

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «МОСКОВСКИЙ
ФИЗИКО-ТЕХНИЧЕСКИЙ ИНСТИТУТ (НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)»

На правах рукописи



Соколов Василий Александрович

**Методы активного обучения детекторов объектов в задачах
видеоаналитики на основе масштабируемых представлений**

Специальность 1.2.1. –
«Искусственный интеллект и машинное обучение»

Диссертация на соискание учёной степени
кандидата технических наук

Научный руководитель:
Д.Т.Н.
Матвеев Иван Алексеевич

Москва – 2026

Оглавление

	Стр.
Введение	4
Глава 1. Детекция объектов, методы активного обучения и представления объектов в задачах видеоаналитики	12
1.1 Детекция объектов в задачах видеоаналитики	12
1.2 Проблема формирования обучающих данных и требования к критерию активного обучения	18
1.3 Критерии отбора: неопределённость, разнообразие и Plug and Play Active Learning (PPAL)	20
1.3.1 Формальная постановка задачи пакетного активного обучения	20
1.3.2 Дилемма исследования и использования	22
1.3.3 Критерии неопределённости	24
1.3.4 Критерии разнообразия	25
1.3.5 Наилучший известный метод (state of the art): PPAL	26
1.4 Выводы по главе 1	29
Глава 2. AREA: активное обучение на основе локальных эмбеддингов объектов	30
2.1 Локальные эмбеддинги объектов	30
2.2 Предобработка эмбеддингов и критерий отбора AREA	34
2.3 Схема экспериментов	39
2.4 Наборы данных	41
2.5 Результаты AREA	42
2.6 Выводы по главе 2	49
Глава 3. Модификация выходного модуля YOLOv8 по принципу Matryoshka Representation Learning	51
3.1 Вложенные представления объектов	51
3.2 Обзор литературы	53

3.3	Модификация выходного модуля YOLOv8 по принципу Matryoshka Representation Learning	55
3.4	Стабилизация обучения и извлечение вложенных эмбеддингов . . .	57
3.5	Каскадный поиск по вложенным эмбеддингам	61
3.6	Экспериментальная проверка модели с вложенными представлениями	64
3.7	Выводы по главе 3	67
Глава 4. MAREA: активное обучение на основе вложенных эмбеддингов объектов		
4.1	Метод MAREA	69
4.2	Схема экспериментов	70
4.3	Результаты	71
4.4	Сводный анализ и скорость работы	76
4.5	Выводы по главе 4	76
4.6	Ограничения и направления дальнейших исследований	77
Заключение		79
Список сокращений и условных обозначений		81
Словарь терминов		83
Список литературы		89
Список рисунков		97
Список таблиц		99

Введение

В настоящее время методы компьютерного зрения на основе глубоких нейронных сетей стали основным инструментом решения прикладных задач (детекции объектов). Задача детекции объектов состоит в локализации экземпляров объектов на изображении и определении их классов. Такая постановка востребована в видеоаналитике, промышленном контроле, мониторинге транспортной и городской инфраструктуры и других сценариях, где требуется автоматически находить множество объектов в сложной сцене.

Качество нейросетевого детектора определяется не только архитектурой модели и процедурой обучения, но и составом обучающего множества. Для задачи детекции это обстоятельство имеет особое значение. В отличие от классификации изображений, где одному изображению обычно сопоставляется одна метка, разметка для детекции требует построения ограничивающих прямоугольников для каждого объекта и указания его класса. Одно изображение может содержать множество объектов разных категорий, масштаба и степени видимости. Поэтому получение разметки является трудоёмким процессом, а стоимость расширения обучающего набора может стать главным ограничением при переносе детектора в новый домен.

Реализация рационального выбора данных для разметки называется задачей активного обучения. Вместо того чтобы размечать весь доступный набор изображений, в активном обучении итеративно выбираются те изображения, добавление которых должно дать наибольший прирост качества. На каждом раунде активного обучения детектор обучается на текущем размеченном множестве, затем неразмеченный набор оценивается по некоторому критерию, после чего выбранные изображения передаются на разметку и добавляются к обучающим данным. Критерий отбора (также называемый методом активного обучения) представляет собой правило, по которому из неразмеченного множества выбираются изображения для передачи на разметку. Он является ключевым компонентом схемы активного обучения, поскольку определяет, какими данными будет пополняться обучающая выборка при ограниченном бюджете разметки.

В литературе критерии активного обучения обычно разделяют на три группы: методы, основанные на неопределённости модели, методы, основанные на разнообразии данных, и гибридные методы. Для детекции объектов эти груп-

пы критериев имеют дополнительные ограничения. Неопределённость детектора может быть распределена между несколькими объектами изображения, фоном и ошибками локализации. Разнообразие изображений также не сводится к глобальному сходству кадров: два изображения могут быть похожи в целом, но содержать разные редкие объекты, и наоборот. Поэтому в критерии отбора для детекции должна учитываться локальная структура объектов на изображении.

Одним из естественных способов такого учёта является переход от глобальных признаков изображения к локальным эмбедингам объектов. В этой постановке каждому объекту, найденному или размеченному на изображении, сопоставляется векторное представление, извлечённое из внутренних или выходных представлений детектора. Затем ценность изображения может определяться через расстояния между объектами из неразмеченного набора и объектами, уже представленными в размеченном множестве. При таком подходе выбираются изображения, содержащие объекты, недостаточно покрытые текущей обучающей выборкой, и критерий отбора становится более согласованным со структурой задачи детекции.

Однако использование локальных эмбедингов порождает две отдельные проблемы. Первая связана с тем, как именно извлекать представление объекта из детектора. В архитектурах современных детекторов используются многомасштабные карты признаков, и объект на изображении должен быть корректно сопоставлен с соответствующей областью на этих картах. Алгоритм билинейной интерполяции по суб-пикселям (RoI Align) обеспечивает извлечение признаков из области интереса без грубого округления координат и поэтому является естественным инструментом для формирования локального эмбединга объекта. Вторая проблема связана с размерностью и качеством самих эмбедингов. Представления высокой размерности могут быть информативными, но их размер вызывает трудности при хранении, сравнении и поиске ближайших соседей; слишком компактные представления, наоборот, могут терять семантическую информацию, то есть сведения о смысловом содержании и классе объекта, и пространственную информацию, то есть сведения о его расположении и взаимном расположении частей.

В настоящей диссертационной работе рассматривается последовательное развитие этого направления исследований. Предлагается метод AREA (Active learning with Region Embedding Alignment), в котором локальные эмбединги объектов извлекаются с использованием RoI Align и применяются для отбора изображений.

ражений по новизне относительно уже размеченного множества. Далее предлагается архитектурное развитие источника эмбедингов объектов: модификация выходного модуля YOLOv8 по принципу Matryoshka Representation Learning. Этот принцип состоит в совместном обучении полного вектора признаков и его начальных частей заранее выбранной длины, каждая из которых должна служить самостоятельным информативным представлением объекта. В такой модификации формируется семейство вложенных эмбедингов разной размерности в рамках одной модели, а укороченные подвекторы могут использоваться для более быстрого поиска. Получаемые представления являются масштабируемыми в двух смыслах: их размерность можно изменять выбором длины вложенного подвектора, а каждый подвектор сохраняет признаки с разных уровней пирамиды, соответствующих разным пространственным масштабам объекта. На основе этих представлений строится метод MAREA (Matryoshka Active learning with Region Embedding Alignment), применяющий вложенные эмбединги объектов в задаче активного обучения.

Методы активного обучения широко исследуются во многих научных коллективах в России и за рубежом. Отбор по неопределённости для обучения классификаторов текстов исследовали Д. Льюис и У. Гейл [40], отбор представительного подмножества данных в пространстве признаков нейронной сети – О. Сенер и С. Савареze [56]. Системный обзор семейств методов активного обучения проведен Сеттлзом [57]. В задаче активного обучения детекторов объектов Т. Юань и соавторы исследовали отбор по неопределённости отдельных объектов [49], В. Юй и соавторы – по согласованности предсказаний на исходных и преобразованных изображениях [19]. Одноэтапный детектор YOLO предложили Дж. Редмон и соавторы [78], пирамидальную сеть признаков FPN предложили Т.-И. Лин и соавторы [28], а операцию выравнивания областей интереса RoI Align – К. Хэ, Г. Гкиоксари, П. Доллар и Р. Гиршик в работе Mask R-CNN [43]. Основы метода обучения вложенных векторных представлений разработали А. Кусупати, Г. Бхатт, А. Реge, П. Джайн и другие соавторы [47]. Метод приближённого поиска ближайших соседей HNSW на основе иерархических графов предложили Ю. А. Малков и Д. А. Яшунин [41].

