводит дополнительную функцию потерь, которая позволяет сети-генератору учиться тонким и толстым сосудам, в то время как сеть-дискриминатор помогает сегментировать оба сосуда как объединенную целевую функцию. По сравнению с передовыми техниками, предложенная модель демонстрирует улучшенную

точность, чувствительность, специфичность и площадь под кривой ROC на наборах данных STARE, DRIVE и CHASEDB1.

Для повышения точности и чувствительности существующих методов сегментации сосудов К. Фу и др., предложили многошкальную сверточную нейронную сеть с механизмами внимания (MSCNN-AM) [21]. Для извлечения сосудов разных масштабов мы вводим атрофические отдельные свертки с различными коэффициентами растяжения, которые могут лучше захватывать глобальную и многошкальную информацию о сосудах. В то же время, чтобы снизить количество ложноположительных прогнозов для маленьких пикселей сосудов, используются механизмы внимания, чтобы предложенная MSCNN-AM могла уделять больше внимания пикселям сосудов сетчатки, а не фоновым пикселям. Экспериментальные результаты показывают, что предложенный метод превосходит большинство существующих методов с чувствительностью 0.8342/0.8412/0.8132 и точностью 0.9555/0.9658/0.9644 на наборах данных DRIVE, STARE и CHASE\_DB1 соответственно.

Х. Динг и др., предложили улучшенный метод сегментации сосудов сетчатки с использованием U-образной нейронной сети (MRU-NET) [19]. Во-первых, для решения проблем низкого контраста и недостаточного объема данных изображения используются алгоритм улучшения изображения и случайный метод сегментации. Более того, меньшие блоки изображений после случайной сегментации помогают уменьшить сложность модели U-образной нейронной сети; во-вторых, введено остаточное обучение в кодер и декодер для улучшения эффективности использования признаков и снижения потерь информации, а также введен модуль слияния признаков между кодером и декодером для извлечения признаков изображения с разной гранулярностью; и, наконец, добавлен модуль балансировки признаков к пропускающим соединениям для разрешения семантического разрыва между признаками низкой размерности в кодере и признаками высокой размерности в декодере.

К. Б. Пак и др., предлагают новую условную генеративно-сопоставительную сеть под названием M-GAN для точной и точной сегментации сосудов сетчатки путем балансировки потерь через стековые глубокие полностью сверточные сети [52]. Она состоит из нового конструктора M с глубокими остаточными блоками для более надежной сегментации и дискриминатора M с более глубокой сетью для более эффективного обучения адверсарной модели. Для проверки предложенного метода мы использовали наборы данных DRIVE, STARE, HRF и CHASE-DB1 и сравнили предложенный M-GAN с другими исследованиями. Измерены точность, пересечение объединения (IoU), F1-оценку и коэффициент корреляции Мэттьюса (MCC) для сравнительного анализа. Результаты сравнения доказали, что предложенный M-GAN обеспечивает более высокую производительность, чем другие исследования.

Алгоритм от Ж. Луо и др., вводит механизм внимания и плотно связанную сеть в оригинальную сеть U-Net и реализует автоматическую сегментацию сосудов сетчатки [45]. Согласно результатам тестирования алгоритма на широко используемых наборах данных DRIVE и STARE фундусных изображений, соответственно, точность составляет 0.9663 и 0.9684; чувствительность - 0.8075 и 0.8437; специфичность - 0.9814 и 0.9762; значения AUC - 0.9846 и 0.9765; и F-меры - 0.8203 и 0.8419, соответственно. В статье алгоритм Attention-Dense-UNet (AD-UNet) применяется для сегментации микрососудов конъюнктивы глаза человека. Экспериментальные результаты показывают, что алгоритм может достигать идеальных результатов сегментации.

Ж. Гу и др., предложили сеть контекстного кодировщика (CE-Net) для захвата более высокоуровневой информации и сохранения пространственной информации для сегментации 2D медицинских изображений [23]. CE-Net в основном содержит три основных компонента: модуль кодировщика признаков, извлекатель контекста и модуль декодировщика признаков. В ней используется предварительно обученный блок ResNet в качестве фиксированного извлекателя

признаков. Модуль извлекателя контекста формируется с помощью нового предложенного блока плотной атрофической свертки и блока остаточного многоканального пулинга. SE-Net была предложена к различным задачам сегментации 2D медицинских изображений. Обширные результаты показывают, что предложенный метод превосходит исходный метод U-Net и другие передовые методы для сегментации диска зрительного нерва, обнаружения сосудов, сегментации легких, сегментации контура клеток и сегментации слоев ретиальной оптической когерентной томографии.

### **1.5 Постановка задачи управления организационной системы**

Целью данной работы является совершенствование модели глубокой нейронной сети (Deep Neural Network), которая может выполнять надежную сегментацию кровеносных сосудов сетчатки для медицинских изображений, полученных с глазного дна сетчатки. Организационная система, выступающая объектом управления - медицинское учреждение, в котором есть необходимость в анализе результатов снимков глазного дна сетчатки для диагностики и профилактики заболеваний, определяемых по результатам анализа кровеносных сосудов сетчатки. Объектом управления может считаться любое медучреждение, занимающееся офтальмологией и имеющее запрос на автоматизацию процесса сегментации кровеносных сосудов сетчатки.

Внедрение автоматической системы сегментации кровеносных сосудов сетчатки на медицинских снимках глазного дна позволить медучреждению более оперативно анализировать снимки и диагностировать заболевания. Среди параметров, которые будут улучшены в процессе работы:

- 1) снижение времени принятия решений (постановка диагноза);
- 2) увеличение точности анализа.

Разрабатываемая инновация относится к преобразованию информационного ресурса. Для реализуемой инновации применима модель

«чёрного ящика», со схематическим представлением данной модели можно ознакомиться на рисунке 3. Модель состоит из следующих компонентов:

1. Выходные данные: автоматически отфильтрованные, сгенерированные и сегментированные, при помощи модели искусственного интеллекта, изображения кровеносных сосудов сетчатки в изображениях глазного дна.
2. Внешняя среда: техническая документация, системные требования для работы нейронной сети.
3. Входящий поток из внутренней среды: поток информации – данные о пациенте, изображения кровеносных сосудов сетчатки глазного дна.
4. Обратная связь: отзывы, реакции, жалобы пользователей на результаты работы автоматической сегментации, оценка точности полученных изображений. Обучение и адаптация: процесс обучения персонала по использованию системы сегментации изображений глазного дна.
5. Процесс: Сбор необходимых медицинских изображений глазного дна. Обработка изображений при помощи различных методов и фильтров, редактирование размера изображений для подстановки в модель. Пользователь загружает изображение в определенную папку, где хранятся изображения глазного дна, далее изображение из этой папки подается в обученную нейронную сеть, обрабатывается и на выходе предоставляется сгенерированное нейронной сетью изображение с выделенными кровеносными сосудами, по которому можно оценить состояние глазного дна пациента и диагностировать имеющиеся заболевания.



Рисунок 3 - Модель «чёрного ящика»

Надсистемой процесса является медицинское учреждение, будучи объектом, заинтересованным в автоматизации процессов, требующих больших временных затрат. Автоматическая сегментация кровеносных сосудов сетчатки в изображениях глазного дна позволит эффективно и быстро обрабатывать изображения, и диагностировать заболевания. Схема организационной структуры надсистемы представлена на рисунке 4.

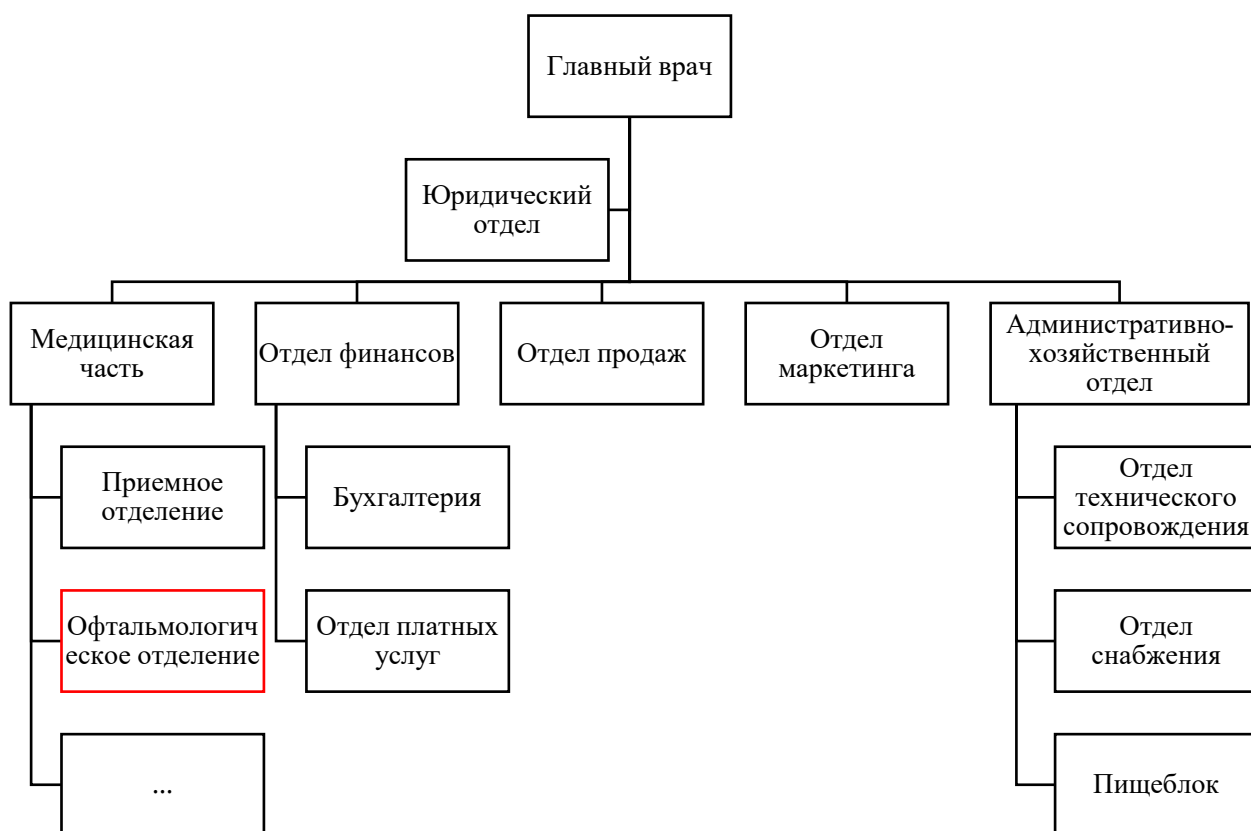


Рисунок 4 - Организационная структура надсистемы

Основываясь на модели «чёрного ящика», можно выделить подсистему, к которой относится наша инновация, ей является отдел офтальмологии медицинского учреждения. Офтальмолог получает информацию о состоянии кровеносных сосудов глазного дна пациента после анализа фотографии, сделанной при помощи специальной фундус-камеры. Схематично организационная структура подсистемы представлена на рисунке 5.

Среди основных участников подсистемы были выделены: заведующий отделением, лечащий врач, средний медицинский персонал и младший медицинский персонал. Для более точного понимания процесса внедрения инновации в подсистему следует рассмотреть роли каждого участника подсистемы:

1. Заведующий отделением отвечает за общее руководство и координацию работы офтальмологического отделения. Он занимается планированием

деятельности, распределением задач между сотрудниками, контролем качества работы и решением административных вопросов.

2. Лечащий врач является основным медицинским специалистом, который проводит диагностику и лечение заболеваний глаз. Он использует инновационную глубокую нейронную сеть для анализа изображений сетчатки и определяет состояние кровеносных сосудов.
3. Средний медицинский персонал включает медсестер, которые помогают лечащему врачу в проведении медицинских процедур и обследований. Они также могут использовать инновационную технологию для обработки изображений и анализа результатов.
4. Младший медицинский персонал, такой как санитары и помощники медсестер, выполняют задачи, связанные с уборкой, транспортировкой пациентов и обеспечением доступа к оборудованию. Они также могут быть обучены использованию инновационной технологии для выполнения простых задач, таких как обработка изображений или ввод данных.

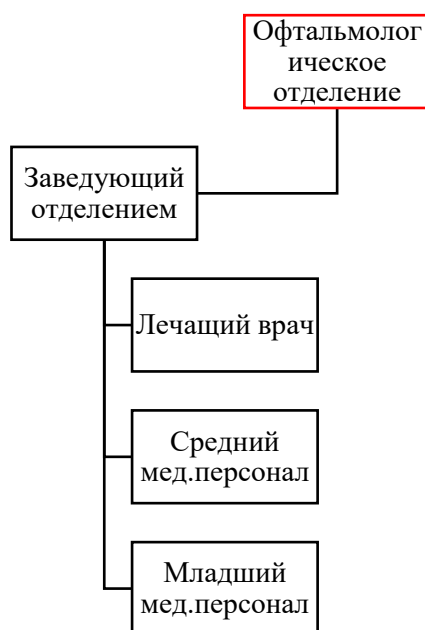


Рисунок 5 - Организационная структура подсистемы

Принцип работы инновации описан в виде концептуальной модели и представлен в таблице 6.

Таблица 6 – Описание концептуальной модели инновации

| Название                 | Описание                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|--------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Основная функция системы | Основной функцией инновации является повышение точности и эффективности сегментации кровеносных сосудов сетчатки на медицинских изображениях глазного дна. Модель глубокой нейронной сети нацелена на достижение количественных показателей accuracy 98%, precision 97%, recall 97 % и F-score 96 %. Такое улучшение сегментации будет способствовать более надежному и быстрому анализу изображений глазного дна сетчатки, что поможет в диагностике и профилактике заболеваний, связанных с кровеносными сосудами сетчатки.                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
| Путь реализации системы  | <p>Сбор и предварительная обработка данных. Сбор данных изображений сетчатки с аннотациями, полученными на основе реальных данных. Предварительная обработка данных для улучшения характеристик и уменьшения шума на изображениях.</p> <p>Разработка модели. Разработка архитектуры глубокой нейронной сети, предназначенной для сегментации кровеносных сосудов сетчатки. Обучение модели на подготовленном наборе данных с упором на достижение заданных показателей эффективности.</p> <p>Проверка и оптимизация. Проверка модели на независимом наборе данных, чтобы убедиться в ее обобщенности и надежности. Оптимизация гиперпараметров и тонкая настройка модели для повышения показателей эффективности.</p> <p>Интеграция в систему медучреждения. Разработка интерфейса для беспрепятственной интеграции с существующими медицинскими системами в офтальмологических отделениях. Обеспечение совместимости с широко используемыми устройствами визуализации и форматами данных.</p> |

Продолжение таблицы 6

| Название                                | Описание                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|-----------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Структура системы                       | <p>Надсистема - Медицинская клиника. Отвечает за общий уход за пациентами и диагностику. Интегрирует подсистему сегментации кровеносных сосудов сетчатки в диагностический рабочий процесс. Подробная организационная структура надсистемы представлена на рисунке 7.</p> <p>Подсистема - Офтальмологическое отделение. Занимается проблемами здоровья глаз, в том числе обследованием сетчатки. Внедряет инновационную глубокую нейронную сеть для автоматизированной сегментации кровеносных сосудов сетчатки. Взаимодействует с основной системой для беспрепятственной интеграции и передачи информации. Структура подсистемы в виде схемы изображена на рисунке 8.</p> |
| Направленность функционирования системы | Информационная. Инновация направлена на улучшение и автоматизацию процессов, связанных со сбором, обработкой, анализом и обменом данных.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
| Результат функционирования              | Автоматизация процесса сегментации кровеносных сосудов сетчатки в медицинских изображениях глазного дна пациента.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |

Надсистема (Медицинская клиника) и подсистема (Офтальмологическое отделение) взаимодействуют через передачу информации о состоянии кровеносных сосудов на сетчатке.

В рамках подсистемы инновационная глубокая нейронная сеть используется для автоматизации процесса анализа изображений сетчатки и определения состояния кровеносных сосудов. Полученная информация затем передается в надсистему для дальнейшей обработки и принятия решений о лечении пациентов.

Состояние системы в текущий момент выглядит следующим образом:

$$S_0 = \{T_0, A_0\}, \quad (1)$$

где  $T_0$  и  $A_0$  – это соответственно начальные значения параметров времени обработки и точности анализа, требующие модернизации;

Для решения задачи управления система должна прийти к конечному состоянию, описываемому как

$$S_{new} = \{minT_{new}, maxA_{new}\}, \quad (2)$$

где  $T_{new}$  и  $A_{new}$  – это соответственно конечные значения параметров времени обработки и точности анализа системы.

Для отслеживания достижения целей, связанных с ускорением процессов отдела аналитики и уменьшением времени принятия бизнес-решений будут использоваться следующие параметры:

1. Время обработки данных: Инновационная глубокая нейронная сеть может значительно сократить время обработки изображений сетчатки, что позволит быстрее получать результаты анализа и принимать решения о лечении. Это можно выразить формулой:

$$T_{new} = T_{old} - \Delta T, \quad (3)$$

где  $T_{new}$  - новое время обработки;

$T_{old}$  - старое время обработки;

$\Delta T$  - сокращение времени обработки.

2. Точность анализа: Глубокие нейронные сети способны помочь более точно анализировать изображения сетчатки и определять состояние кровеносных сосудов, что может повысить качество диагностики и лечения. Можно выразить формулой:

$$A_{new} = A_{old} - \Delta A, \quad (4)$$

где  $A_{new}$  - новая точность анализа;

$A_{old}$  - старая точность анализа;

$\Delta A$  - увеличение точности анализа.

Рассмотрим, какие факторы влияют на показатели эффективности:

1. Долгое время обработки изображений глазного дна пациента. Это может привести к более длительному ожиданию постановки диагноза, а также занимает значительную часть времени работы врача.
2. Отсутствие автоматизации в сегментации кровеносных сосудов сетчатки на изображениях глаз может привести к высокой загруженности специалиста, который занимается выделением. Это приводит к увеличению риска неточности при сегментации.
3. Средний и младший медицинский персонал не обладает необходимыми навыками для сегментации сосудов вручную, при помощи автоматизации можно обучить персонал пользованию программой, что разгрузит более высококвалифицированных специалистов.

В таблице 7 представлены ожидаемые изменения параметров системы после внедрения инновации.

Таблица 7 - Влияние инновации на параметры оценки эффективности

| Параметр системы                                          | Параметр системы AS IS |      |     | Параметр системы TO BE |      |     |
|-----------------------------------------------------------|------------------------|------|-----|------------------------|------|-----|
|                                                           | Min                    | Avg  | Max | Min                    | Avg  | Max |
| T - время, затрачиваемое на обработку изображений, минуты | 3                      | 16.5 | 30  | 0.5                    | 0.75 | 1   |
| A, доля правильных, безошибочных фактов сегментации       | 0,94                   |      |     | 0,96                   |      |     |

Представленная инновация оказывает значительное влияние на параметры эффективности промышленного предприятия. По расчетам благодаря внедрению инновации предприятие сможет добиться снижения времени, затрачиваемого на обработку изображений данных (T). При параметрах системы AS IS минимальный показатель времени, которое необходимо затратить на выгрузку данных равен 3 минутам, средний 16.5 минут, а максимальный – 30 минут. Такой высокий разброс времени возникает из-за того, что время обработки одного изображения сетчатки (выделение кровеносных сосудов) вручную может

варьироваться в зависимости от опыта специалиста и требуемой точности анализа. Если требуется обработать большое количество изображений, то это может занять значительное время. Автоматизированные алгоритмы могут выполнять эту задачу гораздо быстрее. За счет внедрения инновации можно снизить временные затраты до 1 минуты в случае обработки нескольких изображений.

Также, при помощи внедрения нашей инновации можно увеличить показатель  $A$ , описывающий долю правильных, безошибочных сегментаций. Этого можно добиться за счет совершенствования архитектуры нейронной сети, изменив структуру слоев под нашу задачу. По нашим подсчетам можно добиться улучшения данного показателя со значения 0.97 до 0.98.

Ускорение и оптимизация процессов позволит снизить количество монотонной работы для врачей, что в свою очередь должно положительно сказаться на эффективности их труда.

Завершающий этап постановки задачи управления заключается в представлении системы как объекта управления в момент времени  $t$ . Схематическая репрезентация системы как объекта управления представлена на рисунке 6.

