

Содержание

Введение	2
1 Методы бинаризации и протоколы томографии под контролем реконструкции	5
1.1 Томография под контролем реконструкции	5
1.2 Математическая формулировка задачи бинаризации . . .	6
1.3 Методы бинаризации	7
2 Подготовка данных и окружения для численных экспериментов в рамках протокола мониторинговой реконструкции	12
2.1 Подготовка данных для экспериментов	12
2.2 Алгоритмы бинаризации	21
2.3 Метрики качества и оценка их чувствительности	24
2.4 Проектирование и реализация программного комплекса конвейера экспериментов	32
3 Разработка и исследование эффективности правила останова в задаче томографии под контролем реконструкции	38
3.1 Разработка правила останова	38
3.2 Методология проведения численных экспериментов по исследованию эффективности правила останова	41
3.3 Анализ эффективности правила останова	45
4 Выводы	54
Приложение А. Результаты экспериментов для классического порогового алгоритма	60
Приложение Б. Результаты экспериментов для алгоритма Отсу	62
Приложение В. Результаты экспериментов для алгоритма аффинного Ниблэка	65

Введение

Компьютерная томография (КТ) является одним из ключевых методов неразрушающего контроля и широко применяется как в медицинской диагностике, так и в промышленности. Основной принцип КТ заключается в восстановлении внутренней структуры объекта по множеству его проекций, полученных под различными углами.

В промышленной томографии метод используется для анализа геометрии, выявления скрытых дефектов, построения САД-моделей и контроля соответствия изделий стандартам качества. Одной из важнейших задач при этом остаётся снижение дозовой нагрузки и времени проведения эксперимента, особенно в случаях, когда объект чувствителен к излучению или когда процесс сканирования занимает достаточно длительное время.

Современным направлением развития в этой области является применение протоколов мониторинговой реконструкции, при которых процесс сканирования и реконструкции выполняется итеративно с возможностью анализа промежуточных результатов. Такой подход делает процесс сканирования anytime-алгоритмом так как позволяет остановить сканирование при достижении достаточного качества реконструкции с уже имеющимися проекциями, потенциально уменьшая дозу излучения и время сканирования.

Задача снижения дозы является актуальной как в медицине, так и в инженерии и микроэлектронике. В медицине она возникает, например, при исследовании онкологических заболеваний, некротических поражений и других патологий в органах, чувствительных к радиации. При этом в научной литературе часто рассматриваются частные или эвристические подходы. На данный момент существует лишь одно исследование, в котором протокол мониторинговой реконструкции был применён к конкретной задаче — снижению дозовой нагрузки при обнаружении COVID-19 в лёгких на КТ-изображениях [1].

В инженерии и электронике потребность в подобных протоколах особенно выражена при контроле высокоточных и чувствительных изделий — например, в микроэлектронике, оптоэлектронике и спутниковых технологиях, где применение других методов контроля не всегда позво-

ляет гарантировать надёжность.

Одной из базовых задач в индустриальной томографии является задача бинаризации реконструкций. Её решение позволяет уменьшить выраженность артефактов, сформировать САД-модели объектов, автоматизировать поиск дефектов и выполнить контроль соответствия изделий требованиям. Задача бинаризации имеет важное прикладное значение, и точность её решения напрямую влияет на результаты последующей обработки.

С развитием вычислительных средств и появлением протоколов мониторинговой реконструкции появилась возможность перехода к более гибким стратегиям, включая построение правила останова — критерия, определяющего момент завершения сбора данных, когда достигнута реконструкция удовлетворительного качества.

Следует отметить, что правила останова, выбор метрик качества и архитектура протокола могут существенно различаться в зависимости от поставленной задачи. Однако, систематическое исследование таких протоколов в рамках конкретных задач — в частности, бинаризации — остаётся недостаточно проработанным.

Актуальность указанных проблем определяет цель настоящего исследования.

Цель работы — разработка и исследование протоколов мониторинговой реконструкции в рамках задачи бинаризации.

Для достижения данной цели были поставлены следующие задачи:

1. Выбор алгоритмов бинаризации и метрик, подходящих для практического применения в условиях мониторинговой реконструкции.
2. Подбор и генерация набора изображений для исследования поведения бинаризации в процессе томографии под контролем реконструкции.
3. Построение правила останова.
4. Проведение численных экспериментов и анализ их результатов

Объектом исследования является процесс томографии под контролем реконструкции и процедура бинарной сегментации полученных

реконструкций.

Предметом исследования является разработка правила останова, определяющего минимально необходимый набор углов сканирования для получения качественной бинарной реконструкции.

1 Методы бинаризации и протоколы томографии под контролем реконструкции

В рамках текущей главы будут сформулированы основные определения, необходимые в исследовании протоколов мониторинговой реконструкции в рамках задачи бинаризации. Также проведен литературный обзор алгоритмов бинаризации.

1.1 Томография под контролем реконструкции

Компьютерная томография — это один из наиболее эффективных неразрушающих методов исследования внутренней структуры объектов. В процессе компьютерной томографии объект просвечивается рентгеновским излучением под разными углами, а ослабленное излучение регистрируется детектором. Каждое такое измерение называется проекцией. При достаточном количестве проекций возможно воссоздать объёмное изображение внутренней структуры объекта.

Метод компьютерной томографии широко применяется в медицине, промышленности, материаловедении и других областях, где требуется неразрушающее исследование внутренней структуры. Он позволяет выявлять дефекты, измерять геометрию, строить цифровые модели и анализировать многослойные объекты. Точность и универсальность метода делают его важным инструментом как в научных исследованиях, так и в прикладных задачах.

Одной из наиболее актуальных проблем компьютерной томографии является доза излучения, которой подвергается объект в процессе исследования. Многие материалы в индустриальной томографии достаточно чувствительны к излучению и могут охрупчать или разрушиться впоследствии сканирования. В медицине эта проблема не менее актуальна, так как превышение минимально допустимой дозы излучения сопряжено с рисками для здоровья.

На основе этой проблемы сформулирована одна из наиболее актуальных задач в компьютерной томографии - снижение дозы излучения, которой подвергается объект исследования, по принципу ALARA

(As Low As Reasonably Achievable) [2].

Тесно связанной задачей в компьютерной томографии является снижение времени, необходимого для исследования объекта. Эта проблема особенно актуальна в индустриальной томографии, где такой процесс может длиться более часа.

Одним из подходов к решению сформулированных задач является томография под контролем реконструкции, также называемая мониторинговой реконструкцией.

В таком подходе проекционные данные собираются постепенно и после каждой новой итерации выполняется реконструкция по имеющимся проекциям. Далее оценивается качество полученной реконструкции. В случае удовлетворительного результата реконструкции процесс сканирования может быть остановлен досрочно.

Таким образом томография под контролем реконструкции может рассматриваться как anytime-процесс, при котором результат постепенно улучшается по мере накопления проекций, и на любом этапе можно принять решение об остановке [3]. Такой подход является уместным когда цена вычислений, в данном случае доза облучения и количество времени на исследование, сопоставимы с ценой неточностей и ошибок в реконструкции.

В рамках данной работы будет исследован процесс томографии под контролем реконструкции в рамках задачи бинарной сегментации реконструкции. Во многих прикладных задачах конечной целью томографии является не визуализация самой реконструкции, а получение бинарной маски для построения САД-модели, обнаружения дефектов и других задач, требующих выделения объекта исследования и его дальнейшей обработки.

1.2 Математическая формулировка задачи бинаризации

Базовым объектом, изучаемым в рамках текущей работы, является изображение.

Изображением размера $n = (n_1, n_2, n_3)$ будем называть отображение $I_n^m : P_3 \rightarrow G$, в котором элементы области определения $P \subset Z^3$ называются пикселями (вокселями), а $n_i \in N, i = 1, 2, 3$ и $(G, +)$ - аддитивная абелева

группа [4].

Формально задачу бинаризации, также называемую бинарной сегментацией, можно поставить следующим образом [5]: Дано изображение I_n^m и предикат однородности $H : P \rightarrow 0, 1$. Найти такие два конечных, непересекающихся подмножества P_1, P_2 множества P , которые удовлетворяют следующим условиям:

1. Объединение множеств P_1 и P_2 является исходным множеством P :
$$P_1 \cup P_2 = P$$
2. Множества P_1 и P_2 являются однородными по предикату H :
$$H(P_1) = H(P_2) = 1$$
3. Объединение множеств P_1 и P_2 не однородно: $H(P_1 \cup P_2) = 0$

Результатом решения задачи бинаризации является бинарная маска изображения I_n^m . Бинарной маской изображения I_n^m будем называть отображение $\tilde{I}_n^m : P \rightarrow \{0, 1\}$, множество определения которого совпадает с множеством определения I_n^m .

Множеством объекта назовём множество пикселей, при которых бинарная маска изображения принимает значение 1: $\{x \in P | \tilde{I}_n^m(x) = 1\}$.

Далее проведем литературный обзор и рассмотрим существующие алгоритмы бинаризации.

1.3 Методы бинаризации

В рамках данной работы рассматривается поведение сегментации изображения реконструкции в процессе томографии под контролем реконструкции. При этом выбор алгоритмов сегментации является важным шагом для дальнейшего проведения экспериментов.

С целью выбора алгоритма необходимо провести обзор доступных алгоритмов сегментации.

В научной периодике доступно большое количество публикаций на тему алгоритмов сегментации. Исследование [6] показало устойчивый рост количества ежегодно предложенных алгоритмов с 1995 года по 2006 года.

Систематизация и классификация доступных алгоритмов необходима для выбора наиболее подходящих для проведения экспериментов.

Хорошей отправной точкой в изучении доступных алгоритмов сегментации является рассмотрение схем их классификации.

Существует множество различных схем классификации алгоритмов сегментации. Например, часто используется классификация по одному признаку, такому как способ обработки изображения или степень участия человека в процессе сегментации [7].

В работе [8] также предложена схема обобщённой классификации, объединяющая несколько одно-признаковых подходов в единую структуру.

В рамках проведённого литературного обзора поиск алгоритмов сегментации был проведен на основе их классификации по принципу работы.

Первым классом алгоритмов, встречающихся в литературе, является семейство пороговых алгоритмов.

Пороговые алгоритмы бинаризации выполняют классификацию вокселей изображения на основе заданного порогового значения интенсивности. Воксели с интенсивностью ниже порога относятся к фону, тогда как воксели с интенсивностью выше или равной порогу классифицируются как принадлежащие объекту.

Существует большое количество разнообразных пороговых алгоритмов, основные из которых изложены в работы [7].

Например, часть алгоритмов устанавливают порог глобально, другие - устанавливают его индивидуально для каждого пикселя.

Другим широко используемым подходом к сегментации является метод роста области (region growing), основанный на объединении вокселей с близкими характеристиками по мере распространения от заданных начальных точек [9].

Этот метод является полуавтоматическим, то есть он требует участия пользователя в процессе своей работы. В частности требуется ввести набор начальных точек.

В литературе встречается множество модификаций этого метода, включая автоматический вариант алгоритма, не требующий указания

начальной точки [10].

Несмотря на такое множество модификаций, метод роста области достаточно трудно реализуем в контексте задачи сегментации под контролем реконструкции.

Полуавтоматический характер метода и ресурсоёмкость его автоматической модификации делают его непрактичным для использования в рамках данного исследования.

Следующим направлением, широко представленным в литературе, являются методы кластеризации. Схожесть формальных постановок задач сегментации и кластеризации способствует применению кластеризационных алгоритмов в контексте сегментации.

В частности, алгоритмы K-means [11] и mean shift [12] нередко применяются при решении задач сегментации.

Применимость алгоритмов кластеризации к сегментации изображений реконструкции требует дополнительного анализа, поскольку реализация их вычислительно эффективных версий представляет собой нетривиальную задачу.

Представленные ранее классы алгоритмов во многом однородны по своей структуре и принципам работы. В то время как последующие группы объединяют существенно более разнородные методы, классифицированные по более общим признакам.

Такой группой алгоритмов являются методы на базе теории графов [13].

Принципы работы этих методов достаточно разнообразны. Некоторые работы применяют алгоритмы поиска разрезов в графе [14; 15], другие работы адаптируют алгоритм поиска максимального потока под задачу сегментации [16].

Также встречаются и нестандартные подходы к сегментации в этой группе алгоритмов.

Один из таких нестандартных подходов предложен в работе [17], где реализован алгоритм, концептуально близкий к Unseeded Region Growing, но основанный на представлении изображения в виде взвешенного графа.

Следующей группой являются вероятностные алгоритмы. Прин-

цип работы методов в этой группе основан на некотором априорном предположении о распределении значений пикселей в рамках изображения.

На пример методы, изложенные в работах [18; 19] основаны на предположении, что значение пикселей объектов в изображении имеет распределение вейбюля.

Последней крупной группой алгоритмов, встречающейся в литературе, это методы на основе машинного обучения.

К алгоритмам на основе машинного обучения относятся как нейросетевые методы, применяемые в общей задаче сегментации изображений [20; 21], так и подходы, адаптированные под изображения реконструкции в рамках томографии [22].

Также встречаются менее распространённые методы, включая клеточные нейронные сети [23] и отдельные примеры полуавтоматических алгоритмов, использующих классические модели, такие как метод опорных векторов [24; 25].

Некоторые выявленные в литературе методы не вписываются в описанные выше категории и базируются на оригинальных, зачастую уникальных подходах.

Алгоритмы этого типа представлены в небольшом числе работ и не формируют обособленного направления.

Одним из таких алгоритмов является Полуавтоматический SegMo [26], разработанный для сегментации изображений реконструкции индустриальной томографии.

Сам алгоритм достаточно сложный в реализации и требует активного участия пользователя, соответственно не подходит для исследования томографии под контролем реконструкции, однако авторы заявляют достаточно высокое качество сегментации на выходе.

Среди редких методов также встречаются алгоритмы на основе множеств уровня, использующие эволюцию поверхностей для выделения объектов в объёме. В работе [27] такой подход применён для трёхмерной сегментации сосудистой системы на данных магнитно-резонансной ангиографии.

В рамках литературного обзора были изучены пороговые алгоритмы, методы роста области, алгоритмы на основе кластеризации, теории

графов, вероятностные подходы, а также методы, использующие машинное обучение и нестандартные оригинальные принципы.

Из большого разнообразия доступных алгоритмов была выбрана группа пороговых методов.

Причина выбора пороговых методов заключается в их высокой степени изученности. В литературе представлено множество работ, посвящённых их модификации, анализу и практическому применению, обладающих высокой цитируемостью.

Применение алгоритмов других групп в контексте томографии под контролем реконструкции сопряжено с рядом ограничений.

Многие из них являются полуавтоматическими и требуют участия пользователя, что делает их непригодными для включения в итеративный процесс реконструкции.

Классы алгоритмов с высокой вычислительной сложностью, такие как методы кластеризации и алгоритмы на основе множеств уровня, затруднительно использовать в силу необходимости их повторного запуска на каждой итерации томографического сканирования.

Методы, основанные на нейронных сетях, требуют большого объёма размеченных данных, которые на текущий момент недоступны для рассматриваемой задачи.

Следующим этапом после выбора класса является формирование перечня конкретных алгоритмов, из которых будет производиться отбор для последующего эксперимента.

Следующая глава будет посвящена сбору необходимых материалов для конвейера экспериментов и его реализации.

2 Подготовка данных и окружения для численных экспериментов в рамках протокола мониторинговой реконструкции

В данной главе будет описан процесс подготовки входных данных, выбран набор алгоритмов бинаризации, проведён анализ метрик качества на модельных ошибках и сформирован итоговый набор. Затем будет спроектирован и реализован программный конвейер экспериментов, включающий этапы реконструкции, бинаризации, оценки и визуализации результатов.

В целях исследования итеративного процесса томографии под контролем реконструкции с точки зрения качества сегментации необходимы следующие компоненты:

1. Набор данных, на которых проводятся эксперименты.
2. Алгоритмы сегментации.
3. Метрики для оценки качества сегментации.

Далее рассмотрим каждую из этих компонент.

2.1 Подготовка данных для экспериментов

Под набором данных в рамках данной работы подразумевается множество из n изображений $(I_m^n)_1, (I_m^n)_2, \dots, (I_m^n)_n$.

Так как основной целью работы является исследование правила останова в итеративной томографии под контролем реконструкции, набор данных должен моделировать различные варианты соотношения между углами сканирования и угловой структурой изображений.