В главе 1 рассмотрены критерии активного обучения, основанные на неопределённости, разнообразии и сочетании этих оценок, а также многомасштабные архитектуры детекторов. Критерии неопределённости оценивают уверенность предсказаний, но сами по себе не обеспечивают разнообразия отбирае-

мых объектов; сравнение глобальных представлений изображений не позволяет отдельно оценить новизну каждого объекта. При использовании локальных представлений необходимо дополнительно учитывать признаки разных пространственных масштабов и затраты на сравнение большого числа объектов. Поэтому в настоящей работе совместно исследуются оценка новизны отдельных объектов относительно размеченного множества, извлечение их признаков с нескольких уровней детектора и ускорение поиска за счёт вложенных представлений.

Целью диссертационной работы является исследование и разработка метода активного обучения, обеспечивающего быстрый и эффективный отбор данных для обучения нейросетевых детекторов с учётом признаков отдельных объектов, их внутриклассовой вариативности и различий в масштабе.

Для достижения поставленной цели решены следующие **задачи**:

1. Сформулировать требования к критериям отбора данных при активном обучении нейросетевых детекторов объектов с учётом специфики задач видеоаналитики и установить, насколько существующие подходы удовлетворяют этим требованиям.
2. На основе сформулированных требований разработать и реализовать метод отбора изображений по локальным представлениям содержащихся в них объектов. Исследовать различные метрические оценки полезности изображения. Экспериментально оценить влияние предложенного метода на качество детекции при фиксированном объёме разметки.
3. Разработать и реализовать способ формирования вложенных векторных представлений объектов, позволяющий использовать начальные части вектора как самостоятельные представления разной размерности с сохранением информации с различных пространственных масштабов. Верифицировать метод на выбранных наборах данных, оценить влияние предложенного способа на качество детекции, качество и время поиска похожих объектов.
4. Разработать и реализовать метод активного обучения детекторов объектов, использующий масштабируемые представления и последовательное уточнение результатов поиска при увеличении размерности сравниваемых векторов. Определить его преимущества и ограничения относительно метода на основе представлений фиксированной размерности и существующих методов активного обучения по качеству детекции и полному времени подготовки данных для разметки.

Объектом исследования являются нейросетевые методы детекции объектов и процедуры итеративного формирования обучающих множеств для таких моделей. Предметом исследования являются критерии активного обучения, основанные на локальных эмбедингах объектов, а также архитектурные способы формирования вложенных представлений объектов в детекционной модели.

Научная новизна:

1. Предложен новый метод активного обучения детекторов объектов, в котором отбор изображений определяется за счёт расположения локальных представлений отдельных объектов относительно уже размеченного множества, а сами представления объединяют признаки из всех соответствующих объекту областей интереса на выходах модели, соответствующих разному пространственному разрешению.
2. Впервые предложен способ обучения детектора объектов, формирующего семейство двумерных вложенных согласованных представлений объектов, извлекаемых из карт активации. Предложенная структура эмбедингов позволяет эффективно и быстро использовать их в поиске похожих изображений.
3. Предложен метод активного обучения на основе вложенных представлений объектов, сочетающий чередование признаков разных уровней пирамиды и каскадный поиск ближайших соседей, что позволяет ускорить и повысить качество поиска кандидатов для последующей разметки.

Практическая значимость результатов работы состоит в расширении инструментария активного обучения для детекции объектов за счёт предложенных методов. Разработанный критерий удовлетворяет требованиям, накладываемым спецификой обучения детекторов для видеоаналитики. Так, он совместим с наиболее быстрыми и точными архитектурами детекторов, т.е. детекторов, использующих карты активации на нескольких масштабах. Кроме того, он позволяет различать визуально разные группы объектов внутри одного и того же класса. Наконец, предлагаемый метод активного обучения опережает по скорости и качеству работы лучший из представленных на данный момент в литературе методов активного обучения детекторов.

Методология и методы исследования. Методы исследования включают в себя методы активного обучения, методы детекции объектов на основе глубоких нейронных сетей с помощью YOLOv8, методы извлечения локальных признаков с помощью RoI Align, методы метрического сравнения эмбедингов, метод пони-

жения размерности PCA (Principal Component Analysis), методы обучения вложенных представлений на основе Matryoshka Representation Learning, приближённый и каскадный поиск ближайших соседей, а также экспериментальную оценку качества детекции по мере качества mAP50–95 (mean Average Precision).

Основные положения, выносимые на защиту:

1. Метод активного обучения детекторов AREA, отличающийся способностью распознавать внутриклассовые визуальные различия объектов при отборе изображений по новизне отдельных объектов относительно размеченного множества за счёт объединения локальных эмбедингов объектов, извлекаемых со всех уровней выходов модели без округления координат, за счет чего совместимый с наиболее распространенным, мультимасштабным классом детекторов.
2. Принцип модификации детекторов для формирования вложенных пространственных представлений объектов по схеме матрешки, основанный на совместном обучении выходного модуля детектора с общими весами на полных и сокращённых наборах каналов и чередовании уровней пирамиды признаков, что позволяет ускорить задачу нахождения похожих объектов за счет использования каскадного поиска и делает вложенные эмбединги пригодными для использования в различных приложениях.
3. Метод активного обучения детекторов MAREA, основанный на вложенных представлениях объектов из карт активации нейросети, каскадном поиске и обеспечивающий преимущество перед AREA и стадией отбора по разнообразию PPAL (Plug and Play Active Learning) по совокупности качества детекции и скорости отбора данных в проведённых экспериментах.

Содержание диссертации соответствует паспорту научной специальности 1.2.1. Искусственный интеллект и машинное обучение, в частности, пунктам:

- пункт 4 – «Разработка методов, алгоритмов и создание систем искусственного интеллекта и машинного обучения для обработки и анализа текстов на естественном языке, для изображений, речи, биомедицины и других специальных видов данных». В части: «обработки и анализа изображений»;
- пункт 13 – «Методы и средства формирования массивов данных и прецедентов, включая «большие данные», необходимых для решения

- задач искусственного интеллекта и машинного обучения. Проблемно-ориентированные коллекции данных для важных прикладных областей». В части: «формирования массивов данных и прецедентов, необходимых для решения задач искусственного интеллекта и машинного обучения»;
- пункт 16 – «Исследования в области специальных методов оптимизации, проблем сложность и элиминации перебора, снижения размерности». В части: «специальных методов оптимизации и снижения размерности»;
 - пункт 17 – «Исследования в области многослойных алгоритмических конструкций, в том числе – многослойных нейросетей».

Достоверность результатов обеспечивается корректной постановкой задачи активного обучения, использованием общепринятых архитектур детекции объектов, сравнением с базовыми стратегиями отбора, экспериментальной проверкой на нескольких наборах данных, применении стандартных мер качества детекции и апробацией на практике. Статистическая значимость полученных результатов проверена с помощью многократных повторных запусков экспериментов, подсчёта средних значений и доверительного интервала.

Апробация работы. Основные результаты диссертационной работы докладывались и обсуждались на следующих конференциях: 15-й Международной конференции «Интеллектуализация обработки информации» (ИОИ-2024, Гродно, Беларусь, 2024); 67-й Всероссийской научной конференции Московского физико-технического института (Долгопрудный, Россия, 2025); 22-й Всероссийской конференции «Математические методы распознавания образов» (Муром, Россия, 22–26 сентября 2025 г.); 28-й Международной конференции по распознаванию образов (ICPR-2026) в рамках 10-го Международного семинара «Image Mining. Theory and Applications» (IMTA-X-2026, Лион, Франция, 21 августа 2026 г.).

Личный вклад. Все результаты, представленные в работах, включённых в диссертационное исследование, получены лично соискателем под непосредственным научным руководством доктора технических наук Матвеева Ивана Алексеевича. В работах с соавторами личный вклад соискателя заключается в постановке задач, реализации методов, постановке основных численных экспериментов, анализе результатов и подготовке материалов к публикации.

Публикации. Результаты по теме диссертации опубликованы в 5 научных работах, включая 2 статьи в журналах из списка ВАК, индексируемых в Web of Science и Scopus [4; 61], и 3 публикации в сборниках тезисов докладов конферен-

ций [5–7]. Получены 3 свидетельства о государственной регистрации программ для ЭВМ [1–3].