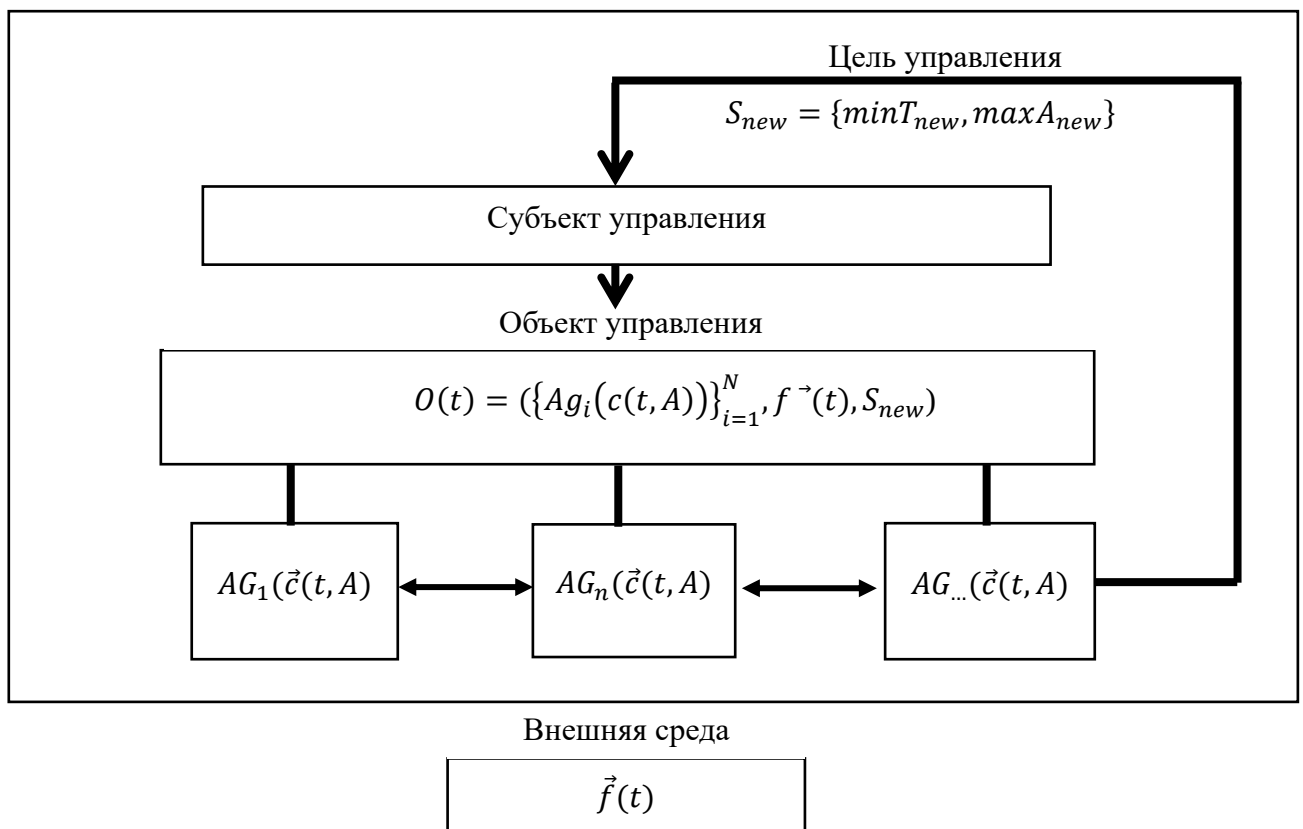


Рисунок 6 - Система, как объект управления

## 1.6 Результаты обзора литературы. Обсуждение выбранных методов сегментации кровеносных сосудов

По результатам анализа литературы была создана таблица, представленная в приложении А, подчеркивающая основные архитектуры моделей нейронных сетей с соответствующими наборами данных, на которых они прошли обучение, и их показателями эффективности. Основные показатели эффективности моделей были представлены в виде гистограммы на рисунке 7.

В частности, SERR-U-net, разработанный Дж. Ванг показал лучшие результаты по точности и специфичности, достигнув 0,980 и 0,993 соответственно, что означает, что он хорошо работает в случае классификации отрицательных случаев и имеет высокую общую точность [63]. Авторы построили свою архитектуру на основе трех ключевых компонентов: блока Encoder Booster Block для минимизации пространственных потерь

микрососудистых структур, модуля улучшения узких мест для улучшения функций перед повышением дискретизации и алгоритма адаптивного порога для идентификации пикселей. Однако SERR-U-Net показал относительно низкую оценку чувствительности. Такое явление возникает, когда модель консервативна в прогнозировании положительных случаев, пытаясь свести к минимуму ложноположительные результаты, но потенциально увеличивая количество ложноотрицательных результатов.

Наилучший результат по чувствительности показала модель HT-Net, достигнув максимума 0,982 [27]. Модель была разработана С. Ху., Л. Ванг и Ю. Ли в 2022 году и использует два основных элемента: Feature Fusion Block (FFB) для повышения сложности представления и Feature Refinement Block (FRB) для объединения функций из различных расширенных сверток для мульти-информация о масштабе. FFB улучшает такие функции, как края и текстуры, важные для изображений глазного дна, в то время как FRB собирает многомасштабную информацию посредством расширенных извилин, что приводит к улучшению результатов сегментации.

MCPANet, предложенный Ю. Цзян. и др., показал наиболее стабильные результаты, так как имеет наибольшее среднее значение по всем трем параметрам — 0,952 [32]. Архитектура MCPANet включает в себя многомасштабные механизмы внимания для решения проблем сегментации сосудов сетчатки. Сосредоточив внимание на функциях в разных положениях и масштабах входных данных, модель может собирать как локальную, так и глобальную информацию,

необходимую для точного определения сосудов, демонстрируя стабильный результат при сегментации.

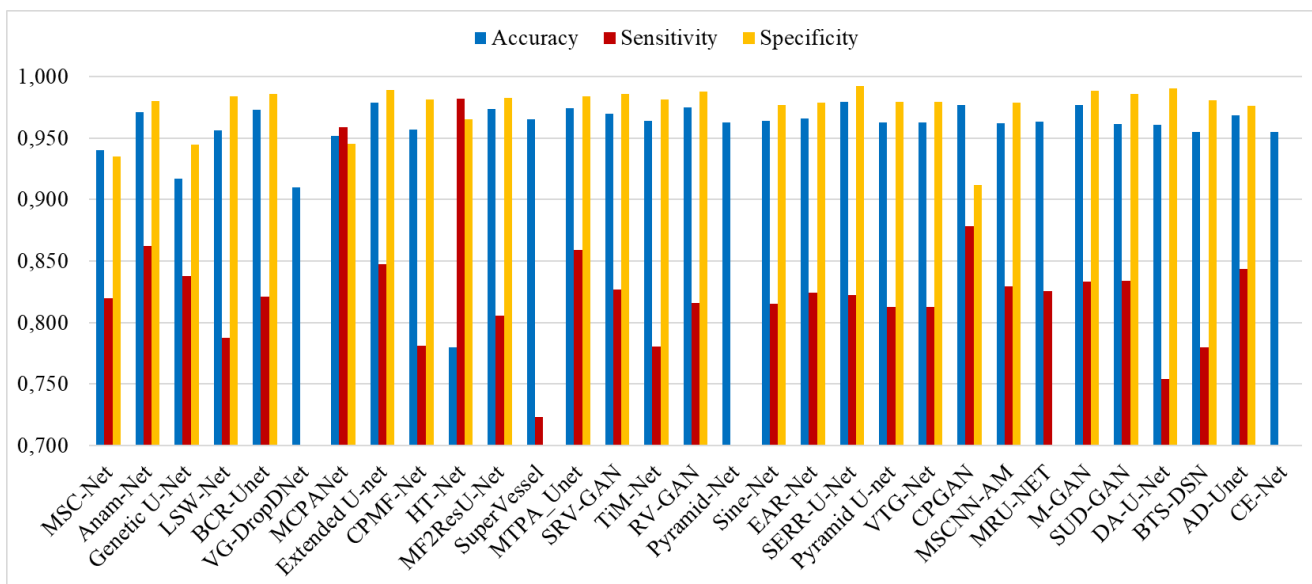


Рисунок 7 - Гистограмма показателей эффективности проанализированных моделей

## **2 РАЗРАБОТКА МЕТОДОЛОГИИ ДЛЯ СЕГМЕНТАЦИИ КРОВЕНОСНЫХ СОСУДОВ СЕТЧАТКИ ГЛАЗА**

### **2.1 Схематическое описание алгоритма сегментации кровеносных сосудов сетчатки**

Для понимания процесса сегментации кровеносных сосудов сетчатки в изображениях глазного дна следует разобраться в самом процессе сегментации. Его можно описать стандартизированной схемой, представленной на рисунке 8. Чтобы описать стандартную схему необходимо выполнить несколько ключевых этапов, основанных на результатах исследований:

1. Подготовка массива данных и предобработка изображений. Исследователи создают наборы данных с изображениями сетчатки, содержащими аннотации кровеносных сосудов. Качественные наборы данных очень важны для обучения и тестирования моделей глубокого обучения. Среди самых распространенных наборов можно выделить STARE, DRIVE, и CHASE [6, 25, 58].
2. Выбор модели. Для сегментации кровеносных сосудов обычно используются свёрточные нейронные сети (CNN). Среди самых распространенных архитектур CNN можно выделить, такие как U-Net, DenseU-Net, LadderNet, Pyramid U-net и Genetic U-Net и другие [43, 56, 62, 65, 76].
3. Процесс обучения: Выбранные модели глубокого обучения обучаются на подготовленных наборах данных. Обучение включает в себя оптимизацию параметров модели для точной сегментации кровеносных сосудов на изображениях сетчатки.
4. Тестирование и оценка: после обучения модели тестируются на отдельных наборах данных для оценки их эффективности. Для оценки качества сегментации используются такие показатели, как точность, чувствительность, специфичность, F1 score, precision и AUC. Также можно использовать

узконаправленные и специализированные тесты. Например, В. Уммади [61] предлагает использовать коэффициент Сёренсена-Дайса или его эквивалентный коэффициент Жаккара для оценки U-Net модели.

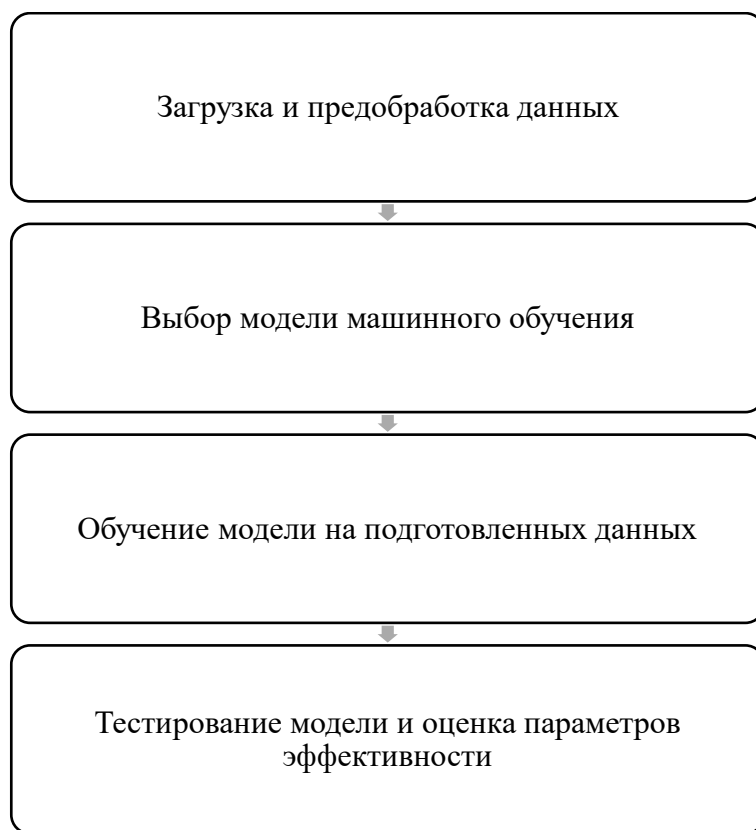


Рисунок 8 – Схема процесса разработки модели сегментации кровеносных сосудов

В целом, стандартная схема сегментации кровеносных сосудов на изображениях сетчатки с помощью глубоких нейронных сетей включает подготовку набора данных, выбор модели, обучение, тестирование, оценку с помощью различных метрик, анализ производительности и применение для помощи медицинским работникам в диагностике глазных заболеваний.

В процессе работы над задачей сегментации кровеносных сосудов исследователи сталкиваются со множеством проблем. Так, М. Р. К. Моокiah и другие [48] проведя масштабное обзорное исследование в области сегментации кровеносных сосудов выделяют следующие трудности в предобработке изображений глазного дна:

- 1) центральное освещение на фотографиях разделяет сосуды на две параллели;
- 2) сосуды с плохой контрастностью упускаются на изображениях;
- 3) разбитие сосудов на местах пересечения;
- 4) близкорасположенные, параллельные сосуды зачастую выделяются в один большой сегмент, вместо двух разных;
- 5) артефакты на изображениях, например: шумы, засветы, слияние двух сосудов в один;
- 6) такие повреждения, как микроаневризмы, экссудаты, пятна и кровоизлияния, создают ложные срабатывания или прерывают сосуды.

В результате анализа различных методов предобработки изображений и алгоритмов нейронных сетей были выбраны современные методы для улучшения алгоритма сегментации кровеносных сосудов сетчатки в изображениях глазного дна. Стандартная схема алгоритма сегментации, представленная ранее на рисунке 8 была модифицирована, новая схема с перечислением используемых методов представлена на рисунке 9. Для улучшения процесса сегментации предлагается:

1. Использовать набора данных – DRIVE. При предобработке изображений будут использованы такие методы как фильтр Кирша и CLANE. Совокупность данных методов поможет справиться с артефактами на изображениях и выделить сосудистую систему.
2. Использование улучшенной архитектуры Unet для достижения более высоких показателей эффективности. В качестве улучшения было выбрано создание дополнительных слоев Dropout в архитектуре сети для решения проблемы маленького количества входных данных. Такой метод показал свою эффективность в работе Су. Ю и др., над нейронной сетью BCR-Unet [68].

3. Для оптимизации процесса обучения предлагается подбор оптимальных гиперпараметров и вывод основных показателей – accuracy и loss для каждой тренировочной эпохи.
4. Для оценки качества сегментации предлагается использовать показатели, accuracy, sensitivity, specificity, F1 score, AUC и коэффициент Сёренсена-Дайса.

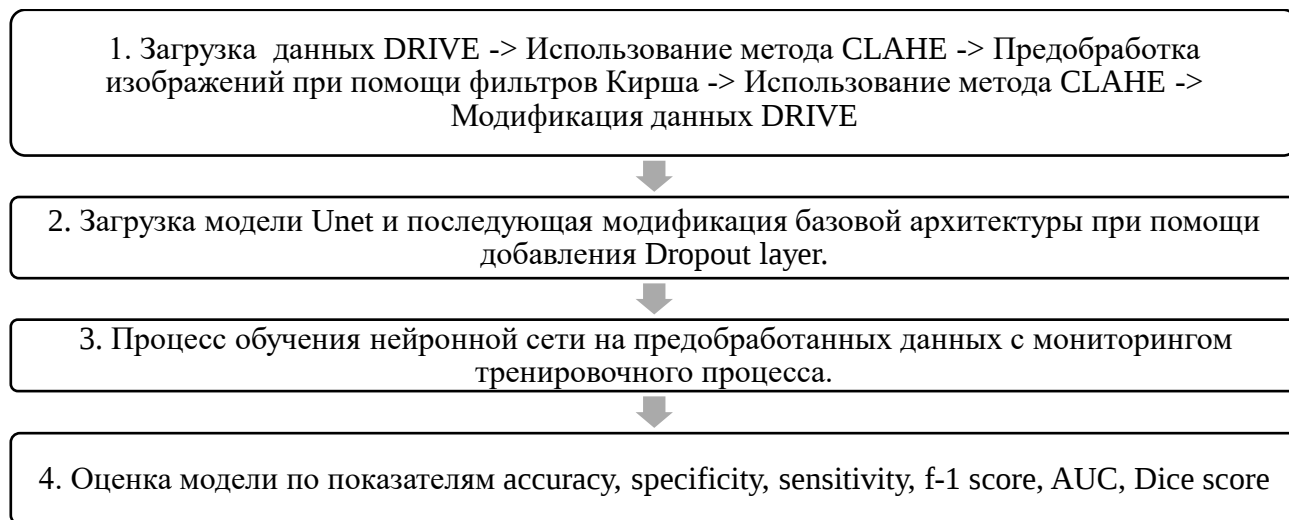


Рисунок 9 - Схема с использованием предлагаемых методов для сегментации кровеносных сосудов сетчатки

## 2.2 Описание используемых данных для задачи сегментации кровеносных сосудов сетчатки в изображениях глазного дна

Для задач сегментации кровеносных сосудов сетчатки используют специальные медицинские изображения глазного дна (фундуса). Такие изображения возможно получить при помощи специальных фундус-камер. В наборах данных для сегментации кровеносных сосудов в изображениях глазного дна зачастую используют связку цветных изображений и набора специально отрисованных черно-белых масок, на которых сосуды выделены вручную независимыми медицинскими специалистами. На данный момент существуют 4 основных набора данных для задач выделения и классификации кровеносных сосудов, среди них DRIVE [58], STARE [25], CHASE [6] и HRF [11].

Для тренировки модели был выбран набор данных DRIVE. Это самый распространенный и сбалансированный набор данных для задач выделения кровеносных сосудов сетчатки в изображениях глазного дна. В таблице 8 представлено описание набора данных с указанием общего количества изображений, количества изображений для тестирования и тренировки, размер снимков, а также соотношение количества научных работ к количеству представленных моделей на сайте Papers With Code.

Таблица 8 – Описание набора данных DRIVE

| Набор данных | Общее количество изображений | Количество изображений для тренировки | Количество изображений для тестирования | Размер изображений, пиксели | Количество проведенных исследований / Количество моделей |
|--------------|------------------------------|---------------------------------------|-----------------------------------------|-----------------------------|----------------------------------------------------------|
| DRIVE [58]   | 40                           | 20                                    | 20                                      | 564 x 584                   | 267 / 19                                                 |

Всего данный набор состоит из 40 изображений глазного дна, среди которых 7 изображений с патологиями (изменения пигментного эпителия, кровоизлияния, экссудаты и т. д.). Все 40 изображений поровну разделены для тренировки и тестирования моделей, в тренировочном наборе 3 изображения с патологией, в наборе для тестирования 4. Помимо изображений в набор включены маски для проверки точности модели, они отрисованы вручную под руководством независимых врачей. Все изображения были получены в рамках программы выявления ретинопатии в Нидерландах. Изображения имеют размер 565×584 пикселей и получены с помощью фундус-камеры (Canon CR5 nonmydriatic 3CCD, New York, NY, USA) с полем зрения (FOV) в 45 градусов. На рисунке 10 представлен пример изображения глазного дна с соответствующей маской.

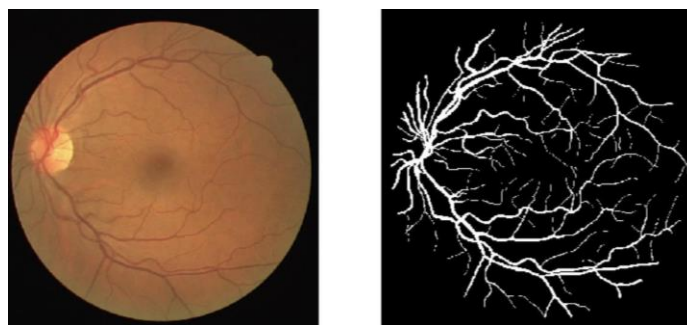


Рисунок 10 - Изображение глазного дна с истинной картой сосудов из набора DRIVE

Блок-схема, описывающая процесс загрузки данных представлена на рисунке 11. Полный программный код, используемый для загрузки данных из набора DRIVE представлен в приложении Б. Для реализации процесса загрузки были использованы следующие библиотеки:

- 1) NumPy (np) для работы с массивами и математических операций;
- 2) OpenCV (cv2) для решения задачи обработки изображений и возможности чтения файлов формата .tif;
- 3) Pillow. (PIL.Image) для возможностей обработки изображений и чтения масок, представленных в формате gif.

После установки необходимых библиотек код приступает к всестороннему исследованию структуры путей данных с помощью функции `os.path`. Путем перебора каталогов корневой папки (`/content/drive/MyDrive/BKP/DRIVE2`) код выводит полные пути ко всем файлам в подкаталогах. Этот предварительный шаг позволяет изучить данные и облегчить отладку.

