

Сравнение эффективности применения различных подходов в задаче детекции объекта на изображении низкого качества

А.А. Проворова¹, И.Ю. Полякова², Е.В. Кузьмичева³

Центр когнитивных нейронаук, Национальный исследовательский университет «Высшая школа экономики», Пермь, Россия

¹ ORCID: 0009-0009-1847-9498, aaprovorova@hse.ru

² ORCID: 0009-0006-2811-823X, iyupolyakova@edu.hse.ru

³ ORCID: 0009-0000-6380-4688, EVKuzmicheva@hse.ru

Аннотация

Машинные методы анализа изображений набирают всё большую популярность в самых различных сферах жизни. Однако остается открытым вопрос, насколько эффективна работа таких алгоритмов на данных низкого качества, например, таких, какие могут использоваться в сфере телемедицины. В работе проведен сравнительный анализ различных подходов к детекции объектов на фотографиях МРТ-снимков головного мозга, сделанных с экрана компьютера. Для распознавания контуров головного мозга на изображении были использованы классический морфометрический подход (библиотека OpenCV), алгоритм Виолы-Джонса и два алгоритма глубокого обучения: YOLOv8 и EfficientDet. Сравнение данных методов проводилось с точки зрения качества обнаружения объекта на изображении, оцениваемого при помощи IoU-метрики, а также с точки зрения скорости работы и объема занимаемой памяти. В результате сравнения оказалось, что модель YOLOv8 продемонстрировала наилучший показатель качества обнаружения объектов, однако её работа была нестабильной в случаях некачественных изображений с высоким уровнем шума. Также среди рассмотренных подходов YOLOv8 является самой габаритной с точки зрения используемой памяти. Наилучшим кандидатом для дальнейшего практического применения как с точки зрения среднего показателя качества работы, так и с точки зрения устойчивости к шумам, можно считать сетевую архитектуру YOLOv8.

Ключевые слова: компьютерное зрение; детекция; OpenCV; Виола-Джонс; YOLOv8; EfficientDet.

1. Введение

В современном мире анализ изображений и обнаружение объектов на них играют ключевую роль во многих областях, включая промышленность, робототехнику и медицинскую диагностику [1-3]. Использование цифровых технологий рассматривается как один из подходов, который может помочь улучшить качество медицинских услуг и, как следствие, опыт пациентов, врачей и других получателей услуг [4]. Среди услуг электронного здравоохранения все большую популярность как среди специалистов, так и среди широкой общественности приобретают услуги, основанные на телемедицине [5] и мобильном здравоохранении. По оценкам исследователей IBM, в 2016 году 90% всех медицинских данных представляют собой изображения [6]. Для анализа медицинских изображений разработано огромное количество прикладных методов, которые хорошо описаны в литературе [7].

В настоящее время широкое распространение получают методы анализа медицинских данных с помощью алгоритмов машинного обучения [8]. В частности, с их помощью решаются задачи классификации типов поражения головного мозга, лёгких и постановки диагноза [9-11]. Однако, зачастую алгоритмы машинного обучения разрабатываются и тестируются на качественных данных, таких как МРТ и КТ-снимки, представленные в виде 3D или 4D изображений с низким уровнем шума и высоким пространственным разрешением. В то же время визуализация и анализ таких данных достаточно трудозатратны, а также требуют использования специализированного программного обеспечения, позволяющего выгрузить, а затем просмотреть данные в специфичных форматах, в которых хранятся медицинские изображения (NIfTI, DICOM). В реальной жизни для быстрого обмена информацией о состоянии пациентов, а также для телемедицинского консультирования, медицинские специалисты зачастую фотографируют экран компьютера, на который выведен результат МРТ-сканирования. Телемедицина сегодня только начинает набирать обороты на российском рынке, а увеличенная потребность в дистанционной передаче медицинских данных во время пандемии COVID-19 лишь подчеркнула важность развития надежных алгоритмов для обеспечения качественного медицинского обслуживания на расстоянии [11]. Одним из главных плюсов телемедицины является экономия времени и финансовых ресурсов, что позволяет проводить консультации с врачами узкой специализации без их физического перемещения [12]. Также при использовании телемедицины пациенты имеют возможность дистанционно передавать данные о своем состоянии здоровья, в том числе фотографии, сделанные с помощью камеры смартфона. Однако, в таком случае качество снимков ухудшается и на них появляются дополнительные шумы, которые могут привести к нестабильной работе стандартных методов анализа изображений [13]. Кроме этого, отметим, что разрабатываемые алгоритмы не должны быть слишком трудозатратными с точки зрения производимых вычислений, чтобы быть пригодными для мобильного использования.

Целью данного исследования является сравнение работы различных алгоритмов в задаче обнаружения структур головного мозга на фотографиях МРТ снимков, сделанных смартфоном с экрана монитора. Критериями для сравнения станут точность обнаружения объекта на изображении, объём памяти, занимаемый моделью, и количество элементарных операций, затрачиваемых моделью на обработку изображений.

В работе были проведены анализ и сравнение четырех подходов к решению задачи обнаружения объекта на изображении: морфометрический подход [14], алгоритм Виолы-Джонса [15, 16] и две нейросетевые модели: YOLOv8 [17] и EfficientDet [18]. Морфометрический подход — это один из распространённых методов предобработки медицинских изображений для машинного обучения [19], который позволяет находить объекты на изображении с помощью стандартных функций библиотеки OpenCV без использования алгоритмов глубокого обучения. Однако, пока остается неизвестным, насколько этот алгоритм устойчив при работе с изображениями плохого качества. Алгоритм Виолы-Джонса является также распространённым и давно используемым методом автоматического распознавания объектов [20,21], поэтому его особенно интересно сравнить с наиболее популярными современными алгоритмами глубокого обучения, такими как YOLOv8 и EfficientDet. Анализ указанных выше подходов проводился с учётом их преимуществ и ограничений в контексте поставленной задачи.

2. Описание и формализация исходных данных

Для обучения и тестирования рассматриваемых алгоритмов был подготовлен специальный набор изображений. За основу был взят открытый набор клинических данных BraTS-19 [22-24]. Эта база данных содержит обширные рутинные клинические предоперационные МРТ-снимки глиобластомы и глиомы головного мозга. Среди различных модальностей снимков, представленных в формате файлов NIfTI (.nii), были вы-

браны изображения, записанные с использованием последовательности FLAIR, так как они наиболее широко используются в контексте выявления злокачественной глиобластомы. Из выбранных 3D-изображений были визуализированы несколько сечений, которые затем были преобразованы в PNG-изображения с помощью библиотеки NiBabel [25] и сфотографированы с использованием камер различных смартфонов (Samsung Galaxy S10e, 12 Мп; Samsung Galaxy A50, 25 Мп; Redmi note 10 pro, 108 Мп). Всего было подготовлено 631 изображение. Примеры получившихся изображений представлены на Рис.1.