Большинство изображений в наборе данных являются синтетическими и были сгенерированы с помощью языка программирования Python.

Набор данных состоит из десяти изображений, сгруппированных по два на основе схожести моделируемой угловой структуры.

Первые 4 пары изображений имеют одинаковый размер — 256 срезов, каждый из которых 512×512 пикселей. Объём каждого изображения соответственно имеет размер $512 \times 512 \times 256$.

Первая пара представляет собой два изображения реконструкции бетонных плит, предоставленные в работе [28].

Изображение “Бетон-1” содержит характерные горизонтально ориентированные дефекты, интерпретируемые как трещины. На рисунке 1 приведён срез этого изображения и его трёхмерная визуализация.

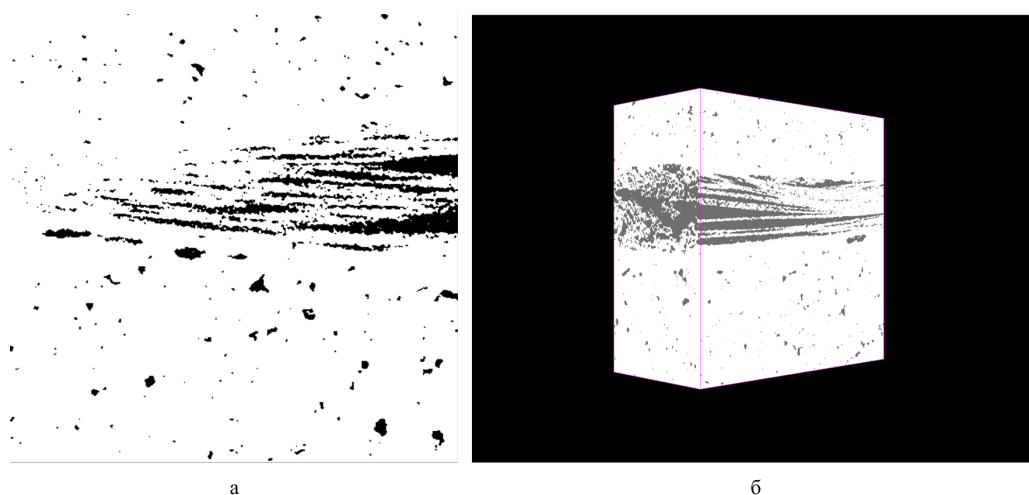


Рисунок 1 Изображение реконструкции “Бетон-1”; а — срез трёхмерного изображения, б — трёхмерная визуализация

Изображение “Бетон-2” представляет собой бетонную плиту с дефектом типа “раковина” (пустое пространство внутри плиты).

На рисунке 2 приведён срез изображения “Бетон-2” и соответствующая трёхмерная модель.

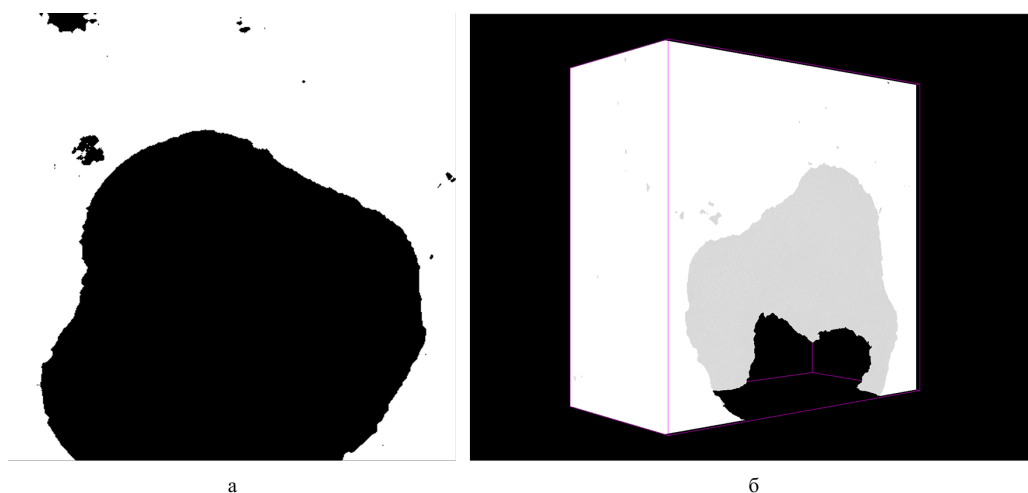


Рисунок 2 Изображение реконструкции “Бетон-2”; а — срез трёхмерного изображения, б — трёхмерная визуализация

В этих изображениях стоит отметить преобладание объекта над фоном по количеству пикселей.

Следующей парой объектов в наборе данных являются синтетические изображения трёхмерной решётки, сформированной пересекающимися семействами параллельных плоскостей.

В одном случае решётка строго ортогональна координатным осям. Это изображение в дальнейшем обозначается как “Решётка”. На рисунке 3 представлен срез её реконструкции и объёмная модель.

Такой объект включён в набор данных как модель идеального случая: углы ориентации структур внутри объекта полностью совпадают с начальными углами сканирования в процессе томографии под контролем реконструкции, соответственно процесс должен остановиться уже при небольшом количестве проекций.

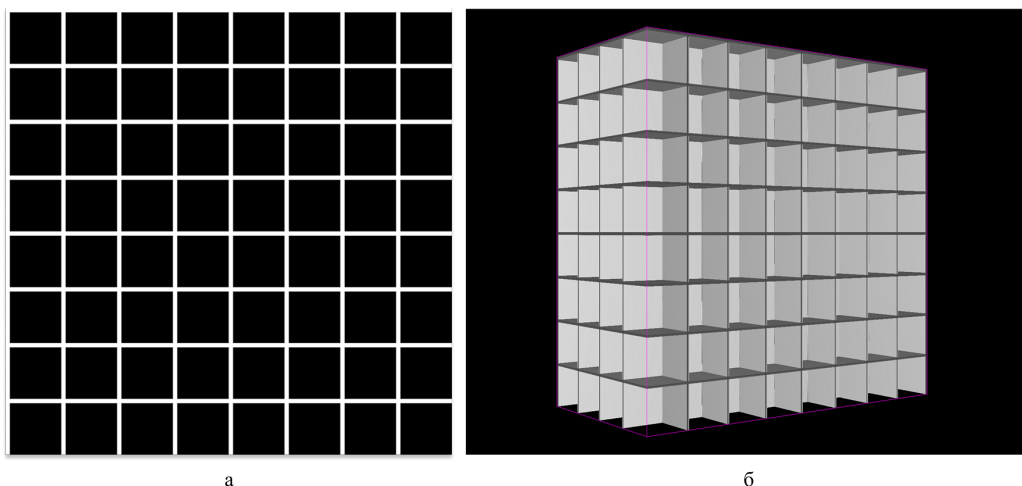


Рисунок 3 Изображение реконструкции "Решётка"; а — срез трёхмерного изображения, б — трёхмерная визуализация

Вторая решётка, обозначаемая как "Наклонная решётка", формируется по тому же принципу, что и предыдущая, но все её плоскости наклонены относительно координатных осей.

Каждое семейство плоскостей в этой решётке изначально было ортогонально одному из направлений декартовой системы координат. Затем к каждому семейству был применён поворот вокруг одной из осей: первое семейство — на угол $1,31$ радиан вокруг оси X , второе — на угол $0,13$ радиан вокруг оси Y , третье — на угол $1,31$ радиан вокруг оси Z .

Срез изображения и трёхмерная визуализация приведены на рисунке 4.

Данный объект моделирует случай, когда начальный набор углов сканирования не совпадает с ориентацией внутренних структур объекта. При этом число таких ориентаций ограничено: решётка состоит из трёх групп плоскостей, каждая из которых наклонена под фиксированным углом.

Ожидается, что при включении в процесс реконструкции проекций под соответствующими углами произойдёт резкое улучшение качества бинаризации.

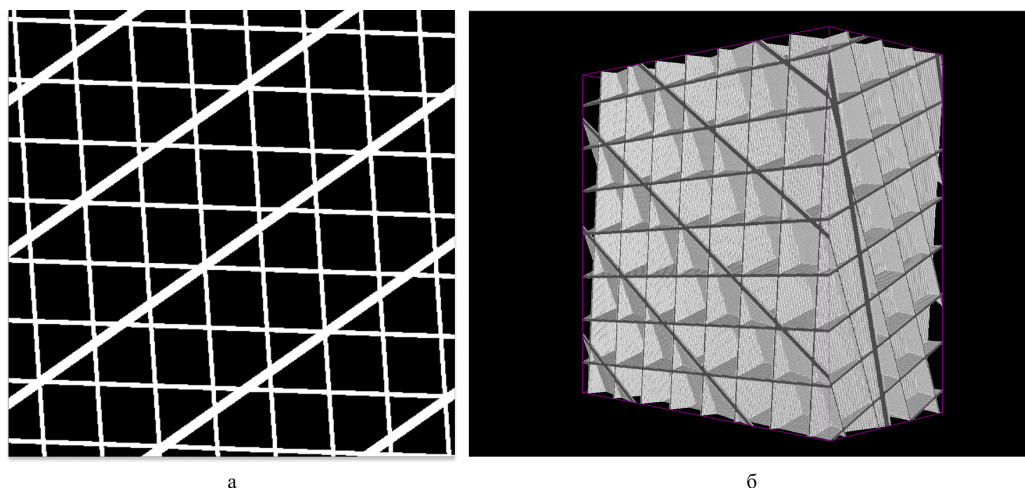


Рисунок 4 Изображение реконструкции "Наклонная решётка"; а — срез трёхмерного изображения, б — трёхмерная визуализация

Далее рассмотрим пару синтетических изображений, содержащих объекты с плавными очертаниями, без резких переходов или острых углов.

В отличие от структур с резкими границами, для них не ожидается резкого скачка качества при включении определённых углов сканирования — качество улучшается постепенно с включением большего количества проекций.

Первое изображение из этой пары — "Эллипсы". Оно представляет собой объём, разделённый на 27 параллелепипедов, в каждый из которых вписан трёхмерный эллипсоид. Размеры эллипсоидов варьируются, и в случае, если фигура не помещается в отведённую область, её выходящие за границы части отсекаются.

Срез и трёхмерная визуализация этого изображения представлены на рисунке 5.

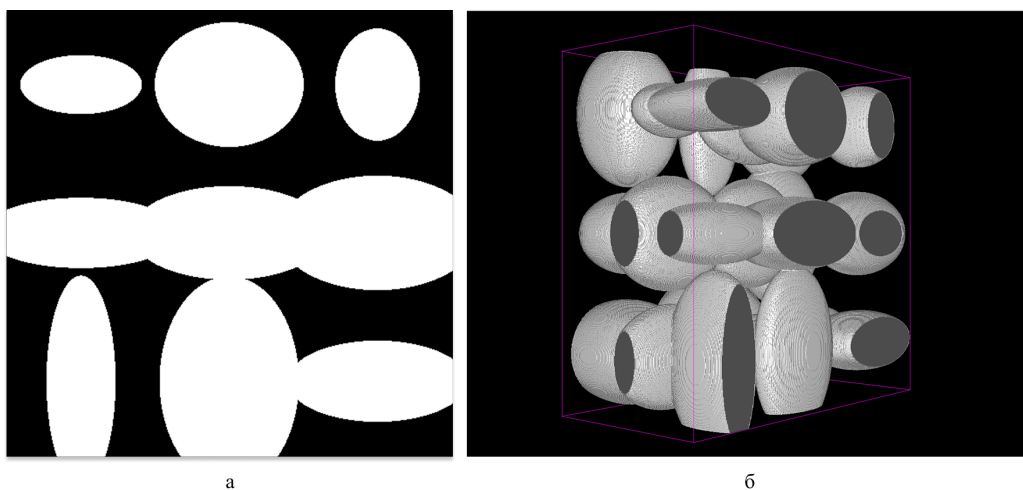


Рисунок 5 Изображение реконструкции "Эллипсы"; а — срез трёхмерного изображения, б — трёхмерная визуализация

Вторым изображением из данной пары является "Гауссиана". Оно представляет собой синтетическую трёхмерную структуру, состоящую из двух объёмных образований, форма которых аппроксимирует гауссовы колокола.

Первая фигура расположена в нижней части объёма, начинаясь с плоскости $z = 0$, и ориентирована так, что её максимум находится вблизи центра, а ось симметрии направлена вверх. Вторая фигура начинается с верхней части объёма, от плоскости $z = 255$, и ориентирована в противоположную сторону — вниз.

Обе фигуры подвергнуты повороту: первая — вокруг оси x , вторая — вокруг оси y . Угол поворота в обоих случаях составляет $\frac{\pi}{22}$ радиан.

Рисунок 6 иллюстрирует изображение "Гауссиана": слева показан его срез, справа — трёхмерная визуализация.

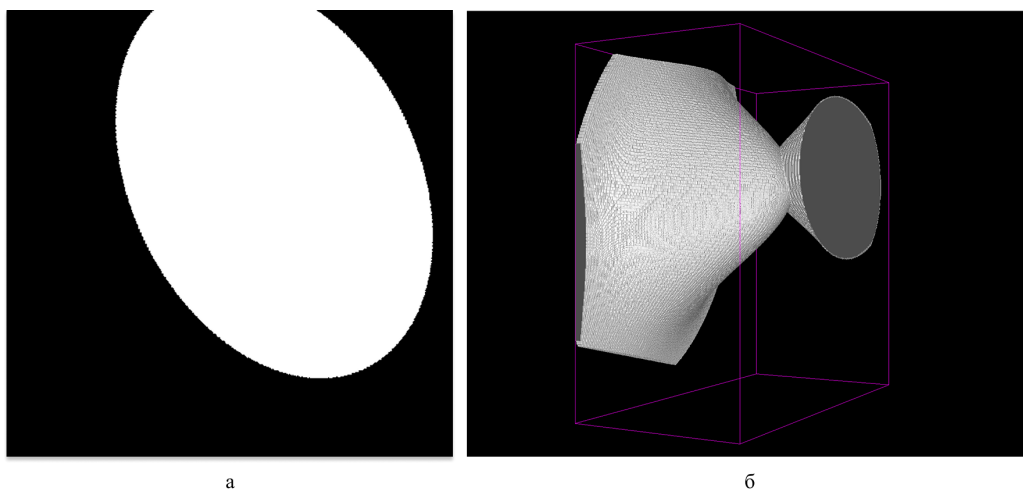


Рисунок 6 Изображение реконструкции "Гауссиана"; а — срез трёхмерного изображения, б — трёхмерная визуализация

Следующая пара синтетических изображений является противоположностью предыдущей — состоит из резких углов и ровных линий.

У таких изображений ожидаются скачки в качестве реконструкции при включении проекций под соответствующими углами. В отличие от пары "Решётка" и "Наклонная решётка", таких скачков может быть несколько, поскольку изображение содержит несколько трёхмерных объектов с различными ориентациями.

В состав изображения входят две пересекающиеся пирамиды: одна с прямоугольным основанием, другая — с треугольным. Оси симметрии пирамид ориентированы по-разному: первая повёрнута вокруг оси z , вторая — вокруг оси y .

В дополнение к ним в свободное пространство между пирамидами встроен трёхмерный гексагон, не пересекающийся с другими объектами. Все фигуры размещены в объёме случайным образом, что исключает регулярность и добавляет сложности к реконструкции.

В случае первого изображения "Полигоны-1", приведённого на рисунке 7, пирамида с прямоугольным основанием имеет поворот вокруг оси z на угол $1,005$ радиан, а пирамида с треугольным основанием имеет угол поворота вокруг оси y , равный $2,89$ радиан.