Следующий шаг – чтение изображений для тестирования и тренировки из каталогов корневой папки. Это достигается за счет использования итеративных циклов `for`. В случае данных для обучения модели цикл структурирован для чтения файлов изображений, обозначенных как `{index}_training.tif`, и `{index}_manual1.gif` где значения `index` варьируются от 21 до 40 включительно. Цикл для чтения тестовых данных устроен аналогичным образом и настроен на

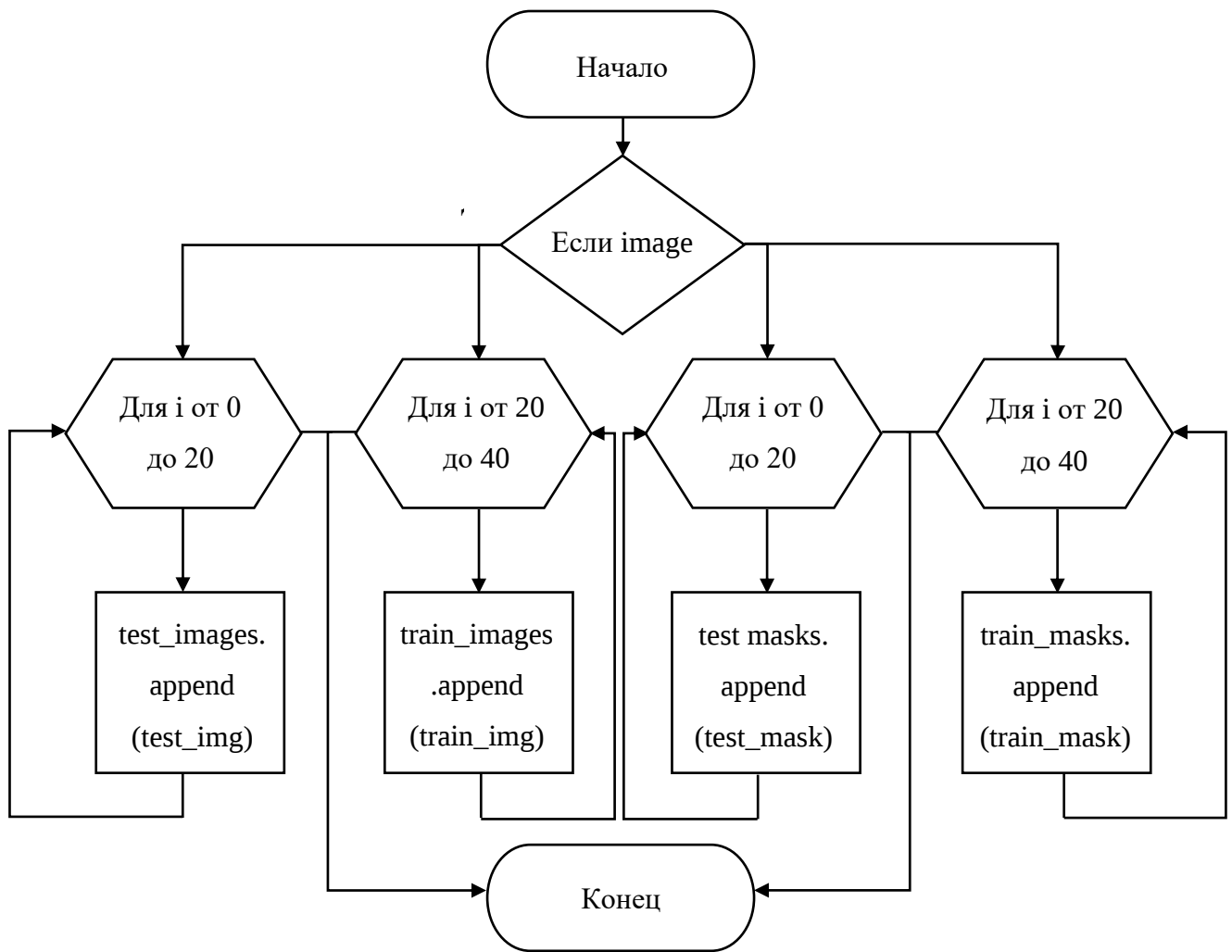


Рисунок 11 – Блок-схема процесса загрузки данных

чтение файлов с названиями  $\{index\}_{test.tif}$  и  $\{index\}_{manual.gif}$  для снимков сетчатки и масок соответственно.

Для решения проблемы ограниченного числа тренировочных данных были применены техники аугментации данных. Процесс аугментации необходим для «раздутия» выборки, путем применения различных техник можно сгенерировать дополнительные данные, основанные на изначальном наборе. Помимо увеличения числа изображений и решения проблемы нехватки данных такой подход может быть полезен для более глубокого обучения нейронной сети и увеличения её способности к генерализации. Алгоритм процесса описан блок-схемой на рисунке 12.

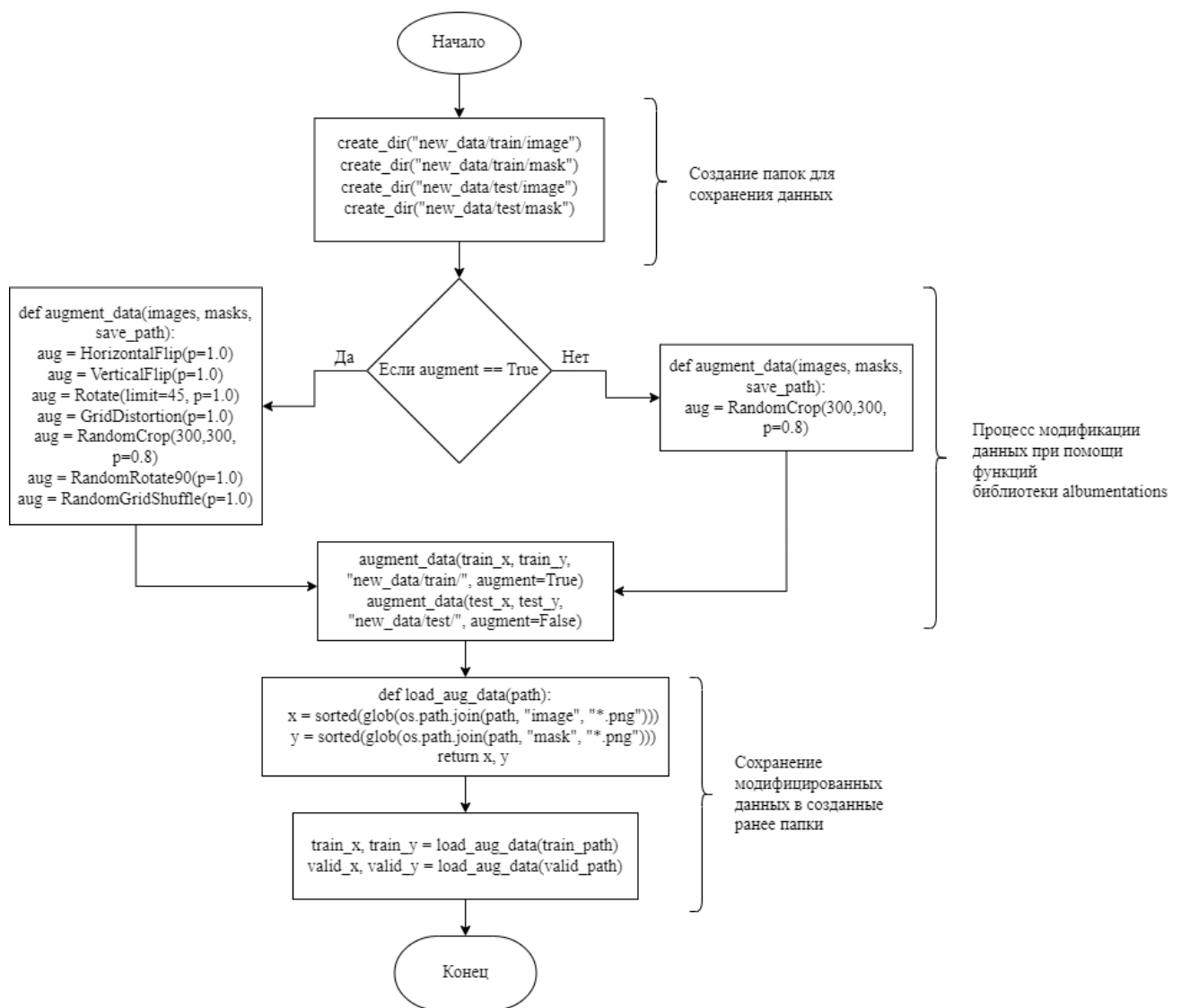


Рисунок 12 - Блок-схема алгоритма аугментации данных

Для реализации процесса аугментации была использована библиотека Albumentations с входящими в неё функциями, исходный код представлен в приложении Д. Так, для набора тренировочных данных были использованы следующие функции, параметр «р» отвечает за вероятность применение модификации к изображению:

1. HorizontalFlip (p = 1.0) отвечает за отражение изначального изображения по горизонтали.
2. VerticalFlip (p = 1.0) отражает изначальное изображение по вертикали.

3. Rotate (limit=45, p=1.0) отвечает за поворот изображения, вращение изображения происходит вокруг центра изображения, а не в левом верхнем углу, как в некоторых других фреймворках.
4. GridDistortion(p = 1.0) применяет увеличение искажений сетки к изображениям, такая техника предполагает разбиение изображения на сетку ячеек и случайное смещение точек пересечения сетки, что приводит к локальным искажениям.
5. RandomCrop (300, 300, p = 1.0) отвечает за случайное обрезание части входного изображения, по заданным параметрам высоты и ширины, в работе применяются параметры в 300 пикселей как для длины, так и для ширины изображения.
6. RandomRotate90 (p = 1.0) отвечает за случайный поворот изображения на 90 градусов.
7. RandomGridShuffle (p = 1.0) отвечает за преобразование, которое делит изображение на сетку, а затем переставляет ячейки сетки на основе случайного отображения.

На набор данных, предназначенный для тестирования модели был применен только метод RandomCrop, это позволит более детально взглянуть на маску, отрисованную через нейронную сеть для её сравнения с истинной маской.

### **2.3 Предобработка изображений. Применение фильтров Кирша с методом адаптивного гистограммного выравнивания CLAHE**

Фильтры Кирша представляют собой одну из важнейших техник обработки изображений, используемых для выделения краев и границ объектов. Разработанные в 1970-х годах Кеном Киршом, эти фильтры представляют собой набор 3x3 ядер свертки, каждое из которых нацелено на обнаружение горизонтальных, вертикальных и диагональных краев на изображении. При применении к изображению каждое ядро выделяет определенный тип границы или края. Фильтры Кирша широко применяются в компьютерном зрении,

медицинском образовании, обработке изображений и других областях. Они являются чрезвычайно полезными для обнаружения объектов на изображениях и выделения ключевых признаков.

Принцип работы фильтров Кирша основан на операции свертки изображения с ядрами Кирша. Каждое ядро представляет собой матрицу коэффициентов, которая проходит по изображению с определенным шагом. На каждом шаге выполняется умножение элементов ядра на соответствующие пиксели изображения, а затем результаты суммируются. Этот процесс позволяет выделить различные края и границы на изображении.

Фильтры Кирша являются эффективным инструментом для обработки изображений благодаря своей способности выделять различные типы краев и границ. Например, вертикальные фильтры Кирша могут помочь обнаружить вертикальные линии или границы на изображении, такие как края зданий или столбцы. Горизонтальные фильтры Кирша обычно используются для обнаружения горизонтальных элементов, таких как горизонтальные линии или края пола. Диагональные фильтры Кирша могут быть использованы для обнаружения диагональных краев или элементов на изображении.

Одним из главных преимуществ фильтров Кирша является их простота и эффективность. Они могут быть легко реализованы в различных программных средах и обеспечивают быстрое и точное выделение краев и границ на изображениях.

Кроме того, фильтры Кирша также могут быть использованы в комбинации с другими методами обработки изображений, такими как фильтры Гаусса или операции морфологической обработки, для дополнительного улучшения качества обработки изображений и получения более точных результатов.

В данной работе предлагается использовать фильтры Кирша в совокупности с адаптивной гистограммной коррективкой с ограничением контрастности (CLAHE).

CLANE (Адаптивная гистограммная коррективная с ограничением контрастности) — это усовершенствованный метод АНЕ (Адаптивная гистограммная коррективная) [36]. Данный метод устраняет проблему алгоритма АНЕ усиливать шум, ограничивая усиление контраста. CLANE работает локально с небольшими участками изображения, называемыми плитками, чтобы повысить контрастность, сохраняя при этом глобальную информацию. Алгоритм CLANE состоит из трех основных этапов:

- 1) разделение изображения на плитки;
- 2) выравнивание гистограммы каждой плитки с помощью ограничения клипа;
- 3) сшивание плиток вместе с помощью билинейной интерполяции.

Выравнивание гистограммы включает в себя вычисление гистограммы, вырезание значений в бинах, превышающих предел, перераспределение избытка в другие бины, вычисление CDF (кумулятивной функции распределения) и сопоставление значений пикселей на основе CDF. Такое локализованное выравнивание гистограммы улучшает контрастность изображения и уменьшает усиление шума.

На основе метода CLANE и фильтров Кирша был разработан алгоритм предобработки изображений, который осуществляет подготовку изображений к последующей тренировке модели. Этот процесс предварительной обработки позволяет улучшить качество изображений и извлечь больше признаков по сравнению со стандартными методами.

Для реализации алгоритма в виде программного кода были использованы библиотеки `numpy`, `pandas`, `opencv`, `matplotlib`, `statistics`, `pillow` и `imageio`. Всего было создано 8 функций, благодаря которым выполняется предобработка изображений, полная версия программного кода представлена в приложении В. В процессе предобработки можно выделить 5 основных шагов:

1. Первым шагом предварительной обработки изображений является их преобразование из цветового пространства BGR в градации серого. Для этого применяется функция `cv2.cvtColor` из библиотеки OpenCV. Этот шаг упрощает последующую обработку, поскольку изображение приводится к одному каналу, обеспечивая единообразие на последующих этапах.
2. После преобразования изображений в градации серого применяется адаптивное уравнивание гистограммы с ограничением контраста (CLAHE). Эта техника повышает контрастность, перераспределяя интенсивность пикселей, что улучшает видимость мелких деталей сосудистой структуры сетчатки. Процесс включает инициализацию объекта CLAHE с параметрами, заданными пользователем, такими как `clipLimit` и `gridsize`, для регулирования степени повышения контрастности.
3. Следующий шаг предобработки определен функциями `create_kirsch_filter`, `apply_kirsch` и `mean_kirsch_responses`. Функция `create_kirsch_filter` создает ядро для извлечения признаков с помощью фильтра Кирша. Она включает в себя определение параметров для фильтра Кирша, таких как размер ядра, что важно для его корректной работы. Затем функция генерирует маски Кирша, используя методы создания масок свертки. С помощью функции `apply_kirsch` сгенерированные маски Кирша применяются к улучшенным результатам CLAHE, используя процесс свертки. Для каждого изображения применяются все маски Кирша, выделяя границы объектов на изображении. После этого, при помощи функции `mean_kirsch_responses`, вычисляется среднее значение отклика отфильтрованных изображений, что потенциально облегчает последующий анализ или агрегирование признаков, указывающих на границы объектов на изображении.
4. После фильтрации с использованием фильтров Кирша производится применение CLAHE (функция `CLAHE_after`). Этот шаг включает в себя повторное применение CLAHE к оттенковым выходам фильтров Кирша с

целью дополнительного увеличения контрастности и выделения границ объектов на изображении. Независимая обработка каждого канала откликов фильтров Кирша помогает сократить возможные расхождения между каналами и формировать улучшенное изображение с более ярко выраженными контурами, готовое к дальнейшему анализу или визуализации.

5. Функция `preprocess_visualize` играет ключевую роль в завершении процесса предобработки изображений. Она упрощает визуализацию каждого этапа обработки, отображая исходное изображение, его оттенковую версию, результаты применения CLAHE, лучший результат CLAHE после фильтрации с помощью фильтра Кирша, средний отклик фильтра Кирша и окончательный результат после применения CLAHE после фильтрации с помощью фильтра Кирша. Это обеспечивает наглядное представление каждого этапа обработки, что помогает лучше понять, как каждый этап влияет на итоговое изображение.

Функция `preprocess` является комплексной функцией, которая последовательно вызывает все вышеупомянутые этапы обработки изображений. Она включает в себя вызовы функций для преобразования изображения в оттенки серого, применения CLAHE, фильтрации с помощью фильтров Кирша и визуализации каждого этапа обработки с использованием функции `preprocess_visualize`. Эта функция представляет собой основной механизм обработки изображений, обеспечивая их подготовку к дальнейшему анализу или использованию в приложениях медицинского образования.

## **2.4 Архитектура модели свёрточной нейронной сети. Адаптация архитектуры Unet под задачу сегментации кровеносных сосудов сетчатки**

Архитектура Unet была выбрана в качестве основной сверточной нейронной сети (CNN) для данного исследования. Она была представлена Олафом Роннебергером и коллегами в 2015 году. Эта архитектура CNN специально разработана для задач сегментации биомедицинских изображений,

таких как изображения сетчатки глаза. Основная идея этой архитектуры заключается в сохранении пространственной информации при одновременном захвате высокоуровневых характеристик, необходимых для точной сегментации. Unet состоит из двух основных частей: "сужающего контура" (Encoder) и "расширяющего контура" (Decoder):

1. В сужающем контуре (Encoder) применяются типичные сверточные нейронные сети. Здесь многократно применяются операции свертки, активации ReLU и операции объединения, такие как Max Pooling. По мере продвижения через слои количество каналов признаков увеличивается, а пространственное разрешение изображения (ширина и высота) уменьшается. Это способствует извлечению высокоуровневых характеристик из входного изображения.
2. Расширяющий контур (Decoder) восстанавливает пространственную информацию, которая была потеряна в сужающемся контуре. Он использует операции транспонированных сверток (сверток с повышающей дискретизацией) для увеличения разрешения при одновременном уменьшении числа каналов признаков. Особенно важно, что пропускные соединения напрямую связывают карты признаков из сужающегося пути с картами признаков из расширяющегося пути на том же уровне разрешения. Эти соединения сохраняют точные пространственные детали, извлеченные на предыдущих этапах, что обеспечивает более точную сегментацию.

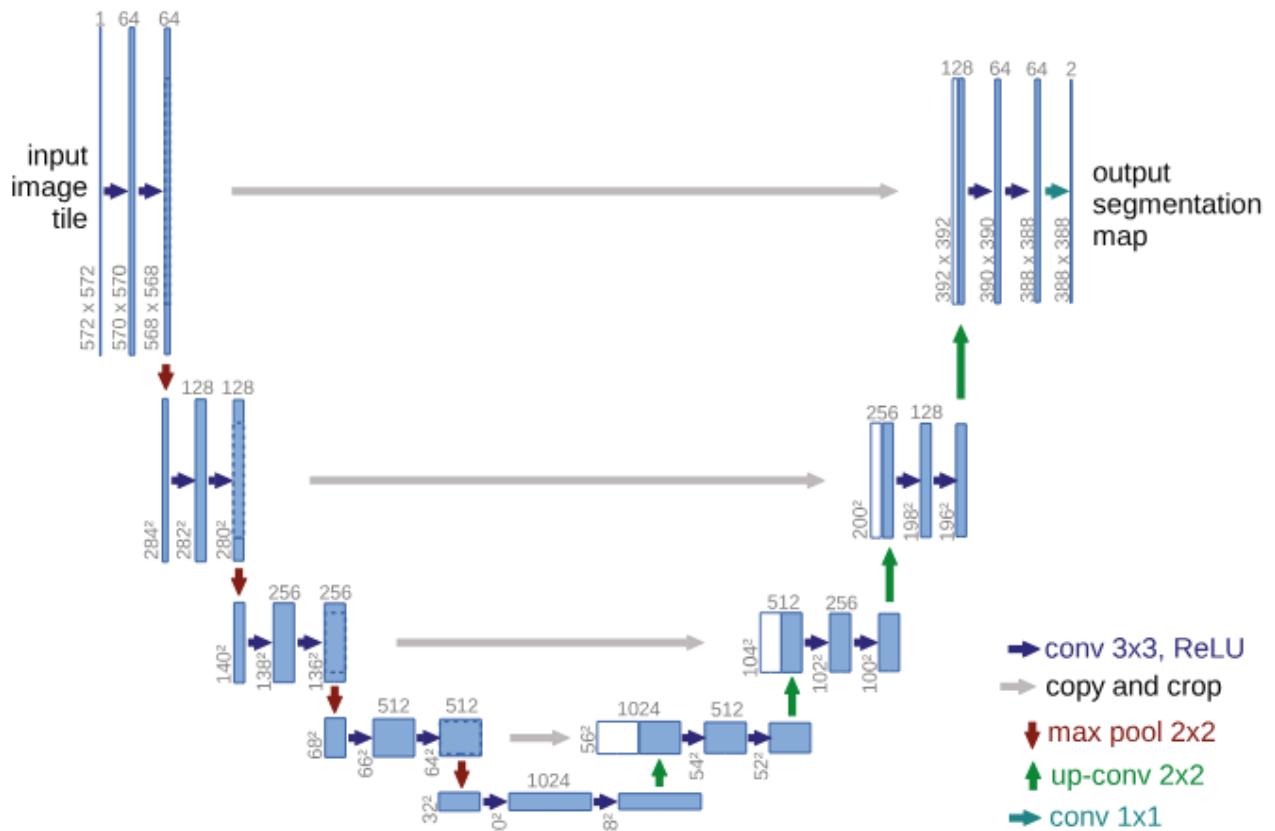


Рисунок 13 - Архитектура нейронной сети Unet [57]

Для построения нейронной сети использовалась библиотека TensorFlow, фреймворк машинного обучения с открытым исходным кодом, разработанный компанией Google. TensorFlow предоставляет инструменты и библиотеки для эффективного создания, обучения и развертывания моделей машинного обучения. Его высокоуровневый API упрощает задачи построения, обучения, оценки и развертывания нейронных сетей на различных платформах и устройствах. В процессе работы были импортированы не только основные библиотеки, но и вспомогательные слои, такие как Input, Conv2D, MaxPooling2D, Dropout, concatenate, Conv2DTranspose. Полная версия программного кода, используемого для построения модели, представлена в приложении Г. Визуальное представление предлагаемой архитектуры изображено на рисунке 17.