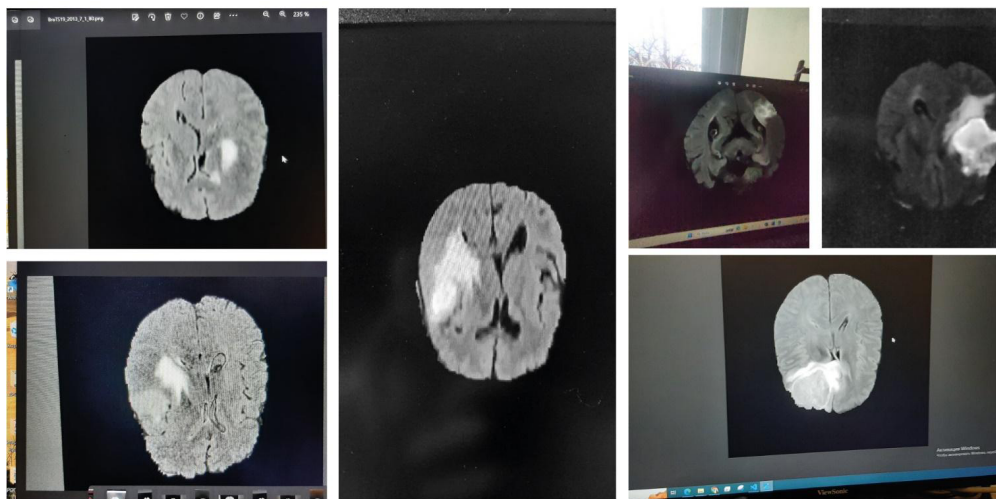


Рис.1. Примеры изображений для обучения моделей

Стоит обратить внимание на то, что на снимках могут содержаться разнообразные посторонние объекты: иконки рабочего стола, блики, отражения и другие шумы, - с целью максимально близко симулировать изображения, которые могут возникать в ходе реальной клинической практики. Затем все изображения были размечены авторами вручную с помощью инструмента Roboflow [26].

Введем понятие ограничивающей рамки (англ. bounding box) – это прямоугольник, в котором содержатся интересующие аналитическую модель объекты. Границы такой рамки задаются двумя координатами: левого верхнего и правого нижнего угла. Ограничивающие рамки широко используются в задачах детекции благодаря четкой формализации описания местоположения объекта на изображении. Это позволяет точно определять границы объектов и использовать эти рамки в различных задачах, связанных с анализом изображений.

Таким образом, контуры мозга были обведены ограничивающими рамками и классифицированы как класс «brains», остальные объекты на изображении никак не были классифицированы. Пример размеченного изображения приведен на Рис.2.

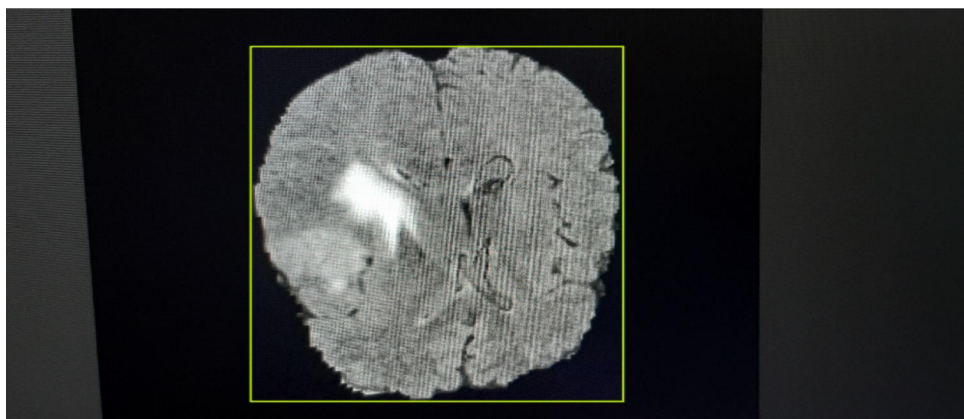


Рис.2. Пример аннотации объекта с помощью ограничивающих рамок

Так как эффективность обучения нейронной сети зависит от объема данных и становится более устойчивой, когда в части набора обучающих данных присутствует шум [19], при создании тестовой выборки были использованы методы аугментации изображений, такие как поворот на 90° , добавление точек на изображении, размытие изображения, зеркальное отражение и искажение перспективы. Примеры изображений с перечисленными аугментациями представлены на Рис.3.

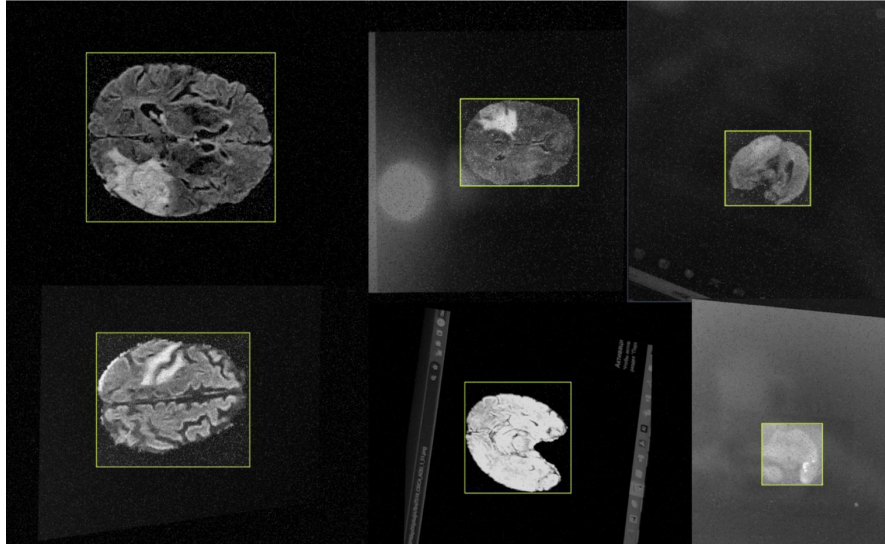


Рис.3. Примеры аугментированных изображений

После проведения аугментации был получен набор из 1038 изображений. Они были разделены на обучающую и тестовую выборки в пропорции 8:2. В результате в обучающую выборку попали 814 изображений, а в тестовую — 224 изображения.

3. Методика эксперимента

В исследовании сравниваются четыре подхода к распознаванию объектов на изображении: использование стандартных методов библиотеки OpenCV (таких, как применение морфологических преобразований к бинаризованному изображению), называемое морфометрическим подходом, применение алгоритма Виолы-Джонса и два метода глубокого обучения: нейронные сети с архитектурой YOLOv8 и с архитектурой EfficientDet. Ниже подробно описан каждый из рассматриваемых подходов.

3.1. Морфометрический подход

В ряде работ, связанных с обработкой медицинских изображений [14, 27], для обработки изображения или для подготовки изображения к использованию при обучении нейронной сети используют некоторые функции из библиотеки OpenCV.

OpenCV (Open Source Computer Vision Library) [28] — это открытая библиотека для работы с алгоритмами компьютерного зрения, машинным обучением и обработкой изображений. Эта библиотека реализована на C/C++, но также она разрабатывается и для Python, Java, Matlab и других языков. В работе была использована версия библиотеки для языка Python.

Опираясь на информацию, описанную в вышеуказанных источниках, авторы создали 6-шаговый алгоритм, позволяющий извлечь контур самого большого объекта на изображении и нарисовать вокруг него ограничивающую рамку.