Объём и структура работы. Диссертация состоит из введения, 4 глав и заключения. Первая глава посвящена анализу методов активного обучения для детекции объектов и обоснованию перехода от глобальных признаков изображения к локальным эмбедингам объектов. Во второй главе представлен метод AREA, использующий локальные эмбединги объектов и критерии отбора по расстоянию до размеченного множества. В третьей главе описана модификация выходного модуля YOLOv8 по принципу Matryoshka Representation Learning, формирующая семейство масштабируемых вложенных эмбедингов объектов. В четвёртой главе предложен метод MAREА, применяющий вложенные эмбединги в задаче активного обучения детекторов объектов.

Полный объём диссертации составляет 99 страниц, включая 28 рисунков и 13 таблиц. Список литературы содержит 83 наименования.

Глава 1. Детекция объектов, методы активного обучения и представления объектов в задачах видеоаналитики

1.1 Детекция объектов в задачах видеоаналитики

Задача детекции объектов заключается в одновременной классификации и локализации всех объектов на изображении [51]. Результат работы детектора представляет собой множество ограничивающих прямоугольников с метками классов и оценками уверенности. По сравнению с задачей классификации изображения детекция существенно сложнее: одно изображение содержит произвольное число объектов разных классов, объекты перекрываются, появляются при различных условиях освещения и ракурсах, занимают разные доли площади кадра.

Детекция объектов является основной инструментальной задачей в системах видеоаналитики [51]. К типичным сценариям применения относятся контроль соблюдения требований безопасности на производстве, мониторинг транспортных потоков, охрана периметра и автоматизированный контроль технологических процессов. Во всех этих сценариях данные поступают в виде непрерывного видеопотока с фиксированных или подвижных камер.

Промышленные системы видеоаналитики функционируют в условиях высокой вариативности визуальной обстановки. Один и тот же объект может выглядеть принципиально по-разному в зависимости от погоды (снег, дождь, туман, пар), освещения (яркий солнечный свет, тени, блики, ночная подсветка), ракурса камеры, высоты её установки и дальности до объекта. Следствием этой вариативности является наличие у каждого семантического класса нескольких визуально различимых внутриклассовых подгрупп. Например, пешеход в защитном снаряжении в условиях задымления и тот же пешеход на открытой площадке при дневном свете принадлежат одному классу, но существенно различаются по внешнему виду. Детектор остаётся уязвимым к таким подгруппам, пока они слабо представлены в обучающем множестве – даже при большом числе уже размеченных кадров с объектами того же класса.

Системы видеоаналитики требуют работы в режиме реального времени. Для задач контроля безопасности и управления процессом допустимая задержка зачастую не превышает десятков миллисекунд, что ограничивает выбор архи-

тектуры детектора [51]. На практике это условие выдвигает на первый план *одностадийные детекторы* (single-stage detectors) – архитектуры, в которых классификация и локализация объектов выполняются за единственный прямой проход через сеть без явного этапа формирования областей-кандидатов [34]. К этому классу относятся семейство You Only Look Once (YOLO), Single Shot MultiBox Detector (SSD) и RetinaNet. В отличие от *двухстадийных детекторов* (two-stage detectors), таких как Faster Region-based Convolutional Neural Network (Faster R-CNN) и Mask Region-based Convolutional Neural Network (Mask R-CNN), которые сначала формируют области-кандидаты, а затем классифицируют каждую из них, одностадийные архитектуры достигают значительно более высокой скорости вывода при сопоставимой точности.

Для описания частей архитектуры детектора в работе используются следующие термины. *Основной модуль* (backbone) извлекает из входного изображения иерархию карт признаков. *Переходный модуль* (neck) объединяет и преобразует карты признаков основного модуля для передачи выходному модулю. *Выходной модуль* (detection head) формирует предсказания ограничивающих прямоугольников и классов объектов [28; 34].

Ещё одной характерной чертой современных детекторов является мультимасштабная архитектура. Объекты на промышленных изображениях одновременно присутствуют в широком диапазоне размеров: мелкие удалённые экземпляры, крупные объекты переднего плана, частично перекрытые объекты у края кадра. Ранние слои свёрточной сети сохраняют высокое пространственное разрешение и мелкие детали, глубокие слои дают пространственно огрублённые, но семантически насыщенные карты. Для одновременной детекции объектов всех масштабов применяется архитектура пирамиды признаков Feature Pyramid Network (FPN), введённая в работе Lin et al. [28]. В FPN признаки нескольких слоёв объединяются через нисходящий проход с боковыми соединениями, а предсказания формируются независимо на каждом уровне пирамиды выходного модуля. В YOLOv8 и большинстве современных быстрых детекторов используется переходный модуль типа FPN, объединяющий признаки нескольких уровней [28; 34]. Наличие мультимасштабного переходного модуля стало фактическим стандартом современных детекторов, обеспечивая высокую точность при работе в реальном времени. Следовательно, представления объектов, используемые в критериях активного обучения, должны учитывать мультимасштабные карты признаков, чтобы сохранять информацию об объектах разных размеров.

Анализ 36 детекторов позволяет выделить несколько семейств по способу организации мультимасштабных признаков.

YOLO-семейство. Первые версии YOLO [53; 78] работали с единственной картой признаков и не имели явного мультимасштабного компонента. В YOLOv3 [54] применяется объединение признаков по схеме, аналогичной FPN: карты глубоких слоёв увеличиваются до разрешения более ранних карт и объединяются с ними. Предсказания формируются на трёх уровнях P3, P4 и P5 с шагами 8, 16 и 32 пикселя относительно входного изображения. В последующих моделях, в частности YOLOv8 [34], нисходящий путь дополняется восходящим путём объединения признаков по схеме Path Aggregation Network (PANet). В PP-YOLOE [52] переходный модуль PAN построен из блоков CSPRepResStage. Во всех этих моделях объекты детектируются на трёх уровнях пирамиды.

Классические одностадийные детекторы. В RetinaNet [29] FPN одной из первых была применена в одностадийном детекторе с добавлением уровней P6 и P7 для охвата более крупных объектов. В Fully Convolutional One-Stage Object Detection (FCOS) [27] и Generalized Focal Loss (GFL) [31] используется аналогичная пирамида из пяти уровней (P3–P7), полностью сохраняющая топологию FPN с нисходящим проходом и боковыми соединениями.

Двухстадийные детекторы. В Faster R-CNN [26] с FPN сеть формирования областей-кандидатов (Region Proposal Network, RPN) использует пирамиду P2–P5: на разных уровнях пирамиды обрабатываются якоря разных размеров. В Cascade R-CNN [17] эта схема расширяется тремя последовательными каскадами с нарастающим порогом Intersection over Union (IoU) при сохранении переходного модуля FPN. В CenterNet2 [80] идея вероятностной двухстадийной детекции объединяется с переходным модулем FPN: на первом этапе по всем уровням пирамиды формируются центры объектов, на втором – по FPN-признакам уточняются классы и рамки.

Одностадийные детекторы без FPN. В SSD [64] напрямую используется несколько карт основного модуля без нисходящего прохода – это примитивная многомасштабность, в которой мелкие уровни лишены семантики глубоких слоёв. В YOLOv1 [78] и YOLOv2 [53] использовался единственный уровень признаков; мультимасштабности в современном смысле у них нет. В YOLOF [75] FPN осознанно не используется: обрабатывается только C5-уровень основного модуля, что компенсируется дилатированными свёртками с разными коэффициентами расширения. В CenterNet-DLA [81] основной модуль Deep Layer Aggregation

(DLA) объединяет карты признаков разных пространственных разрешений без отдельного переходного модуля FPN. В CornerNet [38] и ExtremeNet [82] эту функцию выполняет основной модуль Hourglass. В этих моделях предсказания объектов формируются по итоговой карте признаков одного разрешения; в принятой классификации такие модели относятся к *одномасштабным*.