Структура модели состоит из 9 слоев и её можно разделить на 5 основных частей:

1. Входной слой (Input Layer) инициализирует сеть и определяет форму входных изображений для обработки. В данной архитектуре нейронной сети входной слой принимает изображения в формате (512, 512, 3), что означает, что разрешение входного изображения составляет 512 на 512 пикселей, и изображение содержит 3 канала цвета.
2. Сужающий контур (Encoder) включает в себя сверточные слои (Conv2D), которые используются для извлечения признаков из входных изображений. Эти слои состоят из повторяющихся применений фильтров размером 3x3 с активацией функции выпрямленного линейного блока (ReLU). Каждый сверточный слой извлекает особенности изображения, сохраняя его пространственную информацию. По мере увеличения глубины сети количество фильтров в слоях увеличивается, что облегчает извлечение иерархических признаков. В сужающий контур также входят слои объединения MaxPooling2D. Эти операции понижают пространственные размеры карт признаков путем уменьшения выборки, что помогает захватывать наиболее значимые признаки и снижает сложность вычислений. В конце сужающего контура располагаются слои Dropout. Они применяются к самому глубокому сверточному слою в процессе сужения, чтобы предотвратить переобучение путем случайного исключения части входных единиц во время обучения.
3. Зона "Bottleneck" описывает наиболее узкое место в модели, которое служит связующим звеном между сужающимся и расширяющимся путями. В "Bottleneck" используется сверточный слой Conv2D, который извлекает абстрактные признаки из уменьшенных карт признаков.
4. Расширяющий контур (Decoder) начинается с использования сверточных слоев Conv2DTranspose в начале каждого блока, которые выполняют апсемплинг для

постепенного увеличения пространственных размеров карт признаков. Эти слои помогают восстановить пространственное разрешение, которое было утрачено в процессе сужения. Затем используется слой concatenate для объединения карт признаков из сужающегося контура с картами из расширяющегося контура. Пропускные соединения позволяют передавать детали низкого уровня в слои с более высоким разрешением, что содействует более точной сегментации. В конце каждого блока располагаются сверточные слои Conv2D, которые уточняют карты признаков и собирают дополнительную информацию для сегментации.

5. Модель завершается выходным слоем, который включает в себя сверточный слой с сигмоидальной функцией активации (Conv2D). Этот слой обеспечивает вероятность классификации пикселей, где значения находятся в диапазоне от 0 до 1, указывая на вероятность принадлежности каждого пикселя к целевому классу.

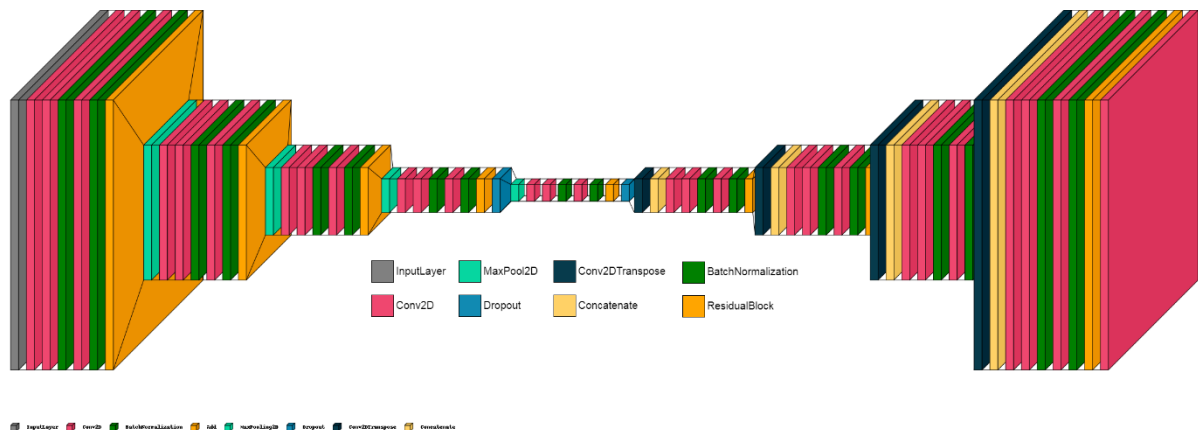


Рисунок 14 - Архитектура предлагаемой свёрточной нейронной сети

## 2.5 Модификация стандартного процесса сегментации кровеносных сосудов сетчатки. Результаты разработки методологии

В ходе разработки методологии были предложены методы предобработки данных и создания архитектуры модели нейронной сети, нацеленной на решение

задачи сегментации кровеносных сосудов сетчатки на медицинских изображениях глазного дна.

Предложенный алгоритм предобработки изображений направлен на решение основных проблем, связанных с подготовкой медицинских изображений сетчатки к сегментации и выделению сосудистой системы. Применение фильтров Кирша эффективно устраняет артефакты, которые могут затруднять извлечение признаков из изображения. Метод CLANE для повышения контрастности позволяет четко выделить сосудистую систему, не искажая при этом структуру изображения, что имеет важное значение при обучении нейронной сети. Модифицированная схема сегментации сосудов включает использование набора данных DRIVE и усовершенствованной архитектуры ResUnet.

### **3 РАЗРАБОТКА СВЕРТОЧНОЙ НЕЙРОННОЙ СЕТИ ДЛЯ СЕГМЕНТАЦИИ КРОВЕНОСНЫХ СОСУДОВ СЕТЧАТКИ ГЛАЗА НА ИЗОБРАЖЕНИЯХ ГЛАЗНОГО ДНА**

#### **3.1 Программная реализация алгоритма предобработки изображений.**

##### **Результаты применения фильтров Кирша с методом CLANE**

Программный код в приложении В описывает алгоритм предобработки набора данных DRIVE с использованием языка программирования Python и импортированных библиотек для обработки изображений.

Первый этап предобработки включает перевод изначального цветного изображения в оттенки серого, что уменьшает его размерность. Это достигается с помощью функции `to_grayscale`, которая применяет преобразование цветового пространства BGR в оттенки серого с использованием функции `cv.COLOR_BGR2GRAY` из библиотеки OpenCV. На выходе получается изображение формата (584, 565, 1), где каждый пиксель представлен одним значением интенсивности серого.

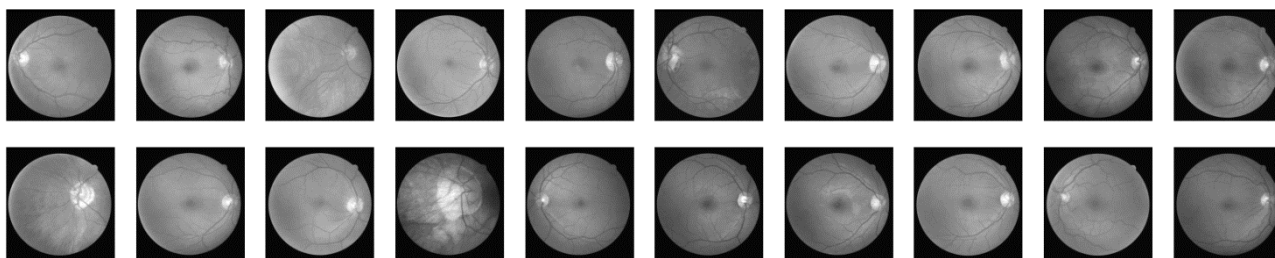


Рисунок 15 – Результат применения функции `to_grayscale` к тренировочным данным

В следующем этапе применяется метод CLANE (Contrast Limited Adaptive Histogram Equalization) к массиву изображений в оттенках серого. Этот метод позволяет улучшить контрастность изображений, что может быть полезным для выделения деталей в сложных областях. Функция `apply_CLANE` применяет встроенную функцию `cv.createCLANE` из библиотеки OpenCV с заданными параметрами, такими как `clipLimit=2.0` и `tileGridSize=(8,8)`, которые определяют

характеристики применяемого метода. Далее, функция CLAHE\_options используется для выбора оптимальных значений параметров clipLimit, при этом параметр tileGridSize остается неизменным. значения параметров clipLimit тестируются, а их результаты представлены в таблице 9.

Таблица 9 - Значения параметров clipLimit для метода CLAHE

| Название переменной | Значение clipLimit | Значение tileGridSize |
|---------------------|--------------------|-----------------------|
| CLAHE1              | 1.0                | 8,8                   |
| CLAHE2              | 2.0                | 8,8                   |
| CLAHE3              | 3.0                | 8,8                   |
| CLAHE4              | 4.0                | 8,8                   |
| CLAHE5              | 5.0                | 8,8                   |
| CLAHE6              | 6.0                | 8,8                   |
| CLAHE7              | 7.0                | 8,8                   |
| CLAHE8              | 8.0                | 8,8                   |
| CLAHE9              | 9.0                | 8,8                   |
| CLAHE10             | 10.0               | 8,8                   |

Выбор наилучшего варианта CLAHE был осуществлен путем сравнения гистограммных распределений полученных изображений. В результате сравнения был выбран вариант CLAHE с параметром, равным 3. Визуализация применения метода CLAHE с различными параметрами представлена на рисунке 16.

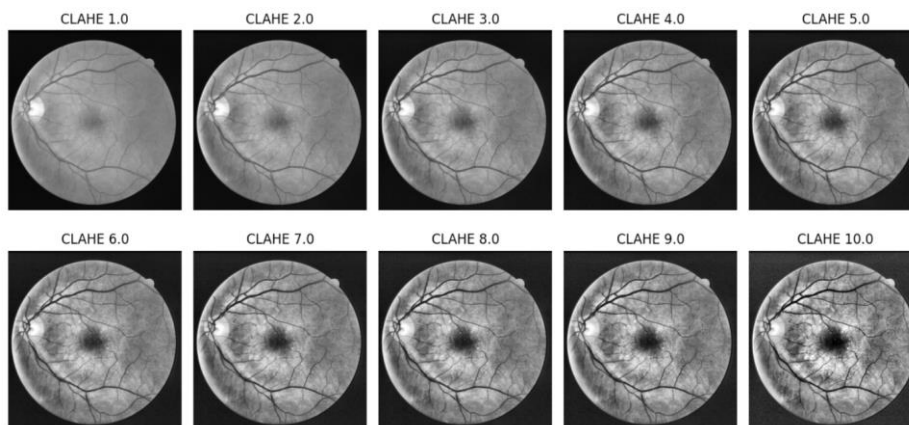


Рисунок 16 - Применение метода CLAHE с различными показателями clipLimit

В рамках последующего этапа предобработки изображений используется серия фильтров Кирша с целью более эффективного выявления важных характеристик на изображении. Этот этап предполагает последовательное применение нескольких функций. Сначала создается массив фильтров, исходя из заданных параметров, включая размер ядра, количество масштабов и ориентаций. Далее, применение фильтров Кирша включает в себя последовательное применение нескольких сверточных ядер к изображению. Каждое ядро выделяет конкретный тип границ, после чего результаты суммируются или комбинируются для получения окончательного изображения с усиленными контурами и выделенными структурами. Такой подход к обработке изображений позволяет более точно и детально анализировать содержание изображения и выделять его ключевые особенности.

Закрывающий шаг предобработки заключается в повторном применении метода адаптивной гистограммной коррективной к изображениям со средним откликом фильтров Кирша. Предобработанное фильтрами Кирша изображение имеет меньшее количество артефактов, но низкий уровень контрастности. За счет применения CLAHE можно добиться улучшения контраста изображения без потери качества признаков. Результат повторного применения метода CLAHE представлен на рисунке 17.

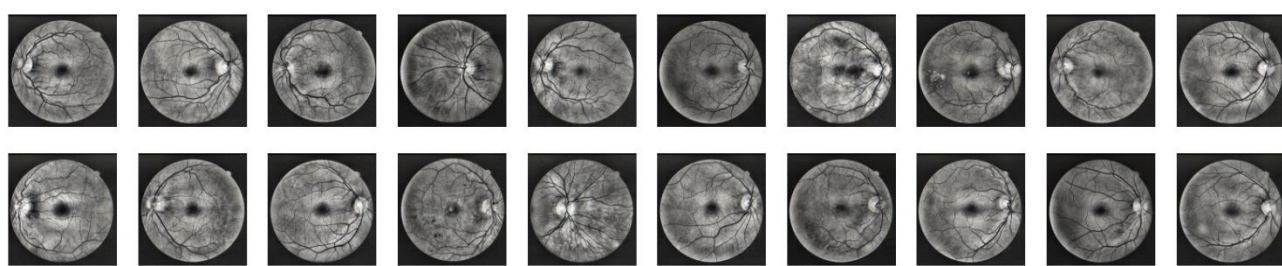


Рисунок 17 - Результат повторного применения метода CLAHE

Полный процесс обработки изображений реализован функцией preprocess и представлен на рисунке 18. Слева направо визуализированы результаты

функций предобработки, описанные ранее. Форма изображения на финальном этапе равна (584, 565, 3), что повторяет изначальный формат.

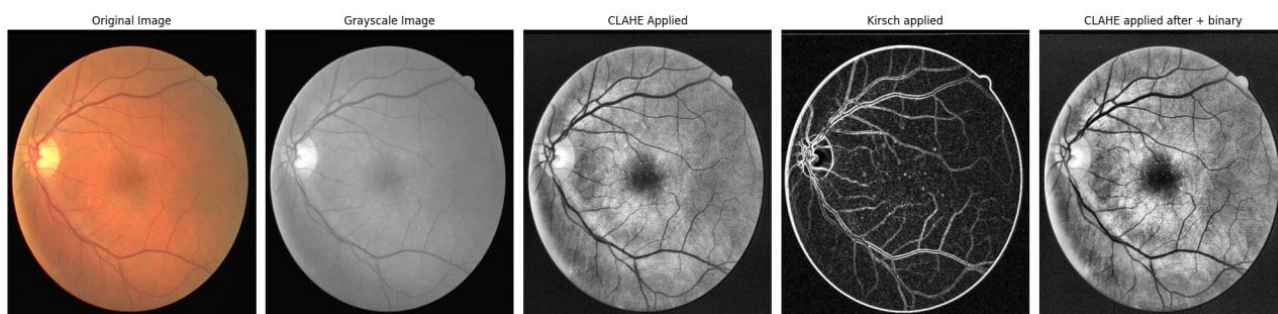


Рисунок 18 – Пошаговая визуализация процесса предобработки данных

### 3.2 Процесс обучения модели свёрточной нейронной сети.

Процесс реализации обучения можно разделить на несколько этапов. На первом этапе предобработанные данные подвергаются модификации, что включает аугментацию изображений для увеличения объема выборки.

Благодаря применению техник аугментации удалось существенно увеличить объем исходных данных: количество изображений в обучающей выборке возросло с 20 до 160, а в проверочной выборке — с 20 до 40. Примеры использованных техник аугментации на изображениях из обучающего набора представлены на рисунке 19, а на проверочных данных — на рисунке 20.

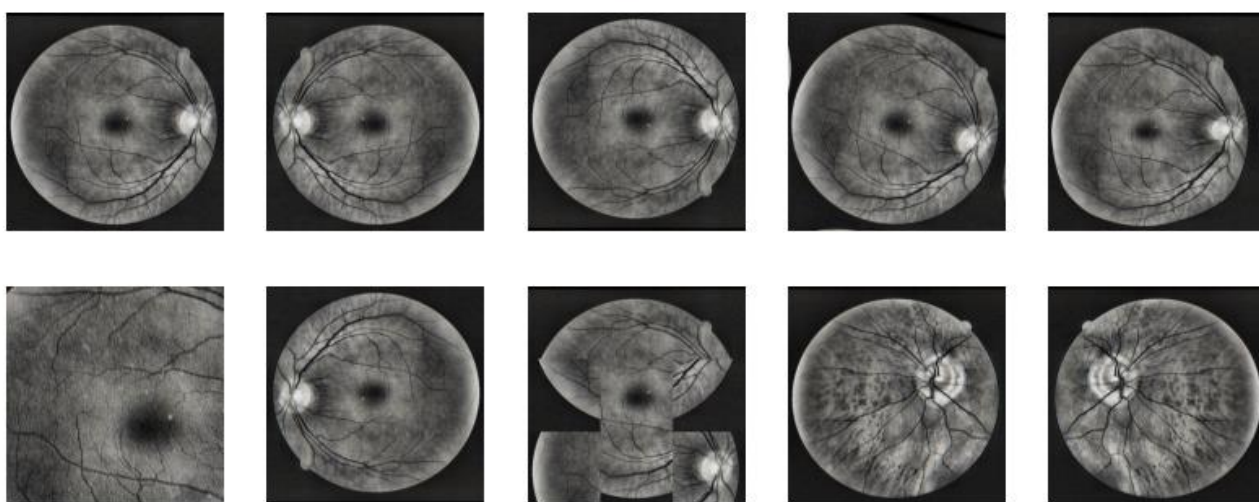


Рисунок 19 - Использование техник аугментации на обучающих данных

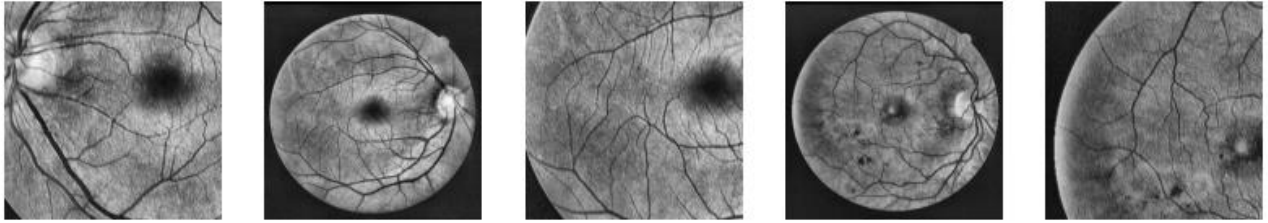


Рисунок 17 - Использование техник аугментации на проверочных данных

На следующем этапе с помощью функции **tf\_dataset** предобработанные и модифицированные данные были преобразованы в формат TensorFlow и подготовлены к передаче в модель. Для реализации процесса обучения были использованы гиперпараметры, представленные в таблице 10. Программный код, использованный для выполнения этого этапа, представлен в приложении Е.

Таблица 10 - Описание гиперпараметров обучения модели

| Название переменной | Значение параметра | Описание параметра                                            |
|---------------------|--------------------|---------------------------------------------------------------|
| num_epochs          | 50                 | Количество эпох для обучения модели                           |
| batch_size          | 2                  | Общее число тренировочных изображений, представленных в батче |
| Lr                  | 1e-4               | Размер шага на каждой эпохе                                   |

Количество шагов, используемых для прохождения моделью одной эпохи, определяется по специально заданной формуле

$$S_t = \frac{l_t}{B}, \quad (6)$$

где,  $S_t$  – количество шагов тренировочной эпохи;

$l_t$  – длина массива с изображениями для обучения модели;

$B$  – переменная batch\_size.

$$S_v = \frac{l_v}{B}, \quad (7)$$

где,  $S_v$  – количество шагов для проверки эпохи;

$l_v$  – длина массива с изображениями для проверки модели;

$B$  – переменная `batch_size`.