В исходном наборе данных содержатся изображения, на которых есть один крупный объект — контур головного мозга и фон. В данном случае непосредственный фон изображения чёрного цвета, но помимо него в кадр могут попадать и другие объекты, опи-

санные выше (иконки рабочего стола, рамка экрана и прочие). Предположим, что в таком случае контуры мозга на изображении являются самыми большими. Тогда для его поиска предлагается использовать следующий алгоритм:

1. Преобразовать изображение из цветного в градации серого.
2. К полученному изображению применить бинаризацию. Для нахождения порога бинаризации используется алгоритм Оцу [29]. Таким образом, в «полезный» класс должны попасть пиксели, принадлежащие к более светлой области, – контуру мозга, а остальные попадают в «фоновые».
3. Применить операцию морфологического закрытия к бинаризованному изображению с ядром 3×3 . Эта операция помогает устранить небольшие разрывы в объектах и объединить близко расположенные объекты, такие как мелкие точки или линии.
4. На преобразованном изображении найти все внешние контуры [30]. После применения предыдущих операций, основной объект стал абсолютно белым, а фон – черным. В данном случае под контурами будем понимать кривые, соединяющие непрерывные точки (вдоль границы), имеющие одинаковый цвет и яркость, то есть будут обнаружены объекты, имеющие наибольшую яркость на изображении.
5. Среди найденных контуров выбрать контур, ограничивающий фигуру максимальной площади. Для этого необходимо вычислить площади всех найденных контуров по формуле Грина-Остроградского:

$$\oint (Pdx + Qdy) = \iint \left(\frac{\partial Q}{\partial x} - \frac{\partial P}{\partial y} \right),$$

где функции $P = P(x, y)$, $Q = Q(x, y)$ определены в области D , ограниченной некой кривой C и имеют частные производные $\frac{\partial Q}{\partial x}$, $\frac{\partial P}{\partial y}$.

6. Нарисовать ограничивающую рамку вокруг найденного контура.

В результате применения этого алгоритма были получены координаты прямоугольников, предположительно, ограничивающих мозг на изображении.

3.2. Алгоритм Виолы-Джонса

Алгоритм Виолы-Джонса является одним из первых методов детекции объектов [20], получившим широкое распространение. Однако, в последнее время этот метод теряет популярность в связи с появлением новых методов машинного обучения, демонстрирующих более высокую точность распознавания исходных объектов [21]. Существенным преимуществом метода является то, что он демонстрирует низкую нагрузку на вычислительные мощности системы [31], что может быть весьма полезно в случае его применения как части более сложной системы по обработке изображений.

Данный метод основан на интегральном представлении изображения, которое позволяет эффективно вычислить суммарную яркость пикселей в прямоугольных областях изображения путем применения предварительной обработки [15,16]. Это значительно сокращает вычислительную сложность операций и позволяет быстро вычислять признаки на различных масштабах и различных положениях объекта на изображении.

Данный алгоритм включает в себя алгоритм адаптивного бустинга [20], который представляет собой метод обучения с последовательной настройкой слабых классификаторов (простых моделей, которые не имеют высокую точность предсказания) на основе ошибок предыдущего классификатора. Таким образом, точность классификации постепенно улучшается. Комбинирование множества слабых классификаторов вместе с определенным весом для каждого из них позволяет получить более сильную модель. Обычно вес каждого слабого классификатора определяется итеративно в процессе обучения адаптивного бустинга, где веса обновляются на основе того, как хорошо модель справляется с определенными образцами данных.

Одним из ключевых компонентов алгоритма Виолы-Джонса, помимо перечисленных выше, является использование признаков Хаара для поиска искомого объекта. Признаки Хаара представляют собой прямоугольные области разного размера, такие

как горизонтальные, вертикальные и диагональные полосы. Эти признаки используются для вычисления различных характеристик изображения, по которым и будет осуществляться классификация объекта. Комбинации признаков формируют каскадную структуру классификатора. Визуализация признаков Хаара приведена на Рисунке 4.

Общая каскадная структура, получаемая по методу Виолы-Джонса, называется Каскадом Хаара. Каскад широко используется для обнаружения таких объектов, как лица, глаза, автомобили и т.д. Он также способен находить передвижения искомого объекта на видео и отслеживать их.

Каскад Хаара является наиболее часто используемым из всех рассматриваемых в данной статье методов распознавания объектов. Однако его существенным ограничением является то, что этот алгоритм требователен к количеству изображений в обучающей выборке из-за использования в обучении бустинга [17], а также к яркости изображений из-за использования метода, учитывающего яркость в алгоритме обучения.

В данной работе при обучении каскада использовались 814 положительных изображения (407 исходных + 407 аугментированных), содержавшие объект класса «brains», и 1000 негативных изображения без объектов искомого класса. Негативные изображения включали в себя снимки экранов рабочего стола, курсоров мышки и т.п. — в целях снижения чувствительности реагирования каскада на присутствие данных объектов в положительной выборке, а также изображения нерелевантных объектов: животных, фруктов, овощей и орехов (Рис. 5).

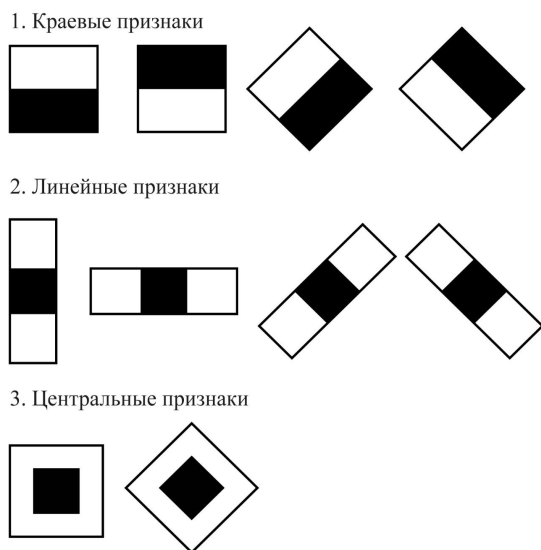


Рис.4. Признаки Хаара



Рис.5. Примеры негативных изображений

В данном исследовании был создан каскад с использованием программы Cascade Trainer GUI, являющейся графическим интерфейсом для применения алгоритмов из библиотеки OpenCV.

Каскад обучался в течение 20 эпох, после чего была достигнута требуемая частота ложных срабатываний классификатора - 10%. При этом точность классификатора составила 0,0002, что означает, что классификатор является достаточно точным и еще не произошло переобучение модели. Таким образом, создание каскада заняло 1 442 минут. Обучение производилось на процессоре AMD Ryzen 5, 4 производительных ядра, частота 2,10 ГГц.

3.3. YOLOv8

YOLO (You Only Look Once, англ. «Вы смотрите только один раз») представляет собой семейство моделей для решения задачи многоклассовой детекции. Модели из семейства YOLO широко применяются для обнаружения объектов на изображении и обработки медицинских изображений [32]. Задача многоклассовой детекции в глубоком обучении условно состоит из двух подзадач. Первая – нахождение «прямоугольников-кандидатов», то есть таких ограничивающих рамок на изображении, внутри которых наиболее вероятно находится хотя бы один из искомых объектов. Вторая задача – классификация выявленных «прямоугольников-кандидатов» с помощью выбора класса с наибольшей рассчитанной вероятностью нахождения его в данной ограничивающей рамке.

Более ранние архитектуры, например, такие, как R-CNN [33], Fast R-CNN [34] или Faster R-CNN [35] решали описанные выше задачи за два отдельных друг от друга этапа, тогда как YOLO решает обе задачи за один этап, что и нашло отражение в английском названии данного метода.

