

Министерство образования Республики Беларусь
Учреждение образования
Белорусский государственный университет
информатики и радиоэлектроники

УДК 616.831-079.2:616.073.756.8

Джанибеков
Хамзат Шарофиддинович

Алгоритм применения углубленного обучения в диагностике
нейроонкологических заболеваний

АВТОРЕФЕРАТ

на соискание степени магистра

по специальности 1-39 80 03 «Электронные системы и технологии»

(подпись магистранта)

Научный руководитель

Чураков Андрей Владимирович

(фамилия, имя, отчество)

Кандидат медицинских наук, доцент

(ученая степень, ученое звание)

(подпись научного руководителя)

Минск 2024

КРАТКОЕ ВВЕДЕНИЕ

В данной диссертации представлены результаты исследований, направленных на улучшение методов цифровой нейровизуализации снимков головного мозга. Работа основана на анализе существующих подходов к обработке и анализу медицинских изображений, а также на разработке новых методов и алгоритмов, учитывающих специфику структуры и функции мозга.

Основной целью данного исследования является повышение точности и эффективности диагностики и лечения заболеваний головного мозга с использованием современных цифровых технологий. Для достижения этой цели в работе рассматриваются такие аспекты, как автоматическое выявление аномалий на изображениях, сегментация различных областей мозга, визуализация новообразований с целью дальнейшего диагностирования их злокачественности.

Главным условием является использование углубленного обучения нейронных сетей для цифровой нейровизуализации снимков головного мозга. Исследования в этой области направлены на разработку и адаптацию алгоритмов обучения нейронных сетей, специально адаптированных для работы с медицинскими изображениями. Это включает в себя создание нейронных архитектур, способных автоматически обнаруживать и сегментировать патологические изменения, а также генерировать дополнительную информацию о структуре мозга на основе имеющихся данных.

Практическая значимость исследований по применению нейронных сетей в нейровизуализации заключается в возможности улучшения скорости и точности диагностики различных заболеваний головного мозга. Разработанные алгоритмы могут помочь врачам в выявлении патологий на ранних стадиях, что позволит своевременно начать лечение и улучшит прогнозы для пациентов. Кроме того, использование нейронных сетей в процессе нейровизуализации может существенно сократить время, затрачиваемое на анализ больших объемов медицинских данных, что особенно важно в условиях современной медицинской практики.

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Научное и тематическое обоснование выбора диссертационной работы

Тема диссертационной работы утверждена приказом ректора учреждения образования «Белорусский государственный университет информатики и радиоэлектроники» (приказ от 03.10.2022 г.) и соответствует разделу 2 «Биологические, медицинские, фармацевтические и химические технологии и производства» перечня приоритетных направлений фундаментальных и прикладных научных исследований Республики Беларусь на 2021-2025 годы (Постановление Совета Министров Республики Беларусь от 7 мая 2020 г. №156).

Цель и задачи исследования

Целью диссертационной работы является обучение нейронной сети обнаружению и сегментированию новообразований головного мозга.

Для достижения поставленной цели потребовалось решить следующие задачи:

- провести литературный обзор на тему «Алгоритм применения углубленного обучения в диагностике нейроонкологических заболеваний»;
- провести экспериментальные работы и тесты по обучению нейронной сети определению локализации новообразования и его последующей сегментации;
- разработать программное средство, определяющее новообразование и сегментирующее его по каждому из срезов.

Научная новизна и значимость полученных результатов

Теоретически установлено и экспериментально подтверждено, что благодаря применению эффективного этапа предварительной обработки, удаляющего порядка 80% неинформативных данных из каждого входного изображения, нет необходимости в использовании сложной глубокой сети. Отбирая 20% данных всего изображения для каждой из четырех входных модальностей и соответствующего нормализованного изображения, для улучшения оценки результатов сегментации количество пикселей, требующих анализа – сокращается.

Положения, выносимые на защиту

Методика использования углубленного обучения в диагностике

нейроонкологических заболеваний.

Программное средство в виде обученной нейронной сети.

Апробация результатов диссертации

Результаты диссертационной работы были представлены на следующих конференциях:

– 59-я научная конференция аспирантов, магистрантов и студентов «Электронные системы и технологии», Минск, 17-21 апреля 2023 года;

– 60-я научная конференция аспирантов, магистрантов и студентов «Электронные системы и технологии», Минск (принято к публикации).

Опубликованность результатов диссертации

По материалам диссертации опубликовано 3 статьи в сборниках научных статей и материалов конференций, в том числе 1 статья в научном журнале, рекомендованном ВАК для опубликования результатов диссертационных исследований. Без соавторов опубликовано 2 статьи в научных журналах.