Семейство Detection Transformer (DETR). В исходном DETR [25] используется единственный уровень C5; DETR-DC5 [25] – его вариант с дилатированным основным модулем C5, улучшающий разрешение, но по-прежнему без FPN. В Anchor DETR [13] и Conditional DETR [18] также используются преимущественно одноуровневые признаки. В Deformable DETR [21] деформируемый механизм внимания объединяет признаки четырёх разрешений, полученные из уровней C3–C5 основного модуля и дополнительного уровня C6. Обмен информацией между уровнями выполняется механизмом внимания без нисходящего пути FPN. В DETR with Improved deNoising anchOr boxes (DINO) [23], Collaborative DINO (CoDINO) [83] и Real-Time Detection Transformer (RT-DETR) [22] также используются многомасштабные признаки и специальные способы инициализации запросов. В частности, при обучении DINO применяется восстановление исходных меток классов и ограничивающих прямоугольников по запросам с добавленными случайными возмущениями (denoising); запросы с более сильными возмущениями используются как отрицательные примеры [23].

InternImage. В InternImage-XL [33] используется основной модуль с деформируемыми свёртками Deformable Convolutional Network v3 (DCN v3), обеспечивающими адаптивное поле восприятия, а также стандартный переходный модуль FPN для мультимасштабного вывода. В принятой классификации эта модель относится к классу мультимасштабных, однако занимает нишу крупных моделей с высокой вычислительной стоимостью.

EfficientDet. В EfficientDet [66] вводится Bidirectional Feature Pyramid Network (BiFPN) – двунаправленная взвешенная пирамида признаков с обучаемыми весами слияния на каждом уровне, применяемая несколько раз подряд. За счёт такой глубокой пирамиды обеспечивается плотный обмен информацией между уровнями и улучшается точность локализации мелких объектов.

Таким образом, эволюция архитектур демонстрирует единую тенденцию: одностадийные и двухстадийные детекторы сходятся к схемам с явным мультимасштабным переходным модулем (FPN, PANet, BiFPN или деформируемое внимание на нескольких уровнях). Архитектура FPN показана на рисунке 1.1.

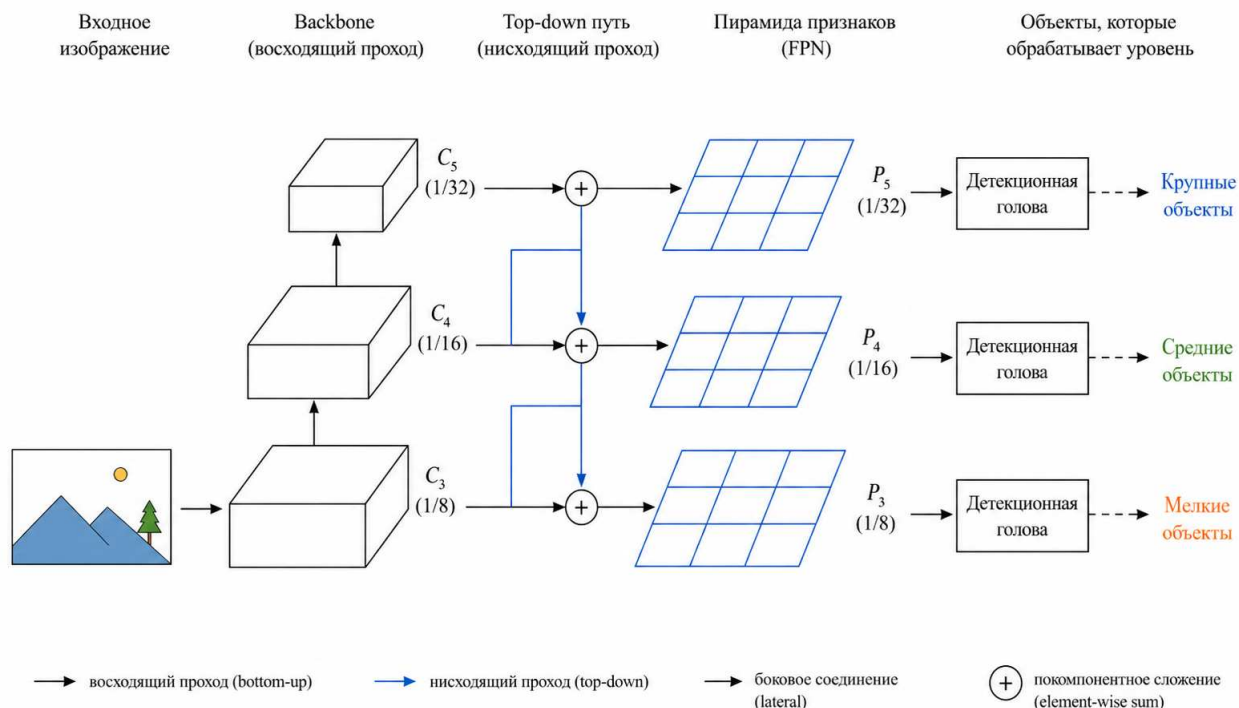


Рисунок 1.1 – Архитектура пирамиды признаков (FPN) [28].

Детекторы без такого переходного модуля (красные квадраты на рисунке 1.2) представлены преимущественно старыми архитектурами или специализированными решениями для узких задач. На рисунке 1.2 цифры на точках соответствуют номерам в таблице 1. Данные по задержке и mean Average Precision (mAP) приведены из публикаций авторов.

Таблица 1 – Детекторы объектов, представленные на рисунке 1.2

№	Модель	mAP, %	№	Модель	mAP, %	№	Модель	mAP, %
1	YOLOv10-L [76]	53.3	13	EfficientDet-D1 [66]	39.6	25	Faster R-CNN [26]	36.2
2	SSD300 [64]	25.1	14	YOLOv1 [78]	23.7	26	Anchor DETR [13]	44.2
3	YOLO11-L [35]	53.4	15	SSD512 [64]	28.8	27	DINO-4scale [23]	51.3
4	YOLOv12-L [70]	53.7	16	PP-YOLOE-L [52]	50.9	28	EfficientDet-D4 [66]	49.7
5	YOLOv8-L [34]	52.9	17	RT-DETRv2-X [55]	54.3	29	Cond. DETR [18]	43.8
6	YOLOX-L [77]	50.1	18	YOLOv3 [54]	33.0	30	DETR [25]	42.0
7	YOLOv7 [72]	51.4	19	CenterNet-DLA [81]	37.4	31	Cascade R-CNN [17]	42.8
8	YOLOv5-L [71]	49.0	20	CenterNet2 [80]	42.5	32	DETR-DC5 [25]	43.3
9	YOLOv9-C [73]	53.0	21	FCOS [27]	41.5	33	Co-DINO [83]	60.7
10	YOLOv2 [53]	21.6	22	RetinaNet [29]	39.1	34	InternImage-XL [33]	56.2
11	RT-DETR-L [22]	53.1	23	GFL [31]	45.0	35	CornerNet [38]	40.5
12	YOLOF [75]	37.7	24	Def. DETR [21]	46.2	36	ExtremeNet [82]	40.2

Таким образом, детекция объектов в видеоаналитике характеризуется высокой внутриклассовой визуальной вариативностью, требованием реального времени и мультимасштабной архитектурой детектора. Эти свойства непосредственно