Базовая версия модели YOLO [17] представляет собой свёрточную нейросетевую архитектуру из семейства «Darknet» [36] и два полносвязных слоя. В базовой версии YOLO используется архитектура «Darknet-17». Отметим, что под «Darknet» в данном случае подразумевается именно архитектура, а не фреймворк для глубокого обучения с аналогичным названием.

На вход YOLO подается изображение, на выходе получают векторы для каждой из оставшихся прямоугольных рамок после процедуры «Non-maximum suppression», которая из нескольких очень похожих ограничивающих рамок оставляет одну наиболее релевантную, исключая из финальной детекции остальные.

С 2015-го года в семействе моделей YOLO наблюдалось несколько «пополнений» [32]. Последнее из них на момент написания статьи – это восьмая версия YOLOv8. В этой модели контрольные точки детекции объектов предобучены на наборе MS-COCO 2017 [37], а модели для классификации изображений – на наборе ImageNet [38], что позволило обеспечить более стабильное качество многоклассовой детекции.

Для работы с моделью была использована официальная библиотека разработчиков YOLOv8 – Ultralytics [39]. Предобученная модель была дообучена на созданном авторами наборе данных с помощью стохастического градиентного спуска, а именно с помощью алгоритма AdamW со скоростью обучения равной 0,002 и инерцией равной 0,9 с подбором гиперпараметров после каждой эпохи на валидационной выборке размером в 100 изображений. Скорость обучения – это параметр, который отвечает за размер шага в процессе обновления весов. Инерция – параметр, отвечающий за ускорение обновления весов в определенном направлении и позволяющий уменьшить колебания весов, добавляя долю предыдущего шага к текущему градиенту.

Данные для валидационной выборки случайным образом отбирались из обучающей выборки. Модель дообучалась в течение 50 эпох, после чего значение функции ошибки перестало убывать и возникла опасность переобучения. Процесс изменения значения функции потерь в течение 50 эпох проиллюстрирован на Рис. 6.

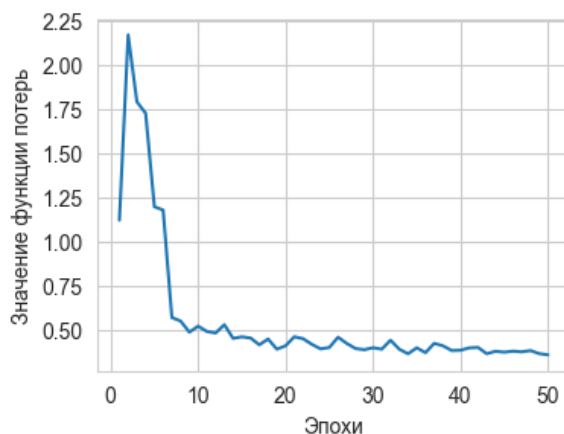


Рис. 6. График сходимости результатов обучения модели YOLOv8

На вероятность нахождения объекта класса «brains» было поставлено пороговое значение 0,7. Значение подбиралось как гиперпараметр с целью достижения наибольшей средней метрики Intersection over Union (IoU).

YOLOv8 обучалась в интерактивной среде Google Colaboratory с помощью графического процессора T4, время обучения составило 25 минут.

3.4. EfficientDet

EfficientDet (Do-D7) [18] — это модели из инновационного класса нейросетевых моделей EfficientNet [40], предназначенные для обнаружения объектов на изображениях. В основе этой модели лежит архитектура EfficientNet, на вершину которой добавлен слой, работающий с пирамидой признаков Bidirectional Feature Pyramid Network (BiFPN), за ним следует сеть-классификатор для генерации предсказаний класса объекта и блочная сеть для предсказания ограничивающей рамки. Ориентируясь на мобильные и встроенные приложения, TensorFlow Lite (TF Lite) [41] разработали семейство моделей обнаружения объектов EfficientDet-lite с использованием стандартных для этого семейства моделей сверточных архитектур, но отформатированных для небольшого размера модели и быстрого вывода. Однако, их производительность немного уступает исходному аналогу семейства EfficientDet. В данном исследовании использовалась модель обнаружения объектов EfficientDet-lite (lited), предобученная на наборе данных MS-COCO 2017 [37]. Эта модель была выбрана в связи с тем, что по результатам данного сравнения авторы планируют интегрировать выбранный подход в большую систему автоматической диагностики. Поэтому скорость работы и малый объем модели заинтересовали нас в исследовательском плане. У модели было изменено количество выходных классов (набор данных MS-COCO 2017 содержит 91 класс объектов, а в поставленной задаче модель должна обнаруживать только один класс - «brains») и пороговое значение вероятности обнаружения объекта на изображении было установлено равным 0,7, аналогично модели YOLOv8, описанной выше. Это было сделано для корректности сравнения финальных результатов.

Модель была дообучена методом обратного распространения ошибки с использованием стохастического градиентного спуска с инерцией 0,9, начальным шагом $2e-2$ и множителем косинусного затухания $4e-5$. Дообучение модели производилось в течение 20 эпох, после чего функция потерь перестала убывать. Этот процесс проиллюстрирован на Рис. 7.

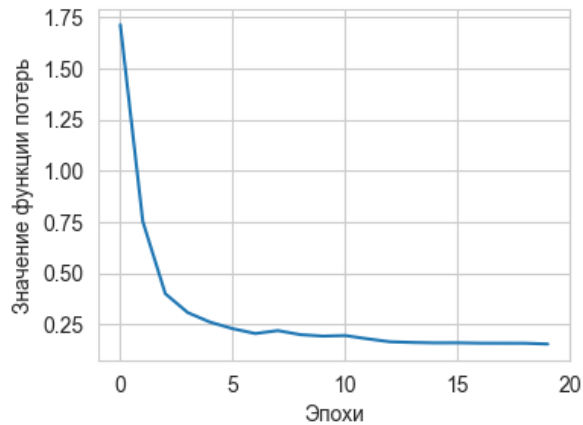


Рис. 7. График сходимости результатов обучения модели EfficientDet-lite (liteo)

Модель обучалась на ЭВМ с процессором AMD Ryzen 3, 4 производительных ядра, частота 2,60 ГГц. На описанной ЭВМ обучение заняло 162 минуты.

3.5. Метрики качества

Для задач обнаружения объектов на изображении важным является и выбор метрики качества, которая показывает, насколько хорошо модель решает поставленную задачу. Метрика Intersection over Union (IoU) широко используется для оценки качества работы моделей в задачах сегментации и обнаружения объектов в компьютерном зрении. IoU имеет простую и интерпретируемую формулу, которая отражает соотношение между площадью области пересечения прямоугольников, ограничивающих истинный и предсказанный объекты, и их общей площадью (Рис.8).

$$IoU = \frac{\text{Area of Intersection}}{\text{Area of Union}}$$

Рис. 8. Графическая интерпретация IoU-метрики

Это позволяет легко интерпретировать, насколько качественно модель обнаруживает объекты. Для сравнения результатов работы разных методов было использовано среднее значение данного параметра на тестовой выборке. В разделе 4 в таблице 1 приведены не только средние значения метрики, но и максимальное и минимальное по выборке. Так как в данном исследовании на изображении ищется только один класс объектов и результатом является единственная ограничивающая рамка, то использование этой метрики является более целесообразным, чем использование таких распространённых метрик, как AP (average precision) и mAP (mean average precision).

Кроме метрики IoU, в качестве меры производительности алгоритмов было измерено количество элементарных операций, которые потребовались для работы алгоритмов на тестовой выборке. Данная метрика не зависит от устройства, на котором запускается модель. Для анализа производительности программ и вычисления количества элементарных операций был использован стандартный модуль языка python - cProfile.