Структура и объём диссертации

Диссертация состоит из введения, общей характеристики работы, трёх глав с краткими выводами по каждой из глав, заключения и списка использованной литературы. Общий объём диссертационной работы составляет 61 страниц, 27 иллюстраций на 21 странице, 2 таблиц на 2 страницах, список использованной литературы на 25 наименований на 2 страницах.

ОБЗОР ЛИТЕРАТУРЫ ПО ТЕМЕ ДИССЕРТАЦИОННОЙ РАБОТЫ

Диагностика опухолей головного мозга по МР-изображениям с использованием усиленного многоградиентного машинного классификатора опорных векторов

Предложенный подход был реализован в виде шагов, перечисленных ниже, каждый из которых включает исчерпывающую информацию о всей процедуре. Сбор изображений МРТ, анизотропная фильтрация и предварительная обработка АНЕ, сегментация изображений на основе EFO-OTSU, извлечение признаков на основе PCA и DWT, а также классификация BMG-SVM составляют следующие пять этапов стратегии. На рисунке 1 и 2 изображены: эволюция предложенной стратегии; образцы МР-изображений четырех классов [4].

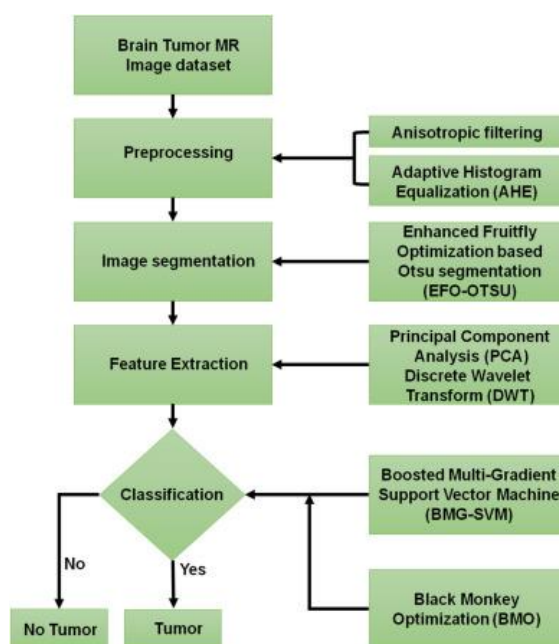


Рисунок 1 – Ход предлагаемой техники

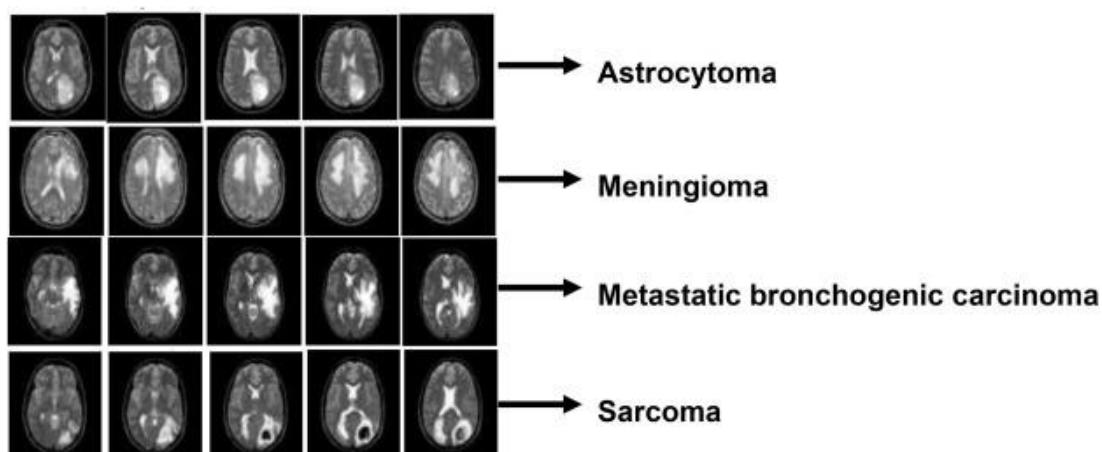


Рисунок 2 – Пять образцов МР-изображений четырех классов

Отзыв – это вероятность положительного результата теста при условии, что он положительный. Это часто называют реальным положительным показателем. Он показывает, что запоминание предлагаемого подхода лучше по сравнению с запоминанием старых методов. Чувствительность предлагаемой стратегии к различным факторам сравнивается с традиционными подходами на следующем рисунке 3 [4].