Для отслеживания процесса обучения были заданы параметры для отслеживания, на каждом обучающем шаге эпохи. Параметры представлены в таблице 11.

Таблица 11 - Описание показателей для отслеживания процесса обучения модели

| Название параметра | Описание параметра                                                                                  |
|--------------------|-----------------------------------------------------------------------------------------------------|
| Dice coefficient   | Измеряет среднее между precision и recall, обеспечивая баланс между двумя метриками                 |
| Iou                | Измеряет насколько хорошо граница предсказания совпадает с границей истины                          |
| Recall             | Измеряет долю истинно положительных случаев среди реально положительных случаев                     |
| Precision          | Измеряет долю истинных положительных результатов среди предсказанных положительных случаев          |
| Accurasy           | Измеряет долю истинно положительных и истинно отрицательных предсказаний среди общего числа случаев |

Дополнительно была создана переменная **Callback**, содержащая функции для мониторинга и регулирования процесса тренировки модели. На каждой обучающей эпохе создается специальный чекпоинт, сохраняющий модель в файл, а результаты обучения параллельно записываются в CSV-файл. Чтобы предотвратить перетренированность модели, используется функция **EarlyStopping()**, которая отслеживает изменение параметра "val\_loss" и сохраняет лучший результат на каждой пройденной эпохе. Если лучший

результат не обновляется в течение 10 подряд идущих эпох, функция завершает процесс тренировки модели.

В таблице 12 представлено разбиение процесса обучения модели по эпохам. В таблицу включены результаты каждой пятой эпохи. Всего обучение состояло из 50 эпох. Начиная с 45-й эпохи, показатель "loss" сохраняет свои значения, что указывает на то, что модель практически достигла предела обучения, и дальнейшие изменения показателей эффективности будут незначительными. Так, показатели "iou" и "dice\_coef" остались без изменений между 45-й и 50-й эпохами, а показатели "precision" и "recall" увеличились лишь на 0.001.

Таблица 12 - Процесс обучения модели по эпохам

| epoch | accuracy | dice_coef | iou   | loss  | precision_1 | recall_1 |
|-------|----------|-----------|-------|-------|-------------|----------|
| 0     | 0,009    | 0,377     | 0,247 | 0,623 | 0,259       | 0,619    |
| 5     | 0,140    | 0,716     | 0,559 | 0,284 | 0,860       | 0,603    |
| 10    | 0,197    | 0,750     | 0,601 | 0,250 | 0,895       | 0,633    |
| 15    | 0,248    | 0,771     | 0,627 | 0,229 | 0,917       | 0,649    |
| 20    | 0,254    | 0,785     | 0,646 | 0,215 | 0,931       | 0,661    |
| 25    | 0,270    | 0,795     | 0,659 | 0,205 | 0,939       | 0,669    |
| 30    | 0,351    | 0,802     | 0,669 | 0,198 | 0,945       | 0,675    |
| 35    | 0,452    | 0,811     | 0,682 | 0,189 | 0,952       | 0,681    |
| 40    | 0,505    | 0,818     | 0,692 | 0,182 | 0,959       | 0,684    |
| 45    | 0,525    | 0,821     | 0,697 | 0,179 | 0,961       | 0,686    |
| 50    | 0,532    | 0,821     | 0,697 | 0,179 | 0,962       | 0,687    |

Рисунок 21 иллюстрирует процесс тренировки в виде графика для каждого отслеживаемого показателя. Наибольший прирост показателей iou, dice\_coef, precision и recall наблюдается в течение первых 5 эпох, после чего график выравнивается, и прирост показателей значительно замедляется. Показатель accuracy демонстрирует иную динамику, показывая заметный рост на протяжении всего процесса обучения.

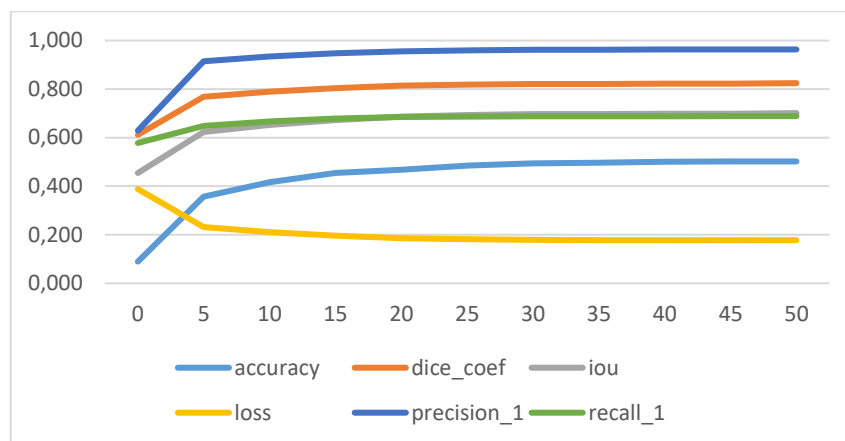


Рисунок 21 - Визуализация тренировочного процесса модели

### 3.3 Результаты обучения и анализ полученных результатов.

Рисунок 22 демонстрирует общие результаты параметров эффективности модели, рассчитанные с помощью программного кода, представленного в приложении Е. Показатель точности (Accuracy) находится на высоком уровне и составляет 0,961, что указывает на то, что 96,1% пикселей на изображениях сетчатки были правильно классифицированы как сосуды или не сосуды. Высокая точность свидетельствует о хорошей общей производительности модели. Однако следует отметить, что этот показатель может вводить в заблуждение при работе с несбалансированными наборами данных, такими как сегментация сосудов сетчатки, где преобладает большинство пикселей фона.

Показатель F1-Score составляет 0.729, что указывает на хорошую работу модели, но также свидетельствует о возможности дальнейшей настройки и улучшения.

Значение Recall, равное 0.872, означает, что 87,2% пикселей сосудов были правильно идентифицированы моделью. Высокий показатель Recall особенно важен в медицинских приложениях, поскольку он гарантирует, что большинство сосудов будет обнаружено и правильно классифицировано.

Значение Precision, равное 0.631, указывает на то, что 63.1% пикселей, идентифицированных моделью как сосуды, действительно являются

кровеносными сосудами сетчатки. Более низкая точность означает большее количество ложных срабатываний, что может привести к чрезмерной сегментации, когда пиксели, не относящиеся к сосудам, ошибочно маркируются как сосуды.

Показатель AUC, равный 0.919, измеряет способность модели различать пиксели с сосудами и без сосудов при различных пороговых значениях. Показатель AUC, равный 0.919, свидетельствует о превосходной дискриминационной способности модели, позволяющей эффективно отличать сосуды от не сосудов. Высокое значение AUC указывает на надежность и устойчивость модели в различных сценариях сегментации.

Коэффициент Дайса составляет 0.729, что указывает на хороший уровень согласия модели сегментации с реальными данными: 72.9% предсказанных пикселей сосудов совпадают с реальными пикселями сосудов.

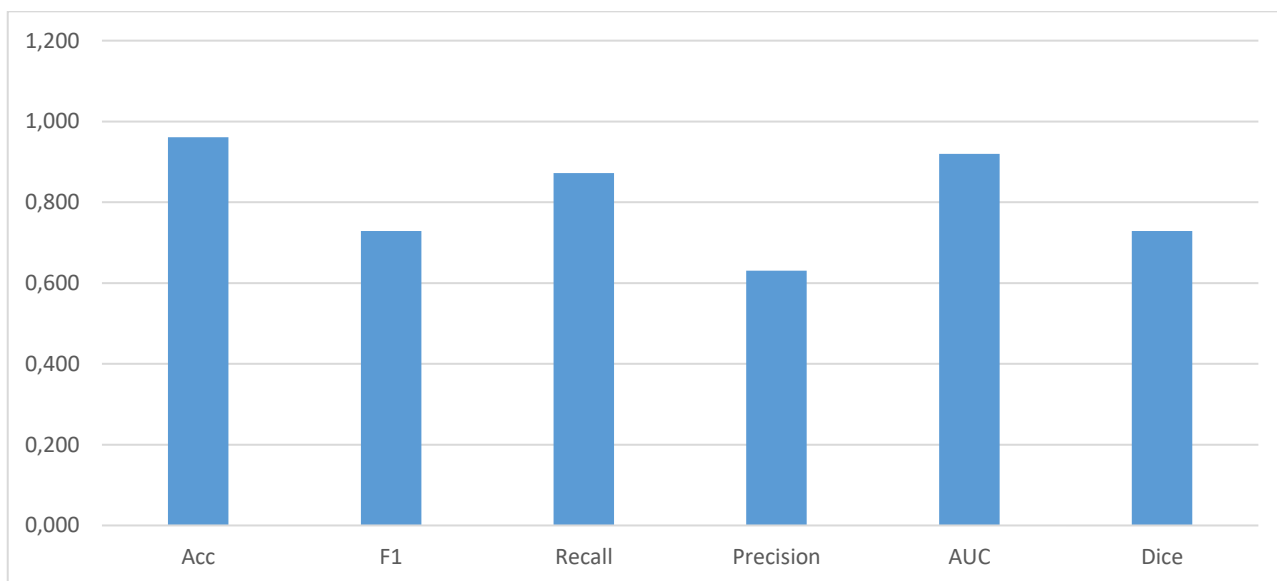


Рисунок 18 - Оценка показателей модели нейронной сети

На рисунке 23 представлен пример полученных результатов. Слева находится изображение, подаваемое на вход модели, по центру — проверочная маска, нарисованная вручную специалистом, а справа — маска, сгенерированная нейронной сетью. Заметны различия в количестве мелких сосудов и их

ответвлений по сравнению с проверочным эталоном. Вероятная причина заключается в том, что модель неспособна определить тонкие кровеносные сосуды, особенно те, что находятся в центре изображения из-за пятна в центре.

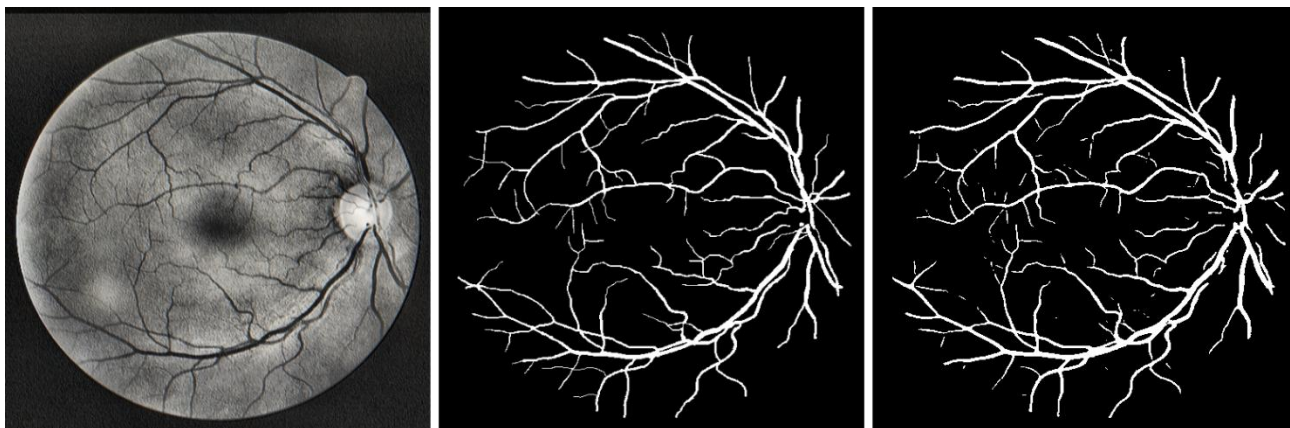


Рисунок 19 - Пример сегментации полноразмерного изображения

На рисунке 24 показано увеличенное изображение из предыдущего снимка. Большинство сосудов, видимых на изображении, правильно выделены моделью, включая мелкие ответвления. Однако наблюдаются значительные расхождения с ручной сегментацией.

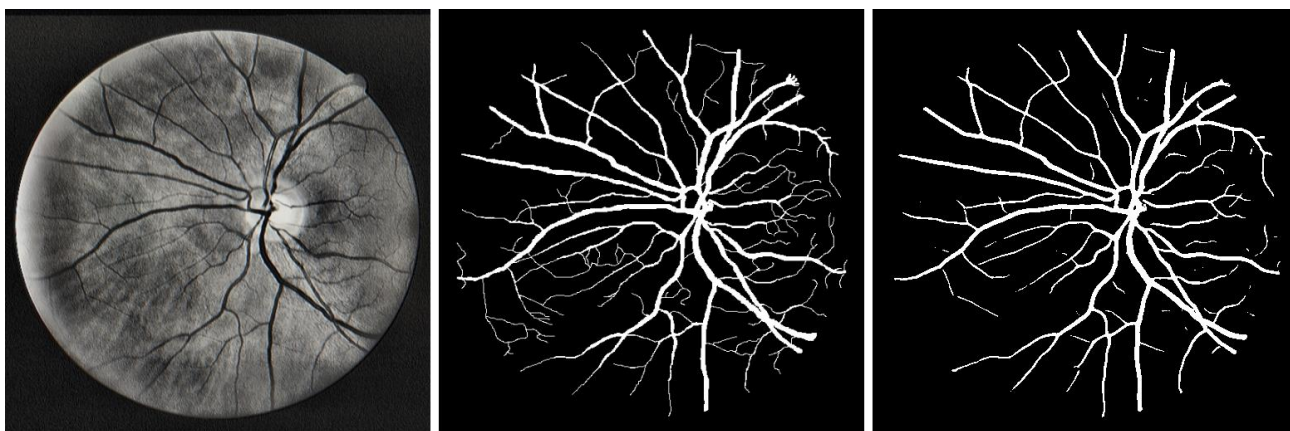


Рисунок 20 - Пример сегментации обрезанного изображения

В таблице 13 представлены подробные результаты оценки модели для каждого из проверочных изображений в изначальном формате. Лучший показатель Ассигасу, равный 0.974102, достигнут при сегментации сосудов шестого изображения (image\_6), в то время как наихудший показатель, составляющий 0.956654, наблюдается у изображения 1 (image\_1).

Таблица 13 - Показатели оценки работы модели на полноразмерных изображениях

| Image      | Acc      | F1       | Jaccard  | Recall   | Precision |
|------------|----------|----------|----------|----------|-----------|
| image_10_0 | 0,970127 | 0,747883 | 0,597295 | 0,907351 | 0,63609   |
| image_11_0 | 0,965267 | 0,733842 | 0,579582 | 0,937346 | 0,60294   |
| image_12_0 | 0,966888 | 0,739277 | 0,586391 | 0,891481 | 0,631466  |
| image_13_0 | 0,966797 | 0,761965 | 0,615463 | 0,900168 | 0,66055   |
| image_14_0 | 0,969166 | 0,740071 | 0,587392 | 0,907993 | 0,624566  |
| image_15_0 | 0,967518 | 0,705964 | 0,545552 | 0,931559 | 0,568331  |
| image_16_0 | 0,961349 | 0,714253 | 0,555517 | 0,880904 | 0,600626  |
| image_17_0 | 0,966717 | 0,698504 | 0,536693 | 0,743928 | 0,658308  |
| image_18_0 | 0,961201 | 0,688894 | 0,525429 | 0,887182 | 0,56305   |
| image_19_0 | 0,964066 | 0,712226 | 0,553067 | 0,923766 | 0,579518  |
| image_1_0  | 0,956654 | 0,691399 | 0,52835  | 0,913521 | 0,556167  |
| image_20_0 | 0,962994 | 0,679571 | 0,514659 | 0,893822 | 0,548172  |
| image_2_0  | 0,962723 | 0,77103  | 0,627378 | 0,943893 | 0,651681  |
| image_3_0  | 0,967327 | 0,761374 | 0,614693 | 0,800187 | 0,726152  |
| image_4_0  | 0,973492 | 0,79834  | 0,664364 | 0,893182 | 0,721706  |
| image_5_0  | 0,972553 | 0,787438 | 0,649401 | 0,892692 | 0,704387  |
| image_6_0  | 0,974102 | 0,787399 | 0,649347 | 0,816152 | 0,760603  |
| image_7_0  | 0,969913 | 0,768716 | 0,624321 | 0,884592 | 0,679683  |
| image_8_0  | 0,971516 | 0,749774 | 0,599711 | 0,805632 | 0,70116   |
| image_9_0  | 0,972973 | 0,755192 | 0,606673 | 0,87881  | 0,662062  |

На рисунке 25 изображено шестое изображение из проверочного набора, которое показало наилучший показатель точности (ассигасу) при проверке модели. Помимо высокой точности, это изображение также демонстрирует лучшие результаты по показателям "F1" и "Jaccard", равные 0.787 и 0.649 соответственно. Такой результат, вероятно, связан со структурой сосудов на изображении, которая значительно отличается от других снимков. Сосудистая структура сетчатки на данном изображении представлена выраженными прямыми линиями, что способствует более точной сегментации.

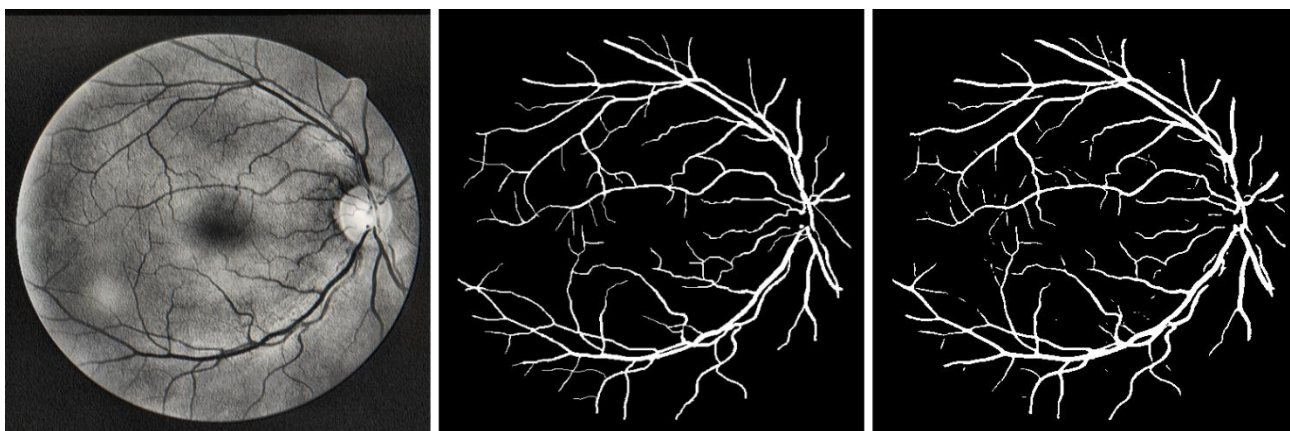


Рисунок 25 - Результат сегментации изображения image\_6\_0

На рисунке 26 представлено первое изображение из проверочного набора, которое продемонстрировало наихудший результат по совокупности показателей при проверке модели. Структура кровеносных сосудов сетчатки на данном изображении характеризуется большим количеством витиеватых тонких сосудов, что может затруднить точную сегментацию из-за своей сложности.

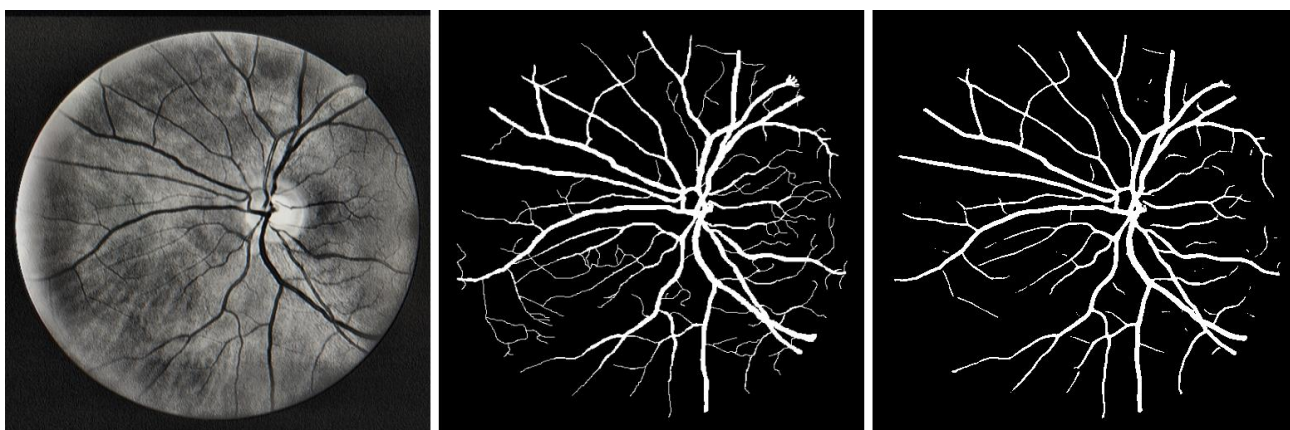


Рисунок 26 - Результат сегментации изображения image\_1\_0

В таблице 14 приведены подробные результаты оценки модели для каждого из проверочных изображений в изначальном формате. Лучший показатель точности (Accuracy), равный 0.973, достигнут при сегментации сосудов четвертого изображения (image\_4), в то время как наихудший показатель точности, равный 0.929, наблюдается у изображения 18 (image\_18). Это может быть обусловлено существенными различиями в сосудистой системе и степени ветвления сосудов на различных изображениях.