4. Результаты

Результаты проведенных вычислительных экспериментов описаны в Таблице 1.

Таблица 1. Результаты сравнения подходов

Название алгоритма	IoU (min; max)	Количество элементарных операций	Размер модели(Мб)
Морфометрический подход	0,795 (0,0; 0,99)	98 710	-
Алгоритм Виолы-Джонса	0,453 (0,0; 0,85)	97 366	0,11
YOLOv8	0,913 (0,0; 0,99)	686 900	22,49
EfficientDet-lite0	0,817 (0,04; 0,94)	197 823	4,23

Наилучшее среднее качество обнаружения объекта на изображении на тестовой выборке продемонстрировала модель YOLOv8. Однако, было обнаружено, что качество работы всех рассматриваемых моделей сильно зависело от наличия шумов на изображении. Для более подробного анализа качества работы алгоритмов были построены гистограммы распределения качества определения объектов на изображении (Рис.9)

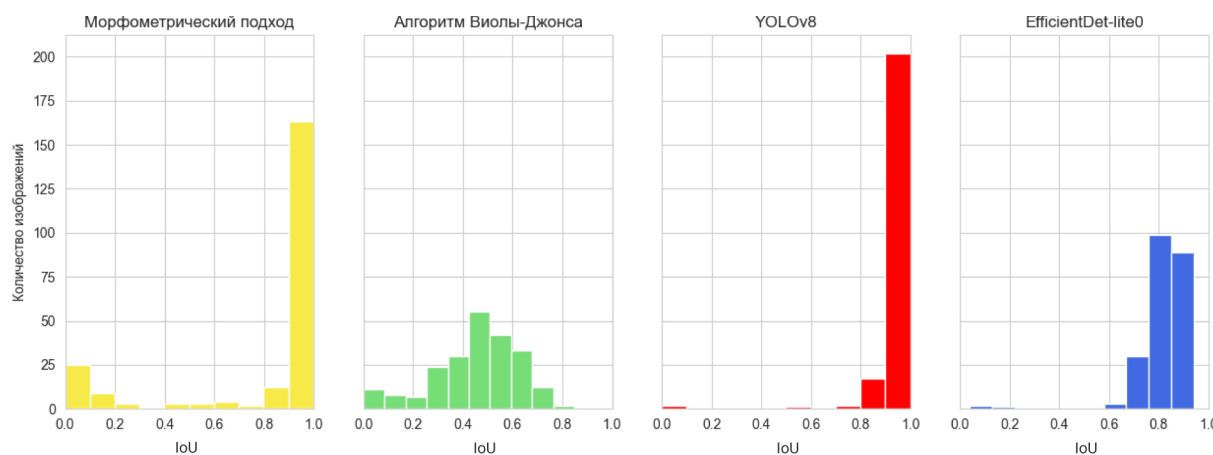


Рис. 9. Гистограммы распределения значений IoU по результатам работы моделей детекции объектов на изображении

Для изображений с небольшим количеством шума и достаточной контрастностью между объектом (мозгом) и фоном морфометрический алгоритм обнаруживал объект с точностью до 99,9%. Однако, присутствие сильного шума на изображении снижало точность его работы до 0%. Таким образом, можно сделать заключение, что большую часть изображений этот алгоритм обработал с очень высокой точностью. Алгоритмы глубокого машинного обучения оказались более устойчивыми к шумам на изображениях и демонстрировали стабильные результаты практически на всей выборке. Тем не менее и для них фактор присутствия шума на изображении являлся существенным. Распределение IoU для модели YOLOv8 оказалось наиболее смещенным вправо, то есть YOLOv8 практически всегда показывает результат 0.8 и выше, за исключением одной фотографии (Рис. 8), когда модель не способна найти объект. На данном изображении сильный шум создаёт вспышка от фотоаппарата, что, очевидно, сбивает модель.

Модель EfficientDet успешно обнаруживала контуры мозга на всех изображениях, даже в случае присутствия сильного шума. Однако, среднее качество обнаружения объекта на изображении у этой модели оказалось хуже, чем у модели YOLOv8, что связано с более низкой точностью нахождения ограничивающей рамки. Алгоритм Виолы-Джонса на представленной выборке показал наихудший результат с точки зрения IoU-

метрики, однако можно отметить определённую стабильность в его работе: с его помощью удаётся обнаружить мозг на изображениях любого качества.

На Рис.8 приведены примеры результатов работы алгоритмов на изображениях хорошего качества и на изображениях с большим количеством шума.

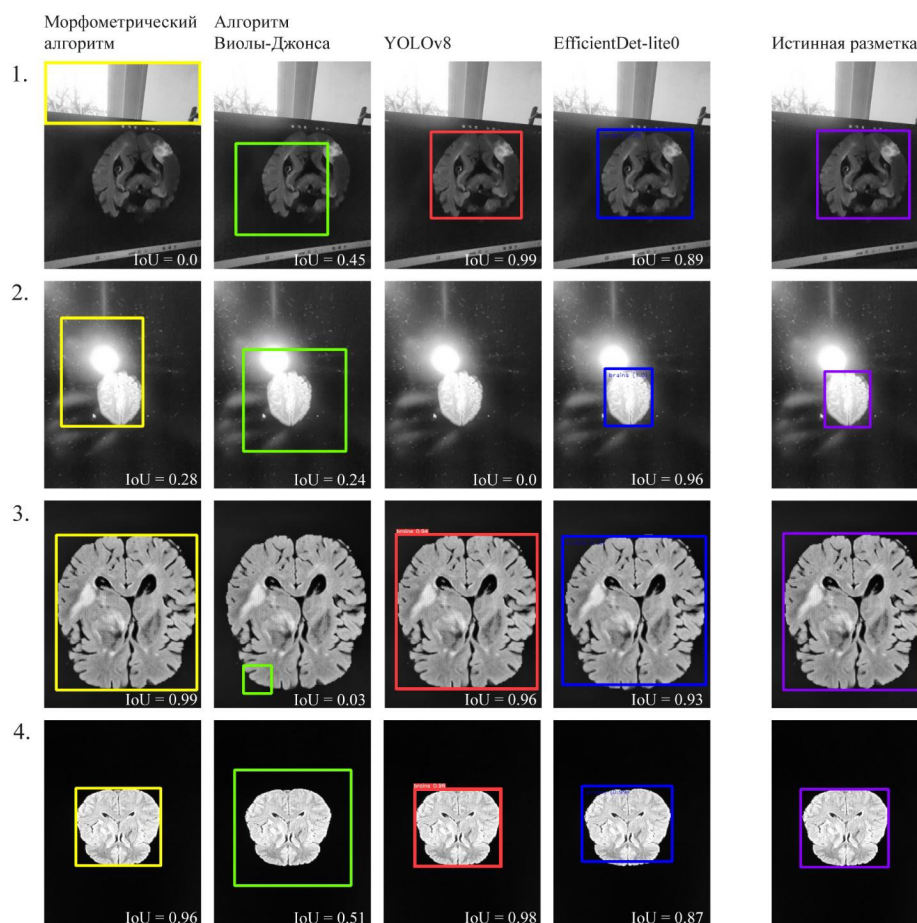


Рис. 10. Результаты работы алгоритмов на изображениях с различным уровнем шума. 1, 2 - сильно зашумлённые изображения, 3, 4 - качественно сфотографированные изображения

Также в рамках исследования было выявлено, что наименее затратным по количеству операций является алгоритм Виолы-Джонса и он занимает меньше всего места в памяти компьютера. По этому показателю очень близок к нему и морфометрический подход. Однако, для использования обоих этих методов возникают серьёзные препятствия: в случае алгоритма Виолы-Джонса — очень низкая точность, а в случае морфометрического подхода — неустойчивость алгоритма к сильным шумам на изображении.