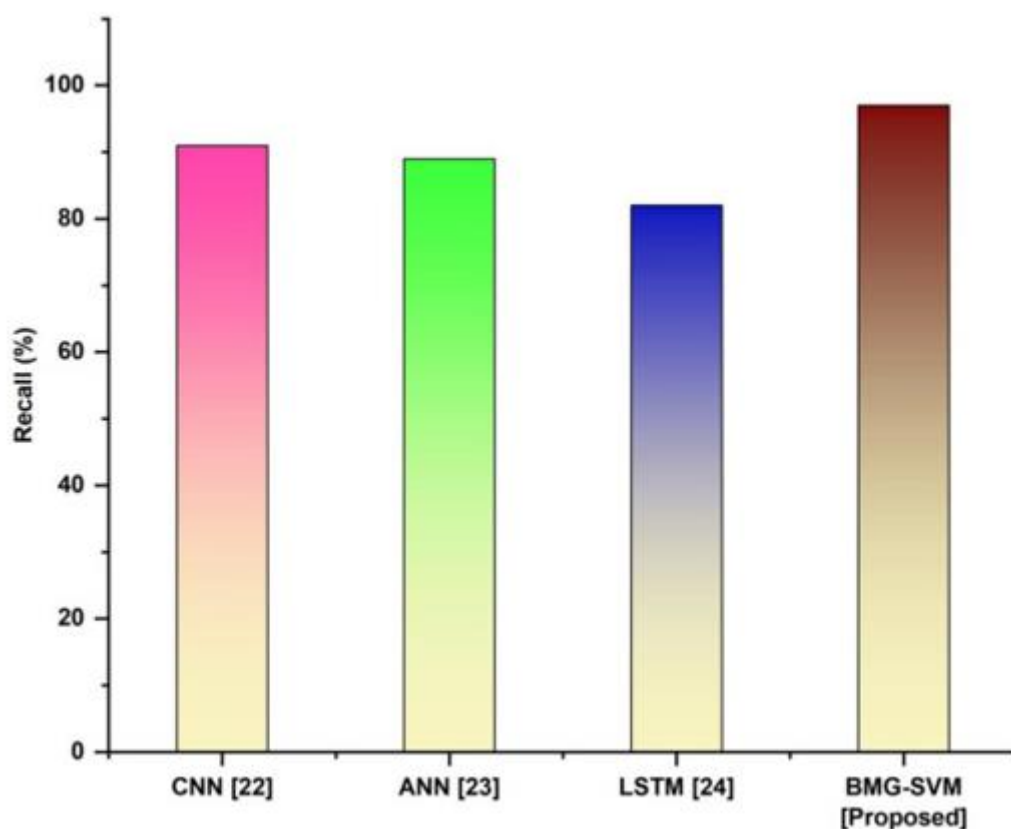


Рисунок 3 – Чувствительность предлагаемой стратегии к различным факторам

Диагностика опухолей головного мозга с помощью МРТ на основе Mobilenetv2, оптимизированного с помощью алгоритма оптимизации контрактной лисы

После применения повышения контрастности к отдельным плиткам они впоследствии объединяются для восстановления окончательного изображения. Выбор метода слияния зависит от конкретной реализации (рисунок 4) [5].