Таблица 14 - Показатели оценки работы модели на обрезанных изображениях

| Image      | Acc      | F1       | Jaccard  | Recall   | Precision |
|------------|----------|----------|----------|----------|-----------|
| image_10_1 | 0,968704 | 0,708995 | 0,549181 | 0,844017 | 0,611216  |
| image_11_1 | 0,965267 | 0,733842 | 0,579582 | 0,937346 | 0,60294   |
| image_12_1 | 0,952583 | 0,618829 | 0,448046 | 0,824818 | 0,495166  |
| image_13_1 | 0,938755 | 0,703766 | 0,542931 | 0,759287 | 0,655812  |
| image_14_1 | 0,947094 | 0,738533 | 0,585456 | 0,91005  | 0,621415  |
| image_15_1 | 0,949783 | 0,696738 | 0,534611 | 0,923255 | 0,559473  |
| image_16_1 | 0,9296   | 0,694964 | 0,532524 | 0,853934 | 0,585893  |
| image_17_1 | 0,942467 | 0,688633 | 0,525126 | 0,720868 | 0,659157  |
| image_18_1 | 0,929214 | 0,688658 | 0,525155 | 0,881037 | 0,565235  |
| image_19_1 | 0,940487 | 0,657008 | 0,489212 | 0,929981 | 0,50792   |
| image_1_1  | 0,956654 | 0,691399 | 0,52835  | 0,913521 | 0,556167  |
| image_20_1 | 0,962994 | 0,679571 | 0,514659 | 0,893822 | 0,548172  |
| image_2_1  | 0,962723 | 0,77103  | 0,627378 | 0,943893 | 0,651681  |
| image_3_1  | 0,961048 | 0,710712 | 0,551244 | 0,806987 | 0,63496   |
| image_4_1  | 0,973492 | 0,79834  | 0,664364 | 0,893182 | 0,721706  |
| image_5_1  | 0,951344 | 0,773071 | 0,630086 | 0,875625 | 0,692021  |
| image_6_1  | 0,957619 | 0,785048 | 0,646156 | 0,801961 | 0,768834  |
| image_7_1  | 0,95343  | 0,759068 | 0,611692 | 0,869394 | 0,67359   |
| image_8_1  | 0,971516 | 0,749774 | 0,599711 | 0,805632 | 0,70116   |
| image_9_1  | 0,972416 | 0,701457 | 0,540188 | 0,867899 | 0,588582  |

На рисунке 27 изображено «image\_2\_1», которое показало наилучшие средние результаты среди всех изображений из проверочного набора DRIVE. Это можно объяснить тем, что на обрезанном изображении присутствует большое количество широких сосудов, которые правильно распознаются нейронной сетью. Это обстоятельство способствует высоким показателям модели при тестировании. Кроме того, сосудистая структура на этом изображении четко выражена и не содержит значительных артефактов или проблем с контрастностью.

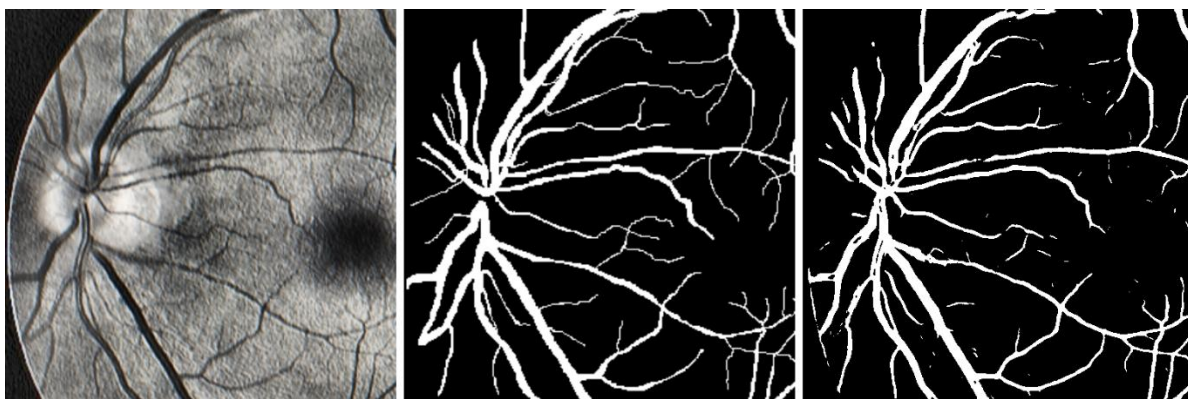


Рисунок 27 - Результат сегментации изображения image\_2\_1

На рисунке 28 представлено изображение «image\_8\_1», которое показало наихудшие средние результаты среди всех изображений проверочного набора DRIVE. На нем видно большое количество микрососудов и точек разветвления, что затрудняет точную сегментацию нейронной сетью. Кроме того, изображение содержит искажения сетчатки, видимые слева. Маска, сгенерированная нейросетью, не включает микрососуды в области искажения сетчатки.

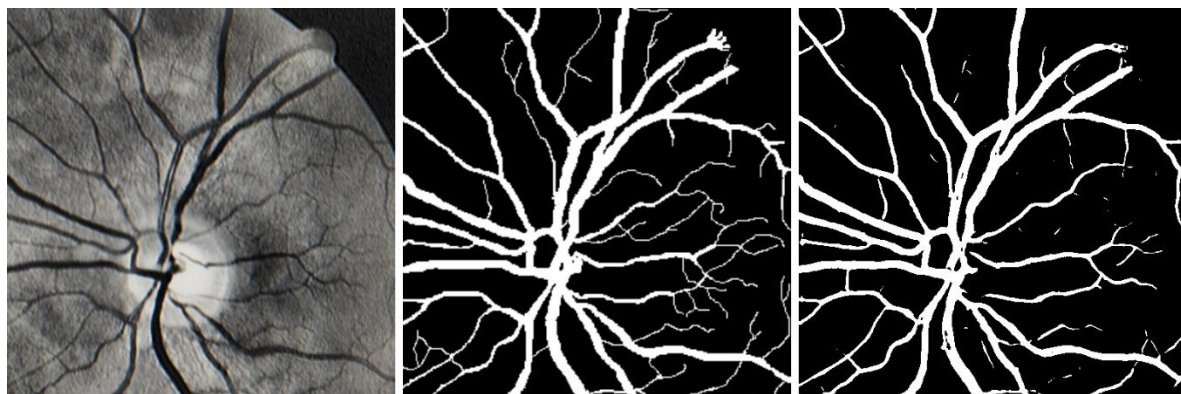


Рисунок 28 - Результат сегментации изображения image\_8\_1

### 3.4 Обсуждение и анализ результатов обучения модели нейронной сети.

В таблице 15 представлен сравнительный анализ предложенной архитектуры нейронной сети с архитектурами, ранее представленными. Предложенная модель демонстрирует самый высокий показатель точности Ассурасу (0,961) среди анализируемых моделей, что свидетельствует о высокой общей эффективности и отличной способности различать пиксели с сосудами и

без сосудов. Однако модель также показывает самый низкий показатель Recall (0,872), что указывает на ее недостаточную способность правильно идентифицировать истинные пиксели сосудов и возможность получения ложноотрицательных результатов. Это критически важно для медицинской диагностики, где отсутствие обнаружения пикселей сосудов может привести к неправильному диагнозу. Кроме того, модель обладает самым низким коэффициентом Dice (0,729), что указывает на неоптимальный баланс между ложноположительными и ложноотрицательными результатами.

Среди других показателей и их значений можно выделить следующие:

1. Показатель Precision модели составляет 0.631, в то время как модель демонстрирует самый высокий показатель Recall в размере 0.872. Повышение точности могло бы привести к сокращению количества ложных срабатываний, что в свою очередь улучшило бы чистоту карт сегментации.
2. F1-Score модели составляет 0.729, а Jaccard Score 0.575, что также является самыми высокими показателями.

Таблица 15 - Сравнение разработанной архитектуры Unet с другими вариантами

| Model                                  | Accuracy | F1-Score | Recall | Precision | Jaccard Score | AUC   | Dice Coefficient |
|----------------------------------------|----------|----------|--------|-----------|---------------|-------|------------------|
| Предложенная модель                    | 0.961    | 0.729    | 0.872  | 0.631     | 0.575         | 0.919 | 0.729            |
| Стандартная U-Net                      | 0.947    | 0.675    | 0.915  | 0.533     | 0.515         | 0.930 | 0.750            |
| Улучшенная U-Net с механизмом внимания | 0.952    | 0.710    | 0.920  | 0.585     | 0.545         | 0.940 | 0.760            |
| Deep Residual U-Net                    | 0.950    | 0.705    | 0.925  | 0.570     | 0.540         | 0.938 | 0.755            |
| Многомасштабная U-Net                  | 0.951    | 0.715    | 0.930  | 0.580     | 0.550         | 0.939 | 0.765            |

Предложенная нейросетевая модель на основе Unet демонстрирует выдающиеся показатели общей точности (Accuracy), запоминания (Recall), AUC и коэффициента Dice, что подтверждает ее значительный потенциал в сегментации кровеносных сосудов сетчатки. Эти результаты указывают на способность модели точно выявлять пиксели сосудов и добиваться

существенного соответствия с реальными данными. Однако для дальнейшего улучшения производительности стоит сосредоточиться на повышении точности (Precision) и достижении баланса между точностью и запоминанием (как показывают F1-Score и Jaccard Score). Это может помочь сократить количество ложных срабатываний и обеспечить более точные результаты сегментации.

Предлагается внедрить метод гибридной постобработки полученного изображения, изображенный на рисунке 29, для улучшения текущей модели. Этот метод предполагает наложение маски, полученной от нейронной сети, на исходное изображение. Такой подход позволит специалисту отслеживать возможные неточности в сгенерированной маске и вносить корректировки при необходимости. Такое сочетание автоматической генерации маски и контроля специалиста обещает высокую точность сегментации, снижая трудозатраты на ручную обработку.

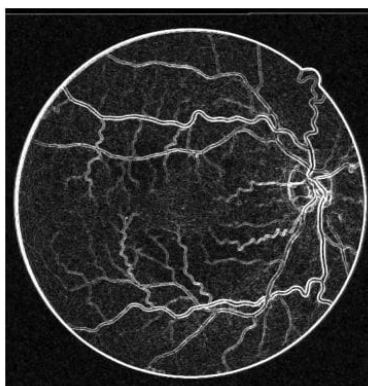


Рисунок 29 – Результат наложения сгенерированной маски на предобработанное изображение

Данный подход обещает достижение максимальной точности при выделении кровеносных сосудов. Непосредственное участие медицинского специалиста позволит оперативно корректировать результаты сегментации, обеспечивая высокое качество окончательного результата. Такая комбинация автоматизированной генерации маски и экспертного контроля существенно сократит временные и трудовые затраты, что делает процесс более эффективным и доступным для практического использования.

## ЗАКЛЮЧЕНИЕ

В данной статье представлены результаты проведенного исследования, направленного на достижение целей, обозначенных во введении. Основная цель работы заключалась в совершенствовании существующих методов и алгоритмов сегментации кровеносных сосудов сетчатки глаза на медицинских изображениях с применением нейронных сетей. Для достижения поставленной цели были установлены следующие задачи: поиск научной литературы и патентной документации и последующий проводимый ее анализ, постановка задачи управления для медицинских организаций, определение наиболее эффективных и точных подходов к сегментации кровеносных сосудов, а также выделение новаторских элементов предложенного метода.

Обзор литературы явно подчеркивает сложности, с которыми сталкиваются исследователи при сегментации кровеносных сосудов сетчатки. Низкий контраст изображений, наличие артефактов и сложная структура сосудов – все это представляет серьезные вызовы для точного выделения сосудистой системы. Однако в обзоре также рассматриваются различные методы решения этой проблемы, включая применение разнообразных нейросетевых архитектур, таких как SERR-U-Net, HT-Net и MCPANet. Каждая из этих архитектур обладает своими сильными сторонами в различных показателях производительности, таких как точность, специфичность и чувствительность.

Предложенная методология успешно справилась с вышеупомянутыми сложностями, выделив важность использования надежных методов предварительной обработки изображений и усовершенствованных модификаций нейронных сетей. Особое внимание уделено улучшению качества изображений перед сегментацией, в частности, рекомендованы методы, такие как фильтрация серого, адаптивная гистограммная эквализация с ограничением контраста (CLANE) и фильтры Кирша.

Эмпирические результаты разработанной нейросетевой модели свидетельствуют о значительном прогрессе в сегментации кровеносных сосудов сетчатки. В сравнении с другими моделями, разработанная модель показала самый высокий показатель точности (0,961), однако она также продемонстрировала самый низкий показатель площади под кривой (AUC) (0,919). Самый низкий показатель recall (0,872) указывает на то, что модель все еще имеет потенциал для улучшения в минимизации ложных результатов и точном определении пикселей сосудов.

Точность (precision) модели (0,631) превышает показатели других моделей, что указывает на высокую чувствительность данной модели. Однако для достижения более высокой точности и общего качества сегментации требуется дополнительная настройка. Показатели F1-Score (0,729) и Dice score (0,729) также оказались выше, чем у других моделей, что свидетельствует о высоком потенциале для роста при последующей настройке.

Предложенный в работе гибридный подход к постобработке представляет собой перекрытие маски, созданной нейронной сетью, с изначальным изображением. Этот метод позволяет специалистам вмешаться в процесс и вручную исправить любые неточности, сочетая эффективность автоматической сегментации с точностью ручного контроля.

В итоге, данное исследование успешно достигло своей цели, улучшив процесс сегментации кровеносных сосудов сетчатки с помощью новых методов предварительной обработки и модификации нейросетевой модели. Разработанные методики и улучшения модели привели к значительному повышению ключевых показателей эффективности, включая точность, запоминание и AUC, и выявили перспективы для дальнейшего развития, особенно в области точности. Гибридный метод постобработки представляет собой обещающее направление для будущих исследований, обеспечивая баланс между автоматизированной эффективностью и ручной коррекцией.

Эти достижения имеют огромное значение на практике, поскольку они не только сокращают время и усилия, затрачиваемые на обработку изображений сетчатки, но и повышают точность диагностики заболеваний глазного дна. Улучшенная сегментация кровеносных сосудов помогает более точно выявлять патологии и болезни, что может привести к более раннему обнаружению и эффективному лечению. Кроме того, эти результаты играют ключевую роль в развитии медицинского образования и научных исследований, предоставляя ценные данные для будущих инноваций в области медицинской диагностики.

## СПИСОК ЛИТЕРАТУРЫ

1. М. А. Нафиков Алгоритмы сегментации кровеносных сосудов сетчатки глаза. 2016.
2. Насонов А.В. [и др.]. ПРИМЕНЕНИЕ МЕТОДА МОРФОЛОГИЧЕСКИХ АМЁБ ДЛЯ ВЫДЕЛЕНИЯ СОСУДОВ НА ИЗОБРАЖЕНИЯХ ГЛАЗНОГО ДНА. 2011.
3. A Isavand Rahmani, H Akbari, S Saraf Esmaili Retinal blood vessel segmentation using gabor filter and morphological reconstruction 2020.
4. Ahmed M. N. [и др.]. A Modified Fuzzy C-Means Algorithm for Bias Field Estimation and Segmentation of MRI Data. 2002.
5. Ali A. [и др.]. Segmenting Retinal Blood Vessels with Gabor Filter and Automatic Binarization. 2018.
6. Alyoubi W. L., Shalash W. M., Abulkhair M. F. Diabetic retinopathy detection through deep learning techniques: A review // Informatics in Medicine Unlocked. 2020. T. 20.
7. Arora J., Tushir M. An Enhanced Spatial Intuitionistic Fuzzy C-means Clustering for Image Segmentation Elsevier B.V., 2020.C. 646–655.
8. Atli İ., Gedik O. S. Sine-Net: A fully convolutional deep learning architecture for retinal blood vessel segmentation // Engineering Science and Technology, an International Journal. 2021. № 2 (24). С. 271–283.
9. Aurangzeb K., Haider S. I., Alhussein M. Retinal Vessel Segmentation Based on the Anam-Net Model // Elektronika ir Elektrotechnika. 2022. № 3 (28). С. 54–64.
10. Boudegga H. [и др.]. Extended U-net for retinal vessel segmentation 2022.
11. Budai A. [и др.]. Robust vessel segmentation in fundus images // International Journal of Biomedical Imaging. 2013. (2013).
12. Cai W., Chen S., Zhang D. Fast and Robust Fuzzy C-Means Clustering Algorithms Incorporating Local Information for Image Segmentation. 2007.

13. Chan Y., Cheng C. Y., Sabanayagam C. Eyes as the windows into cardiovascular disease in the era of big data // Taiwan Journal of Ophthalmology. 2023. T. 13. № 2. С. 151–167.
14. Chatterjee S. [и др.]. Retinal blood vessel segmentation using edge detection method IOP Publishing Ltd, 2021.
15. Cheong H. [и др.]. OCT-GAN: single step shadow and noise removal from optical coherence tomography images of the human optic nerve head // Biomedical Optics Express. 2021. № 3 (12). С. 1482.
16. Cherif S. [и др.]. Novel Intuitionistic-Based Interval Type-2 Fuzzy Similarity Measures with Application to Clustering // IEEE Transactions on Fuzzy Systems. 2022. № 5 (30). С. 1260–1271.
17. Cui Z. [и др.]. MF2ResU-Net: A Multi-Feature Fusion Deep Learning Architecture for Retinal Blood Vessel Segmentation 2021.
18. Desiani A. [и др.]. VG-DropDNet a Robust Architecture for Blood Vessels Segmentation on Retinal Image // IEEE Access. 2022. (10). С. 92067–92083.
19. Ding H. [и др.]. MRU-NET: A U-shaped network for retinal vessel segmentation // Applied Sciences (Switzerland). 2020. № 19 (10).
20. Fedotov E. A., Chernomorets D. A., Mikhelev V. M. On the efficiency of the consistent retinal thick and thin blood vessels segmentation method // Scientific bulletins of the Belgorod State University. Series: Economics. Computer Science. 2018. № 2 (45). С. 312–321.
21. Fu Q., Li S., Wang X. MSCNN-AM: A multi-scale convolutional neural network with attention mechanisms for retinal vessel segmentation // IEEE Access. 2020. (8). С. 163926–163936.
22. Granlund G. H. In Search of a General Picture Processing Operator. 1978.
23. Gu Z. [и др.]. CE-Net: Context Encoder Network for 2D Medical Image Segmentation 2019.

24. Guo S. [и др.]. BTS-DSN: Deeply Supervised Neural Network with Short Connections for Retinal Vessel Segmentation 2018.
25. Hoover A., Kouznetsova V., Goldbaum M. Locating Blood Vessels in Retinal Images by Piecewise Threshold Probing of a Matched Filter Response. 2000.
26. Hoque M. E., Kipli K. Deep Learning in Retinal Image Segmentation and Feature Extraction: A Review // International journal of online and biomedical engineering. 2021. № 14 (17). С. 103–118.
27. Hu X., Wang L., Li Y. HT-Net: A Hybrid Transformer Network for Fundus Vessel Segmentation // Sensors. 2022. № 18 (22).
28. Hu Y. [и др.]. SuperVessel: Segmenting High-resolution Vessel from Low-resolution Retinal Image 2022.
29. Jaafar Belaid L., Mourou W. IMAGE SEGMENTATION: A WATERSHED TRANSFORMATION ALGORITHM. 2009.
30. Jiang Y. [и др.]. ASRNet: Adversarial Segmentation and Registration Networks for Multispectral Fundus Images // Computer Systems Science and Engineering. 2021. № 3 (36). С. 537–549.
31. Jiang Y. [и др.]. MTPA\_Unet: Multi-Scale Transformer-Position Attention Retinal Vessel Segmentation Network Joint Transformer and CNN // Sensors. 2022. № 12 (22).
32. Jiang Y. [и др.]. MCPANet: Multiscale Cross-Position Attention Network for Retinal Vessel Image Segmentation // Symmetry. 2022. № 7 (14).
33. Jin Q. [и др.]. DUNet: A deformable network for retinal vessel segmentation ☆ 2019. (178). С. 149–162.
34. JN Kapur, PK Sahoo, AKC Wong A new method for gray-level picture thresholding using the entropy of the histogram 1985.
35. Kamran S. A. [и др.]. RV-GAN: Segmenting Retinal Vascular Structure in Fundus Photographs using a Novel Multi-scale Generative Adversarial Network 2021.