5. Заключение

В данной работе было проведено сравнение четырёх способов обнаружения мозга на фотографии МРТ-снимка с экрана компьютера. Все протестированные подходы демонстрируют достаточно высокие средние характеристики обнаружения объекта на изображении. Наивысшее качество обнаружения объектов на изображении по критерию среднего Intersection over Union (IoU) продемонстрировала модель YOLOv8, но качество её работы оказалось чувствительным к шумам на изображении. Эта модель является достаточно объемной с точки зрения занимаемого места на диске. Модель EfficientDet-lite0 продемонстрировала чуть более низкое качество распознавания объектов в среднем, но показала себя более устойчивой к шумам. Ещё одним достоинством

этой модели является её малый вес. Объем модели важен для дальнейшего её включения в более сложные нейросетевые диагностические системы, для которых обнаружение объекта на изображении станет только первым этапом работы. Классический морфометрический подход к обнаружению крупного объекта на изображении также продемонстрировал высокие результаты, но он не может быть использован для решения поставленной задачи, т.к. этот алгоритм распознаёт самый крупный светлый участок на изображении и в случае фотосъёмки со вспышкой или попадания в кадр случайного светлого объекта работает неверно. Алгоритм Виолы-Джонса плохо справился с задачей обнаружения структур головного мозга на фотографии МРТ снимка и не рекомендуется к использованию в рамках поставленной задачи.

6. Благодарности

Публикация подготовлена в ходе проведения исследования № 23-00-026 «Разработка автоматических подходов для определения этиологии криптогенного инсульта с целью профилактики вторичных острых нарушений мозгового кровообращения» в рамках Программы «Научный фонд Национального исследовательского университета «Высшая школа экономики» (НИУ ВШЭ)».

Авторы благодарят за помощь в подготовке статьи Куликову Софью Петровну, заведующего Центром когнитивных нейронаук НИУ ВШЭ, Пермь.

Литература

1. Song S. et al. EfficientDet for fabric defect detection based on edge computing //Journal of Engineered Fibers and Fabrics, Vol. 16, 2021 (<https://doi.org/10.1177/15589250211008346>).
2. Afif M. et al. An evaluation of EfficientDet for object detection used for indoor robots assistance navigation //Journal of Real-Time Image Processing, Vol. 19 (3), 2022, pp. 651-661 (<http://dx.doi.org/10.1007/s11554-022-01212-4>).
3. Mercaldo F. et al. Object Detection for Brain Cancer Detection and Localization //Applied Sciences, Vol. 13 (16), 2023, pp. 9158 (<http://dx.doi.org/10.3390/app13169158>).
4. Moghaddasi H. et al., E-health: a global approach with extensive semantic variation //Journal of medical systems, Vol. 36, 2012, pp. 3173-3176 (<http://dx.doi.org/10.1007/s10916-011-9805-z>).
5. Pol P., Deshpande A.M. Telemedicine mobile system //2016 Conference on Advances in Signal Processing (CASP), IEEE, 2016, pp. 484-487 (<http://dx.doi.org/10.1109/CASP.2016.7746220>).
6. Landi H. IBM Unveils Watson-Powered Imaging Solutions at RSNA, тRSNA, 2016. Accessed: November, 29st, 2023. [Online]. Available: <https://www.hcinnovationgroup.com/populationhealth-management/news/13027814/ibm-unveilswatsonpowered-imaging-solutions-at-rsna>
7. Недзьведь А.М., Абламейко С. В. Анализ изображений для решения задач медицинской диагностики, Минск : ОИПИ НАН Беларуси, 2012, С.240 (ISBN 978-985-6744-75-7).
8. Papp L. et al. Personalizing medicine through hybrid imaging and medical big data analysis //Frontiers in Physics, Vol. 6, 2018, pp. 51 (<http://dx.doi.org/10.3389/fphy.2018.00051>).
9. Kumar S., Paliania U., Nandal N. A systematic study of artificial intelligence-based methods for detecting brain tumors //Информатика и автоматизация, Vol. 22 (3), 2023, pp. 541-575 (<http://dx.doi.org/10.15622/ia.22.3.3>).
10. Krishnapriya S., Karuna Y. Pre-trained deep learning models for brain MRI image classification //Frontiers in Human Neuroscience, Vol. 17, 2023, pp. 1150120 (<http://dx.doi.org/10.3389/fnhum.2023.1150120>).

11. Moreira L.P. Automated Medical Device Display Reading Using Deep Learning Object Detection //arXiv preprint arXiv:2210.01325, 2022 (<https://doi.org/10.48550/arXiv.2210.01325>).
12. Козлова А. С., Тараскин Д. С. Тенденции развития телемедицины и ее влияние на страховой рынок России //Промышленность: экономика, управление, технологии, 2018, Т.2 (71), С.144-148.
13. Mabotuwana T. et al. Detecting technical image quality in radiology reports //AMIA Annual Symposium Proceedings, American Medical Informatics Association, Vol. 2018, 2018, pp. 780.
14. Widodo C.E., Adi K., Gernowo R. Medical image processing using python and open cv //Journal of Physics: Conference Series, IOP Publishing, Vol. 1524 (1), 2020, pp. 012003 (<http://dx.doi.org/10.1088/1742-6596/1524/1/012003>).
15. Viola P., Jones M.J. Robust real-time face detection //International journal of computer vision, Vol. 57, 2004, pp. 137-154 (<https://doi.org/10.1023/B:VISI.0000013087.49260.fb>).
16. Freund Y., Schapire R., Abe N. A short introduction to boosting //Journal-Japanese Society For Artificial Intelligence, Vol. 14(771-780), 1999, pp. 1612 (<https://doi.org/10.4236/jamp.2021.911186>).
17. Redmon J. et al. You only look once: Unified, real-time object detection //Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 779-788 (<http://dx.doi.org/10.1109/CVPR.2016.91>).
18. Tan M., Pang R., Le Q.V. Efficientdet: Scalable and efficient object detection //Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, pp. 10781-10790 (<http://dx.doi.org/10.1109/CVPR42600.2020.01079>).
19. Z. et al. Improving synthetic CT accuracy by combining the benefits of multiple normalized preprocesses //Journal of Applied Clinical Medical Physics, 2023, pp. e14004 (<http://dx.doi.org/10.1002/acm2.14004>).
20. Viola P., Jones M. Rapid object detection using a boosted cascade of simple features //Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001, IEEE, Vol. 1 2001, pp. I-I (<http://dx.doi.org/10.1109/CVPR.2001.990517>).
21. Sanjay T., Priya W.D. Criminal Identification System to Improve Accuracy of Face Recognition using Innovative CNN in Comparison with HAAR Cascade //Journal of Pharmaceutical Negative Results, 2022, pp. 218-223.
22. Menze B. H. et al. The multimodal brain tumor image segmentation benchmark (BRATS) //IEEE transactions on medical imaging, Vol. 34 (10), 2014, pp. 1993-2024 (<http://dx.doi.org/10.1109/TMI.2014.2377694>).
23. Bakas S. et al. Advancing the cancer genome atlas glioma MRI collections with expert segmentation labels and radiomic features //Scientific data, Vol. 4 (1), 2017, pp. 1-13 (<https://doi.org/10.1038/sdata.2017.117>).
24. Bakas S. et al. Identifying the best machine learning algorithms for brain tumor segmentation, progression assessment, and overall survival prediction in the BRATS challenge //arXiv preprint arXiv:1811.02629, 2018 (<https://doi.org/10.48550/arXiv.1811.02629>).
25. Brett M. et al. Nipy/nibabel:. 4.0.0, Zenodo, 17.06.2022, (<http://doi.org/10.5281/zenodo.6658382>).
26. Dwyer B. et al. Roboflow (version 1.0)[software]. 2022. Accessed: November, 29st, 2023. [Online]. Available: <https://roboflow.com>.
27. Yousif M.J. Enhancing The Accuracy of Image Classification Using Deep Learning and Preprocessing Methods //Artificial Intelligence & Robotics Development Journal, 2023 (<http://dx.doi.org/10.52098/airdj.2023348>).
28. Bradski G. et al. OpenCV //Dr. Dobb's journal of software tools, Vol. 3 (2), 2000.