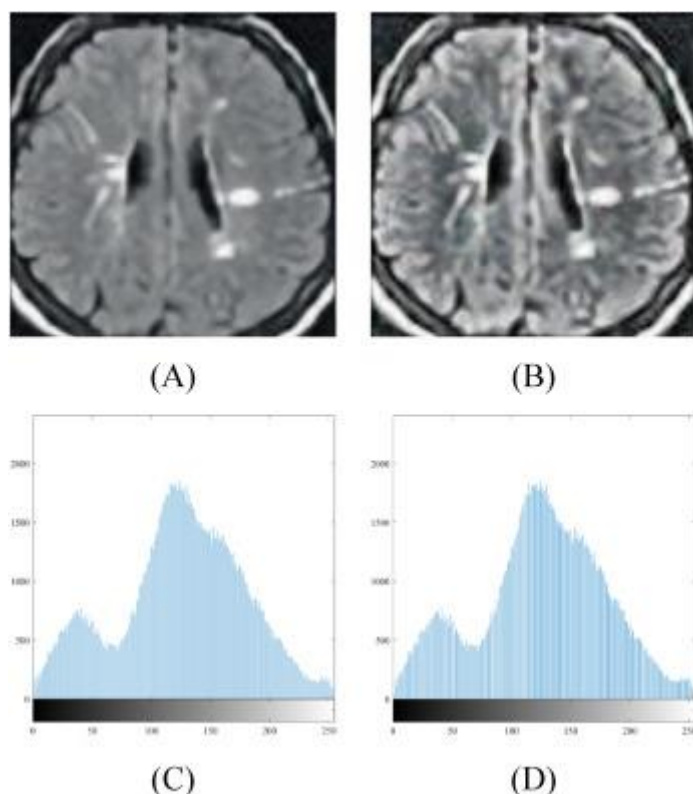


Рисунок 4 – CLAHE применительно к МРТ опухоли головного мозга: А – входное изображение; В – изображение после CLAHE; С – гистограмма А; D – гистограмма В.

Путем сравнительного анализа исходного входного изображения (А) и впоследствии улучшенного изображения (В) становится очевидным заметное повышение контрастности и увеличение деталей изображения. Гистограммы, изображенные на рисунках (С) и (D), дают графическое изображение изменений в распределении интенсивности до и после внедрения адаптивного выравнивания гистограмм с ограниченным контрастом (CLAHE) соответственно. Эффективность адаптивного выравнивания гистограмм с ограниченным контрастом (CLAHE) в улучшении видимости и качества сканирований магнитно-резонансной томографии (МРТ) опухолей головного мозга иллюстрируется улучшенным изображением и соответствующей ему

гистограммой [5].

Нормализация `min-max`, также называемая масштабированием признаков или нормализацией данных, представляет собой метод, используемый для настройки масштаба числовых данных до заранее определенного диапазона. Основная цель нормализации – стандартизировать масштаб всех функций или переменных в наборе данных, тем самым способствуя улучшению возможностей анализа и моделирования [5].

После реализации нормализации `min-max` значения переменной претерпят преобразование, соответствующее заранее определенному диапазону. Минимальному значению переменной будет присвоено сопоставление 0, максимальному значению будет присвоено сопоставление 1, а все остальные значения масштабируются пропорционально между этими двумя крайними значениями. Рисунок 5 иллюстрирует влияние нормализации `min-max` на изображения МРТ головного мозга: (А) исходное изображение и (В) изображение после нормализации `min-max` [5].

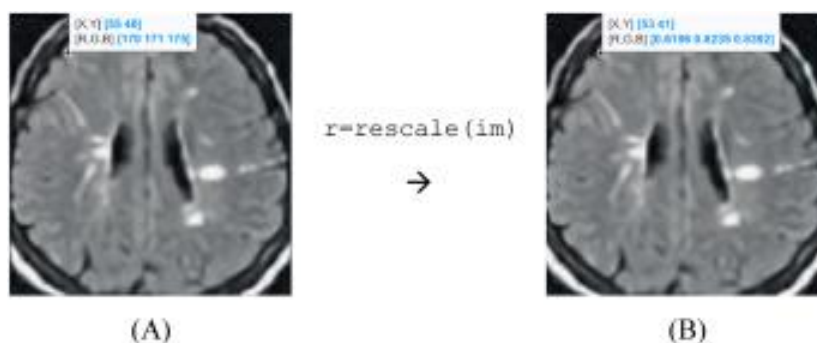
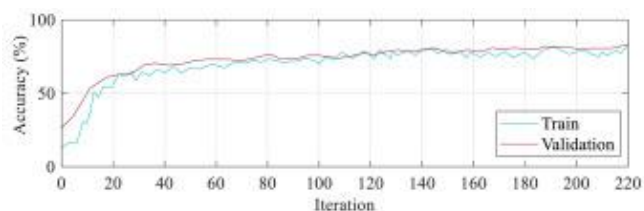
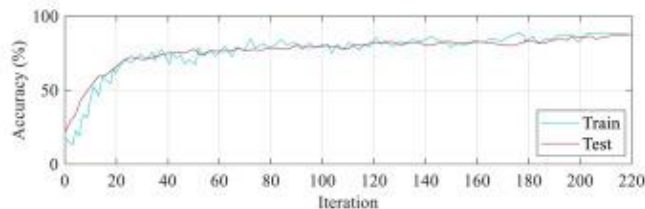