36. Karel Zuiderveld Contrast Limited Adaptive Histogram Equalization 1994.
37. Khan K. B. [и др.]. A robust technique based on VLM and Frangi filter for retinal vessel extraction and denoising // PLoS ONE. 2018. Т. 13. № 2.
38. Kittler J., Illingworth J. MINIMUM ERROR THRESHOLDING. 1986.
39. Kornilov A., Safonov I., Yakimchuk I. A Review of Watershed Implementations for Segmentation of Volumetric Images // Journal of Imaging. 2022. Т. 8. № 5.
40. Kumar K. S., Singh N. P. Analysis of retinal blood vessel segmentation techniques: a systematic survey // Multimedia Tools and Applications. 2023. № 5 (82). С. 7679–7733.
41. Liu J. [и др.]. Automatic segmentation of foveal avascular zone based on adaptive watershed algorithm in retinal optical coherence tomography angiography images // Journal of Innovative Optical Health Sciences. 2022. № 1 (15).
42. Liu R. [и др.]. LSW-Net: A Learning Scattering Wavelet Network for Brain Tumor and Retinal Image Segmentation // Electronics (Switzerland). 2022. № 16 (11).
43. Liu Y. P. [и др.]. Feature pyramid U-Net for retinal vessel segmentation // IET Image Processing. 2021. № 8 (15). С. 1733–1744.
44. Liu Y. P. [и др.]. Feature pyramid U-Net for retinal vessel segmentation // IET Image Processing. 2021. № 8 (15). С. 1733–1744.
45. Luo Z. [и др.]. Micro-Vessel Image Segmentation Based on the AD-UNet Model // IEEE Access. 2019. (7). С. 143402–143411.
46. Ma Y. [и др.]. ROSE: A Retinal OCT-Angiography Vessel Segmentation Dataset and New Model // IEEE Transactions on Medical Imaging. 2021. № 3 (40). С. 928–939.
47. Mishra S. [и др.]. VTG-Net: A CNN Based Vessel Topology Graph Network for Retinal Artery/Vein Classification // Frontiers in Medicine. 2021. (8).

48. Mookiah M. R. K. [и др.]. A review of machine learning methods for retinal blood vessel segmentation and artery/vein classification // Medical Image Analysis. 2021. Т. 68.
49. Nair A. T., Muthuvel D. K., Haritha K. S. Blood vessel segmentation for diabetic retinopathy IOP Publishing Ltd, 2021.
50. NOBUYUKI OTSU A Threshold Selection Method from Gray-Level Histograms 1979.
51. Pan L. [и др.]. MSC-Net: Multitask Learning Network for Retinal Vessel Segmentation and Centerline Extraction // Applied Sciences (Switzerland). 2022. № 1 (12).
52. Park K. B., Choi S. H., Lee J. Y. M-GAN: Retinal Blood Vessel Segmentation by Balancing Losses through Stacked Deep Fully Convolutional Networks // IEEE Access. 2020. (8). С. 146308–146322.
53. Radha K., Karuna Y. Retinal vessel segmentation to diagnose diabetic retinopathy using fundus images: A survey // International Journal of Imaging Systems and Technology. 2024. № 1 (34).
54. Ramesh K. K. D. [и др.]. A review of medical image segmentation algorithms // EAI Endorsed Transactions on Pervasive Health and Technology. 2021. № 27 (7).
55. Rammy S. A. [и др.]. CPGAN: Conditional patch-based generative adversarial network for retinal vessel segmentation // IET Image Processing. 2020. № 6 (14). С. 1081–1090.
56. Ronneberger O., Fischer P., Brox T. U-net: Convolutional networks for biomedical image segmentation Springer Verlag, 2015. С. 234–241.
57. Ronneberger O., Fischer P., Brox T. U-Net: Convolutional Networks for Biomedical Image Segmentation 2015.
58. Staal J. [и др.]. Ridge-based vessel segmentation in color images of the retina // IEEE Transactions on Medical Imaging. 2004. № 4 (23). С. 501–509.

59. Taimur Hassan [и др.]. RAG-FW: A hybrid convolutional framework for the automated extraction of retinal lesions and lesion-influenced grading of human retinal pathology 2020.
60. Tang W., Deng H., Yin S. CPMF-Net: Multi-Feature Network Based on Collaborative Patches for Retinal Vessel Segmentation // Sensors. 2022. № 23 (22).
61. Ummadi V. U-Net and its variants for Medical Image Segmentation: A short review 2022.
62. Wang C. [и др.]. Dense U-net based on patch-based learning for retinal vessel segmentation // Entropy. 2019. № 2 (21).
63. Wang J. [и др.]. SERR-U-Net: Squeeze-and-Excitation Residual and Recurrent Block-Based U-Net for Automatic Vessel Segmentation in Retinal Image // Computational and Mathematical Methods in Medicine. 2021. (2021).
64. Wang J. [и др.]. EAR-NET: Error Attention Refining Network For Retinal Vessel Segmentation 2021.
65. Wei J., Fan Z. Genetic U-Net: Automatically Designed Deep Networks for Retinal Vessel Segmentation Using a Genetic Algorithm 2020.
66. Wu C., Zou Y., Zhan J. DA-U-Net: Densely Connected Convolutional Networks and Decoder with Attention Gate for Retinal Vessel Segmentation Institute of Physics Publishing, 2019.
67. Xu L., Luo S. A novel method for blood vessel detection from retinal images. 2010.
68. Xu Y. [и др.]. BCR-UNet: Bi-directional ConvLSTM residual U-Net for retinal blood vessel segmentation. 2022.
69. Yang T. [и др.]. SUD-GAN: Deep Convolution Generative Adversarial Network Combined with Short Connection and Dense Block for Retinal Vessel Segmentation // Journal of Digital Imaging. 2020. № 4 (33). С. 946–957.

70. Yue C. [и др.]. SRV-GAN: A generative adversarial network for segmenting retinal vessels // Mathematical Biosciences and Engineering. 2022. № 10 (19). С. 9948–9965.
71. Zhang D. Q., Chen S. C. A novel kernelized fuzzy C-means algorithm with application in medical image segmentation // Artificial Intelligence in Medicine. 2004. № 1 (32). С. 37–50.
72. Zhang H. [и др.]. TiM-Net: Transformer in M-Net for Retinal Vessel Segmentation // Journal of Healthcare Engineering. 2022. (2022).
73. Zhang J. [и др.]. Pyramid-Net: Intra-layer Pyramid-Scale Feature Aggregation Network for Retinal Vessel Segmentation // Frontiers in Medicine. 2021. (8).
74. Zheng X. [и др.]. Image segmentation based on adaptive K-means algorithm // Eurasip Journal on Image and Video Processing. 2018. № 1 (2018).
75. Zhu S. C., Yuille A. Region Competition: Unifying Snakes, Region Growing, and Bayes/MDL for Multi-band Image Segmentation. 1996.
76. Zhuang J. LadderNet: Multi-path networks based on U-Net for medical image segmentation 2018.
77. Научная электронная библиотека eLIBRARY.RU.
78. Elsevier [Электронный ресурс]. URL: <https://www.elsevier.com> (дата обращения: 07.03.2024).
79. eLibrary source code [Электронный ресурс]. URL: [https://github.com/MaksBasher/Thesis/blob/main/Elibrary\\_parcer.ipynb](https://github.com/MaksBasher/Thesis/blob/main/Elibrary_parcer.ipynb) (дата обращения: 04.03.2024).
80. Google Patents [Электронный ресурс]. URL: <https://patents.google.com> (дата обращения: 07.03.2024).
81. Rospatent [Электронный ресурс]. URL: [rospatent.gov.ru](https://rospatent.gov.ru) (дата обращения: 07.03.2024).

## ПРИЛОЖЕНИЕ А

(справочное)

### СРАВНИТЕЛЬНЫЙ АНАЛИЗ АРХИТЕКТУР НЕЙРОННЫХ СЕТЕЙ ДЛЯ ЗАДАЧ СЕГМЕНТАЦИИ КРОВЕНОСНЫХ СОСУДОВ СЕТЧАТКИ

Таблица А.1 - Современные модели для сегментации кровеносных сосудов сетчатки

| Название модели     | Набор данных                    | Accuracy | Sensitivity | Specificity | AUC   | F1    |
|---------------------|---------------------------------|----------|-------------|-------------|-------|-------|
| MSC-Net [51]        | DRIVE, STARE, CHASE             | 0.940    | 0.820       | 0.935       | 0.960 | -     |
| Anam-Net [9]        | STARE, DRIVE, CHASE_DB          | 0.971    | 0.862       | 0.980       | 0.989 | 0.817 |
| Genetic U-Net [65]  | DRIVE, CHASE-DB1, HRF,          | 0.917    | 0.838       | 0.945       | -     | 0.821 |
| LSW-Net [42]        | DRIVE, BraTS2020                | 0.957    | 0.788       | 0.984       | -     | -     |
| BCR-UNet [68]       | STARE, IOSTAR, DRIVE, CHASE-DB1 | 0.973    | 0.821       | 0.986       | 0.988 | 0.823 |
| VG-DropDNet [18]    | DRIVE, CHASE, STARE, KvasirESG  | 0.910    | -           | -           | 0.960 | 0.910 |
| MCPANet [32]        | DRIVE, CHASE, STARE             | 0.952    | 0.959       | 0.945       | -     | -     |
| Extended U-net [10] | DRIVE, HRF                      | 0.979    | 0.847       | 0.989       | -     | -     |
| CPMF-Net [60]       | DRIVE, STARE                    | 0.958    | 0.781       | 0.981       | 0.980 | -     |
| HT-Net [27]         | CHASE-DB1, DRIVE, STARE         | 0.780    | 0.982       | 0.965       | 0.983 | -     |
| MF2ResU-Net [17]    | DRIVE, CHASE-DB1                | 0.974    | 0.806       | 0.982       | 0.982 | 0.818 |
| SuperVessel [28]    | PRIME-FP20, OCTA, HRF           | 0.965    | 0.723       | -           | 0.851 | -     |
| MTPA_Unet [31]      | CHASE-DB1, DRIVE, STARE         | 0.974    | 0.859       | 0.984       | -     | -     |
| SRV-GAN [70]        | CHASE-DB1, DRIVE, STARE         | 0.970    | 0.827       | 0.986       | 0.988 | 0.825 |
| TiM-Net [72]        | CHASE-DB1, DRIVE, STARE         | 0.964    | 0.781       | 0.982       | 0.968 | -     |

|                    |                              |       |        |       |       |       |
|--------------------|------------------------------|-------|--------|-------|-------|-------|
| RV-GAN [35]        | CHASE-DB1, DRIVE, STARE      | 0.975 | 0.816  | 0.988 | 0.990 | 0.866 |
| RAG-FW [59]        | Zhang, BIOMISA, DUKE         | -     | -      | -     | -     | -     |
| Pyramid-Net [73]   | CHASE-DB1, DRIVE, STARE      | 0.963 | -      | -     | 0.983 | -     |
| Sine-Net [8]       | CHASE-DB1, DRIVE, STARE      | 0.964 | 0.815  | 0.977 | -     | -     |
| EAR-Net [64]       | DRIVE, STARE                 | 0.966 | 0.8245 | 0.979 | -     | -     |
| SERR-U-Net [63]    | DRIVE, STARE                 | 0.980 | 0.822  | 0.993 | 0.986 | -     |
| ROSE [46]          | ROSE                         | -     | -      | -     | -     | -     |
| ASRNet [30]        | MSI                          | -     | -      | -     | -     | -     |
| Pyramid U-net [44] | DRIVE, CHASE-DB1             | 0.963 | 0.812  | 0.980 | 0.982 | -     |
| VTG-Net [47]       | AV-DRIVE                     | 0.963 | 0.812  | 0.980 | -     | -     |
| OCT-GAN [15]       | Own dataset                  | -     | -      | -     | -     | -     |
| CPGAN [55]         | DRIVE, STARE, CHASE-DB1      | 0.977 | 0.879  | 0.912 | 0.984 | -     |
| MSCNN-AM [21]      | DRIVE, STARE, CHASE-DB1      | 0.962 | 0.830  | 0.979 | -     | 0.830 |
| MRU-NET [19]       | DRIVE, STARE                 | 0.964 | 0.825  | -     | 0.985 | 0.829 |
| M-GAN [52]         | DRIVE, STARE, HRF, CHASE-DB1 | 0.977 | 0.834  | 0.989 | 0.986 | 0.819 |
| SUD-GAN [69]       | DRIVE, STARE                 | 0.961 | 0.834  | 0.986 |       |       |
| DA-U-Net [66]      | DRIVE                        | 0.961 | 0.754  | 0.990 | -     | -     |
| BTS-DSN [24]       | DRIVE, STARE, CHASE-DB1      | 0.955 | 0.780  | 0.981 | 0.980 | 0.821 |
| DUNet [33]         | DRIVE, STARE, CHASE-DB1, HRF | -     | -      | -     | -     | -     |
| AD-UNet [45]       | DRIVE, STARE                 | 0.968 | 0.844  | 0.976 | 0.977 | 0.842 |
| CE-Net [23]        | DRIVE, STARE                 | 0.955 | -      | -     | 0.978 | -     |

## ПРИЛОЖЕНИЕ Б

(обязательное)

### ПРОГРАММНЫЙ КОД ЗАГРУЗКИ ДАННЫХ

```
import numpy as np
import os
import cv2
from PIL import Image

# Reading the training data
train_images= []
for i in range(21,41):
    train_img = cv.imread(f'/content/drive/MyDrive/BKP/DRIVE2/training/images/
{i}_training.tif')
    train_images.append(train_img)

# Reading the testing data
test_images = []
for i in range(1,21):
    test_img = cv.imread(f'/content/drive/MyDrive/BKP/DRIVE2/test/images/{i}_
test.tif')
    test_images.append(test_img)

# Reading the training manuals (manually created masks)
root = '/content/drive/MyDrive/BKP/'
train_target_data = os.path.join(root, 'DRIVE2/training/1st_manual')
train_masks = sorted(
    [
```

```

        os.path.join(train_target_data, f'{i}_manual1.gif')
        for i in range(21, 41)
    ]
)

test_target_data = os.path.join(root, 'DRIVE2/test/mask')
test_masks = (
    [
        os.path.join(test_target_data, f'{i}_manual1.gif')
        for i in range(1, 21)
    ]
)

```

```

def load_masks(masks_list):
    loaded_masks = []
    for mask_path in masks_list:
        with Image.open(mask_path) as mask_image:
            # Convert the image to grayscale
            mask_gray = mask_image.convert('L')
            # Convert to numpy array
            mask_array = np.array(mask_gray)
            # Append to the list
            loaded_masks.append(mask_array)
    return loaded_masks

```

```

# Load the masks
loaded_train_masks = load_masks(train_masks)
loaded_test_masks = load_masks(test_masks)

```

```
train_masks = np.array(loaded_train_masks)
train_masks = train_masks.reshape(-1, 584, 565, 1) # Add a channel dimension
test_masks = np.array(loaded_test_masks)
test_masks = test_masks.reshape(-1, 584, 565, 1) # Add a channel dimension
```

## ПРИЛОЖЕНИЕ В

(обязательное)

### ПРОГРАММНЫЙ КОД ПРЕДОБРАБОТКИ ДАННЫХ

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import statistics
import cv2 as cv
import os
import cv2

for dirname, _, filenames in os.walk('/content/drive/MyDrive/BKP/DRIVE2'):
    for filename in filenames:
        print(os.path.join(dirname, filename))

# Reading the training data
train_images= []
for i in range(21,41):
    train_img = cv.imread(f'/content/drive/MyDrive/BKP/DRIVE2/training/images/{i}_training.tif')
    train_images.append(train_img)

# Reading the testing data
test_images = []
for i in range(1,21):
    test_img = cv.imread(f'/content/drive/MyDrive/BKP/DRIVE2/test/images/{i}_test.tif')
```

```
test_images.append(test_img)
```

```
print(train_images[1].shape)
```

```
print(test_images[1].shape)
```

```
from PIL import Image
```

```
# Reading the training manuals (manually created masks)
```

```
root = '/content/drive/MyDrive/BKP/'
```

```
train_target_data = os.path.join(root, 'DRIVE2/training/mask')
```

```
train_masks = sorted([
```

```
    [
```

```
        os.path.join(train_target_data, f'{i}_manual1.gif')
```

```
        for i in range(21, 41) # Assuming there are 20 images in total
```

```
    ]
```

```
    )
```

```
test_target_data = os.path.join(root, 'DRIVE2/test/mask')
```

```
test_masks = (
```

```
    [
```

```
        os.path.join(test_target_data, f'{i}_manual1.gif')
```

```
        for i in range(1, 21) # Assuming there are 20 images in total
```

```
    ]
```

```
    )
```

```
# Assuming you already have the 'masks' list containing paths to the .gif images
```

```
def load_masks(masks_list):
```

```

loaded_masks = []
for mask_path in masks_list:
    with Image.open(mask_path) as mask_image:
        # Convert the image to grayscale
        mask_gray = mask_image.convert('L')
        # Convert to numpy array
        mask_array = np.array(mask_gray)
        # Append to the list
        loaded_masks.append(mask_array)
return loaded_masks

# Load the masks
loaded_train_masks = load_masks(train_masks)
loaded_test_masks = load_masks(test_masks)

train_masks = np.array(loaded_train_masks)
train_masks = train_masks.reshape(-1, 584, 565, 1) # Add a channel dimension

test_masks = np.array(loaded_test_masks)
test_masks = test_masks.reshape(-1, 584, 565, 1) # Add a channel dimension

print(train_masks[0].shape)
print(test_masks[0].shape)

# Plotting all the images from training set (DRIVE)
plt.figure(figsize=(20,4))
for i in range(0,20):

```

```

plt.subplot(2,10, i+1)
plt.imshow(cv.cvtColor(train_images[i] , cv.COLOR_BGR2RGB))
plt.axis('off')

plt.show()
plt.imshow(cv.cvtColor(train_images[0] , cv.COLOR_BGR2RGB))
print(train_images[0].shape)

def to_grayscale(train_images):
    # Grayscale
    trains_gs=[]

    for i in range(0,20):
        train_gray = cv.cvtColor(train_images[i] , cv.COLOR_BGR2GRAY)
        trains_gs.append(train_gray)

    # Display the filtered images
    plt.figure(figsize=(20,4))
    for i in range(0,20):
        plt.subplot(2,10, i+1)
        plt.imshow(trains_gs[i] , cmap='gray')
        plt.axis('off')

    plt.imshow(train_gray[0])
    plt.show()

# trains_gs = np.array(trains_gs) # Convert the list to a NumPy array
# trains_gs = trains_gs.reshape(-1, 584, 565, 1) # Add a channel dimension

```

```

# print(trains_gs[1].shape)

return trains_gs

# Grayscale
trains_gs=[]

for i in range(0,20):
    train_gray = cv.cvtColor(train_images[i] , cv.COLOR_BGR2GRAY)
    trains_gs.append(train_gray)

# Display the filtered images
plt.figure(figsize=(20,4))
for i in range(0,20):
    plt.subplot(2,10, i+1)
    plt.imshow(trains_gs[i] , cmap='gray')
    plt.axis('off')
plt.show()

plt.imshow(trains_gs[1], cmap='gray')

import cv2
import numpy as np
import matplotlib.pyplot as plt

# Define the Kirsch kernels
kirsch_kernels = [

```

```
np.array([[5, 5, 5],  
         [-3, 0, -3],  
         [-3, -3, -3]]),
```

```
np.array([[5, 5, -3],  
         [5, 0, -3],  
         [-3, -3, -3]]),
```

```
np.array([[5, -3, -3],  
         [5, 0, -3],  
         [5, -3, -3]]),
```

```
np.array([[-3, -3, -3],  
         [5, 0, -3],  
         [5, 5, -3]]),
```

```
np.array([[-3, -3, -3],  
         [-3, 0, -3],  
         [5, 5, 5]]),
```

```
np.array([[-3, -3, -3],  
         [-3, 0, 5],  
         [-3, 5, 5]]),
```

```
np.array([[-3, -3, 5],  
         [-3, 0, 5],  
         [-3, -3, 5]]),
```

```

np.array([[ -3,  5,  5],
          [ -3,  0,  5],
          [ -3, -3, -3]])
]

def apply_kirsch_filter(image):
    # Apply each kernel to the image
    kirsch_images = [cv2.filter2D(image, -1, kernel) for kernel in kirsch_kernels]