29. Otsu N. A threshold selection method from gray-level histograms //IEEE transactions on systems, man, and cybernetics, Vol. 9 (1), 1979, pp. 62-66 (<https://doi.org/10.1109/TSMC.1979.4310076>).
30. Suzuki S. et al. Topological structural analysis of digitized binary images by border following //Computer vision, graphics, and image processing, Vol. 30 (1), 1985, pp. 32-46 ([https://doi.org/10.1016/0734-189X\(85\)90016-7](https://doi.org/10.1016/0734-189X(85)90016-7)).
31. Ганиев М.А. Методы детекции объектов для анализа //Актуальные исследования, Т. 10, 2024, С. 48.
32. Qureshi R. et al. A Comprehensive Systematic Review of YOLO for Medical Object Detection (2018 to 2023), 2023 (<https://doi.org/10.36227/techrxiv.23681679.v1>).
33. Girshick R. et al. Rich feature hierarchies for accurate object detection and semantic segmentation //Proceedings of the IEEE conference on computer vision and pattern recognition, 2014, pp. 580-587 (<https://doi.org/10.1109/CVPR.2014.81>).
34. Girshick R. Fast R-CNN //Proceedings of the IEEE international conference on computer vision, 2015, pp. 1440-1448 (<https://doi.org/10.1109/ICCV.2015.169>).
35. Ren S. et al. Faster R-CNN: Towards real-time object detection with region proposal networks //Advances in neural information processing systems, Vol. 28, 2015 (<https://doi.org/10.48550/arXiv.1506.01497>).
36. Weng L. Object Detection Part 4: Fast Detection Models // lilianweng.github.io, 2018. Accessed: November, 29st, 2023. [Online]. Available: <https://lilianweng.github.io/posts/2018-12-27-object-recognition-part-4/>.
37. Lin T.Y. et al. Microsoft COCO: Common objects in context //Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13. – Springer International Publishing, 2014, pp. 740-755 (https://doi.org/10.1007/978-3-319-10602-1_48).
38. Deng J. et al. Imagenet: A large-scale hierarchical image database //2009 IEEE conference on computer vision and pattern recognition, IEEE, 2009, pp. 248-255. (<https://doi.org/10.1109/CVPR.2009.5206848>)
39. Jocher, G., Chaurasia, A., Qiu, J. (2023). Ultralytics YOLO (Version 8.0.0) [Computer software], Accessed: November, 29st, 2023. [Online]. Available: <https://github.com/ultralytics/ultralytics>
40. Tan M., Le Q. Efficientnet: Rethinking model scaling for convolutional neural networks//International conference on machine learning, PMLR, 2019, pp. 6105-6114. (<https://doi.org/10.48550/arXiv.1905.11946>)
41. Abadi M. et al. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015 (<https://doi.org/10.48550/arXiv.1603.04467>).

Comparison of the Effectiveness of Using Various Approaches in Detecting Objects on Low-Quality Images

A. Provorova¹, I. Polyakova², E. Kuzmicheva³

National Research University «Higher School of Economics», Perm, Russia

¹ ORCID: 0009-0009-1847-9498, aaprovorova@hse.ru

² ORCID: 0009-0006-2811-823X, iyupolyakova@edu.hse.ru

³ ORCID: 0009-0000-6380-4688, EVKuzmicheva@hse.ru

Abstract

Machine methods of image analysis are gaining popularity in various fields of life. However, the question remains as to how effective such algorithms are on low-quality data, such as those that can be used in the field of telemedicine. The work provides a comparative analysis of various approaches to object detection in MRI brain images taken from a computer screen. For the recognition of brain contours in the image, a classical morphometric approach (OpenCV library), the Viola-Jones algorithm, and two deep learning algorithms, YOLOv8 and EfficientDet, were used. The comparison of these methods was conducted in terms of the quality of object detection in the image. To assess the quality, we used the IoU metric, as well as measured the amount of memory used and the speed of algorithm execution. As a result of the comparison, we found that the YOLOv8 model demonstrated the best performance in terms of object detection quality. However, its performance was unstable in cases of low-quality images with high levels of noise. Among the considered approaches, YOLOv8 is also the most memory-intensive. The YOLOv8 network architecture can be considered the best candidate for further practical application in terms of average performance and resistance to noise.

Keywords: computer vision; detection; OpenCV; Viola-Jones; YOLOv8; EfficientDet.

References

1. Song S. et al. EfficientDet for fabric defect detection based on edge computing // Journal of Engineered Fibers and Fabrics, Vol. 16, 2021 (<https://doi.org/10.1177/15589250211008346>).
2. Afif M. et al. An evaluation of EfficientDet for object detection used for indoor robots assistance navigation // Journal of Real-Time Image Processing, Vol. 19 (3), 2022, pp. 651-661 (<http://dx.doi.org/10.1007/s11554-022-01212-4>).
3. Mercaldo F. et al. Object Detection for Brain Cancer Detection and Localization // Applied Sciences, Vol. 13 (16), 2023, pp. 9158 (<http://dx.doi.org/10.3390/app13169158>).
4. Moghaddasi H. et al., E-health: a global approach with extensive semantic variation // Journal of medical systems, Vol. 36, 2012, pp. 3173-3176 (<http://dx.doi.org/10.1007/s10916-011-9805-z>).
5. Pol P., Deshpande A.M. Telemedicine mobile system // 2016 Conference on Advances in Signal Processing (CASP), IEEE, 2016, pp. 484-487 (<http://dx.doi.org/10.1109/CASP.2016.7746220>).
6. Landi H. IBM Unveils Watson-Powered Imaging Solutions at RSNA, tRSNA, 2016. Accessed: November, 29st, 2023. [Online]. Available: <https://www.hcinnovationgroup.com/populationhealth-management/news/13027814/ibm-unveilswatsonpowered-imaging-solutions-at-rsna>