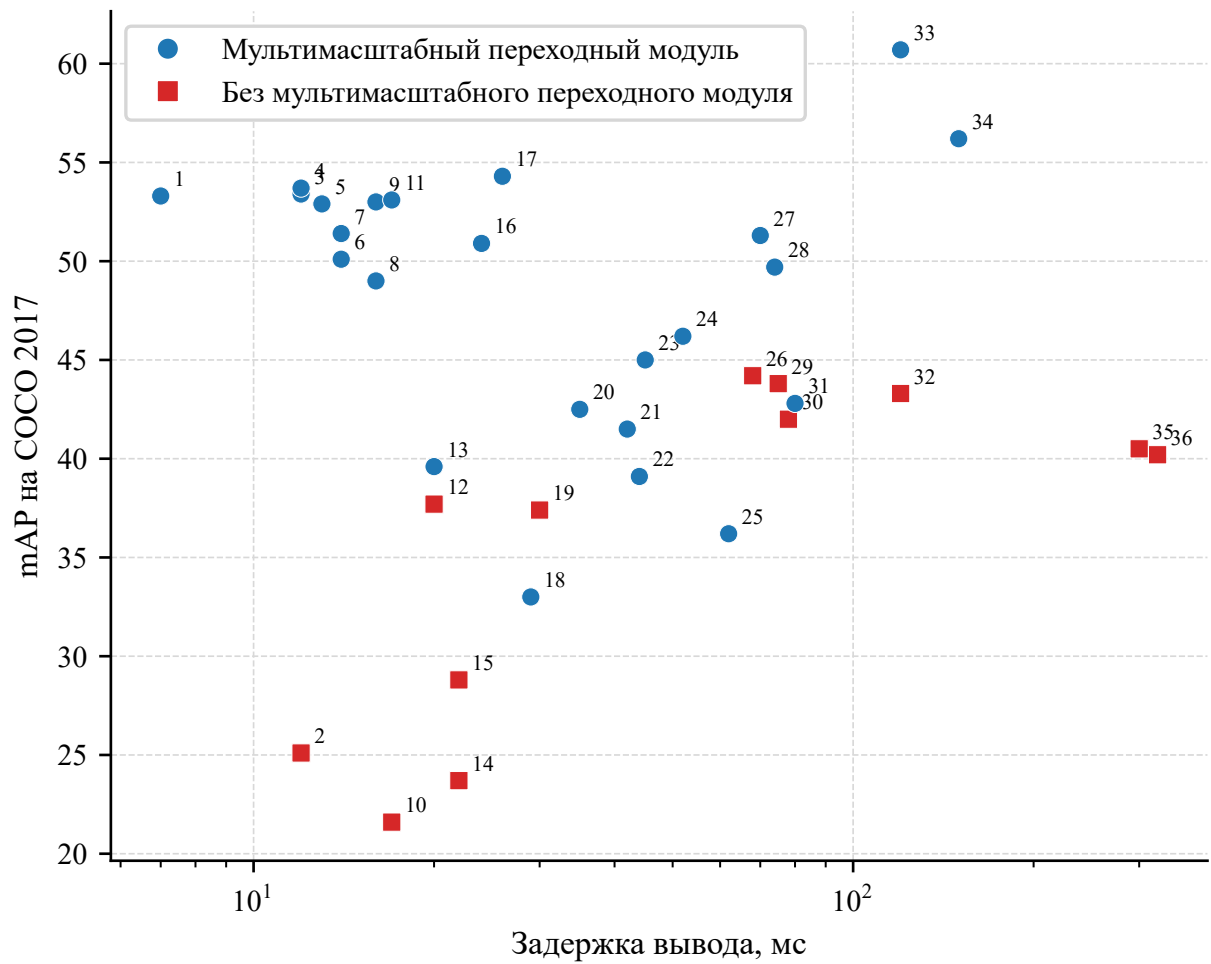


Рисунок 1.2 – Точность и скорость детекторов объектов на Common Objects in Context 2017 (COCO 2017).

определяют требования к представлениям объектов, которые будут использоваться в критериях активного обучения.

1.2 Проблема формирования обучающих данных и требования к критерию активного обучения

Обучение нейросетевых детекторов требует значительного объёма размеченных данных [39]. При этом разметка для задачи детекции существенно более трудоёмкая, чем для классификации изображений. Разметчик обязан найти каждый объект на изображении, провести ограничивающий прямоугольник с требуемой точностью и указать класс каждого экземпляра. На одном кадре промышленной видеосцены может присутствовать несколько десятков объектов, а значит, усилия разметчика растут не только с числом изображений, но и с числом объектов [10; 67].

Видеопотоки обладают свойствами, которые дополнительно осложняют подготовку обучающих данных.

Избыточность. Соседние кадры видеопотока в большинстве случаев фиксируют почти одну и ту же сцену с незначительным смещением объектов. Случайная или равномерная выборка кадров для разметки не устраняет эту избыточность: значительная часть бюджета тратится на изображения, дублирующие уже имеющуюся информацию.

Редкость информативных событий. Нарушения техники безопасности, аварийные конфигурации, нетипичные положения объектов и нестандартные визуальные условия в промышленных видеопотоках появляются редко. Стандартное добавление новых кадров расширяет набор данных, но не гарантирует попадания таких событий в обучающее множество в достаточном количестве.

Частые циклы дообучения. Изменение условий съёмки, ввод нового оборудования или расширение зоны наблюдения требуют регулярного обновления обучающих данных, поскольку домен детектора смещается [67]. Каждый такой цикл дообучения содержит шаг выбора новой порции изображений для разметки при ограниченном бюджете.

Перечисленные свойства делают методы активного обучения (Active Learning, AL) особенно актуальными для детекции объектов в видеоаналитике. С помощью методов AL из неразмеченного пула выбираются те изображения, разметка которых принесёт наибольший прирост качества модели, — вместо случайного или равномерного отбора [57; 67].

Из совокупности свойств задачи вытекают конкретные требования к критерию AL.

1. **Работа на уровне объектов, а не изображений.** Полезность кадра для детектора определяется не общим видом сцены, а тем, какие конкретно объекты в нём содержатся. Для этого каждому найденному объекту сопоставляется *локальный эмбединг* – векторное представление, извлечённое из карты признаков в области ограничивающего прямоугольника этого объекта. Изображение, глобально похожее на уже размеченные, может содержать редкую конфигурацию или новую подгруппу объектов; критерий, работающий только с признаком изображения в целом, пропускает её.
2. **Различение внутриклассовых визуальных подгрупп.** Из-за высокой вариативности промышленных условий объекты одного класса образуют несколько визуально различимых кластеров в пространстве признаков. В критерии отбора должны использоваться представления, отражающие внешний вид объекта, а не только принадлежность к классу – иначе редкие визуальные подгруппы окажутся недопредставленными.
3. **Использование мультимасштабных карт признаков.** Поскольку объекты разных размеров лучше всего описываются различными уровнями пирамиды признаков, локальные представления должны извлекаться с нескольких уровней, чтобы корректно учитывать как мелкие, так и крупные объекты.
4. **Новизна относительно уже размеченного множества.** Ценность кандидата определяется тем, насколько он дополняет уже существующую разметку. В критериях, максимизирующих попарное разнообразие внутри выбираемого пакета, не используется информация о том, какие подгруппы объектов уже достаточно покрыты, поэтому могут выбираться кадры, дублирующие хорошо представленные объекты.

Существующие методы удовлетворяют этим требованиям лишь частично. В следующем разделе рассматриваются основные семейства критериев активного обучения и анализируется их соответствие перечисленным требованиям.

1.3 Критерии отбора: неопределённость, разнообразие и Plug and Play Active Learning (PPAL)

1.3.1 Формальная постановка задачи пакетного активного обучения

В литературе рассматриваются два основных режима активного обучения: потоковый (stream-based, или online) и с пулом (pool-based) [57]. В потоковом режиме данные поступают последовательно, и для каждого нового образца немедленно принимается решение, запрашивать ли его разметку у оракула. Такая постановка естественна для систем реального времени, но плохо согласована с реальным процессом промышленной видеоаналитики: оракул-разметчик – как правило, эксперт-человек – эффективнее работает с пакетами кадров, а не с непрерывным потоком запросов. Кроме того, в потоковом режиме нельзя сравнить кандидата с уже собранными, но ещё не размеченными кадрами, что снижает качество критерия отбора.

В постановке с пулом весь неразмеченный архив доступен заранее, и пакет из b кадров выбирается по текущему состоянию модели. Это соответствует реальному сценарию: видеопоток сначала записывается или предобрабатывается, затем из него отбираются наиболее информативные кадры, которые передаются разметчику в виде пакета. При пакетной стратегии также возможно обучать модель после каждого раунда разметки. В настоящей диссертации используется именно постановка с пулом, которая является стандартом в задачах детекции объектов [61; 67; 74].

Рассмотрим задачу пакетного активного обучения с пулом неразмеченных данных. Пусть задано начальное размеченное множество $L_0 \subset D$, неразмеченный пул $U_0 = D \setminus L_0$ и валидационный набор V . Общий бюджет разметки составляет B изображений, которые выбираются пакетами размера b , так что число раундов равно $T = B/b$. Нейросетевая модель f_θ (с текущим набором параметров θ) обучается на размеченном множестве, а критерий отбора $A(L_{t-1}, U_{t-1}, f_\theta)$ определяет, какие b изображений из пула U_{t-1} следует отправить оракулу-разметчику O на раунде t . Качество модели измеряется мерой $Q(f_\theta, V)$ на валидационной выборке V ; желаемое пороговое значение обозначается Q^* .