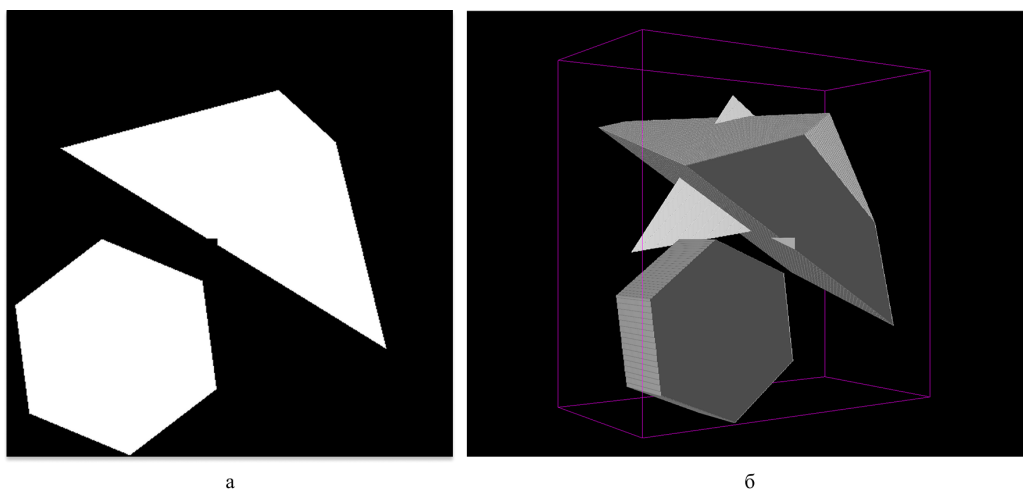


Рисунок 7 Изображение реконструкции "Полигоны-1"; а — срез трёхмерного изображения, б — трёхмерная визуализация

Второе изображение — "Полигоны-2" — проиллюстрировано на рисунке 8. В нём пирамида с прямоугольным основанием повёрнута вокруг оси z на угол $4,71$ радиан, а пирамида с треугольным основанием — вокруг оси y на угол $2,89$ радиан.

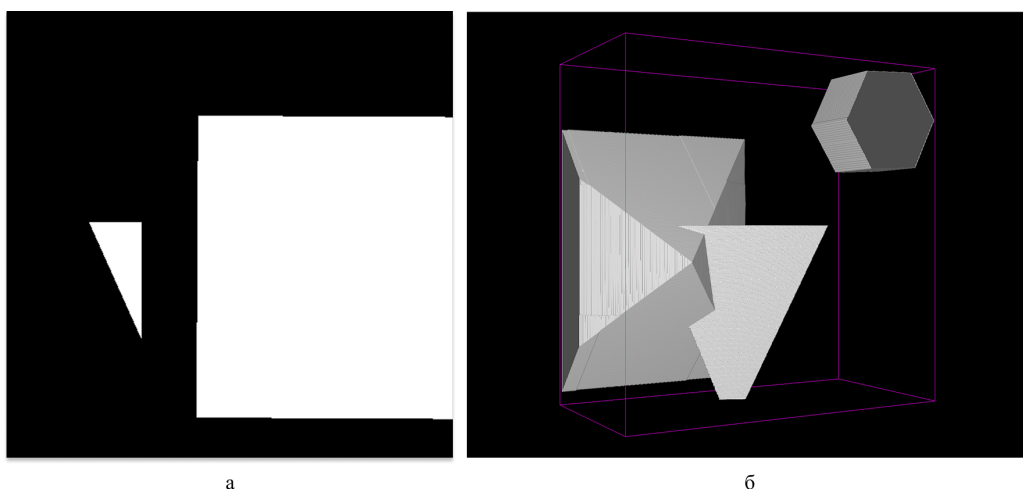


Рисунок 8 Изображение реконструкции "Полигоны-2"; а — срез трёхмерного изображения, б — трёхмерная визуализация

Последней парой изображений являются трехмерные модели построенные из изображений реконструкции при сканировании объектов с помощью сканера Cyberware 3030 MS в лаборатории компьютерной графики Стэнфордского университета [29].

Данные изображения представляют собой сканы реальных объектов с высокой геометрической детализацией, что позволяет оценить поведение правила останова на данных, приближённых к практическим

условиям применения.

Первым изображением из данной пары, которое назовем "Кролик", является скан фигурки кролика и состоит из 100 срезов размером 128×129 , соответственно размер объема $128 \times 129 \times 100$.

Один из срезов изображения и его трехмерная визуализация представлены в рисунке 9.

Фигурка является полой, соответственно особенность данного изображения является преобладание объекта над фоном. Это делает данное изображение противоположностью изображений "Бетон-1" и "Бетон-2".

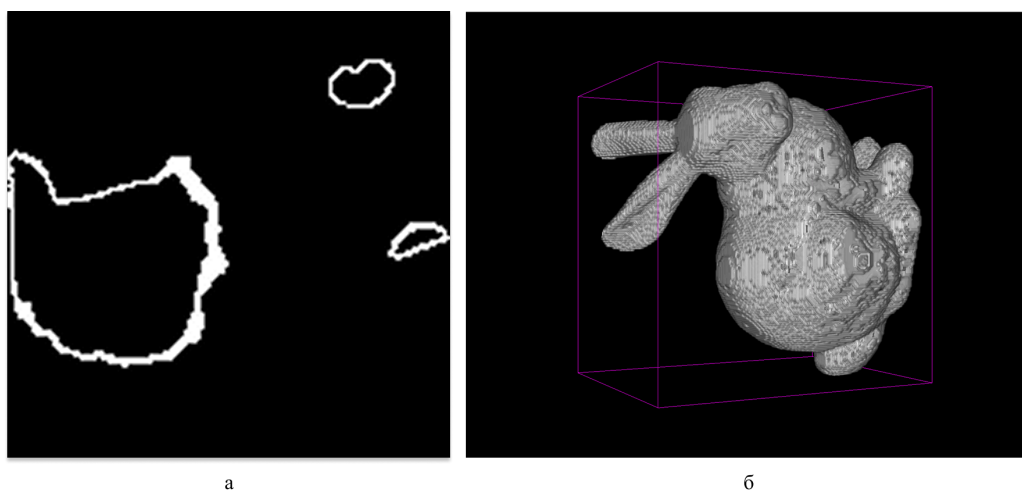


Рисунок 9 Изображение реконструкции "Кролик"; а - срез трёхмерного изображения, б - трехмерная визуализация

Последним изображением является скан небольшой тайской статуэтки и представлено на рисунке 10. Трехмерное изображение формируется из 132 срезов размером 257×153 пикселей, соответственно полный объём имеет размер $257 \times 153 \times 132$ пикселя.

Её особенностью являются достаточно сложные границы объекта. В отличие от изображения "Кролик", данное изображение, которое назовем "Тайская статуэтка", не является полым.

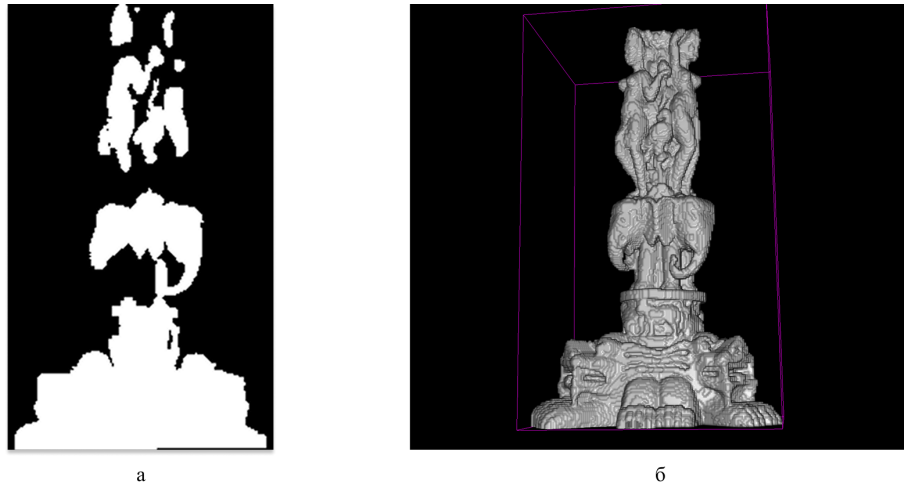


Рисунок 10 Изображение реконструкции "Тайская статуэтка"; а - срез трёхмерного изображения, б - трехмерная визуализация

Собранный набор данных, состоящий из десяти изображений с разнообразной структурой, охватывает широкий спектр сценариев, возникающих при томографической реконструкции: от регулярных и идеально выровненных структур до объектов с произвольной ориентацией и сложной геометрией.

Такой набор данных формирует основу исследования, поскольку исследование эффективности правила останова возможно лишь на сбалансированном наборе данных.

Далее будем рассматривать остальные компоненты, необходимые для конвейера основных экспериментов.

2.2 Алгоритмы бинаризации

С учётом необходимости многократного применения в процессе итеративной реконструкции, для экспериментов были выбраны пороговые алгоритмы, как простые, устойчивые и не требующие сложной настройки.

Основная идея пороговых алгоритмов достаточно проста.

Пусть дано изображение I_m^n размера $n = (n_1, n_2, n_3)$.

Выходом алгоритма будет сегментация \tilde{I}_m^n изображения I_m^n , совпадающая размером с исходным изображением.

Значение каждого пикселя сегментации \tilde{I}_m^n определяется по следующей формуле:

$$\tilde{I}_m^n(i, j, k) = \begin{cases} 1, I_m^n(i, j, k) \geq t \\ 0, I_m^n(i, j, k) < t \end{cases} \quad (1)$$

Соответственно, в сегментации пиксель (i, j, k) классифицируется как объект если значение исходного изображения в этом пикселе имеет значение больше или равное некоторому порогу t , иначе этот пиксель классифицируется как фон.

В зависимости от характера порога t выделяют глобальные и локальные пороговые алгоритмы.

В глобальном случае порог не зависит от пикселя, соответственно все пиксели сравниваются с одним значением порога.

Таковыми алгоритмами являются классический пороговый алгоритм и алгоритм Отсу.

В классическом пороговом алгоритме порог является параметром, соответственно он требует его априорной оценки.

Алгоритм Отсу [30] определяет оптимальное значение порога t , максимизируя межклассовую дисперсию:

$$\sigma^2(t) = \omega_0(t)\omega_1(t) [\mu_0(t) - \mu_1(t)]^2, \quad (2)$$

где $\omega_0(t)$ и $\omega_1(t)$ — вероятности (доли) фона и объекта при пороге t , а $\mu_0(t)$ и $\mu_1(t)$ — соответствующие средние значения интенсивности.

Алгоритм перебирает возможные значения t и выбирает то, при котором значение $\sigma^2(t)$ максимизируется.

Локальные пороговые алгоритмы определяют порог для каждого пикселя, соответственно порог t становится функцией от пикселя $t = t(i, j, k)$.

Распространенным локальным пороговым алгоритмом является метод Ниблэка [31].

Алгоритм Ниблэка определяет порог t для пикселя (i, j, k) по следующей формуле:

$$t(i, j, k) = \mu_r(i, j, k) + k\sigma_r(i, j, k) \quad (3)$$

где $\mu_r(i, j, k)$ и $\sigma_r(i, j, k)$ - среднее и среднеквадратичное отклоне-

ние интенсивности в окрестности r пикселя (i, j, k) .

В общем случае среднее и среднеквадратичное отклонение в окрестности r пикселя (i, j, k) рассчитываются по следующим формулам:

$$\mu_r(i, j, k) = \frac{1}{(2r+1)^3} \sum_{x=i-r}^{i+r} \sum_{y=j-r}^{j+r} \sum_{z=k-r}^{k+r} I_m^n(x, y, z) \quad (4)$$

$$\sigma_r(i, j, k) = \sqrt{\frac{1}{(2r+1)^3} \sum_{x=i-r}^{i+r} \sum_{y=j-r}^{j+r} \sum_{z=k-r}^{k+r} (I_m^n(x, y, z) - \mu_r(i, j, k))^2} \quad (5)$$

Соответственно окном является куб с длинной сторон $2r+1$.

Проблемной частью формул 4 и 5 является ситуация, когда окно частично выходит за границы изображения.

В таких случаях применяются следующие стратегии обработки:

1. Обрезка окна по границам. Вычисления проводятся только по той части окна, которая полностью попадает внутрь изображения.
2. Задание фиксированного значения. За пределами изображения значения пикселей считаются равными фиксированной константе, например, нулю или среднему значению изображения.
3. Отражение по границе. Отсутствующие значения заполняются за счёт зеркального отражения пикселей относительно соответствующей границы изображения.

У каждой стратегии есть свои преимущества и недостатки. Её выбор зависит от характера входных изображений.

У классического алгоритма Ниблэка существует множество модификаций.

Таковыми модификациями являются, например, алгоритмы Сауволя [32] и Фансалкара [33].

В рамках данной работы выбран аффинный вариант алгоритма Ниблэка, описанный в работе [34].

Метод аффинного Ниблэка добавляет в уравнение 3 дополнительный параметр β , который является глобальной оценкой шума.

$$t(i, j, k) = \mu_r(i, j, k) + \sigma_r(i, j, k) + \beta \quad (6)$$

Такая модификация является вычислительно эффективной так как к формуле добавляется константа, однако она добавляет дополнительный параметр.

Таким образом, итоговый набор алгоритмов, используемых в конвейере экспериментов, включает в себя:

1. Классический пороговый алгоритм
2. Алгоритм Отсу
3. Алгоритм аффинного Ниблека

Выбор данных алгоритмов обусловлен их высокой степенью изученности в литературе и активным применением в задачах компьютерной томографии.

Поскольку алгоритм бинаризации запускается на каждой итерации томографии под контролем реконструкции, его вычислительная эффективность напрямую влияет на общую скорость всего процесса.

Следующей необходимой для экспериментов компонентой являются метрики, позволяющие оценить качество полученных бинарных масок.

2.3 Метрики качества и оценка их чувствительности

Последним необходимым элементом экспериментального конвейера является набор метрик оценки качества сегментации.

Метрики должны быть подобраны так, чтобы в совокупности охватывать ключевые аспекты качества бинарных масок — например, точность границ и степень перекрытия с эталоном.

Такой набор метрик должен выявлять различные типы ошибок и предоставлять более объективную оценку результата.

Хотя в области машинного обучения и компьютерного зрения существует множество метрик, не все из них применимы к задачи сегментации изображений.

Например, метрики, ориентированные на числовые значения, текстовые последовательности или графовые структуры, не учитывают пространственные особенности изображений и потому неинформативны в контексте сегментации.

Среди метрик, предназначенных для работы с изображениями, встречается избыточность: несколько показателей могут быть чувствительны к одним и тем же видам ошибок.

В конвейер экспериментов требуется сформировать минимально необходимый набор метрик, который позволит обнаружить распространённые типы ошибок.

Чтобы выбрать такой набор метрик обоснованно, необходимо провести серию экспериментов на модельных данных.

Такие эксперименты позволят изучить поведение метрик при различных типах ошибок.

Для реализации таких экспериментов необходимы следующие шаги:

1. Собрать начальный набор метрик для исследования.
2. Определить набор ошибок, которые должны обнаружить метрики.
3. Сформировать модельные изображения для набора ошибок

Часть метрик для начального набора были найдены в работе [35]. Авторы данного исследования изложили распространённые метрики, применяемые в оценке качества сегментации изображений реконструкции в медицине.

Из таких метрик в начальный набор данных будут включены метрики DICE, Intersection Over Union (IOU), Mean Square Error (MSE).

Пусть даны эталонная бинарная маска $(\tilde{I}_m^{\tilde{n}})_{gt}$ изображения и его сегментация $(\tilde{I}_m^{\tilde{n}})_{seg}$.

Обозначим как $(\tilde{I}_m^{\tilde{n}})_{gt} \cap (\tilde{I}_m^{\tilde{n}})_{seg}$ множество пикселей, на которых значение сегментации и эталонной бинарной маски совпадают и равно 1: $(\tilde{I}_m^{\tilde{n}})_{gt} \cap (\tilde{I}_m^{\tilde{n}})_{seg} := |\{x \in P | (\tilde{I}_m^{\tilde{n}})_{gt}(x) = (\tilde{I}_m^{\tilde{n}})_{seg}(x) = 1\}|$.

Определим также $(\tilde{I}_m^{\tilde{n}})_{gt} \cup (\tilde{I}_m^{\tilde{n}})_{seg}$ как множество пикселей, на которых значение сегментации или эталонной бинарной маски равно 1: $(\tilde{I}_m^{\tilde{n}})_{gt} \cup (\tilde{I}_m^{\tilde{n}})_{seg} := |\{x \in P | (\tilde{I}_m^{\tilde{n}})_{gt}(x) = 1 \vee (\tilde{I}_m^{\tilde{n}})_{seg}(x) = 1\}|$.

Тогда метрика DICE определяется как следующей формулой:

$$DICE((\tilde{I}_m^n)_{seg}, (\tilde{I}_m^n)_{gt}) = \frac{2 \cdot |(\tilde{I}_m^n)_{gt} \cap (\tilde{I}_m^n)_{seg}|}{|(\tilde{I}_m^n)_{gt}| + |(\tilde{I}_m^n)_{seg}|} \quad (7)$$

где $|(\tilde{I}_m^n)|$ является мощностью множества пикселей, на которых маска (\tilde{I}_m^n) имеет значение 1: $|(\tilde{I}_m^n)| = |\{x \in P | (\tilde{I}_m^n)(x) = 1\}|$.