7. Недзьведь А.М., Абламейко С. В. Анализ изображений для решения задач медицинской диагностики, Минск : ОИПИ НАН Беларуси, 2012, С.240 (ISBN 978-985-6744-75-7).
8. Papp L. et al. Personalizing medicine through hybrid imaging and medical big data analysis //Frontiers in Physics, Vol. 6, 2018, pp. 51 (<http://dx.doi.org/10.3389/fphy.2018.00051>).
9. Kumar S., Paliana U., Nandal N. A systematic study of artificial intelligence-based methods for detecting brain tumors //Информатика и автоматизация, Vol. 22 (3), 2023, pp. 541-575 (<http://dx.doi.org/10.15622/ia.22.3.3>).
10. Krishnapriya S., Karuna Y. Pre-trained deep learning models for brain MRI image classification //Frontiers in Human Neuroscience, Vol. 17, 2023, pp. 1150120 (<http://dx.doi.org/10.3389/fnhum.2023.1150120>).
11. Moreira L.P. Automated Medical Device Display Reading Using Deep Learning Object Detection //arXiv preprint arXiv:2210.01325, 2022 (<https://doi.org/10.48550/arXiv.2210.01325>).
12. Козлова А. С., Тараскин Д. С. Тенденции развития телемедицины и ее влияние на страховой рынок России //Промышленность: экономика, управление, технологии, 2018, Т.2 (71), С.144-148.
13. Mabotuwana T. et al. Detecting technical image quality in radiology reports //AMIA Annual Symposium Proceedings, American Medical Informatics Association, Vol. 2018, 2018, pp. 780.
14. Widodo C.E., Adi K., Gernowo R. Medical image processing using python and open cv //Journal of Physics: Conference Series, IOP Publishing, Vol. 1524 (1), 2020, pp. 012003 (<http://dx.doi.org/10.1088/1742-6596/1524/1/012003>).
15. Viola P., Jones M.J. Robust real-time face detection //International journal of computer vision, Vol. 57, 2004, pp. 137-154 (<https://doi.org/10.1023/B:VISI.0000013087.49260.fb>).
16. Freund Y., Schapire R., Abe N. A short introduction to boosting //Journal-Japanese Society For Artificial Intelligence, Vol. 14(771-780), 1999, pp. 1612 (<https://doi.org/10.4236/jamp.2021.911186>).
17. Redmon J. et al. You only look once: Unified, real-time object detection //Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 779-788 (<http://dx.doi.org/10.1109/CVPR.2016.91>).
18. Tan M., Pang R., Le Q.V. Efficientdet: Scalable and efficient object detection //Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, pp. 10781-10790 (<http://dx.doi.org/10.1109/CVPR42600.2020.01079>).
19. Z. et al. Improving synthetic CT accuracy by combining the benefits of multiple normalized preprocesses //Journal of Applied Clinical Medical Physics, 2023, pp. e14004 (<http://dx.doi.org/10.1002/acm2.14004>).
20. Viola P., Jones M. Rapid object detection using a boosted cascade of simple features //Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001, IEEE, Vol. 1 2001, pp. I-I (<http://dx.doi.org/10.1109/CVPR.2001.990517>).
21. Sanjay T., Priya W.D. Criminal Identification System to Improve Accuracy of Face Recognition using Innovative CNN in Comparison with HAAR Cascade //Journal of Pharmaceutical Negative Results, 2022, pp. 218-223.
22. Menze B. H. et al. The multimodal brain tumor image segmentation benchmark (BRATS) //IEEE transactions on medical imaging, Vol. 34 (10), 2014, pp. 1993-2024 (<http://dx.doi.org/10.1109/TMI.2014.2377694>).
23. Bakas S. et al. Advancing the cancer genome atlas glioma MRI collections with expert segmentation labels and radiomic features //Scientific data, Vol. 4 (1), 2017, pp. 1-13 (<https://doi.org/10.1038/sdata.2017.117>).

24. Bakas S. et al. Identifying the best machine learning algorithms for brain tumor segmentation, progression assessment, and overall survival prediction in the BRATS challenge //arXiv preprint arXiv:1811.02629, 2018 (<https://doi.org/10.48550/arXiv.1811.02629>).
25. Brett M. et al. Nipy/nibabel:. 4.0.0, Zenodo, 17.06.2022, (<http://doi.org/10.5281/zenodo.6658382>).
26. Dwyer B. et al. Roboflow (version 1.0)[software]. 2022. Accessed: November, 29st, 2023. [Online]. Available: <https://roboflow.com>.
27. Yousif M.J. Enhancing The Accuracy of Image Classification Using Deep Learning and Preprocessing Methods //Artificial Intelligence & Robotics Development Journal, 2023 (<http://dx.doi.org/10.52098/airdj.2023348>).
28. Bradski G. et al. OpenCV //Dr. Dobb's journal of software tools, Vol. 3 (2), 2000.
29. Otsu N. A threshold selection method from gray-level histograms //IEEE transactions on systems, man, and cybernetics, Vol. 9 (1), 1979, pp. 62-66 (<https://doi.org/10.1109/TSMC.1979.4310076>).
30. Suzuki S. et al. Topological structural analysis of digitized binary images by border following //Computer vision, graphics, and image processing, Vol. 30 (1), 1985, pp. 32-46 ([https://doi.org/10.1016/0734-189X\(85\)90016-7](https://doi.org/10.1016/0734-189X(85)90016-7)).
31. Ганиев М.А. Методы детекции объектов для анализа //Актуальные исследования, Т. 10, 2024, С. 48.
32. Qureshi R. et al. A Comprehensive Systematic Review of YOLO for Medical Object Detection (2018 to 2023), 2023 (<https://doi.org/10.36227/techrxiv.23681679.v1>).
33. Girshick R. et al. Rich feature hierarchies for accurate object detection and semantic segmentation //Proceedings of the IEEE conference on computer vision and pattern recognition, 2014, pp. 580-587 (<https://doi.org/10.1109/CVPR.2014.81>).
34. Girshick R. Fast R-CNN //Proceedings of the IEEE international conference on computer vision, 2015, pp. 1440-1448 (<https://doi.org/10.1109/ICCV.2015.169>).
35. Ren S. et al. Faster R-CNN: Towards real-time object detection with region proposal networks //Advances in neural information processing systems, Vol. 28, 2015 (<https://doi.org/10.48550/arXiv.1506.01497>).
36. Weng L. Object Detection Part 4: Fast Detection Models // lilianweng.github.io, 2018. Accessed: October, 15st, 2023. [Online]. Available: <https://lilianweng.github.io/posts/2018-12-27-object-recognition-part-4/>.
37. Lin T.Y. et al. Microsoft COCO: Common objects in context //Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13. – Springer International Publishing, 2014, pp. 740-755 (https://doi.org/10.1007/978-3-319-10602-1_48).
38. Deng J. et al. Imagenet: A large-scale hierarchical image database //2009 IEEE conference on computer vision and pattern recognition, IEEE, 2009, pp. 248-255. (<https://doi.org/10.1109/CVPR.2009.5206848>)
39. Jocher, G., Chaurasia, A., Qiu, J. (2023). Ultralytics YOLO (Version 8.0.0) [Computer software], Accessed: November, 10st, 2023. [Online]. Available: <https://github.com/ultralytics/ultralytics>
40. Tan M., Le Q. Efficientnet: Rethinking model scaling for convolutional neural networks//International conference on machine learning, PMLR, 2019, pp. 6105-6114. (<https://doi.org/10.48550/arXiv.1905.11946>)
41. Abadi M. et al. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015 (<https://doi.org/10.48550/arXiv.1603.04467>).