На каждом раунде $t = 1, \dots, T$ выполняются следующие шаги:

1. Обучить модель f_θ на текущем размеченном множестве L_{t-1} .
2. Выбрать пакет кандидатов:

$$S_t^* = \arg \max_{S \subset U_{t-1}, |S|=b} A(L_{t-1}, U_{t-1}, f_\theta, S). \quad (1.1)$$

3. Разметить выбранный пакет с помощью оракула:

$$y_x = O(x), \quad \forall x \in S_t^*. \quad (1.2)$$

4. Обновить размеченное и неразмеченное множества:

$$L_t = L_{t-1} \cup \{(x, y_x) \mid x \in S_t^*\}, \quad U_t = U_{t-1} \setminus S_t^*. \quad (1.3)$$

Формально задача активного обучения состоит в максимизации качества итоговой модели при фиксированном бюджете разметки и возможном критерии досрочной остановки [57]:

$$\max_{S_1^*, \dots, S_T^*} Q(f_{\theta_T}, V) \quad \text{при условиях} \quad \begin{cases} |L_T| - |L_0| \leq B, \\ Q(f_{\theta_t}, V) \leq Q^*. \end{cases} \quad (1.4)$$

Цикл завершается либо при исчерпании бюджета B , либо при достижении моделью требуемого качества Q^* . Общий цикл пакетного активного обучения с пулом показан на рисунке 1.3.

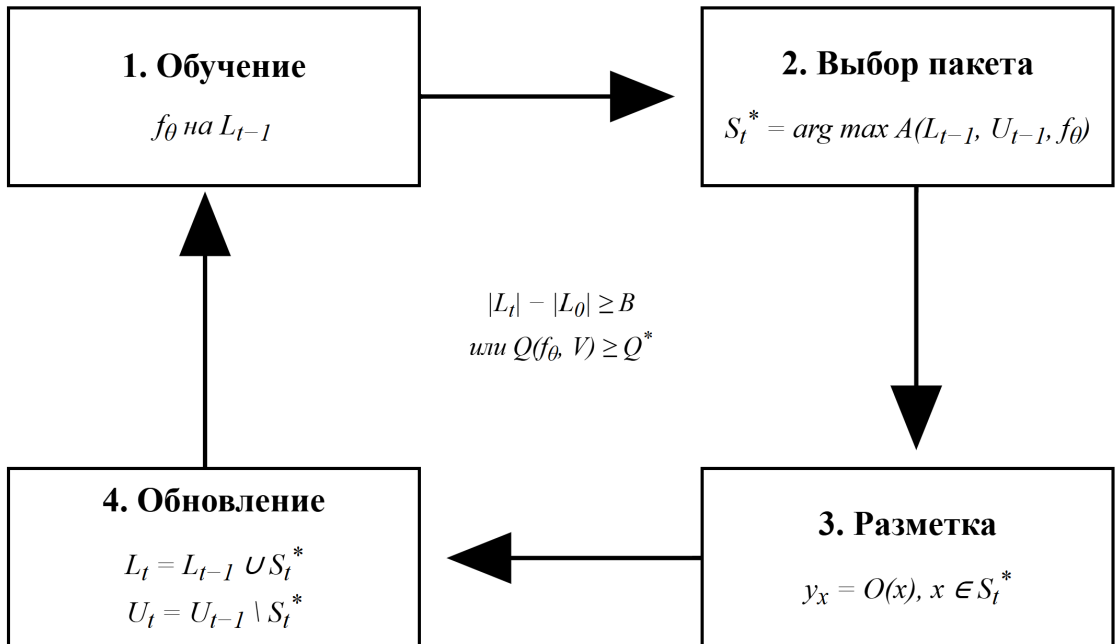


Рисунок 1.3 – Цикл пакетного активного обучения с пулом.

1.3.2 Дилемма исследования и использования

Фундаментальное противоречие при выборе объектов для разметки определяется дилеммой исследования и использования (*exploration/exploitation*), восходящей к задаче многоруких бандитов в теории обучения с подкреплением [15; 65]. В этой задаче агент на каждом шаге должен выбирать между двумя стратегиями: использовать текущее знание, выбирая действие с наибольшей известной наградой, или исследовать малоизученные действия в расчёте на то, что они окажутся ещё более выгодными. Перекос в сторону использования ведёт к застреванию в локальном оптимуме; перекос в сторону исследования – к неэффективному расходованию ресурсов на изучение неперспективных областей.

В активном обучении та же дилемма проявляется как выбор между двумя семействами критериев. Критерии неопределённости реализуют стратегию использования: они концентрируют разметку в окрестности текущей границы решения, где ошибки модели наиболее вероятны и где каждый новый пример максимально уточняет её. Критерии разнообразия реализуют стратегию исследования: они выбирают образцы, равномерно покрывающие пространство признаков, стремясь охватить области, ещё слабо представленные в обучающем множестве [57]. Рисунок 1.4 иллюстрирует разницу в поведении двух стратегий на модельном двумерном примере: стратегия использования выбирает точки вблизи границы решения, тогда как стратегия исследования направляет разметку в отдалённые, плохо покрытые области. Заполненные маркеры обозначают уже размеченные объекты двух классов, полупрозрачные маркеры – неразмеченный пул, пунктир – текущую границу решения. Звёзды обозначают кандидатов, выбираемых каждой стратегией.

На практике ни одна стратегия в чистом виде не является оптимальной [57; 65]. Стратегия использования в чистом виде игнорирует структуру пула: она выбирает для разметки примеры вблизи уже известной границы, оставляя целые кластеры неразмеченными, и в результате модель остаётся уязвимой к новым визуальным условиям. При выборе только стратегии исследования текущее состояние модели не используется: бюджет может направляться на объекты из уже хорошо выученных групп, расположенных далеко от имеющейся разметки, но вовсе не трудных для модели. Именно поэтому в литературе предложены гибридные под-

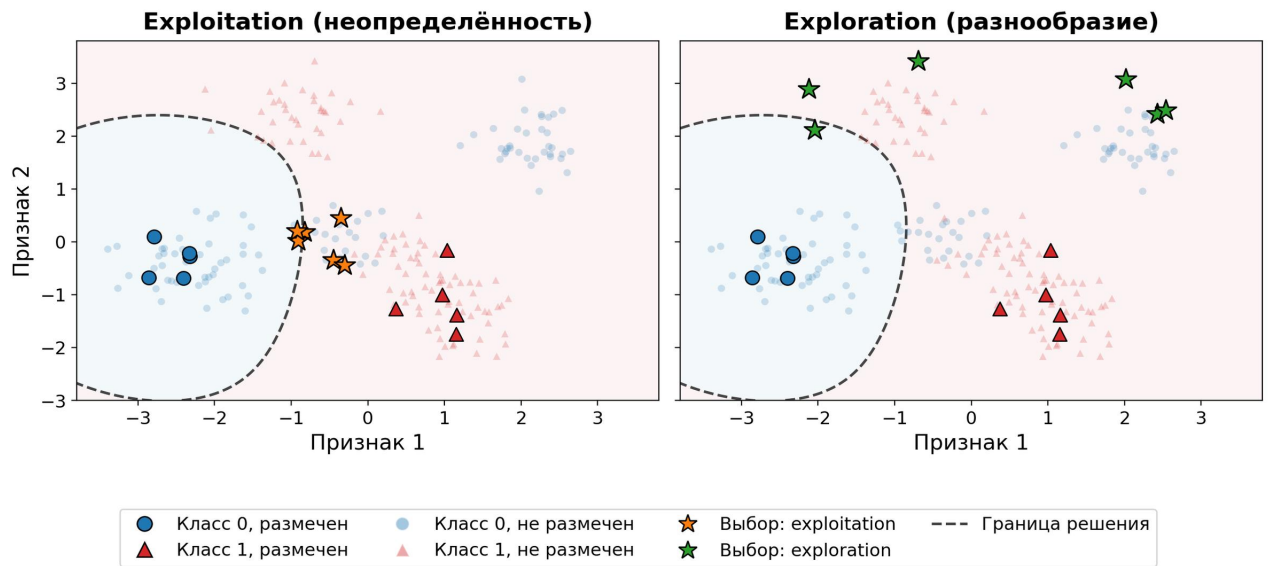


Рисунок 1.4 – Дилемма исследования и использования в активном обучении.

ходы, объединяющие оба сигнала [57; 67], а также методы, специально разработанные под конкретные задачи, – такие как PPAL [74].

1.3.3 Критерии неопределённости

Критерии неопределённости (uncertainty-based) оценивают степень неуверенности модели в собственных предсказаниях и реализуют стратегию использования: разметка направляется туда, где модель ошибается с наибольшей вероятностью, максимизируя выигрыш от каждого нового примера [57; 67]. Для отдельного изображения x вводится функция оценки $a(x)$, на основе которой строится пакетный критерий путём агрегации оценок.