Рисунок 5 – Влияние нормализации `min-max` на изображения МРТ головного мозга

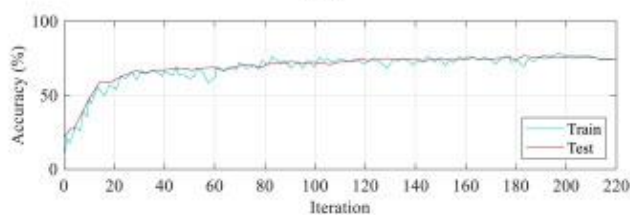
Рисунок 6 позволяет анализировать такие важные факторы, как скорость сходимости, переобучение и стабильность модели. В случае RN на графике показано постепенное уменьшение ошибки обучения, хотя и с более медленной сходимостью по сравнению с другими моделями. Тем не менее, ошибка проверки также уменьшается, что указывает на успешное обобщение невидимых данных. WT/DL демонстрирует быструю сходимость как ошибок обучения, так и ошибок проверки, что предполагает эффективное обучение, а небольшой разрыв между кривыми указывает на хорошие возможности обобщения. VGG19 демонстрирует последовательное обучение посредством постоянного снижения ошибок обучения, но существует заметный разрыв между ошибками обучения и проверки, что предполагает потенциальное переобучение или ограниченное обобщение [5].



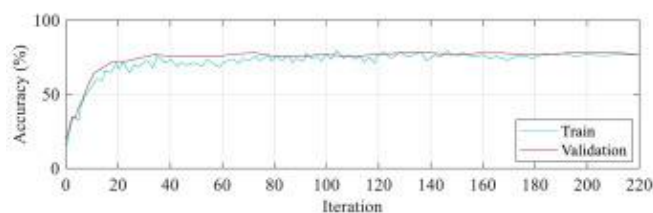
(A)



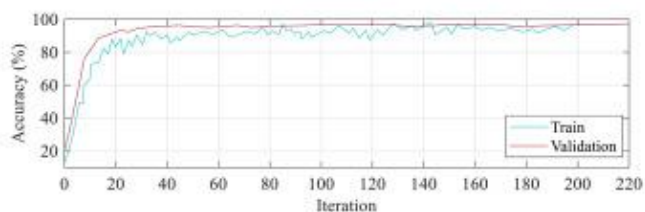
(B)



(C)



(D)



(E)

Рисунок 6 – Профиль обучения и тестирования для: (A) RN (B) WT/DL (C) VGG19 (D) CNN и (E) MN-V2/CFO

CNN демонстрирует быструю начальную сходимость, за которой следует небольшое увеличение ошибки теста, что указывает на возможное переобучение и снижение производительности на невидимых образцах. С другой стороны, MN-V2/CFO демонстрирует плавное уменьшение ошибок обучения и проверки, что означает эффективное обучение и обобщение, а небольшой разрыв между кривыми означает стабильную производительность в разных наборах данных [5].

ЭКСПЕРИМЕНТАЛЬНЫЙ РАЗДЕЛ

Подготовка данных – это ключевой этап в разработке систем компьютерного зрения, поскольку качество и разнообразие данных напрямую влияют на производительность и обобщающую способность модели.

В описание данного шага представлены следующие основные этапы:

1. Сбор данных:

- Определение целей задачи и необходимых данных для ее решения;
- Сбор разнообразных данных, которые покрывают различные условия, сценарии и вариации объектов интереса;
- Внимание на разнообразие данных, чтобы модель обучалась на реалистичных примерах и могла обобщаться на новые данные.

2. Аннотация данных:

- Разметка данных с помощью экспертов или специализированных программ для создания меток, соответствующих задаче компьютерного зрения (например, разметка объектов, сегментация, классификация);
- Обозначение и классификация объектов на изображениях, а также определение их атрибутов или характеристик;
- В случае с сегментацией опухолей головного мозга, аннотации могут включать обозначение границ опухолей и их местоположение на изображении.

3. Предобработка данных:

- Масштабирование и нормализация изображений для обеспечения стабильности и лучшей сходимости во время обучения;
- Удаление шума или артефактов с изображений, если это необходимо;
- Выравнивание разрешения изображений и стандартизация их формата для удобства обработки.

4. Разделение на обучающий, валидационный и тестовый наборы:

- Разделение данных на три отдельных набора для обучения, валидации и тестирования модели;
- Обычно используется соотношение примерно 60-80% для обучения, 10-20% для валидации и 10-20% для тестирования, но это может варьироваться в зависимости от размера и характера данных.

5. Расширение данных (опционально):

- Применение методов аугментации данных для увеличения разнообразия обучающего набора и улучшения обобщающей способности модели;
- Аугментация может включать в себя повороты, масштабирование, отражение, изменение яркости и другие преобразования.