Метрика DICE принимает значения в отрезке от 0 до 1, где 0 - полное несовпадение изображений, а 1 - их полное совпадение.

Индекс DICE является одной из классических метрик перекрытия — класса метрик, которые измеряют степень пересечения двух множеств (в данном случае бинарных масок).

Другой растрасстраненной метрикой перекрытия является IOU, также известная как метрика Жаккарда. Она определяется слудующей формулой:

$$IOU((\tilde{I}_m^n)_{seg}, (\tilde{I}_m^n)_{gt}) = \frac{|(\tilde{I}_m^n)_{gt} \cap (\tilde{I}_m^n)_{seg}|}{|(\tilde{I}_m^n)_{gt} \cup (\tilde{I}_m^n)_{seg}|} \quad (8)$$

Аналогично метрики DICE, значения IOU находятся в отрезке от 0 до 1, где 0 - полное несовпадение изображений, а 1 - их полное совпадение.

Последняя метрика перекрытия несколько менее распрастраненная - Symmetric Boundary DICE (SBD), описанной в работе [36].

Чтобы определить эту метрику необходимо ввести понятия r окрестности пикселя x и границы изображения.

Окрестностью радиуса r пикселя $x \in P$ маски \tilde{I}_m^n будем называть множество

$$M_r(x) = \left\{ x' \in P \left| \max_{i \in \{0,1,2\}} |x_i - x'_i| \leq r \right. \right\}, \quad (9)$$

то есть все пиксели, находящиеся в дискретной окрестности Мура радиуса r от x .

Размер этой окрестности напрямую влияет на чувствительность метрики: при малом r метрика фиксирует только локальные несоответствия границ, а при большом — начинает учитывать отклонения на большем расстоянии, сглаживая мелкие ошибки.

Границей $\partial \tilde{I}_m^n$ маски \tilde{I}_m^n будем называть множество пикселей,

окрестность радиуса r которых содержит как пиксели фона, так и пиксели объекта:

$$\partial I_m^n = \{x \in P \mid \exists x' \in M_r(x) : \tilde{I}_m^n(x') = 0 \wedge \exists x'' \in M_r(x) : \tilde{I}_m^n(x'') = 1\} \quad (10)$$

Заметим, что так как метрики сравнивают бинарную маску $(\tilde{I}_m^n)_{seg}$ и эталонную маску $(\tilde{I}_m^n)_{gt}$ одного и того же исходного изображения, то у них одинаковая область определения P .

Это позволяет выбрать одну окрестность радиуса r пикселя x и сравнивать значения пикселей из этого множества в обоих изображениях.

Обозначим через $DICE(M_r(x))$ значение метрики dice на множестве $M_r(x)$:

$$DICE(M_r(x)) = DICE[(\tilde{I}_m^n)_{seg}(M_r(x)), (\tilde{I}_m^n)_{gt}(M_r(x))] \quad (11)$$

Определив необходимые понятия можно перейти к определению метрики Symmetric Boundary DICE:

$$SBD((\tilde{I}_m^n)_{seg}, (\tilde{I}_m^n)_{gt}) = \frac{\sum_{x \in \partial(\tilde{I}_m^n)_{seg}} DICE(M_r(x)) + \sum_{x \in \partial(\tilde{I}_m^n)_{gt}} DICE(M_r(x))}{|\partial(\tilde{I}_m^n)_{gt}| + |\partial(\tilde{I}_m^n)_{seg}|} \quad (12)$$

В числителе формулы суммируются значения метрики Dice, вычисленные в окрестностях радиуса r каждого граничного пикселя как сегментированной маски, так и эталонной. Знаменатель представляет собой общее количество таких граничных пикселей, обеспечивая нормировку результата. Параметр r задаёт радиус окрестности, используемой для локального сравнения, и определяется ранее в формуле (9).

Следующая метрика в начальном наборе - метрика Normalized Hausdorff - отличается от предыдущих двух по принципу сравнения.

Данная метрика основана не на площади перекрытия, а на расстоянии между множествами объекта в сегментации и в эталонной бинарной маске.

Чтобы определить метрику Normalized Hausdorff, определим рас-

стояние между пикселем $x \in P$ и множеством $(\tilde{I}_m^n) = \{x | (\tilde{I}_m^n)(x) = 1\}$.

$$d(x, (\tilde{I}_m^n)) = \inf_{y \in (\tilde{I}_m^n)} (\sqrt{(x_0 - y_0)^2 + (x_1 - y_1)^2 + (x_2 - y_2)^2}) \quad (13)$$

Операясь на формулу 13 можно определить расстояние Хаусдорфа:

$$HD((\tilde{I}_m^n)_{seg}, (\tilde{I}_m^n)_{gt}) = \max(\sup_{x \in (\tilde{I}_m^n)_{seg}} d(x, (\tilde{I}_m^n)_{gt})) \quad (14)$$

Соответственно значением расстояния Хаусдорфа является максимум из минимумов расстояний от пикселя до множества объектов.

Определив все необходимые формулы можно определить метрику Normalized Hausdorff Distance:

$$NHD((\tilde{I}_m^n)_{seg}, (\tilde{I}_m^n)_{gt}) = 1 - \frac{\max(HD((\tilde{I}_m^n)_{seg}, (\tilde{I}_m^n)_{gt}), HD((\tilde{I}_m^n)_{gt}, (\tilde{I}_m^n)_{seg}))}{\sqrt{n_0^2 + n_1^2 + n_2^2}} \quad (15)$$

Максимальное значение расстояния Хаусдорфа в пределах изображения не превышает его диагонали, поэтому длина диагонали является разумным нормализующим коэффициентом.

Чем меньше значение расстояния Хаусдорфа, тем ближе бинарные маски друг к другу.

Для удобства интерпретации нормализованное расстояние вычитается из единицы, поэтому метрика Normalized Hausdorff Distance принимает значения от 0 (полное несовпадение) до 1 (полное совпадение масок).

Последней метрикой в начальном наборе является MSE - средне-квадратичное расстояние между пикселями сравниваемых масок.

$$MSE((\tilde{I}_m^n)_{seg}, (\tilde{I}_m^n)_{gt}) = \frac{1}{n_0 \cdot n_1 \cdot n_2} \sum_{x \in P} ((\tilde{I}_m^n)_{seg}(x) - (\tilde{I}_m^n)_{gt}(x))^2 \quad (16)$$

Таким образом, начальный набор метрик составлен из метрик, различных по принципу их оценки.

Следующим шагом в формировании итогового набора метрик является составление набора эталонных изображений для оценки чувствительности начального набора к различным видам ошибок.

Будем рассматривать в качестве типовых ошибок "Рябь", соответствие границ объекта, "Выброс" и ошибки при преобладании объекта над фоном.

"Рябь" будем определять как наличие фоновых пикселей, ошибочно попавших в область объекта.

Ошибка соответствия границ возникает в тех случаях, когда сегментированная область в целом совпадает с объектом, но границы объекта смещены относительно эталона.

Следующим типом ошибок является "Выброс" - некоторое небольшое количество пикселей ошибочно классифицированные как объект, при этом находящихся на расстоянии от области объекта.

Последней ситуацией, на которой будем проводить опыты над метриками, является преобладание объекта над фоном.

В такой ситуации некоторые метрики могут быть менее чувствительными к ошибкам так как в абсолютных цифрах большинство изображений является объектам. На пример это может привести к хорошей оценке нечувствительной метрикой если просто классифицировать все изображения как объект.

Модельные изображения, на которых будут проводиться эксперименты с метриками, приведены в рисунке 11.

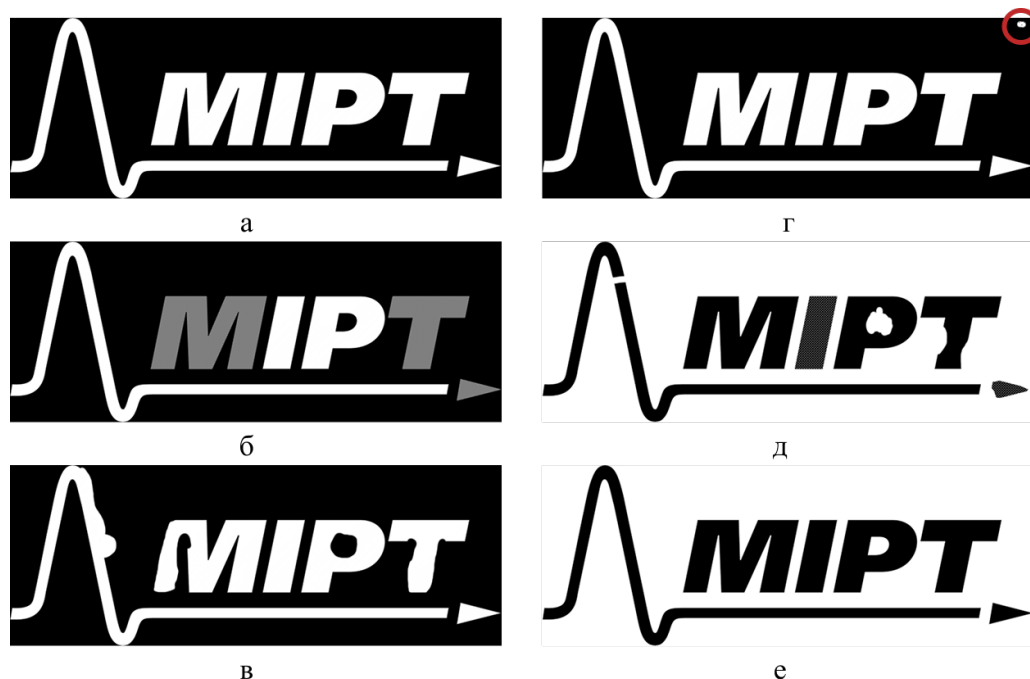


Рисунок 11 Тестовое изображение для экспериментов по оценке чувствительности метрик; а — эталонная маска, б — тест "Рябь", в — тест "Соответствие границ" г — тест "Выброс" д — тест "Преобладание объекта над фоном", е — эталонная маска для теста "Преобладание объекта над фоном"

Собрав начальный набор метрик, определив типы ошибок и модельные изображения, можно провести эксперименты по оценке чувствительности метрик.

Эксперименты реализованы с помощью языка программирования Python с использованием библиотек numpy для работы с изображениями как с многомерным массивом, PIL для операций чтения и записи изображений в файловой системе, а также matplotlib для визуализации результатов.

Настройки экспериментов читаются с файла формата JSON, который представляет собой массив из объектов, описывающих эксперимент.

Каждый объект описания эксперимента содержит его название и пути к модельному и эталонному изображениям.

Программа сравнивает по каждой метрике модельные изображения с эталонными и записывает результат в таблицу, столбцы которой являются значениями метрик, а строки соответствуют экспериментам.

Заполнив таблицу с результатами программа выводит ее визуализацию в виде табличной тепловой карты.

Получившаяся в результате экспериментов тепловая карта представлена на рисунке 12.

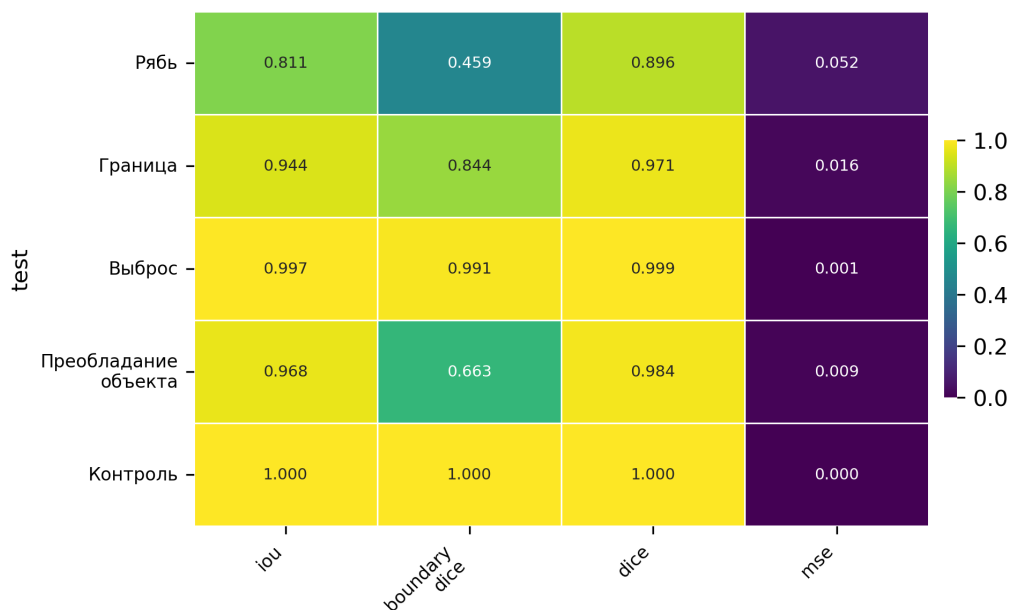


Рисунок 12 Тепловая карта значений метрик для модельных изображений с различными типами ошибок. Цвет и числа отражают чувствительность метрик к отклонениям от эталона.

Сравнение значений метрик показывает, что IOU и DICE в целом демонстрируют схожее поведение на различных типах ошибок, однако IOU в большинстве случаев оказывается более чувствительной. Соответственно в итоговый набор метрик имеет смысл включить IOU и исключить DICE.

Symmetric Boundary DICE показывает высокую чувствительность к ошибкам соответствия границ, "Ряби" и ошибкам при преобладании объекта над фоном.

Метрика MSE реагирует на все виды ошибок, однако её показания в совокупности дублируются метриками IOU и Symmetric Boundary DICE, что делает её избыточной в составе итогового набора.

Для ошибки типа "Выброс" все метрики показывают сравнительно низкую чувствительность, однако наилучшие результаты наблюдаются у метрик IOU и Symmetric Boundary DICE.

В итоговый набор метрик включены IOU и Symmetric Boundary DICE. Первая представляет собой компактную и широко используемую метрику перекрытия, служащую ориентиром для оценки общей точности

сегментации, а вторая — наиболее чувствительна к локальным погрешностям и искажениям границ.

Собрав набор данных, алгоритмы и метрики можно приступить к проектированию и реализации конвейера основных экспериментов.

2.4 Проектирование и реализация программного комплекса конвейера экспериментов

Проектирование и реализация эффективного программного обеспечения начинается с чёткой постановки требований, поэтому первым шагом в разработке конвейера экспериментов является их формулировка.

Первым требованием к конвейеру является устойчивость к прерыванию.

Система должна обеспечивать возможность продолжения экспериментов с места остановки, поэтому результаты уже выполненных вычислений необходимо сохранять, чтобы при последующем запуске можно было продолжить выполнение без необходимости пересчёта уже обработанных данных.

Следующим требованием к конвейеру экспериментов является модульность — система должна позволять легко добавлять новые данные, алгоритмы и метрики без необходимости переработки остального кода.

Такая гибкость особенно важна при проведении серии экспериментов, где может потребоваться быстро изменять состав компонентов для оценки их влияния на итоговый результат.

Последним требованием к конвейеру является поддержка вычислений на графическом процессоре.

Многие используемые алгоритмы и метрики допускают параллельную обработку, поэтому реализация с возможностью использования дискретной видеокарты позволяет существенно ускорить вычисления и сократить общее время экспериментов.

Конвейер экспериментов был спроектирован с учетом вышесформулированных требований.

Система конвейера экспериментов будет состоять из пяти модулей:

1. Модуль реконструкции

2. Модуль бинаризации
3. Модуль оценки результатов
4. Модуль визуализации результатов

Рассмотрим подробнее каждый из этих модулей и сформулируем их входные данные, выход и особенности реализации.

Модуль реконструкции отвечает за симуляцию томографии под контролем реконструкции.

На вход этому модулю требуется трёхмерное изображение в виде набора двумерных срезов, а также файл с настройками, содержащий название изображения, путь к эталонной маске и параметры для выполнения реконструкции.

Кроме того, модулю передаются общие параметры эксперимента: стратегия выбора углов проекций и их максимальное количество.