    # Take the maximum value from each direction
    kirsch_edge = np.max(np.stack(kirsch_images), axis=0)

    # Normalize the output to 0-255 range for visualization
    kirsch_edge_normalized = cv2.normalize(kirsch_edge, None, 0, 255,
cv2.NORM_MINMAX)

    return kirsch_edge_normalized

# Assuming train_images is a list of BGR images

### Convert to grayscale
trains_gs = []

for i in range(20):
    train_gray = cv2.cvtColor(train_images[i], cv2.COLOR_BGR2GRAY)
    trains_gs.append(train_gray)

# CL3 =[]

```

```

# for i in range(0,20):
#   CL3.append(trains_gs[i])

# CLAHE = cv.createCLAHE(clipLimit=3.0, tileGridSize=(8,8))

# for i in range(0,20):
#   CL3[i] = CLAHE.apply(CL3[i])

# Apply Kirsch filter to grayscale images
trains_kirsch = []

for gray_image in trains_gs:
    kirsch_filtered = apply_kirsch_filter(gray_image)
    trains_kirsch.append(kirsch_filtered)

# Display the filtered images
plt.figure(figsize=(20, 4))
for i in range(20):
    plt.subplot(2, 10, i + 1)
    plt.imshow(trains_kirsch[i], cmap='gray')
    plt.axis('off')
plt.show()

# Display one of the filtered images
plt.imshow(trains_kirsch[0], cmap='gray')
plt.axis('off')
plt.show()

```

```

CL5 =[]

for i in range(0,20):
    CL5.append(trains_gs[i])

CLAHE = cv.createCLAHE(clipLimit=10.0, tileGridSize=(8,8))

for i in range(0,20):
    CL5[i] = CLAHE.apply(CL5[i])
steps = [
    cv.cvtColor(train_images[0],cv.COLOR_BGR2RGB),
    trains_gs[0],
    CL3[0],
    trains_kirsch[0],
    CL5[0]
]

# List of titles corresponding to each image
titles = [
    "Original Image",
    "Grayscale Image",
    "CLAHE Applied",
    "Kirsch applied",
    "CLAHE applied after + binary",
]

# Plotting the images in a combined plot

```

```

plt.figure(figsize=(20, 5))
for i in range(len(steps)):
    plt.subplot(1, len(steps), i + 1)
    plt.imshow(steps[i], cmap='gray', aspect='auto', extent=(0, steps[i].shape[1],
steps[i].shape[0], 0))
    plt.title(titles[i])
    plt.axis('off')

plt.tight_layout() # Adjust subplot parameters to give specified padding.
plt.show()

```

```

CL3 =[ np.empty([584, 565])] #Size of an image?

```

```

for i in range(0,20-1):
    CL3.append(np.empty([584, 565]))

```

```

CLAHE = cv.createCLAHE(clipLimit=3.0, tileGridSize=(8,8))

```

```

for i in range(0,20):
    CL3[i] = CLAHE.apply(trains_gs[i])

```

```

plt.figure(figsize=(20,4))
for i in range(0,20):
    plt.subplot(2,10, i+1)
    plt.imshow(CL3[i] , cmap='gray')
    plt.axis('off')

```

```

plt.show()

```

## ПРИЛОЖЕНИЕ Г

(обязательное)

### ПРОГРАММНЫЙ КОД ПОСТРОЕНИЯ МОДЕЛИ НЕЙРОННОЙ СЕТИ

```
from tensorflow.keras.models import Model

from tensorflow.keras.layers import Input, Conv2D, MaxPooling2D, Dropout,
concatenate, Conv2DTranspose, BatchNormalization

from tensorflow.keras.layers import Add

def build_unet(input_shape):
    inputs = Input(input_shape)

    # Add residual connections within convolutional blocks
    def residual_block(x, filters, kernel_size=3):
        res = Conv2D(filters, kernel_size, activation='relu', padding='same',
kernel_initializer='he_normal')(x)
        res = BatchNormalization()(res)
        res = Conv2D(filters, kernel_size, activation='relu', padding='same',
kernel_initializer='he_normal')(res)
        res = BatchNormalization()(res)
        return Add()([x, res])

    def build_unet(input_shape):
        inputs = Input(input_shape)
        conv1 = Conv2D(64, 3, activation='relu', padding='same',
kernel_initializer='he_normal')(inputs)
        conv1 = residual_block(conv1, 64)
        pool1 = MaxPooling2D(pool_size=(2, 2))(conv1)
```

```

        conv2 = Conv2D(128, 3, activation='relu', padding='same',
kernel_initializer='he_normal')(pool1)
        conv2 = residual_block(conv2, 128)
        pool2 = MaxPooling2D(pool_size=(2, 2))(conv2)
        conv3 = Conv2D(256, 3, activation='relu', padding='same',
kernel_initializer='he_normal')(pool2)
        conv3 = residual_block(conv3, 256)
        pool3 = MaxPooling2D(pool_size=(2, 2))(conv3)
        conv4 = Conv2D(512, 3, activation='relu', padding='same',
kernel_initializer='he_normal')(pool3)
        conv4 = residual_block(conv4, 512)
        drop4 = Dropout(0.5)(conv4)
        pool4 = MaxPooling2D(pool_size=(2, 2))(drop4)
        conv5 = Conv2D(1024, 3, activation='relu', padding='same',
kernel_initializer='he_normal')(pool4)
        conv5 = residual_block(conv5, 1024)
        drop5 = Dropout(0.5)(conv5)
        up6 = Conv2DTranspose(512, 2, strides=(2, 2), padding='same')(drop5)
        up6 = concatenate([up6, drop4], axis=3)
        conv6 = Conv2D(512, 3, activation='relu', padding='same',
kernel_initializer='he_normal')(up6)
        conv6 = residual_block(conv6, 512)

        up7 = Conv2DTranspose(256, 2, strides=(2, 2), padding='same')(conv6)
        up7 = concatenate([up7, conv3], axis=3)
        conv7 = Conv2D(256, 3, activation='relu', padding='same',
kernel_initializer='he_normal')(up7)
        conv7 = residual_block(conv7, 256)

```

```

up8 = Conv2DTranspose(128, 2, strides=(2, 2), padding='same')(conv7)
up8 = concatenate([up8, conv2], axis=3)
    conv8 = Conv2D(128, 3, activation='relu', padding='same',
kernel_initializer='he_normal')(up8)
    conv8 = residual_block(conv8, 128)
up9 = Conv2DTranspose(64, 2, strides=(2, 2), padding='same')(conv8)
up9 = concatenate([up9, conv1], axis=3)
    conv9 = Conv2D(64, 3, activation='relu', padding='same',
kernel_initializer='he_normal')(up9)
    conv9 = residual_block(conv9, 64)
conv10 = Conv2D(1, 1, activation='sigmoid')(conv9)
model = Model(inputs=inputs, outputs=conv10)
return model
if __name__ == "__main__":
    input_shape = (512, 512, 3)
    model = build_unet(input_shape)
#model.summary()

```

**ПРИЛОЖЕНИЕ Д**  
(обязательное)  
**ПРОГРАММНЫЙ КОД МОДИФИКАЦИИ ДАННЫХ**

```
# Data augmentation
def create_dir(path):
    if not os.path.exists(path):
        os.makedirs(path)

def load_data(path):
    """ X = Images and Y = masks """

    train_x = sorted(glob(os.path.join(path, "train", "image", "*.png")))
    train_y = sorted(glob(os.path.join(path, "train", "mask", "*.png")))

    test_x = sorted(glob(os.path.join(path, "test", "image", "*.png")))
    test_y = sorted(glob(os.path.join(path, "test", "mask", "*.png")))

    return (train_x, train_y), (test_x, test_y)

# Seeding
np.random.seed(42)

# Load the data
data_path = "/content/drive/MyDrive/BKP/preprocessed_DRIVE"
(train_x, train_y), (test_x, test_y) = load_data(data_path)
```

```

print(f"Train: {len(train_x)} - {len(train_y)}")
print(f"Test: {len(test_x)} - {len(test_y)}")

# Creating directories

create_dir("new_data/train/image")
create_dir("new_data/train/mask")
create_dir("new_data/test/image")
create_dir("new_data/test/mask")

from albumentations import HorizontalFlip, VerticalFlip, Rotate, GridDistortion,
RandomCrop, RandomRotate90, RandomGridShuffle

def augment_data(images, masks, save_path, augment=True):
    H = 512
    W = 512

    for idx, (x, y) in tqdm(enumerate(zip(images, masks)), total=len(images)):
        # Extracting names
        name = x.split("/")[-1].split(".")[0]

        # Reading image and mask
        x = cv2.imread(x, cv2.IMREAD_COLOR)
        y = imageio.mimread(y)[0]

        if augment == True:
            aug = HorizontalFlip(p=1.0)

```

```
augmented = aug(image=x, mask=y)
x1 = augmented["image"]
y1 = augmented["mask"]
```

```
aug = VerticalFlip(p=1.0)
augmented = aug(image=x, mask=y)
x2 = augmented["image"]
y2 = augmented["mask"]
```

```
aug = Rotate(limit=45, p=1.0)
augmented = aug(image=x, mask=y)
x3 = augmented["image"]
y3 = augmented["mask"]
```

```
aug = GridDistortion(p=1.0)
augmented = aug(image=x, mask=y)
x4 = augmented["image"]
y4 = augmented["mask"]
```

```
aug = RandomCrop(300,300, p=0.8)
augmented = aug(image=x, mask=y)
x5 = augmented["image"]
y5 = augmented["mask"]
```

```
aug = RandomRotate90(p=1.0)
augmented = aug(image=x, mask=y)
x6 = augmented["image"]
y6 = augmented["mask"]
```

```
aug = RandomGridShuffle(p=1.0)
augmented = aug(image=x, mask=y)
x7 = augmented["image"]
y7 = augmented["mask"]
```

```
X = [x, x1, x2, x3, x4, x5, x6, x7]
Y = [y, y1, y2, y3, y4, y5, y6, y7]
```

else:

```
aug = RandomCrop(300,300, p=0.8)
augmented = aug(image=x, mask=y)
x5 = augmented["image"]
y5 = augmented["mask"]
```

```
X = [x, x5]
Y = [y, y5]
```

index = 0

for i, m in zip(X, Y):

```
i = cv2.resize(i, (W, H))
m = cv2.resize(m, (W, H))
```

if len(X) == 1:

```
tmp_image_name = f"{name}.png"
tmp_mask_name = f"{name}.png"
```

else:

```
tmp_image_name = f"{name}_{index}.png"
```

```

tmp_mask_name = f"{name}_{index}.png"

image_path = os.path.join(save_path, "image", tmp_image_name)
mask_path = os.path.join(save_path, "mask", tmp_mask_name)

cv2.imwrite(image_path, i)
cv2.imwrite(mask_path, m)

index += 1

# Data augmentation
augment_data(train_x, train_y, "new_data/train/", augment=True)
augment_data(test_x, test_y, "new_data/test/", augment=False)

def load_aug_data(path):
    x = sorted(glob(os.path.join(path, "image", "*.png")))
    y = sorted(glob(os.path.join(path, "mask", "*.png")))
    return x, y

# Dataset
dataset_path = "/content/new_data"
train_path = os.path.join(dataset_path, "train")
valid_path = os.path.join(dataset_path, "test")

train_x, train_y = load_aug_data(train_path)
valid_x, valid_y = load_aug_data(valid_path)

print(f"Train: {len(train_x)} - {len(train_y)}")

```

```
print(f"Valid: {len(valid_x)} - {len(valid_y)}")
```

## **ПРИЛОЖЕНИЕ Е**

(обязательное)

### **ПРОГРАММНЫЙ КОД ТРЕНИРОВОЧНОГО ПРОЦЕССА МОДЕЛИ**

```
from sklearn.utils import shuffle

from tensorflow.keras.callbacks import ModelCheckpoint, CSVLogger,
ReduceLROnPlateau, EarlyStopping, TensorBoard

from tensorflow.keras.optimizers import Adam

from tensorflow.keras.metrics import Recall, Precision, Accuracy

from tensorflow.keras.models import load_model


H = 512

W = 512


def create_dir(path):
    if not os.path.exists(path):
        os.makedirs(path)


def load_data(path):
    x = sorted(glob(os.path.join(path, "image", "*.png")))
    y = sorted(glob(os.path.join(path, "mask", "*.png")))
    return x, y


def shuffling(x, y):
    x, y = shuffle(x, y, random_state=42)
    return x, y
```

```

def read_image(path):
    path = path.decode()
    x = cv2.imread(path, cv2.IMREAD_COLOR)
    # x = cv2.resize(x, (W, H))
    x = x/255.0
    x = x.astype(np.float32)
    return x

def read_mask(path):
    path = path.decode()
    x = cv2.imread(path, cv2.IMREAD_GRAYSCALE) ## (512, 512)
    # x = cv2.resize(x, (W, H))
    x = x/255.0
    x = x.astype(np.float32)
    x = np.expand_dims(x, axis=-1)          ## (512, 512, 1)
    return x

def tf_parse(x, y):
    def _parse(x, y):
        x = read_image(x)
        y = read_mask(y)
        return x, y

    x, y = tf.numpy_function(_parse, [x, y], [tf.float32, tf.float32])
    x.set_shape([H, W, 3])
    y.set_shape([H, W, 1])
    return x, y

```

```
def tf_dataset(X, Y, batch_size=2):  
    dataset = tf.data.Dataset.from_tensor_slices((X, Y))  
    dataset = dataset.map(tf_parse)  
    dataset = dataset.batch(batch_size)  
    dataset = dataset.prefetch(4)  
    return dataset
```

```
if __name__ == "__main__":
```

```
    # Seeding
```

```
    np.random.seed(42)
```

```
    tf.random.set_seed(42)
```

```
    # Directory to save files
```

```
    create_dir("files")
```

```
    # Hyperparameters
```

```
    batch_size = 2
```

```
    lr = 1e-4
```

```
    num_epochs = 50
```

```
    model_path = os.path.join("files", "model.h5")
```

```
    csv_path = os.path.join("files", "data2.csv")
```

```
    # Dataset
```

```
    dataset_path = "new_data"
```

```
    train_path = os.path.join(dataset_path, "train")
```

```
    valid_path = os.path.join(dataset_path, "test")
```

```

train_x, train_y = load_data(train_path)
train_x, train_y = shuffling(train_x, train_y)
valid_x, valid_y = load_data(valid_path)

print(f"Train: {len(train_x)} - {len(train_y)}")
print(f"Valid: {len(valid_x)} - {len(valid_y)}")

train_dataset = tf_dataset(train_x, train_y, batch_size=batch_size)
valid_dataset = tf_dataset(valid_x, valid_y, batch_size=batch_size)

train_steps = len(train_x)//batch_size
valid_setps = len(valid_x)//batch_size

if len(train_x) % batch_size != 0:
    train_steps += 1
if len(valid_x) % batch_size != 0:
    valid_setps += 1

# Model
# For new training
model = build_unet((H, W, 3))
model.compile(loss=dice_loss, optimizer=Adam(lr), metrics=[dice_coef, iou,
Recall(), Precision(), Accuracy()])
# model.summary()

callbacks = [
    ModelCheckpoint(model_path, verbose=1, save_best_only=True),

```

```

        ReduceLROnPlateau(monitor="val_loss", factor=0.1, patience=5, min_lr=1e-
6, verbose=1),
        CSVLogger(csv_path),
        TensorBoard(),
        #EarlyStopping(monitor="val_loss", patience=10,
restore_best_weights=False)
    ]
    model.fit(
        train_dataset,
        epochs=num_epochs,
        validation_data=valid_dataset,
        steps_per_epoch=train_steps,
        validation_steps=valid_setps,
        callbacks=callbacks
    )

```

## ПРИЛОЖЕНИЕ Ж

(обязательное)

### ПРОГРАММНЫЙ КОД ОЦЕНКИ ЭФФЕКТИВНОСТИ МОДЕЛИ

```
import os
os.environ["TF_CPP_MIN_LOG_LEVEL"] = "2"
import numpy as np
import pandas as pd
import cv2
from glob import glob
from tqdm import tqdm
import tensorflow as tf
from tensorflow.keras.utils import CustomObjectScope
from sklearn.metrics import accuracy_score, f1_score, jaccard_score, precision_score,
recall_score, roc_auc_score

H = 512
W = 512

def create_dir(path):
    if not os.path.exists(path):
        os.makedirs(path)

def read_image(path):
    x = cv2.imread(path, cv2.IMREAD_COLOR)
    ori_x = x
    x = x / 255.0
    x = x.astype(np.float32)
    return ori_x, x

def read_mask(path):
    x = cv2.imread(path, cv2.IMREAD_GRAYSCALE)
    ori_x = x
    x = x / 255.0
    x = x.astype(np.int32)
```

```

return ori_x, x

def load_data(path):
    x = sorted(glob(os.path.join(path, "image", "*.png")))
    y = sorted(glob(os.path.join(path, "mask", "*.png")))
    return x, y

def save_results(ori_x, ori_y, y_pred, save_image_path):
    line = np.ones((H, 10, 3)) * 255

    ori_y = np.expand_dims(ori_y, axis=-1)
    ori_y = np.concatenate([ori_y, ori_y, ori_y], axis=-1)

    y_pred = np.expand_dims(y_pred, axis=-1)
    y_pred = np.concatenate([y_pred, y_pred, y_pred], axis=-1) * 255

    cat_images = np.concatenate([ori_x, line, ori_y, line, y_pred], axis=1)
    cv2.imwrite(save_image_path, cat_images)

def dice_coef(y_true, y_pred, smooth=1e-7):
    y_true_f = y_true.flatten()
    y_pred_f = y_pred.flatten()
    intersection = np.sum(y_true_f * y_pred_f)
    return (2. * intersection + smooth) / (np.sum(y_true_f) + np.sum(y_pred_f) + smooth)

if __name__ == "__main__":

    # Save the results in this folder
    create_dir("results")

    # Load the model
    with CustomObjectScope({'iou': iou, 'dice_coef': dice_coef, 'dice_loss': dice_loss}):
        model = tf.keras.models.load_model("files/model.h5")

    # Load the dataset
    dataset_path = os.path.join("new_data", "test")

```

```

test_x, test_y = load_data(dataset_path)
print(len(test_x), "-", len(test_y))

# Make the prediction and calculate the metrics values
SCORE = []
for x, y in tqdm(zip(test_x, test_y), total=len(test_x)):

    # Extracting name
    name = x.split("/")[-1].split(".")[0]

    # Read the image and mask
    ori_x, x = read_image(x)
    ori_y, y = read_mask(y)

    # Prediction
    y_pred = model.predict(np.expand_dims(x, axis=0))[0]
    y_pred = y_pred > 0.5
    y_pred = y_pred.astype(np.int32)
    y_pred = np.squeeze(y_pred, axis=-1)

    # Saving the images
    save_image_path = f"results/{name}.png"
    save_results(ori_x, ori_y, y_pred, save_image_path)

    # Flatten the array
    y = y.flatten()
    y_pred = y_pred.flatten()

    # Calculate the metrics
    acc_value = accuracy_score(y, y_pred)
    f1_value = f1_score(y, y_pred, labels=[0, 1], average="binary")
    #jac_value = jaccard_score(y, y_pred, labels=[0, 1], average="binary")
    recall_value = recall_score(y, y_pred, labels=[0, 1], average="binary")
    precision_value = precision_score(y, y_pred, labels=[0, 1], average="binary")
    auc_value = roc_auc_score(y, y_pred)
    dice_value = dice_coef(y, y_pred)

```

```

SCORE.append([name, acc_value, f1_value, jac_value, recall_value,
precision_value, auc_value, dice_value])

score = [s[1:] for s in SCORE]
score = np.mean(score, axis=0)
print(f"Accuracy: {score[0]:0.5f}")
print(f"F1: {score[1]:0.5f}")
print(f"Jaccard: {score[2]:0.5f}")
print(f"Recall: {score[3]:0.5f}")
print(f"Precision: {score[4]:0.5f}")
print(f"AUC: {score[5]:0.5f}")
print(f"Dice Coefficient: {score[6]:0.5f}")

# Saving
#df = pd.DataFrame(SCORE, columns=["Image", "Acc", "F1", "Jaccard", "Recall",
"Precision", "AUC", "Dice"])
df = pd.DataFrame(SCORE, columns=["Image", "Acc", "F1-score", "Recall",
"Precision", "AUC", "Dice"])
df.to_csv("files/score.csv")

```