6. Проверка качества данных:

- Проведение анализа качества данных для обнаружения ошибок или несоответствий в разметке;
- Достоверность данных, представляющих собой адекватное представление реального мира и включающих в себя разнообразие сценариев, с которыми модель может столкнуться в процессе работы.

Описание алгоритма модели обучения нейронной сети

При создании программного средства использовался язык программирования Python. При разработке модели обучения нейронной сети использовались следующие библиотеки:

1. Matplotlib – это библиотека для создания графиков и визуализации данных в языке программирования Python. Она широко используется для построения статических, анимированных и интерактивных графиков, диаграмм, гистограмм, точечных графиков и других типов визуальных представлений данных. Благодаря своей гибкости, простоте в использовании и мощным возможностям, Matplotlib является одной из наиболее популярных библиотек в Python для визуализации данных и создания графических представлений [9];

2. Nilearn – это библиотека, предназначенная для анализа и визуализации нейрообразовательных данных. Она предоставляет широкий набор инструментов для работы с данными изображений головного мозга, включая функции для загрузки, предварительной обработки, статистического анализа и визуализации таких данных [10];

3. Torchsummary – это инструмент для быстрого обзора архитектуры нейронных сетей, построенных с использованием библиотеки PyTorch. Она предоставляет простой и удобный способ получения сводки о модели, включая информацию о размерах входных и выходных тензоров на каждом слое, количество параметров и объем памяти, используемой для хранения параметров модели [11];

4. Nibabel – это инструмент для работы с файлами медицинских изображений в формате NIfTI. NIfTI (Neuroimaging Informatics Technology Initiative) – это формат файла для хранения медицинских изображений, используемых в нейроимиджинге и нейронауке. Этот формат был разработан и предложен инициативой Neuroimaging Informatics Technology Initiative (NIfTI) в целях стандартизации и улучшения обмена и анализа данных нейроимиджинга. Библиотека nibabel предоставляет удобный интерфейс для чтения, записи и манипуляций с данными, содержащимися в файлах NIfTI и их производных, таких как ANALYZE, NIfTI-1, NIfTI-2 и CIFTI-2 [12];

5. TQDM (от "taqadum", что на арабском языке означает "прогресс") – это библиотека для создания прогресс-баров в Python. Она предоставляет

пользователю возможность визуализировать прогресс выполнения итерируемых операций, что упрощает отслеживание работы кода, особенно в случае долгих или медленных операций [13];

6. NumPy (Numerical Python) – это библиотека для работы с многомерными массивами и матрицами, предоставляющая широкие возможности для работы с числовыми данными в Python [14];

7. OpenCV (Open Source Computer Vision Library) – это библиотека с открытым исходным кодом, разработанная для обработки изображений и компьютерного зрения. Она предоставляет широкий спектр функций для работы с изображениями и видео, включая чтение и запись файлов изображений и видео, обработку изображений (например, фильтрацию, сегментацию, детекцию объектов), а также различные инструменты для работы с камерами и потоками видео [15];

8. PyTorch – это библиотека машинного обучения с открытым исходным кодом, разработанная на платформе Python и предоставляемая Facebook. Она предоставляет гибкие инструменты для создания и обучения глубоких нейронных сетей [16];

9. Torchvision – это популярная библиотека, разработанная на основе PyTorch, предназначенная для обработки изображений и работы с нейросетевыми моделями компьютерного зрения. Она предоставляет различные инструменты и утилиты для работы с наборами данных, предварительно обученными моделями, а также для создания и обучения собственных моделей глубокого обучения [17];

10. TensorBoard – это инструмент визуализации и отслеживания для библиотеки машинного обучения TensorFlow. Он предоставляет набор инструментов для мониторинга и анализа процесса обучения нейронных сетей [18];

11. MedPy – это библиотека, разработанная для медицинского образования и анализа медицинских изображений. Она предоставляет инструменты для работы с изображениями и объемными данными, такими как сканирование мозга, изображения снимков СТ и MRI, медицинские снимки PET и другие [19];

12. Scikit-image – это библиотека, предназначенная для обработки и анализа изображений. Она предоставляет множество функций и инструментов для выполнения различных операций с изображениями, включая фильтрацию, преобразования, сегментацию, измерения, визуализацию и многое другое [20];

13. SciPy – это библиотека для научных вычислений и технических вычислений на языке программирования Python. Она предоставляет множество функций и инструментов для работы с различными задачами, такими как численное интегрирование, оптимизация, решение

дифференциальных уравнений, обработка сигналов, обработка изображений, статистический анализ и многое другое [21];