К числу параметров реконструкции относятся, в частности, размеры проекционного детектора, задаваемые числом строк и столбцов, а также шаг детектора по осям, определяющий физический размер пикселя. Эти параметры влияют на масштаб и разрешение моделируемых проекционных данных. Также указывается используемый алгоритм реконструкции; если он является итеративным, дополнительно задаётся количество итераций, определяющее глубину восстановления.

На выходе модуль формирует реконструкции для каждого набора углов проекций и сохраняет их в формате TIFF, а также некоторые параметры и значение, которые понадобятся для следующего модуля.

Таковыми параметрами являются путь к эталонной маске, среднеквадратичное отклонение уровня шума в реконструкции, название исходного изображения, индекс набора углов и массив с углами в радианах.

Результаты сохраняются в директории с установленной структурой: для каждого входного изображения создаётся отдельная папка, имя которой совпадает с названием изображения. Внутри неё размещаются подкаталоги, пронумерованные от 0 до n , где n - количество наборов углов. Каждый из этих подкаталогов содержит изображение реконструкции, полученное на основе соответствующего набора углов.

С целью обеспечения устойчивости к прерыванию каждая реконструкция сохраняется сразу после завершения расчёта. При повторном запуске модуль проверяет наличие соответствующего результата в папке и, если он уже существует, пропускает данный набор углов.

Модуль считывает входные данные из заданной директории, в которой должны располагаться подкаталоги — по одному на каждое изображение. Каждый такой подкаталог содержит срезы изображения и файл с его настройками.

Такой способ организации данных позволяет легко добавлять или удалять входные изображения без изменения исходного кода, что соответствует требованию модульности.

Реализация модуля выполнена на языке Python. Для моделирования процесса томографии под контролем реконструкции используется библиотека `astra-toolbox` [37], поддерживающая, в том числе, выполнение реконструкции на графическом процессоре.

Модуль бинаризации реализует бинарную сегментацию каждого входного изображения с помощью заданного набора алгоритмов.

Входными данными для модуля являются реконструкции, полученные на предыдущем этапе, а также параметры, необходимые для настройки алгоритмов бинаризации.

Работа модуля организована следующим образом: производится обход директории с результатами модуля реконструкции. Для каждой вложенной папки — соответствующей конкретному изображению и набору углов — считываются файл реконструкции в формате TIFF и сопутствующие параметры, сохранённые на этапе реконструкции.

Затем для каждой полученной реконструкции последовательно применяются все алгоритмы из заданного набора.

Набор алгоритмов формируется из папки `algorithms`, в которой должны быть файлы с реализацией алгоритма сегментации. Реализация должна содержать метод `segment`, принимающий на вход изображение реконструкции и эталонную маску. На выходе функция `segment` должна вернуть бинаризацию и объект с значениями параметров, используемых алгоритмом бинаризации.

Таким образом, требование модульности выполняется: для добав-

ления или удаления алгоритма достаточно поместить или удалить соответствующий файл из директории `algorithms`, без необходимости внесения изменений в остальной код конвейера.

После выполнения бинаризации полученная маска сохраняется в формате TIFF в папке, путь к которой включает название изображения, индекс набора углов и имя алгоритма. В той же папке дополнительно сохраняется JSON-файл, содержащий параметры, использованные при бинаризации, путь к соответствующей эталонной маске, а также список с набором углов проекций, использованных при реконструкции входного изображения.

В целях обеспечения устойчивости алгоритма к прерыванию, при запуске производится проверка на существования пути к результатам и наличия там уже готовой бинаризации. При обнаружении таких файлов изображение пропускается.

Модуль реализован на языке Python с использованием библиотеки CuPy [38], предоставляющей реализации базовых численных операций, таких как свёртки и тензорные преобразования, на графическом процессоре с поддержкой CUDA.

Благодаря этому становится возможным эффективно использовать графический процессор для ускорения алгоритмов бинаризации, поскольку многие из них хорошо поддаются параллелизации.

Далее рассмотрим модуль оценки результатов. Этот модуль отвечает за вычисление значений метрик, характеризующих качество полученных бинаризаций.

Работа модуля организована как последовательный обход структуры папок, сформированной предыдущим этапом. В каждой вложенной директории он поочерёдно обрабатывает результаты бинаризации: загружает бинарную маску и соответствующий JSON-файл с параметрами, содержащими путь к эталонной маске, параметры алгоритма бинаризации и набор углов проекций.

После загрузки данных модуль рассчитывает значения всех метрик из заданного набора и сохраняет результат в CSV-файл. Структура путей при этом сохраняется аналогично результатам бинаризации, но с другой корневой директорией, соответствующей модулю оценки.

При этом модуль оценки совершает расчет метрик в двух режимах - по эталону и по соседней бинаризации.

В режиме эталона значение метрики бинаризации рассчитывается в сравнении с эталонной маской.

В режиме соседней бинаризации значение метрики текущей бинаризации рассчитывается по сравнению с бинаризацией предыдущей реконструкции того же изображения.

Предыдущей реконструкцией считаем реконструкцию изображения по набору углов с индексом на 1 меньше чем индекс набора углов текущей реконструкции.

Соответственно в режиме соседней бинаризации расчет метрик начинается с бинаризации реконструкции, полученной из набора угла с индексом 1 и таких значений будет на единицу меньше, чем в режиме эталона.

Обработка выполняется поэтапно: каждая бинаризация обрабатывается отдельно, результаты сохраняются сразу после расчёта.

При этом, перед расчетом метрик, программа также проверяет наличие CSV файла в директории с результатами, соответствующей текущей входной бинарной маске. Если таковой существует - то программа пропускает данное изображение.

Такой подход обеспечивает устойчивость к прерыванию и позволяет избежать повторной обработки уже обработанных данных.

Аналогично предыдущему модулю, набор метрик формируется на основе содержимого директории `metrics`, в которой располагаются файлы с их реализациями. Название каждой метрики соответствует имени соответствующего файла. Каждый такой файл должен содержать функцию `evaluate`, принимающую на вход бинарную маску и эталонную маску, и возвращающую численное значение метрики.

Стоит также отметить, что метрика `Symmetric Boundary DICE` требует параметра r - радиуса окрестности граничных пикселей (12).

В целях простоты реализации в качестве значения параметра было выбрано $r = 5$.

Модуль был написан на языке Python с использованием библиотеки CuPy, позволяющей ускорить вычисления метрик с помощью графиче-

ского процессора.

Последним модулем конвейера экспериментов является модуль визуализации. Его задача — агрегировать полученные результаты и представить их в наглядной графической форме.

Модуль считывает все CSV-файлы, сформированные на предыдущем этапе, из директории с результатами. В отличие от других модулей, он не обрабатывает файлы по одному, а загружает все данные сразу. Для удобства доступа и группировки реализован интерфейс `ResultData`, обеспечивающий получение результатов в различных представлениях — например, значения метрик по алгоритмам, по изображениям и другим признакам.

Такая конструкция позволяет программе соответствовать требованию модульности: для добавления новой визуализации достаточно разместить соответствующий файл с реализацией в директории `visualizers`, без необходимости внесения изменений в основной код.

Модуль реализован на языке `python` с применением библиотек `numpy` для работы с массивами данных и `matplotlib` для формирования графиков.

Применение графического процессора в рамках данного модуля избыточно, соответственно все операции проходят на центральном процессоре.

В реализации конвейера также присутствует контрольный модуль, который последовательно запускает другие модули. Он необходим так как все другие модули спроектированы как отдельные программы.

Таким образом, в данной главе были рассмотрены все основные компоненты, необходимые для построения эксперимента по исследованию итеративной томографии под контролем реконструкции: входной набор данных, алгоритмы сегментации, метрики оценки качества и программный конвейер.

В следующей главе будет сформулировано правило останова процесса томографии под контролем реконструкции на основе анализа бинаризаций. С помощью разработанного конвейера экспериментов будут проведены эксперименты по исследованию его эффективности.

3 Разработка и исследование эффективности правила останова в задаче томографии под контролем реконструкции

В рамках данной главы будет сформулировано правило останова и методология проведения экспериментов для исследования его эффективности.

Реализованный конвейер экспериментов предоставляет базу для проведения численных экспериментов, на основе которых будет проведен анализ правила останова и сформулированы выводы.

3.1 Разработка правила останова

Томография под контролем реконструкции является итеративным процессом, соответственно, в каждой итерации необходимо принять решение об остановке процесса на текущей реконструкции или добавления большего количества проекций.

В рамках текущей работы изучается автоматическое принятие такого решения заранее сформулированным правилом останова на основе анализа бинаризации.

Для изучения данного вопроса требуется сформулировать такое правило останова и рассмотреть его влияние на среднее качество бинаризации по изображениям на количество углов с помощью реализованного конвейера экспериментов.

В реальных условиях применения правила останова в процессе томографии под контролем реконструкции отсутствует доступ к эталонным данным, соответственно правило останова, основанное на объективной оценке качества, является неприменимым в данном контексте.

Однако, собранный для конвейера экспериментов набор метрик позволяет всесторонне оценить схожесть двух входных изображений.

В процессе томографии под контролем реконструкции, как правило, в первой же итерации с набором из четырёх углов, маловероятно получить достаточно качественное изображение для остановки процесса.

Соответственно, для формулировки осмысленного правила остано-

ва требуется сравнение двух реконструкций — полученной на текущем наборе углов и реконструкции, соответствующей предыдущему набору.

Предполагается, что на каждой итерации доступны изображения $(I_m^n)_{i-1}$ и $(I_m^n)_i$, где $i = 1, \dots, n$ это индекс набора углов, а реконструкция, полученная на первом наборе, имеет индекс 0.

Правилом останова назовем отображение $S : R \rightarrow \{0, 1\}$, где значение 0 означает продолжение процесса томографии под контролем реконструкции, а 1 - завершение процесса на текущей реконструкции.

Сравнивая бинаризации двух соседних реконструкций, можно получить информацию о степени изменения бинаризации по мере добавления большего количества проекций.

Аргументом для правила останова будет являться значение метрики M на бинаризациях $(\hat{I}_m^n)_{i-1}$ и $(\hat{I}_m^n)_i$.

Так как объективная оценка качества бинаризации не доступно при применении правила останова, его формулировка требует некоторых предположений и эвристики.

Первым предположением, лежащим в основе формулировки правила останова, является допущение о том, что при достаточно высокой степени схожести бинаризаций двух соседних реконструкций, добавление новых проекций не приводит к значимым изменениям результата. Формально это выражается условием:

$$S(\hat{I}_{mi-1}^n, \hat{I}_{mi}^n) = \begin{cases} 1, & \text{если } M((\hat{I}_m^n)_{i-1}, (\hat{I}_m^n)_i) \geq c \\ 0, & \text{иначе} \end{cases} \quad (17)$$

где M — выбранная метрика сравнения бинарных масок, а $c = \text{const} \in [0, 1]$ — фиксированное константное пороговое значение.

Описанное правило уже позволяет получить хороший результат в случае, когда качество реконструкции с увеличением количества углов плавно улучшается.

Примерами таких изображений в наборе данных конвейера экспериментов являются изображения "Кролик" и "Статуэтка".

В случаях, когда наблюдается резкий скачок качества при включении проекций под углами, совпадающими с основными структурными

элементами исследуемого объекта, правило останова, описанное в формуле (17), может привести к слишком раннему завершению процесса. Это связано с тем, что до момента скачка несколько соседних реконструкций могут оказаться достаточно схожими между собой, несмотря на то, что качество ещё не достигло оптимального уровня.

Таковыми свойствами в наборе данных обладают изображения "Решётка" и "Наклонная решётка". В первом случае, благодаря ортогональной структуре плоскостей, приемлемое качество достигается уже при первом наборе углов. Во втором случае качество улучшается скачкообразно — при добавлении углов, совпадающих с наклонами отдельных семейств плоскостей.

Такие ситуации можно учесть в правиле останова добавив дополнительное условие на минимальное количество углов в наборе.

Хотя в текущем виде правило останова не может сработать на самом первом наборе углов, для расчёта метрики требуется как минимум две реконструкции — этого недостаточно для устранения проблемы преждевременной остановки. Чтобы её избежать, необходимо дополнительно ограничить минимальный индекс, с которого правило начинает применяться.

Модифицируем формулу (17) - добавим в нее условие останова с определенного индекса:

$$S(\hat{I}_{mi-1}^n, \hat{I}_{mi}^n) = \begin{cases} 1, & \text{если } M((\hat{I}_m^n)_{i-1}, (\hat{I}_m^n)_i) \geq c \text{ и } i \geq \alpha \\ 0, & \text{иначе} \end{cases} \quad (18)$$

где $\alpha = \text{const}$ — минимальный индекс, начиная с которого применяется правило останова.

Такая модификация позволяет снизить вероятность останова на локальном максимуме.

Сформулированное в формуле (18) правило останова носит предварительный характер, однако его достаточно для изучения вопроса возможности применения правила останова в томографии под контролем реконструкции.

Далее спроектируем эксперименты для исследования эффективности правила останова.

3.2 Методология проведения численных экспериментов по исследованию эффективности правила останова

Модуль оценки, в режиме соседних бинаризаций, реализованного конвейера экспериментов уже содержит значения метрик по различию бинаризаций реконструкций с соседними наборами углов. Соответственно, все необходимые данные для изучения эффекта правила останова уже включены в результаты конвейера экспериментов.

Эксперименты с правилом останова заключаются в переборе его параметров: минимального индекса угла α и порога схожести соседних бинаризаций c .

Так как данные уже доступны из результатов предыдущих модулей конвейера экспериментов, для исследования правила останова не требуется изменения текущего исходного кода - достаточно в модуле визуализации добавить новую компоненту в директорию `visualizators`.

Модуль визуализации обеспечивает доступ ко всем экспериментальным данным через единый интерфейс `ResultData` и передает его в качестве аргумента всем компонентам.

Реализация экспериментов с правилом останова предполагает симуляцию его применения в процессе томографии под контролем реконструкции.

Процедура проведения экспериментов с правилом останова описана в рисунке 13.

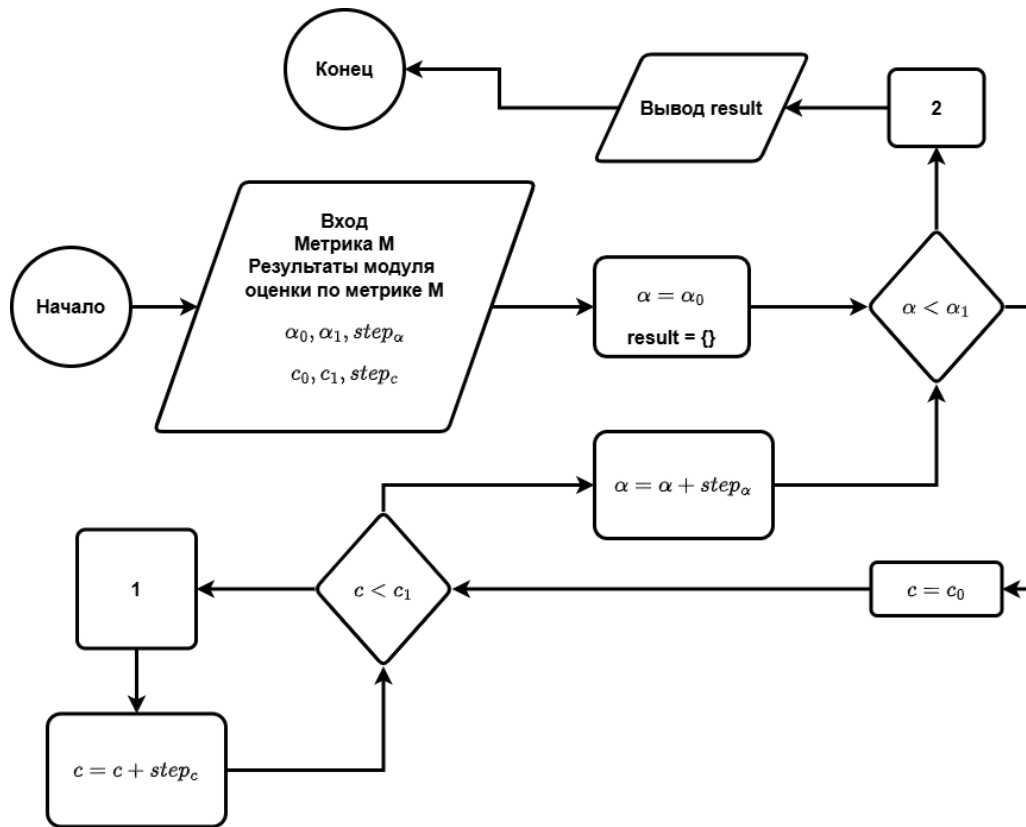


Рисунок 13 Процедура проведения экспериментов по исследованию эффективности правила останова. 1 - Блок расчета среднего по набору данных значение угла и метрики при применении правила останова; 2 - Блок добавления контроля - значений углов и метрик без применения правила останова

Процедура принимает восемь аргументов: объект ResultData, имя метрики сравнения бинарных масок (обозначена переменной M), а также шесть параметров, задающих диапазоны значений для перебора.