Entropy Sampling. Выбираются изображения с максимальной энтропией апостериорного распределения по классам [57; 59]:

$$a(x) = - \sum_{c=1}^C p_c(x) \log p_c(x), \quad (1.5)$$

где $p_c(x) = p(y = c \mid x, f_\theta)$ – вероятность класса c .

Least Confidence. Приоритет получают изображения с наименьшей максимальной вероятностью предсказанного класса [40]:

$$a(x) = 1 - \max_c p(y = c \mid x, f_\theta). \quad (1.6)$$

Margin Sampling. Использует разность двух наибольших вероятностей классов [37]:

$$a(x) = 1 - (p_1(x) - p_2(x)), \quad (1.7)$$

где $p_1(x)$ и $p_2(x)$ – первая и вторая по убыванию компоненты вектора вероятностей. Чем меньше разрыв между ними, тем менее однозначным является решение модели.

Методы, основанные на несогласии (disagreement-based). В этом подходе высокая ценность присваивается изображениям, для которых разные версии или режимы модели дают существенно различающиеся ответы [57; 58]. Стандартная мера несогласия – энтропия голосов комитета из M предсказаний:

$$a(x) = - \sum_{c=1}^C \frac{V(c, x)}{M} \log \frac{V(c, x)}{M}, \quad (1.8)$$

где $V(c, x)$ – число предсказаний класса c для изображения x . Источником несогласия могут выступать: *Monte Carlo Dropout* [30] – многократный прогон изображения через сеть с включённым случайным отключением нейронов, имитирующим байесовскую неопределённость; *ансамбли моделей* [69] – набор независимо обученных копий детектора, предсказания которых агрегируются; *случайные аугментации входных данных* [19] – несколько аугментированных версий одного изображения подаются в одну модель, и разброс предсказаний отражает её чувствительность к преобразованиям; предсказания с разных уровней пирамиды признаков [49].

Применительно к задаче детекции объектов перечисленные критерии требуют дополнительного шага агрегации: уверенность модели определяется на уровне отдельных предсказаний объектов внутри изображения, и итоговую оценку изображения необходимо получать из оценок составляющих его объектов (например, через максимум или среднее). Семейство критериев неопределённости является наиболее изученным в литературе. Комплексные обзоры [9–11] систематизируют десятки методов этой группы, предоставляют эмпирические сравнения и стандартные протоколы воспроизводимой оценки. Совместный вывод этих обзоров: критерии неопределённости работают надёжно на задачах классификации и хорошо адаптируются к детекции через агрегацию оценок по объектам [67; 74]. В настоящей диссертации основное внимание уделяется критериям разнообразия и представлениям объектов., поскольку именно эта область остаётся менее исследованной и открывает возможности для существенных улучшений в задачах детекции.

1.3.4 Критерии разнообразия

Критерии разнообразия (diversity-based) реализуют стратегию исследования: цель – выбрать пакет, покрывающий пространство признаков пула более равномерно и минимизирующий дублирование уже имеющихся данных [56; 67].

CoreSet. Метод формально решает задачу оптимального покрытия: требуется выбрать подмножество S фиксированного объёма, при котором наибольшее

расстояние от любой точки пула до ближайшей точки в $L \cup S$ минимально [56]:

$$\min_{S \subset U, |S|=b} \max_{u \in U} d(u, L \cup S), \quad (1.9)$$

где $d(\cdot, \cdot)$ – расстояние в пространстве признаков. Поскольку точное решение является NP-трудным (nondeterministic polynomial-time hard), применяется жадный алгоритм k -Center-Greedy, итеративно добавляющий в S элемент, наиболее удалённый от уже выбранного ядра.

k -means++. Стохастический алгоритм инициализации, в котором вероятность выбора очередного кандидата пропорциональна квадрату расстояния до ближайшего уже выбранного центра. Обеспечивает равномерное начальное покрытие пространства признаков [14].

Подходы на основе плотности. Ценность изображения определяется плотностью его окрестности в пространстве признаков: выбираются представители плотных, но ещё не покрытых областей пула [50].

Гибридные критерии. Информацию о неопределённости и разнообразии можно объединить в единый взвешенный критерий [57; 67]:

$$A(L_t, U_t, f_\theta) = \lambda A_{\text{unc}}(L_t, U_t, f_\theta) + (1 - \lambda) A_{\text{div}}(L_t, U_t, f_\theta), \quad \lambda \in [0, 1]. \quad (1.10)$$

Альтернативная реализация – двухступенчатая схема: сначала на этапе отбора по неопределённости формируется промежуточный набор кандидатов, затем на этапе отбора по разнообразию из него выбирается итоговый пакет.

1.3.5 Наилучший известный метод (state of the art): PPAL

Наилучшие среди известных результатов на наборах данных Pascal Visual Object Classes (Pascal VOC) и COCO 2017 получены для метода PPAL (Plag and Play Active Learning) [74]. Метод является двухступенчатым гибридным подходом: на этапе PPAL-uncertainty формируется промежуточное множество кандидатов $H_t \subset U_{t-1}$, после чего на этапе PPAL из H_t выбирается финальный пакет S_t^* . В экспериментальных сравнениях настоящей диссертации используется стадия PPAL-diversity; результаты не относятся к полной двухстадийной процедуре PPAL с предварительным отбором по неопределённости.

Мера сходства объектов. Разнообразие в PPAL оценивается на уровне отдельных предсказаний объектов. Обозначим через $f_{a,i}$ и $f_{b,j}$ эмбединги объектов $o_{a,i}$ и $o_{b,j}$ из изображений O_a и O_b соответственно, а через $c_{a,i}$ – предсказанный класс объекта $o_{a,i}$. Сходство объекта $o_{a,i}$ с изображением O_b определяется как:

$$s(o_{a,i}, O_b) = \begin{cases} \max_j \cos(f_{a,i}, f_{b,j}) + 1, & c_{b,j} = c_{a,i}, \\ 0, & \text{если в } O_b \text{ нет объекта класса } c_{a,i}, \end{cases} \quad (1.11)$$

где $\cos(u, v) = u \cdot v / (\|u\| \cdot \|v\|)$ – косинусное сходство. Таким образом, сравниваются только объекты одинакового предсказанного класса.

Взвешенное по уверенности модели $d_{a,i}$ среднее по объектам изображения даёт несимметричную меру сходства:

$$S'(O_a, O_b) = \frac{1}{\sum_i d_{a,i}} \sum_i d_{a,i} s(o_{a,i}, O_b). \quad (1.12)$$

Итоговая симметричная мера сходства между изображениями:

$$S(O_a, O_b) = \frac{1}{2} (S'(O_a, O_b) + S'(O_b, O_a)). \quad (1.13)$$

Два шага этапа отбора по разнообразию. На первом шаге жадный алгоритм k -Center-Greedy решает задачу оптимального покрытия в пространстве, индуцированном мерой $(1 - S)$: итеративно выбирается элемент s_j , наиболее удалённый от уже выбранных элементов,

$$s_j = \arg \max_{u \in H_t \setminus S_{j-1}} \min_{a \in S_{j-1}} d(u, a), \quad (1.14)$$

где $d(u, a) = 1 - S(O_u, O_a)$. Расстояния берутся из заранее вычисленной матрицы попарных расстояний между всеми изображениями из H_t ; её построение требует $O(|H_t|^2 \cdot m^2 \cdot D)$ операций и $O(|H_t|^2)$ памяти, где m – число объектов на изображении, D – размерность признака объекта. На втором шаге в алгоритме k -means++ полученные элементы используются как начальные центры кластеров и уточняются в течение 100 итераций: каждое изображение относится к ближайшему центру, после чего новым центром кластера G_c становится изображение с наименьшей суммой расстояний до остальных изображений кластера,

$$c' = \arg \min_{u \in G_c} \sum_{w \in G_c} d(u, w). \quad (1.15)$$

Итоговые центры образуют отбираемый пакет. В исходной работе авторы используют $|H_t| = 4b$ и исследуют значения до $6b$, поэтому затраты этапа растут квадратично с бюджетом b .

Ограничения RPAL. Несмотря на высокое качество отбора, у метода есть ряд существенных ограничений.