14. Gdown – это инструмент командной строки и библиотека Python, предназначенная для загрузки файлов с Google Drive через их идентификаторы или URL-адреса. Он облегчает загрузку больших файлов с помощью программного интерфейса Google Drive [22];

15. Names – предоставляет инструменты для генерации случайных имён человека. Она может быть полезна при создании тестовых данных, написании скриптов или приложений, где требуется генерация фиктивных персоналий [23];

16. PySide6 – это официальная привязка Qt для языка программирования Python. Она позволяет разработчикам создавать кроссплатформенные приложения с графическим интерфейсом пользователя (GUI) на основе библиотеки Qt, используя синтаксис Python [24];

17. Plotly – это библиотека для создания интерактивных графиков и визуализаций данных. Она предоставляет различные инструменты и API для создания красивых и интерактивных графиков на основе данных, используя языки программирования Python, R и JavaScript [25].

Описание алгоритма реализации тренировки и валидации модели

Класс TrainerArgs

Класс TrainerArgs представляет собой простой контейнер для хранения аргументов, необходимых для настройки процесса тренировки модели. В данном контексте, его целью является удобное хранение и передача параметров в объекты, которые управляют тренировкой модели. Данный класс выглядит следующим образом:

```
class TrainerArgs:
    def __init__(self, n_epochs=50, device="cpu", output_path="",
                 loss="dice"):
        self.n_epochs = n_epochs
        self.device = device
        self.output_path = output_path
        self.loss = loss
```

Таким образом, класс TrainerArgs облегчает организацию и передачу аргументов в процессе тренировки модели, улучшая читаемость и поддерживаемость кода.

Класс Trainer

Класс `Trainer` представляет собой ключевой компонент для обучения модели машинного обучения. Он обычно используется для управления циклом обучения, включая передачу данных модели, вычисление потерь, обновление параметров модели и оценку производительности модели на обучающем и проверочном наборах данных. Данный класс выглядит следующим образом:

```
class Trainer:

    def __init__(self, args, model, optimizer, criterion,
start_epoch, train_loader, val_loader, lr_scheduler, writer):
        self.model = model
        self.optimizer = optimizer
        self.criterion = criterion

        self.train_data_loader = train_loader
        self.number_train_data = len(self.train_data_loader)

        self.valid_data_loader = val_loader
        self.number_val_data = len(self.valid_data_loader)

        self.lr_scheduler = lr_scheduler
        self.writer = writer

        self.start_epoch = start_epoch
        self.args = args

    def start(self, best_loss=1000):
        val_dice_score = 0

        for epoch in range(self.start_epoch, self.args.n_epochs):

            train_dice_loss, train_dice_score,
train_combined_loss, train_ce_loss = self.train_epoch(epoch)
            val_dice_loss, val_dice_score, val_combined_loss,
val_ce_loss = self.val_epoch(epoch)

            val_loss = val_combined_loss if self.args.loss ==
"combined" else val_dice_loss
            if self.lr_scheduler:
                self.lr_scheduler.step(val_loss)

            self._epoch_summary(epoch, train_dice_loss,
val_dice_loss, train_dice_score, val_dice_score,
train_combined_loss,
train_ce_loss, val_combined_loss, val_ce_loss)
```

```

is_best = bool(val_loss < best_loss)
best_loss = val_loss if is_best else best_loss
save_checkpoint({
    'epoch': epoch,
    'model_state_dict': self.model.state_dict(),
    'optimizer_state_dict':
self.optimizer.state_dict(),
    'val_loss': best_loss,
    'val_dice_score': val_dice_score
}, is_best, self.args.output_path)

save_model({
    'epoch': self.args.n_epochs + 1,
    'model_state_dict': self.model.state_dict(),
    'optimizer_state_dict': self.optimizer.state_dict(),
    'val_loss': best_loss,
    'val_dice_score': val_dice_score
}, self.args.output_path)

```

Таблица 1 – Сравнение методов сегментации медицинских изображений по метрикам DICE и HAUSDORFF

Метод	DICE WT	DICE TC	DICE ET	HAUSDORFF WT	HAUSDORFF TC	HAUSDORFF ET
Basic 3D-UNet	0,8411	0,7906	0,6876	13,3658	13,6065	50,9828
Basic 3D-UNet +post	0,8052	0,7749	0,6742	13,0969	14,0047	43,8928
Residual 3D- UNet	0,8072	0,7740	0,6955	16,9635	17,5142	39,9172
Residual 3D- UNet + post	0,8142	0,7748	0,7119	11,8505	18,8146	34,9652
Residual 3D- UNet- multiscale	0,8172	0,7664	0,7071	15,5342	13,9380	38,6098
Residual 3D- UNet- multiscale + post	0,8246	0,7647	0,7163	12,3372	13,1045	37,4224