Для параметра c , используемого в качестве порога схожести между бинаризациями, задаются начальное значение c_0 , конечное значение c_1 и шаг $step_c$. Аналогично, для параметра α , определяющего минимальный индекс набора углов, с которого разрешается применение правила останова, задаются значения α_0 , α_1 и $step_\alpha$.

На основе этих значений формируются соответствующие дискретные множества параметров с равномерным шагом, которые затем используются для перебора всех возможных комбинаций в рамках эксперимента.

Следующий блок инициализирует начальное значение *alpha* и переменную result, в которую будет записываться результат.

На каждый индекс α в переменную `result` будет записано $\frac{(c_1-c_0)}{step_c}$ значений. После итерации по индексам *alpha* в `results` также будут все $\alpha_1 + 1$ значений метрик без применения правила останова в качестве контроля.

После записи данных планируется их изобразить на графике, соответственно необходимо выбрать структуру данных для переменной `result`, которая позволяет наиболее эффективно выполнять ее функцию в процедуре.

В качестве структуры данных была выбрана последовательность пар массивов библиотеки NumPy, организованная в виде списка. Такая форма хранения позволяет удобно добавлять новые массивы с результатами на каждой итерации, а также напрямую передавать их в функции визуализации библиотеки Matplotlib без дополнительной обработки, поскольку формат данных уже соответствует ожидаемому.

В качестве альтернатив рассматривались двумерная матрица и словарь. Однако, использование матрицы затруднено из-за различного количества углов и их значений в каждой итерации, а словарь уступает списку по скорости обработки в типичных сценариях, что делает его менее предпочтительным.

После инициализации переменных в процедуре выполняется цикл по значениям $\alpha \in [\alpha_0, \alpha_1]$.

На каждой его итерации инициализируется начальное значение порога $c = c_0$ и запускается вложенный цикл по значениям порога $c \in [c_0, c_1]$.

В вложенном цикле по порогам c запускается блок расчета среднего по набору данных значение угла и метрики при применении правила останова, обозначенного на рисунке 13 цифрой 1.

В данном блоке для фиксированной пары параметров α, c запускается симуляция процесса томографии под контролем реконструкции с применением правила останова на основе схожести бинаризаций.

В частности, для каждого изображения в наборе данных, список который доступен в `ResultData`, извлекаются значения метрики M как для соседних бинаризаций, так и для эталонной маски.

В вложенном цикле по порогам c инициализируются numpy массивы для записи значений углов и метрик, после чего запускается блок

расчёта среднего по набору данных значения угла и метрики при применении правила останова, обозначенного на рисунке 13 цифрой 1.

В данном блоке для фиксированной пары параметров α, c осуществляется симуляция процесса томографии под контролем реконструкции с применением правила останова, основанного на схожести бинаризаций.

Для каждого изображения в наборе данных, список которых доступен через интерфейс ResultData, извлекаются значения выбранной метрики M , измеряющей степень различия между бинаризациями соседних реконструкций, а также значения метрики, измеряющей отличие от эталонной маски.

Далее запускается цикл по каждому значению индекса набора углов i , начиная с $i = \alpha$. На каждой итерации производится проверка, превышает ли значение метрики между бинаризациями $M((\hat{I}_m^n)_{i-1}, (\hat{I}_m^n)_i)$ заданный порог c . Если условие выполнено, текущий индекс считается моментом останова, и дальнейший перебор для данного изображения прекращается.

После фиксации точки останова записывается соответствующий угол, на котором был выполнен останов, и значение метрики M оценивающее бинаризацию реконструкции под индексом i по сравнению с эталонной маской.

Эти значения добавляются в массивы результатов, которые впоследствии усредняются по всем изображениям.

Полученное среднее значение угла и метрики для порога c добавляются в соответствующие массивы numру, инициализированными перед началом цикла по c .

После завершения цикла по значениям c , заполненные массивы numру с значениями углов и метрик добавляются как пара tuple в исходный список result.

Данные шаги повторяются на каждой итерации внешнего цикла по α и к моменту завершения цикла мы получаем заполненный список results.

После завершения цикла по значениям α , в блоке добавления контрольных значений (обозначен на рисунке 13 цифрой 2), к переменной

result добавляется список средних значений количества углов и соответствующей метрики по эталонам без применения правила останова. Соответственно эти данные являются контролем, с которым можно сравнивать результаты, полученные при различных значениях параметров правила останова.

Далее в процедуре переменная result передается в блок вывода, который формирует графики и визуализацию результата, после чего процесс завершается.

Описанная выше процедура позволяет симулировать применение правила останова с различными параметрами и сравнить их с контрольными значениями без применения правила останова.

Далее будут рассмотрены результаты запуска данной процедуры и проведен их анализ.

3.3 Анализ эффективности правила останова

Реализованного конвейера экспериментов и всех его модулей достаточно чтобы провести главный эксперимент от реконструкции всех изображений по указанной в настройках стратегии, до вывода результата симуляции работы правила останова в томографии под контролем реконструкции.

Перед запуском основного эксперимента необходимо определить все нужные входные параметры.

Запуск модуля реконструкции требует определения стратегии набора углов. Такая настройка принимается модулем в виде JSON файла. Файл с настройками стратегий представлен в листинге 1

Листинг 1: Файл конфигурации стратегий набора углов для конвейера экспериментов

```
[
  {
    "strategy": "binary",
    "max_angles": 2048
  }
]
```

Объект настройки стратегии находится в списке так как модуль

способен выполнять реконструкцию по нескольким стратегиям набора углов.

Список необходимых ключей в самом объекте зависит от стратегии, однако для всех обязательным является ключ `name`. На данный момент реализована стратегия `binary` - удвоение углов, описанная в работе [39].

Стратегия `binary` настраивается максимальным количеством углов. В рамках текущего запуска конвейера экспериментов в качестве такого значения указано 1024, соответственно $2^{10} = 1024$, а начинаем набор углов с $2^2 = 4$, откуда получается 9 наборов углов на каждое изображение.

Запуск реализованной в модуле визуализации процедуры экспериментов по исследованию эффективности правила останова требует определения шести аргументов, задающих границы параметров правила останова. Этими аргументами являются значения начала и конца отрезка соответствующего параметра, а также шаг в каждой итерации.

Для параметра α выбран отрезок $\alpha \in [3, 8], \alpha \in \mathbb{Z}$. Так как индексами являются целые числа, то в качестве шага $step_\alpha$ выбрана единица: $step_\alpha = 1$. В экспериментах из предыдущих модулей 9 наборов углов с индексами от 0 до 8.

Для параметра c , определяющего пороговое значение метрики схожести бинаризаций, выбран отрезок $c \in [0.4, 1.0]$ с шагом $step_c = 0.005$. Таким образом, в цикле перебираются 120 различных значений параметра c .

Конвейер экспериментов был запущен с описанными выше параметрами и в результате его работы были получены результаты в виде графиков и таблиц, описывающих зависимость среднего угла от среднего значения метрик.

Таких результатов девять - по два графика и одной таблице на каждый алгоритм сегментации.

Рассмотрим результаты экспериментов по классическому пороговому алгоритму, определенному в формуле 1.

График зависимости среднего угла от среднего значения метрики IOU представлен на рисунке 14.

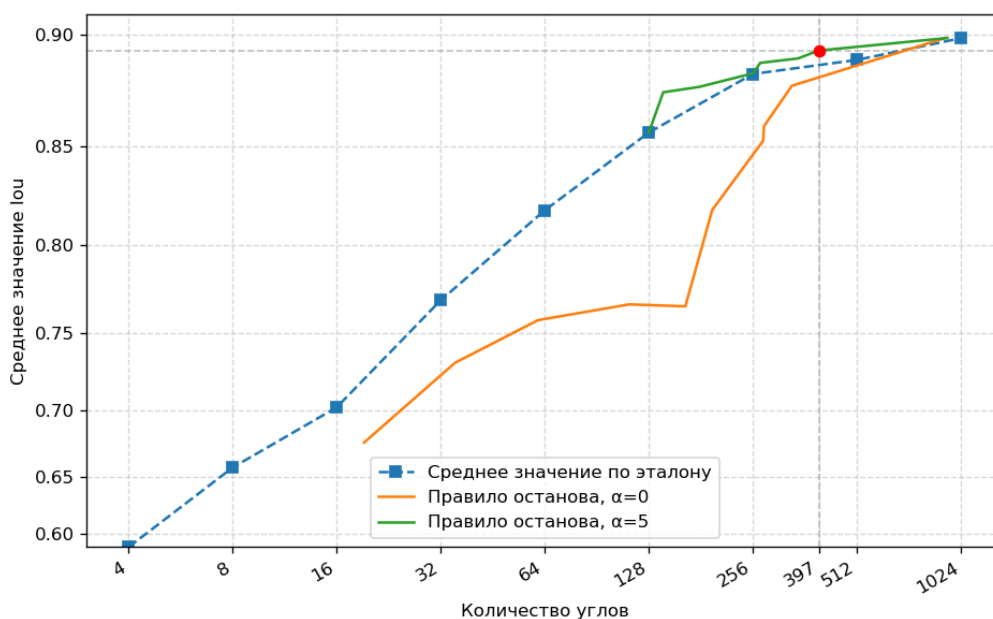


Рисунок 14 График зависимости среднего значения угла от среднего значение метрики IOU при различных параметрах правила останова для классического порогового алгоритма

Синия линия с квадратными точками на графике является контролем - значение угла и метрики без применения правила останова, вместо этого происходит остановка на фиксированных наборах углов.

На графике кривые соответствующие различным значениям $\alpha = 0$ и $\alpha = 5$.

Красным отмечена точка с наибольшим выигрышем, соответствующая порогу $c = 0.95$. В этой точке наблюдается как улучшение по качеству, так и улучшение по количеству углов, по сравнению с остановкой на фиксированном количестве углов 512. В частности, значение метрики 0.8931 достигается за 397 при применении правила останова.

При этом, в сравнении с полным набором углов в 1024, значения метрик при применении правила останова и без него различаются лишь на 0.01, а количество углов, которое необходимо для достижения такого результата сократилось более чем на 2 раза.

В таблице 1 приведены результаты исследования на долю точек, находящихся над кривой контроля для каждой кривой правила останова.

Доля точек над контролем у кривых правила останова достаточно высока. Это означает, что существует большое количество комбинаций пар параметров (α, c) позволяющих получить незначительный, но выиг-

рыш в качестве по метрике IOU при применении правила останова.

Таблица 1: Доля точек кривых, на которых значение метрики IOU превышает значение метрики в контроле

α	Доля точек (%)
3	69.231
4	76.923
5	84.615
6	53.846
7	46.154

График для этой метрики Symmetric Boundary DICE приведен в рисунке 15, а таблица с исследованием доли точек превышающих контроль в таблице.

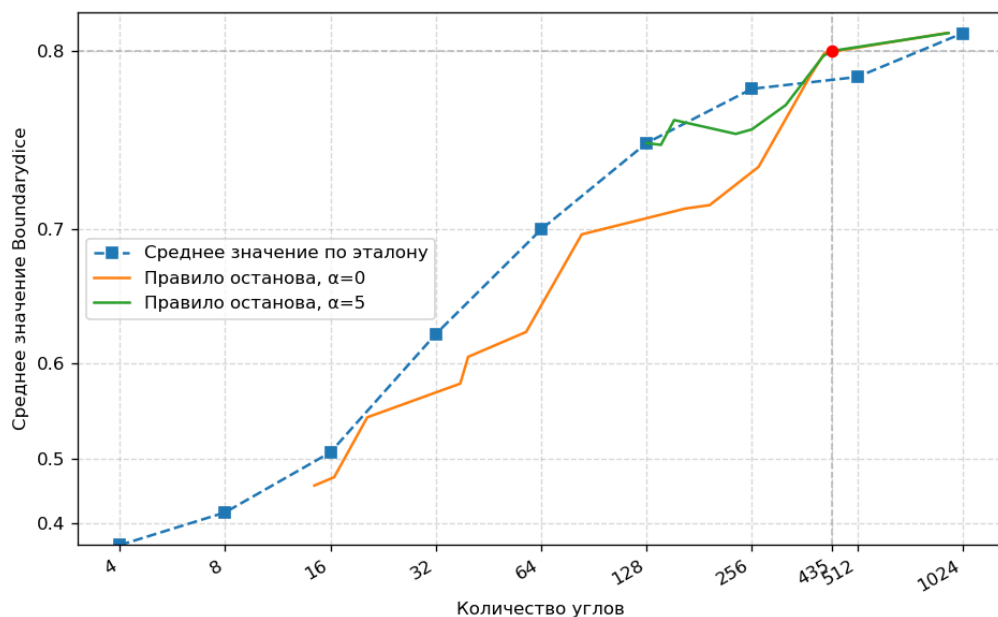


Рисунок 15 График зависимости среднего значения угла от среднего значение метрики Symmetric boundary DICE при различных параметрах правила останова для классического порогового алгоритма

В случае с Symmetric Boundary DICE наблюдается аналогичная картина и наилучший результат постигается при тех же значениях параметров правила останова. Значение метрики в 0.8 удалось достичь за 435 углов, по сравнению с значением в 0.78 при фиксированной остановке на 512 углах. При этом, разница в качестве между применением правила останова и полным набором углов составляет лишь 0.02

Однако, доля точек превышающих контроль, значительно ниже.

Такие результаты означают что правило останова дает выигрыш в качестве, однако, необходимо подбирать его параметры для получения такого эффекта.

Таблица 2: Доля точек кривых, на которых значение метрики SBD превышает значение метрики в контроле

α	Доля точек (%)
3	23.076
4	46.153
5	30.769
6	15.384
7	53.846

Численные значения экспериментов, на базе которых были построены графики, приведены в приложении "А".

Рассмотрим результаты экспериментов по применению правила останова для алгоритма Отсу.

На рисунках 16 и 17 изображены графики результатов применения правила останова.

Из графиков можно сделать вывод, что в отличие от случая с классическим пороговым алгоритмом, выигрыш в качестве от применения правила останова значителен для обеих метрик.

При этом, как показывают таблицы 3 и 4, значения доли точек, превышающих контроль для обеих метрик также достаточно большой.

В случае метрики IOU при значениях параметров правила останова $\alpha = 4$ и $c = 0.95$ достигается качество в 0.9391 при количестве углов в 301. Разница с полным набором углов всего в 0.02, однако количество углов потребовалось также в более чем в два раза меньше.

Аналогичная ситуация с метрикой Symmetric Boundary DICE.

Такие результаты позволяют сделать вывод о высокой эффективности правила останова для алгоритма Отсу.

Выигрыш в применении правила останова в случае алгоритма Отсу наиболее позволяет исключить влияния параметров самого алгоритма, так как у метода Отсу вручную подбираемых параметров нет.

В приложении "Б" представлены численные результаты экспери-

ментов, на основе которых строились графики.