Таблица 2 – Сравнение методов сегментации с использованием различных метрик неопределенности для Residual 3D-UNet-multiscale

MEASURE	METHOD	AUC DICE WT	AUC DICE TC	AUC DICE ET	FTP RATIO WT	FTP RATIO TC	FTP RATIO ET	FTN RATIO WT	FTM RATIO TC	FTN RATIO ET
Variance	TTA Residual 3D- UNet- multiscale	0,8316	0,7715	0,7088	0,0538	0,0449	0,0380	0,0009	0,0002	0,0001

Продолжение таблицы 2

Variance	TTD Residual 3D- UNet- multiscal e	0,8300	0,7582	0,7318	0,1646	0,1558	0,0937	0,0024	0,0015	0,0004
Variance	TTA + TTD Residual 3D- UNet- multiscal	0,8325	0,7632	0,7276	0,1812	0,1588	0,0998	0,0036	0,0020	0,0005

продолжение таблицы 3.2

Entropy	TTA Residual 3D- UNet- multiscal e	0,8326	0,7816	0,7138	0,0635	0,0476	0,0362	0,0011	0,0047	0,0063
Entropy	TTD Residual 3D- UNet- multiscal e	0,8233	0,7797	0,7423	0,1512	0,1285	0,0698	0,0021	0,0082	0,0122

ЗАКЛЮЧЕНИЕ

В ходе проведенных исследований и написания данной работы было показано, что применение технологий углубленного обучения в МРТ диагностике опухолей головного мозга является перспективным подходом, который может значительно улучшить точность и эффективность диагностики; представлены результаты исследований, направленных на улучшение методов цифровой нейровизуализации снимков головного мозга.

Достигнута автоматизация в определении локализации и сегментировании новообразований головного мозга. В ходе достижения поставленной цели были решены такие задачи как проведение обзора по теме работы, сбор и представление теоретической информации, изучение необходимых библиотек для написания алгоритма, написание алгоритма, проведены экспериментальные работы и тесты по обучению нейронной сети, получены и зафиксированы результаты, достигнуто автоматическое выявление аномалий на изображениях, сегментация различных областей мозга, визуализация новообразований. Так же были представлены методы дифференциальной диагностики при многоочаговом поражении ЦНС, стандартный протокол исследований опухолей мозга, выбор наклона плоскости аксиальных изображений, показаны различные патологии, описаны основные задачи компьютерного зрения, представлены основные этапы подготовки данных для обучения нейросети, пошагово описана непосредственно подготовка данных для обучения и представлены библиотеки, используемые при написании алгоритма, предоставление показаний результатов оценки качества сегментации медицинских изображений и эффективности которого так же затронуло несколько отдельных методов сегментации и их метрик для различных классов и/или типов тканей.

Поставленная цель достигнута, задачи выполнены в полной мере.

Список публикаций соискателя

[1] Джанибеков, Х. Ш. Применение технологии углубленного обучения в МРТ диагностике опухолей головного мозга / Джанибеков Х. Ш. // Электронные системы и технологии [Электронный ресурс]: сборник материалов 59-й научной конференции аспирантов, магистрантов и студентов БГУИР, Минск, 17–21 апреля 2023 г. / Белорусский государственный университет информатики и радиоэлектроники ; редкол.: Д. В. Лихаческий [и др.]. – Минск, 2023. – С. 554–556. – Режим доступа : https://www.bsuir.by/m/12_100229_1_173624.pdf

[2] Джанибеков Х., Чураков А., Онгарбаева А., Науменко Д., Шульгина И., Лопатов П. Нейровизуализация интракраниальных новообразований: объемная сегментация трехмерных изображений. Наука и инновации. 2023;1(12):73-79. <https://doi.org/10.29235/1818-9857-2023-12-73-79>

[3] Джанибеков, Х. Ш. Сегментация опухоли головного мозга на основе глубокого обучения с использованием мультимодальных изображений мозга МРТ / Джанибеков Х. Ш. // Электронные системы и технологии [Электронный ресурс]: сборник материалов 60-й научной конференции аспирантов, магистрантов и студентов БГУИР, Минск, 17–21 апреля 2023 г. / Белорусский государственный университет информатики и радиоэлектроники ; редкол.: Д. В. Лихаческий [и др.]. – Минск, 2024 (принята в электронный сборник)