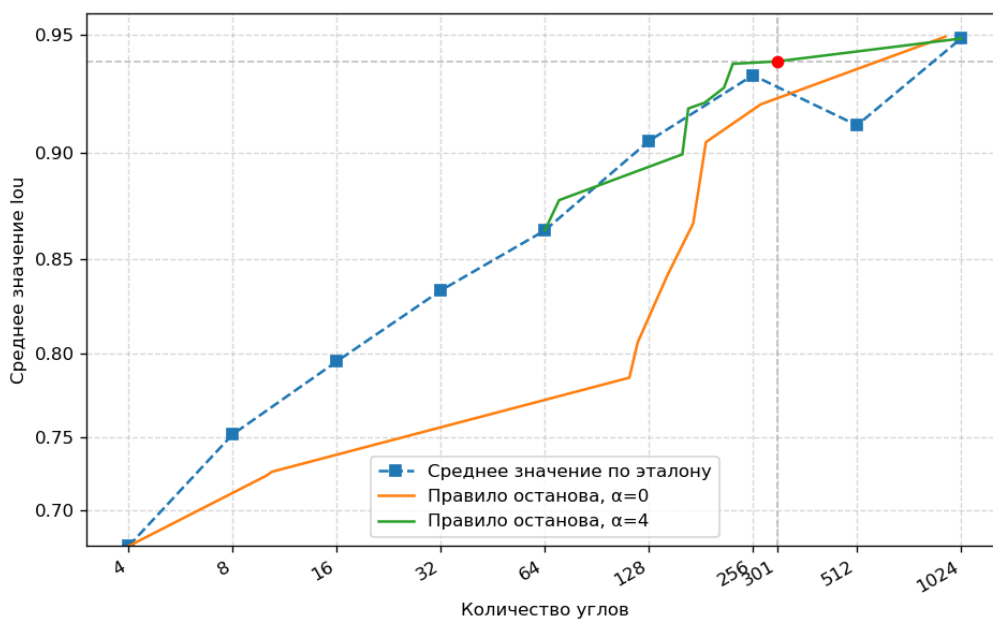


Рисунок 16 График зависимости среднего значения угла от среднего значение метрики IOU при различных параметрах правила останова для алгоритма Отсу

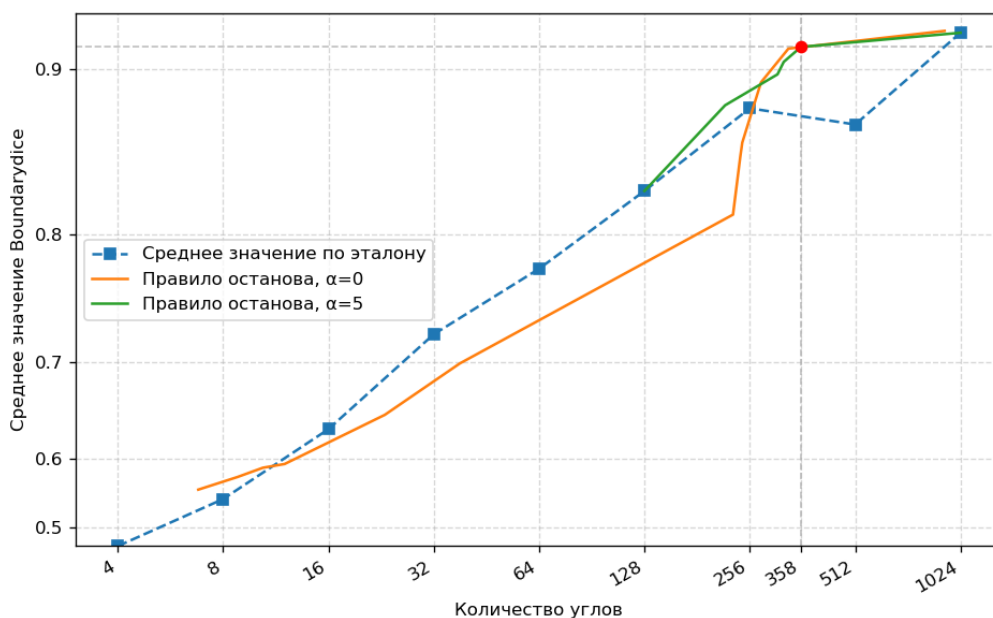


Рисунок 17 График зависимости среднего значения угла от среднего значение метрики Symmetric boundary DICE при различных параметрах правила останова для алгоритма Отсу

Таблица 3: Доля точек кривых, на которых значение метрики IOU превышает значение метрики в контроле для алгоритма Отсу

α	Доля точек (%)
3	46.153
4	46.153
5	53.845
6	76.923
7	76.923

Таблица 4: Доля точек кривых, на которых значение метрики SBD превышает значение метрики в контроле для алгоритма Отсу

α	Доля точек (%)
3	15.384
4	30.769
5	69.230
6	76.923
7	69.230

Последним исследуемым алгоритмом является метод аффинного Ниблэка.

На рисунках 18 и 19 представлены результаты экспериментов по применению правила останова с алгоритмом аффинного Ниблэка.

Из рисунка 18 видно, что хотя правило останова и не дает в случае метрики IOU выигрыш в качестве, однако при параметрах правила останова $\alpha = 3$ и $c = 0.95$ получаем значение метрики в 0.7158, что отличается от полного набора углов на 0.02, но при этом достигается за 630 углов.

В графике метрики Symmetric Boundary DICE, на рисунке 19, аналогичная ситуация - качество практически равное полному набору углов достигается за 819 углов.

При этом доля точек выше контроля у метрики IOU (таблица 5) достаточно низкая, соответственно для этой метрики правило останова дает выигрыш на узком спектре параметров.

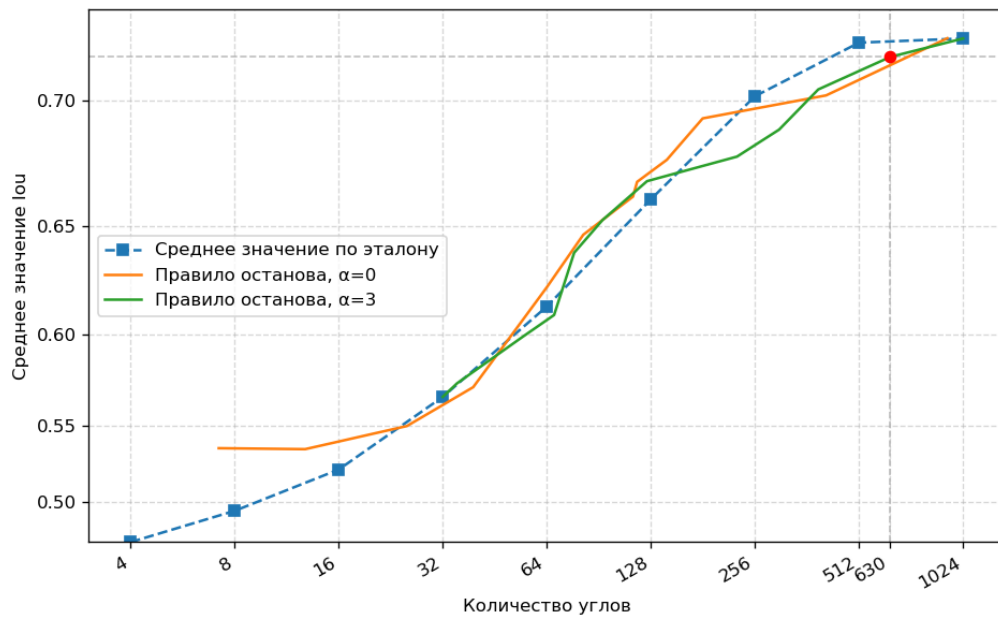


Рисунок 18 График зависимости среднего значения угла от среднего значение метрики IOU при различных параметрах правила останова для алгоритма аффинного Ниблэка

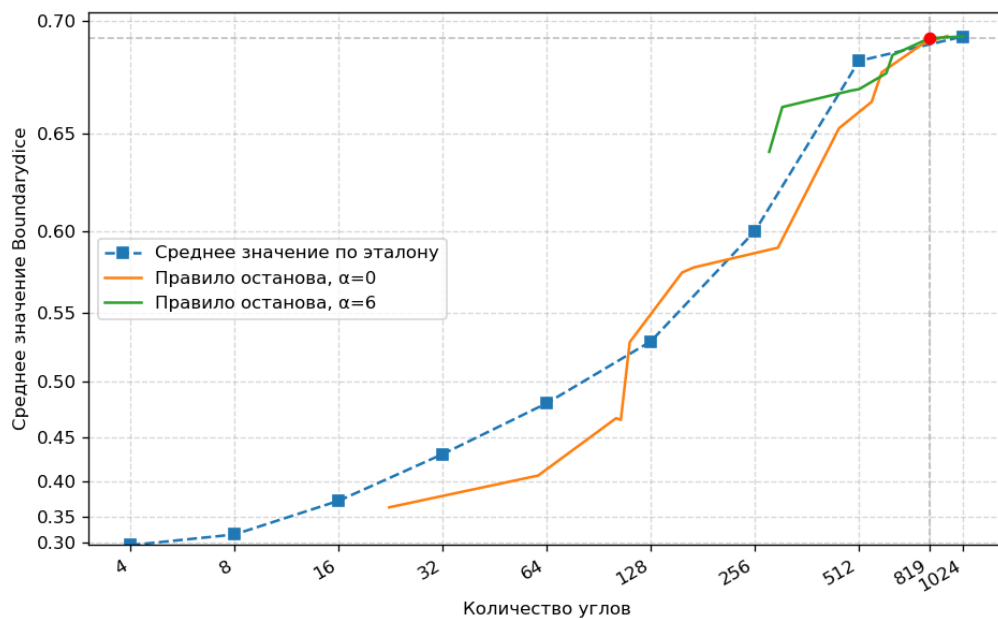


Рисунок 19 График зависимости среднего значения угла от среднего значение метрики Symmetric boundary DICE при различных параметрах правила останова для алгоритма аффинного Ниблэка

Таблица 5: Доля точек кривых, на которых значение метрики IOU превышает значение метрики в контроле для алгоритма аффинного Ниблэка

α	Доля точек (%)
3	30.769
4	23.076
5	15.384
6	15.384
7	0

Таблица 6: Доля точек кривых, на которых значение метрики SBD превышает значение метрики в контроле для алгоритма аффинного Ниблэка

α	Доля точек (%)
3	61.538
4	61.538
5	61.538
6	61.538
7	15.384

В случае метрики Symmetric Boundary DICE таблица 6 напротив показывает выигрыш в качестве для достаточно большого количества параметров правила останова.

В приложении "В" представлены численные результаты экспериментов, на основе которых строились графики.

Сформулируем ряд положений, основанных на результатах экспериментов.

Правило останова 18 позволяет в алгоритмах бинаризации получить в среднем значительный выигрыш в качестве бинаризаций при меньшем количестве необходимых углов проекций. Такой выигрыш наблюдается как по соответствию тонких деталей (согласно метрике Symmetric Boundary DICE) так и по общему количеству правильно классифицированных пикселей (измеренных метрикой IOU).

Также, в случае алгоритмов Отсу и классического порогового не требуется тонкая настройка параметров α и c правила останова.

4 Выводы

Разработка принципов управления процедурами компьютерной томографии и их научная обоснованность - одно из приоритетных направлений междисциплинарных исследований, находящихся на стыке современной прикладной математики, информатики, физики.

Проведённое исследование протоколов мониторинговой реконструкции в рамках задачи бинаризации показывает, что применение правила останова позволяет повысить среднее качество бинаризации по исследуемым метрикам при меньшем числе углов проекций по сравнению с контролем, под которым в данном случае понимается остановка процесса на фиксированном заранее заданном наборе углов.

В исследовании было сформулировано правило останова процедуры компьютерной томографии под контролем реконструкции опираясь на анализ бинаризаций.

В работы был собран набор данных, состоящий из десяти изображений, четыре из которых являются сканированными изображениями реальных объектов, а остальные шесть синтетические.

В исследовании использовались метрики IOU и Symmetric Boundary DICE, а также алгоритмы бинаризации: Отсу, классический пороговый и аффинный Ниблэк.

В работе был спроектирован и разработан программный комплекс для конвейера экспериментов, включающий в себя модули реконструкции, бинаризации, оценки бинаризаций и модуль визуализации результатов. Реализация программного комплекса выполнена на языке Python с использованием библиотек `numpy`, `cupy`, `matplotlib` и `astra toolbox`.

Было сформировано правило останова, настраиваемое двумя параметрами - *alpha* и *c*, первый из которых регулирует минимальный индекс набора углов, для начала действия правила останова, а второй - порог схожести соседних бинаризаций.

Осуществлен комплекс численных экспериментов, позволяющий установить эффективность разработанного правила.

В результате экспериментов было установлено, что применение правила останова позволяет достичь качества, сравнимого с использованием полного набора углов, в среднем за вдвое меньшее количество

проекций.

Разница в качестве при этом, в среднем по всем алгоритмам, не превышает 0.02 ед. по всем метрикам.

Результаты работы показывают, что применение правила останова в томографии под контролем реконструкции позволяет завершить процесс при меньшем числе углов проекций. Это, в свою очередь, способствует снижению дозы излучения и сокращению времени, необходимого для проведения КТ-исследования.

Полученные результаты открывают направления для дальнейших исследований, включая оптимизацию параметров правила останова, его проверку в практических условиях КТ-сканирования и адаптацию под различные типы данных. Перспективным также является изучение альтернативных алгоритмов бинаризации и метрик для применения в условиях anytime-реконструкции.

Список литературы

1. Reducing radiation dose for NN-based COVID-19 detection in helical chest CT using real-time monitored reconstruction / К. В. Bulatov [и др.] // Expert Systems with Applications. — 2023. — Т. 229. — С. 120425.
2. Slovis T. L. The ALARA concept in pediatric CT: myth or reality? // Radiology. — 2002. — Т. 223, № 1. — С. 5—6.
3. Monitored reconstruction: Computed tomography as an anytime algorithm / К. Bulatov [и др.] // IEEE Access. — 2020. — Т. 8. — С. 110759—110774.
4. Николаев Д. П. Методы и алгоритмы Хаф-анализа изображений в системах технического зрения : дис. ... канд. / Николаев Д. П. — 2023.
5. Fu K.-S., Mui J. A survey on image segmentation // Pattern recognition. — 1981. — Т. 13, № 1. — С. 3—16.
6. Zhang Y.-J. Advances in image and video segmentation. — IGI Global, 2006.
7. Wirjadi O. Survey of 3D image segmentation methods. — 2007.
8. Ханыков И. Классификация алгоритмов сегментации изображений // Известия высших учебных заведений. Приборостроение. — 2018. — Т. 61, № 11. — С. 978—987.
9. Adams R., Bischof L. Seeded region growing // IEEE Transactions on pattern analysis and machine intelligence. — 1994. — Т. 16, № 6. — С. 641—647.
10. Lin Z., Jin J., Talbot H. Unseeded Region Growing for 3D Image Segmentation. // ACM International Conference Proceeding Series. Т. 9. — Citeseer. 2000. — С. 31—37.
11. Segmentation and classification of lung tumor from 3D CT image using K-means clustering algorithm / P. Sarker [и др.] // 2017 4th international conference on advances in electrical engineering (ICAEE). — IEEE. 2017. — С. 731—736.

12. Comaniciu D., Meer P. Mean shift: A robust approach toward feature space analysis // IEEE Transactions on pattern analysis and machine intelligence. — 2002. — Т. 24, № 5. — С. 603—619.
13. Camilus K. S., Govindan V. A review on graph based segmentation // International Journal of Image, Graphics and Signal Processing. — 2012. — Т. 4, № 5. — С. 1.
14. Boykov, Kolmogorov. Computing geodesics and minimal surfaces via graph cuts // Proceedings Ninth IEEE international conference on computer vision. — IEEE. 2003. — С. 26—33.
15. Peng Z., Qu S., Li Q. Interactive image segmentation using geodesic appearance overlap graph cut // Signal Processing: Image Communication. — 2019. — Т. 78. — С. 159—170.
16. Topology cuts: A novel min-cut/max-flow algorithm for topology preserving segmentation in N-D images / Y. Zeng [и др.] // Computer vision and image understanding. — 2008. — Т. 112, № 1. — С. 81—90.
17. Felzenszwalb P. F., Huttenlocher D. P. Efficient graph-based image segmentation // International journal of computer vision. — 2004. — Т. 59. — С. 167—181.
18. Volumetric segmentation using Weibull E-SD fields / J. Hu [и др.] // IEEE Transactions on Visualization and Computer Graphics. — 2003. — Т. 9, № 3. — С. 320—328.
19. Ayed I. B., Hennane N., Mitiche A. Unsupervised variational image segmentation/classification using a Weibull observation model // IEEE transactions on Image processing. — 2006. — Т. 15, № 11. — С. 3431—3439.
20. A 3D convolutional neural network for volumetric image semantic segmentation / H. Lu [и др.] // Procedia Manufacturing. — 2019. — Т. 39. — С. 422—428.
21. Ха Л. М. Свёрточная нейронная сеть для решения задачи классификации // Труды Московского физико-технического института. — 2016. — Т. 8, 3 (31). — С. 91—97.

22. Milletari F., Navab N., Ahmadi S.-A. V-net: Fully convolutional neural networks for volumetric medical image segmentation // 2016 fourth international conference on 3D vision (3DV). — Ieee. 2016. — C. 565—571.
23. Liu C. J., Wu X. L. Industrial computerized tomography images segmentation based on cellular neural networks // Applied Mechanics and Materials. — 2011. — T. 66. — C. 2228—2235.
24. Lang T., Sauer T. AI-Supported Segmentation of Industrial CT Data // 11th Conf. Ind. Comput. Tomogr. — Wels. 2022.
25. Semi-automatic Segmentation of MRI Brain Metastases Combining Support Vector Machine and Morphological Operators. / G. Gonella [и др.] // IJCCI. — 2019. — C. 457—463.
26. Nagai Y., Ohtake Y., Suzuki H. SegMo: CT volume segmentation using a multi-level Morse complex // Computer-Aided Design. — 2019. — T. 107. — C. 23—36.
27. 3D volume segmentation of MRA data sets using level sets: image processing and display / A. A. Farag [и др.] // Academic radiology. — 2004. — T. 11, № 4. — C. 419—435.
28. Wagner F., Maas H.-G. A Comparative Study of Deep Architectures for Voxel Segmentation in Volume Images // The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. — 2023. — T. 48. — C. 1667—1676.
29. Marc Levoy. The Stanford 3D Scanning Repository. — 2023. — URL: [https : / / graphics . stanford . edu / data / 3Dscanrep/](https://graphics.stanford.edu/data/3Dscanrep/) (дата обр. 13.06.2025).
30. A threshold selection method from gray-level histograms / N. Otsu [и др.] // Automatica. — 1975. — T. 11, № 285—296. — C. 23—27.
31. Niblack W. An introduction to digital image processing. — Strandberg Publishing Company, 1985.
32. Sauvola J., Pietikäinen M. Adaptive document image binarization // Pattern recognition. — 2000. — T. 33, № 2. — C. 225—236.

33. Adaptive local thresholding for detection of nuclei in diversity stained cytology images / N. Phansalkar [и др.] // 2011 International conference on communications and signal processing. — IEEE. 2011. — С. 218—220.
34. Николаев Д. П., Сараев А. А. Критерии оценки качества в задаче автоматизированной настройки алгоритмов бинаризации // Труды Института системного анализа Российской академии наук. — 2013. — Т. 63, № 3. — С. 85—94.
35. Taha A. A., Hanbury A. Metrics for evaluating 3D medical image segmentation: analysis, selection, and tool // BMC medical imaging. — 2015. — Т. 15. — С. 1—28.
36. Yeghiazaryan V., Voiculescu I. Family of boundary overlap metrics for the evaluation of medical image segmentation // Journal of Medical Imaging. — 2018. — Т. 5, № 1. — С. 015006—015006.
37. Fast and flexible X-ray tomography using the ASTRA toolbox / W. Van Aarle [и др.] // Optics express. — 2016. — Т. 24, № 22. — С. 25129—25147.
38. CuPy Developers. CuPy: A NumPy-compatible array library accelerated by CUDA. — 2024. — URL: <https://cupy.dev/> (дата обр. 13.06.2025).
39. Applicability and potential of monitored reconstruction in computed tomography / M. Gilmanov [и др.] // PloS one. — 2024. — Т. 19, № 7. — e0307231.

ПРИЛОЖЕНИЕ А

Результаты экспериментов для классического порогового алгоритма

Таблица .1: Результаты экспериментов для классического порогового алгоритма: сравнение метрик IOU и Symmetric Boundary DICE

α	порог c	Угол (IOU)	Метрика IOU	Угол (SBD)	Метрика SBD
3	0.4	41	0.7898	35	0.6237
3	0.45	41	0.7898	35	0.6237
3	0.5	57	0.8449	35	0.6237
3	0.55	57	0.8449	51	0.6444
3	0.6	57	0.8449	54	0.6592
3	0.65	105	0.8472	67	0.6582
3	0.7	182	0.8531	89	0.7153
3	0.75	192	0.8542	169	0.7148
3	0.8	211	0.8766	198	0.717
3	0.85	288	0.8786	272	0.7398
3	0.9	288	0.8786	377	0.7871
3	0.95	342	0.8855	416	0.799
3	1.0	934	0.8984	934	0.8088
4	0.4	70	0.8341	64	0.6998
4	0.45	70	0.8341	64	0.6998
4	0.5	83	0.8525	64	0.6998
4	0.55	83	0.8525	76	0.7145
4	0.6	83	0.8525	76	0.7145
4	0.65	128	0.8545	89	0.7135
4	0.7	204	0.8605	108	0.741
4	0.75	211	0.8616	185	0.7334
4	0.8	230	0.884	211	0.7359
4	0.85	307	0.8859	281	0.75
4	0.9	307	0.8859	384	0.7878
4	0.95	358	0.8893	422	0.7997
4	1.0	934	0.8984	934	0.8088

α	порог c	Угол (IOU)	Метрика IOU	Угол (SBD)	Метрика SBD
5	0.4	128	0.8564	128	0.7516
5	0.45	128	0.8564	128	0.7516
5	0.5	140	0.8748	128	0.7516
5	0.55	140	0.8748	128	0.7516
5	0.6	140	0.8748	128	0.7516
5	0.65	179	0.8772	140	0.7507
5	0.7	256	0.8832	153	0.7643
5	0.75	256	0.8832	230	0.7567
5	0.8	268	0.8878	256	0.7592
5	0.85	345	0.8897	320	0.7723
5	0.9	345	0.8897	409	0.7975
5	0.95	396	0.8931	435	0.8001
5	1.0	934	0.8984	934	0.8088
6	0.4	256	0.8829	256	0.7808
6	0.45	256	0.8829	256	0.7808
6	0.5	256	0.8829	256	0.7808
6	0.55	256	0.8829	256	0.7808
6	0.6	256	0.8829	256	0.7808
6	0.65	281	0.8869	256	0.7808
6	0.7	358	0.8929	256	0.7808
6	0.75	358	0.8929	332	0.7733
6	0.8	358	0.8929	358	0.7758
6	0.85	435	0.8948	409	0.781
6	0.9	435	0.8948	486	0.8048
6	0.95	486	0.8982	512	0.8074
6	1.0	1024	0.8984	1024	0.8087
7	0.4	512	0.889	512	0.7869
7	0.45	512	0.889	512	0.7869
7	0.5	512	0.889	512	0.7869
7	0.55	512	0.889	512	0.7869
7	0.6	512	0.889	512	0.7869
7	0.65	512	0.889	563	0.8068

α	порог c	Угол (IOU)	Метрика IOU	Угол (SBD)	Метрика SBD
7	0.7	563	0.8983	614	0.8031
7	0.75	563	0.8983	614	0.8031
7	0.8	563	0.8983	614	0.8031
7	0.85	614	0.8949	665	0.8083
7	0.9	614	0.8949	665	0.8083
7	0.95	665	0.8983	665	0.8083
7	1.0	1024	0.8984	1024	0.8087
8	0.4	1024	0.8984	1024	0.8087
8	0.45	1024	0.8984	1024	0.8087
8	0.5	1024	0.8984	1024	0.8087
8	0.55	1024	0.8984	1024	0.8087
8	0.6	1024	0.8984	1024	0.8087
8	0.65	1024	0.8984	1024	0.8087
8	0.7	1024	0.8984	1024	0.8087
8	0.75	1024	0.8984	1024	0.8087
8	0.8	1024	0.8984	1024	0.8087
8	0.85	1024	0.8984	1024	0.8087
8	0.9	1024	0.8984	1024	0.8087
8	0.95	1024	0.8984	1024	0.8087
8	1.0	1024	0.8984	1024	0.8087

ПРИЛОЖЕНИЕ Б

Результаты экспериментов для алгоритма Отсу

Таблица .2: Результаты экспериментов для алгоритма Отсу: сравнение метрик IOU и Symmetric Boundary DICE

α	порог c	Угол (IOU)	Метрика IOU	Угол (SBD)	Метрика SBD
3	0.4	35	0.8236	32	0.7244
3	0.45	35	0.8236	32	0.7244

α	порог c	Угол (IOU)	Метрика IOU	Угол (SBD)	Метрика SBD
3	0.5	44	0.87	35	0.7118
3	0.55	134	0.8909	35	0.7118
3	0.6	134	0.8909	48	0.7401
3	0.65	134	0.8909	48	0.7401
3	0.7	134	0.8909	236	0.8335
3	0.75	140	0.9108	246	0.8535
3	0.8	163	0.9135	246	0.8535
3	0.85	188	0.9198	249	0.8621
3	0.9	201	0.9297	278	0.893
3	0.95	281	0.9342	332	0.9112
3	1.0	1024	0.9484	1024	0.9196
4	0.4	64	0.8641	64	0.7756
4	0.45	64	0.8641	64	0.7756
4	0.5	70	0.8785	64	0.7756
4	0.55	160	0.8994	64	0.7756
4	0.6	160	0.8994	70	0.7948
4	0.65	160	0.8994	70	0.7948
4	0.7	160	0.8994	256	0.8606
4	0.75	166	0.9194	262	0.8732
4	0.8	185	0.9219	262	0.8732
4	0.85	211	0.9282	262	0.8732
4	0.9	224	0.9381	288	0.8945
4	0.95	300	0.9391	339	0.9117
4	1.0	1024	0.9484	1024	0.9196
5	0.4	128	0.9053	128	0.8287
5	0.45	128	0.9053	128	0.8287
5	0.5	128	0.9053	128	0.8287
5	0.55	217	0.9262	217	0.8802
5	0.6	217	0.9262	217	0.8802
5	0.65	217	0.9262	217	0.8802
5	0.7	217	0.9262	307	0.8974
5	0.75	217	0.9262	307	0.8974

α	порог c	Угол (IOU)	Метрика IOU	Угол (SBD)	Метрика SBD
5	0.8	230	0.9263	307	0.8974
5	0.85	256	0.9326	307	0.8974
5	0.9	268	0.9425	320	0.9042
5	0.95	345	0.9435	358	0.9121
5	1.0	1024	0.9484	1024	0.9196
6	0.4	256	0.9334	256	0.8785
6	0.45	281	0.9083	256	0.8785
6	0.5	332	0.9388	281	0.8972
6	0.55	332	0.9388	409	0.9165
6	0.6	332	0.9388	409	0.9165
6	0.65	332	0.9388	409	0.9165
6	0.7	332	0.9388	409	0.9165
6	0.75	332	0.9388	409	0.9165
6	0.8	332	0.9388	409	0.9165
6	0.85	358	0.9451	409	0.9165
6	0.9	358	0.9451	409	0.9165
6	0.95	435	0.9461	435	0.919
6	1.0	1024	0.9484	1024	0.9196
7	0.4	512	0.9122	512	0.8692
7	0.45	512	0.9122	512	0.8692
7	0.5	563	0.9427	512	0.8692
7	0.55	563	0.9427	614	0.9196
7	0.6	563	0.9427	614	0.9196
7	0.65	563	0.9427	614	0.9196
7	0.7	563	0.9427	614	0.9196
7	0.75	563	0.9427	614	0.9196
7	0.8	563	0.9427	614	0.9196
7	0.85	563	0.9427	614	0.9196
7	0.9	563	0.9427	614	0.9196
7	0.95	614	0.9467	614	0.9196
7	1.0	1024	0.9484	1024	0.9196
8	0.4	1024	0.9484	1024	0.9196

α	порог c	Угол (IOU)	Метрика IOU	Угол (SBD)	Метрика SBD
8	0.45	1024	0.9484	1024	0.9196
8	0.5	1024	0.9484	1024	0.9196
8	0.55	1024	0.9484	1024	0.9196
8	0.6	1024	0.9484	1024	0.9196
8	0.65	1024	0.9484	1024	0.9196
8	0.7	1024	0.9484	1024	0.9196
8	0.75	1024	0.9484	1024	0.9196
8	0.8	1024	0.9484	1024	0.9196
8	0.85	1024	0.9484	1024	0.9196
8	0.9	1024	0.9484	1024	0.9196
8	0.95	1024	0.9484	1024	0.9196
8	1.0	1024	0.9484	1024	0.9196

ПРИЛОЖЕНИЕ В

Результаты экспериментов для алгоритма аффинного Ниблэка

Таблица .3: Результаты экспериментов для аффинного алгоритма Ниблэка: сравнение метрик IOU и Symmetric Boundary DICE

α	порог c	Угол (IOU)	Метрика IOU	Угол (SBD)	Метрика SBD
3	0.4	32	0.5666	38	0.44
3	0.45	32	0.5666	73	0.489
3	0.5	35	0.5741	112	0.5278
3	0.55	54	0.5985	115	0.5262
3	0.6	67	0.6095	121	0.543
3	0.65	76	0.6384	166	0.5872
3	0.7	92	0.6527	198	0.62
3	0.75	124	0.6688	326	0.6317
3	0.8	227	0.6786	451	0.6528
3	0.85	300	0.6891	560	0.665

α	порог c	Угол (IOU)	Метрика IOU	Угол (SBD)	Метрика SBD
3	0.9	390	0.7041	598	0.6783
3	0.95	630	0.7158	784	0.6904
3	1.0	1024	0.7222	1024	0.6937
4	0.4	64	0.6136	64	0.4814
4	0.45	64	0.6136	96	0.5289
4	0.5	64	0.6136	134	0.5677
4	0.55	76	0.6322	134	0.5677
4	0.6	89	0.6431	134	0.5677
4	0.65	96	0.6516	179	0.6119
4	0.7	204	0.6643	211	0.6447
4	0.75	236	0.6804	332	0.6453
4	0.8	256	0.6811	454	0.6528
4	0.85	307	0.6895	563	0.6651
4	0.9	396	0.7045	601	0.6783
4	0.95	633	0.7158	787	0.6904
4	1.0	1024	0.7222	1024	0.6937
5	0.4	128	0.6614	128	0.5302
5	0.45	128	0.6614	140	0.5457
5	0.5	128	0.6614	179	0.5845
5	0.55	128	0.6614	179	0.5845
5	0.6	140	0.6723	217	0.6296
5	0.65	140	0.6723	217	0.6296
5	0.7	243	0.683	243	0.6546
5	0.75	268	0.6942	358	0.6556
5	0.8	281	0.6944	473	0.6628
5	0.85	332	0.7029	576	0.6699
5	0.9	409	0.7072	614	0.6832
5	0.95	640	0.7158	793	0.6904
5	1.0	1024	0.7222	1024	0.6937
6	0.4	256	0.7016	281	0.6414
6	0.45	256	0.7016	281	0.6414
6	0.5	256	0.7016	307	0.6627

α	порог c	Угол (IOU)	Метрика IOU	Угол (SBD)	Метрика SBD
6	0.55	256	0.7016	307	0.6627
6	0.6	256	0.7016	307	0.6627
6	0.65	256	0.7016	307	0.6627
6	0.7	332	0.7001	307	0.6627
6	0.75	358	0.7113	486	0.6702
6	0.8	358	0.7113	512	0.6708
6	0.85	384	0.7108	614	0.6779
6	0.9	460	0.7152	640	0.6858
6	0.95	665	0.7183	819	0.693
6	1.0	1024	0.7222	1024	0.6937
7	0.4	512	0.7208	512	0.6833
7	0.45	512	0.7208	512	0.6833
7	0.5	512	0.7208	512	0.6833
7	0.55	512	0.7208	512	0.6833
7	0.6	512	0.7208	512	0.6833
7	0.65	512	0.7208	512	0.6833
7	0.7	563	0.7194	563	0.6866
7	0.75	563	0.7194	614	0.6799
7	0.8	563	0.7194	614	0.6799
7	0.85	563	0.7194	716	0.687
7	0.9	614	0.7184	716	0.687
7	0.95	768	0.7213	870	0.6931
7	1.0	1024	0.7222	1024	0.6937
8	0.4	1024	0.7222	1024	0.6937
8	0.45	1024	0.7222	1024	0.6937
8	0.5	1024	0.7222	1024	0.6937
8	0.55	1024	0.7222	1024	0.6937
8	0.6	1024	0.7222	1024	0.6937
8	0.65	1024	0.7222	1024	0.6937
8	0.7	1024	0.7222	1024	0.6937
8	0.75	1024	0.7222	1024	0.6937
8	0.8	1024	0.7222	1024	0.6937

α	порог c	Угол (IOU)	Метрика IOU	Угол (SBD)	Метрика SBD
8	0.85	1024	0.7222	1024	0.6937
8	0.9	1024	0.7222	1024	0.6937
8	0.95	1024	0.7222	1024	0.6937
8	1.0	1024	0.7222	1024	0.6937