1. **Попарное разнообразие вместо новизны.** На этапе отбора по разнообразию в RPAL максимизируется взаимное расстояние между кандидатами внутри H_t без учёта того, какие подгруппы объектов уже присутствуют в размеченном множестве L_{t-1} . В результате могут выбираться изображения, содержащие объекты, хорошо представленные в разметке.
2. **Квадратичная вычислительная сложность.** Этап отбора по разнообразию требует вычисления попарных сходств между всеми изображениями, т.е. вычисления матрицы попарных расстояний. Этот этап имеет квадратичную сложность и катастрофически замедляет отбор при работе с большими датасетами. Предварительный отбор $|H_t| \in [4b; 6b]$ снижает константу, но не устраняет квадратичной зависимости.
3. **Несовместимость с мультимасштабными пространствами признаков.** Эмбединги объектов в RPAL извлекаются в пространстве вероятностей предсказаний модели, поскольку сравнение признаков с разных уровней FPN-пирамиды затруднено различием их размерности и масштаба. Это ограничивает информативность представлений: компактные выходы детектора содержат меньше визуальной информации, чем промежуточные карты признаков.
4. **Потеря пространственной информации при извлечении признаков.** В RPAL применяется пространственное усреднение (average pooling) по всем позициям карты признаков внутри ограничивающего прямоугольника объекта. При этом карта размером $h \times w$ преобразуется в один вектор, поэтому явное расположение её пространственных элементов в представлении не сохраняется.

Ни одно из рассмотренных семейств методов не удовлетворяет одновременно всем требованиям, сформулированным в разделе 1.2. Критерии неопределённости не обращаются к структуре пула, нарушая требование 4. CoreSet и k -means++ работают на уровне изображений в целом, нарушая требования 1 и 2. Для RPAL выполняется требование 1 и частично – требование 2, однако нарушается требование 3 (пространство вероятностей вместо мультимасштабных карт)

и требование 4 (попарное разнообразие кандидатов вместо новизны относительно L_{t-1}). В главах 2 и 4 описываются предложенные для решения этих проблем методы AREA и MAREA.

1.4 Выводы по главе 1

1. Задача детекции объектов в видеоаналитике требует работы в реальном времени и характеризуется высокой внутриклассовой вариативностью условий съёмки. Проведённый анализ 36 детекторов разных поколений показал, что во всех точных и быстрых архитектурах используются мультимасштабные карты признаков.
2. Высокая стоимость разметки для задачи детекции, избыточность видеопотока, редкость информативных событий и частые циклы дообучения определяют актуальность методов активного обучения.
3. Сформулированы четыре требования к критерию AL для детекции в видеоаналитике: работа на уровне объектов; различение подгрупп объектов; использование мультимасштабных карт признаков; новизна относительно размеченного множества.
4. Методы неопределённости хорошо изучены, но не учитывают структуру пула. Методы разнообразия (CoreSet, k -means++) работают на уровне изображений. Наилучший известный метод PPAL частично удовлетворяет требованиям 1 и 2, но нарушает требования 3 и 4.
5. Ни один из рассмотренных методов не удовлетворяет всем четырём требованиям одновременно. Это обосновывает разработку методов AREA (глава 2) и MAREA (глава 4), предложенных в настоящей диссертации.

Глава 2. AREA: активное обучение на основе локальных эмбедингов объектов

В данной главе описывается метод *AREA* (Active learning with Region Embedding Alignment – активное обучение с выравниванием представлений областей), разработанный в рамках настоящей диссертации [61]. Основная идея метода состоит в следующем: вместо того чтобы оценивать полезность изображения по его глобальному описанию, в AREA каждому обнаруженному объекту сопоставляется локальный вектор признаков, извлечённый непосредственно из мультимасштабных карт детектора. Для разметки выбираются те изображения, объекты которых удалены в пространстве эмбедингов от объектов уже размеченного множества. Такой подход отвечает требованиям 1, 2, 3 и 4, сформулированным в разделе 1.2.

2.1 Локальные эмбединги объектов

В разделе 1.1 было показано, что современные одностадийные детекторы обрабатывают входное изображение в три этапа. В основном модуле применяется последовательность свёрточных блоков и формируется иерархия карт признаков с убывающим пространственным разрешением: чем глубже слой, тем грубее карта и тем богаче семантика каждого её элемента. В переходном модуле [28; 34] несколько таких карт агрегируются через нисходящий проход с боковыми соединениями, образуя пирамиду уровней, пригодную для детекции объектов разного масштаба. В выходном модуле каждый уровень пирамиды преобразуется в предсказания ограничивающих прямоугольников и классов объектов. Предсказания со всех уровней объединяются в единый выходной набор, после чего применяется подавление немаксимумов (Non-Maximum Suppression, NMS) для удаления дублирующихся детекций.

Пусть переходный модуль формирует L уровней пирамиды признаков. Карта l -го уровня обозначается $\mathbf{F}^{(l)} \in \mathbb{R}^{H_l \times W_l \times C_l}$, где $H_l \times W_l$ – пространственные размеры, C_l – число каналов, $l = 1, \dots, L$ (уровень 1 соответствует наибольшему разрешению). Шаг сетки s_l задаёт расстояние между соседними позициями кар-

ты l -го уровня в координатах входного изображения. В архитектуре YOLOv8 [34] используются три уровня с шагами сетки $s_1 = 8$, $s_2 = 16$, $s_3 = 32$; таким образом, шаг верхнего уровня в четыре раза больше шага нижнего. Площадь соответствующей ячейки сетки больше в 16 раз. Для построения критерия активного обучения необходимо каждому обнаруженному объекту сопоставить векторное представление, извлечённое непосредственно из карт признаков детектора. Пусть объект задан ограничивающим прямоугольником $\mathbf{b} = (x_1, y_1, x_2, y_2)$ в координатах входного изображения. При проецировании на l -й уровень пирамиды координаты рамки принимают вещественные значения:

$$\tilde{x}_1^{(l)} = \frac{x_1}{s_l}, \quad \tilde{y}_1^{(l)} = \frac{y_1}{s_l}, \quad \tilde{x}_2^{(l)} = \frac{x_2}{s_l}, \quad \tilde{y}_2^{(l)} = \frac{y_2}{s_l}. \quad (2.1)$$

Ширина и высота проецируемой рамки, как правило, не кратны шагу дискретизации, поэтому вещественные координаты (2.1) не совпадают с узлами целочисленной сетки карты (рисунок 2.1).

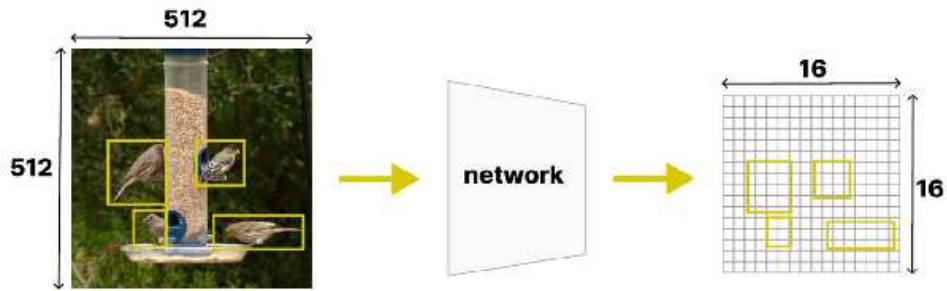


Рисунок 2.1 – Несовпадение границ ограничивающего прямоугольника с сеткой карты признаков.

Наивный подход – заменить вещественные границы рамки (2.1) ближайшими узлами целочисленной сетки карты признаков и взять соответствующий срез тензора $\mathbf{F}^{(l)}$ – приводит к систематическим искажениям, известным как *ошибки округления координат*. Ячейки карты, лишь частично перекрывающиеся с рамкой, либо целиком включаются в срез (захватывая активации фона или соседних объектов), либо полностью исключаются из него (теряя часть признаков самого объекта). Пример таких ошибок показан на рисунке 2.2. Жёлтым контуром на рисунке обозначена истинная область объекта, красным – часть объекта, потерянная при округлении границы внутрь рамки, зелёным – лишний фон, попавший в срез при округлении границы наружу. На верхних уровнях пирамиды, где шаг сетки достигает 32 пикселей входного изображения, ошибка округления вносит стоит еще дороже, т.к. вносит больший шум. Кроме того, срезы рамок разного размера

имеют различные пространственные размеры, что делает невозможным их прямое сравнение или конкатенацию без дополнительного масштабирования; стандартный пулинг при таком масштабировании вновь вводит двойное округление: сначала при проецировании рамки, затем при разбиении на ячейки пулинга.

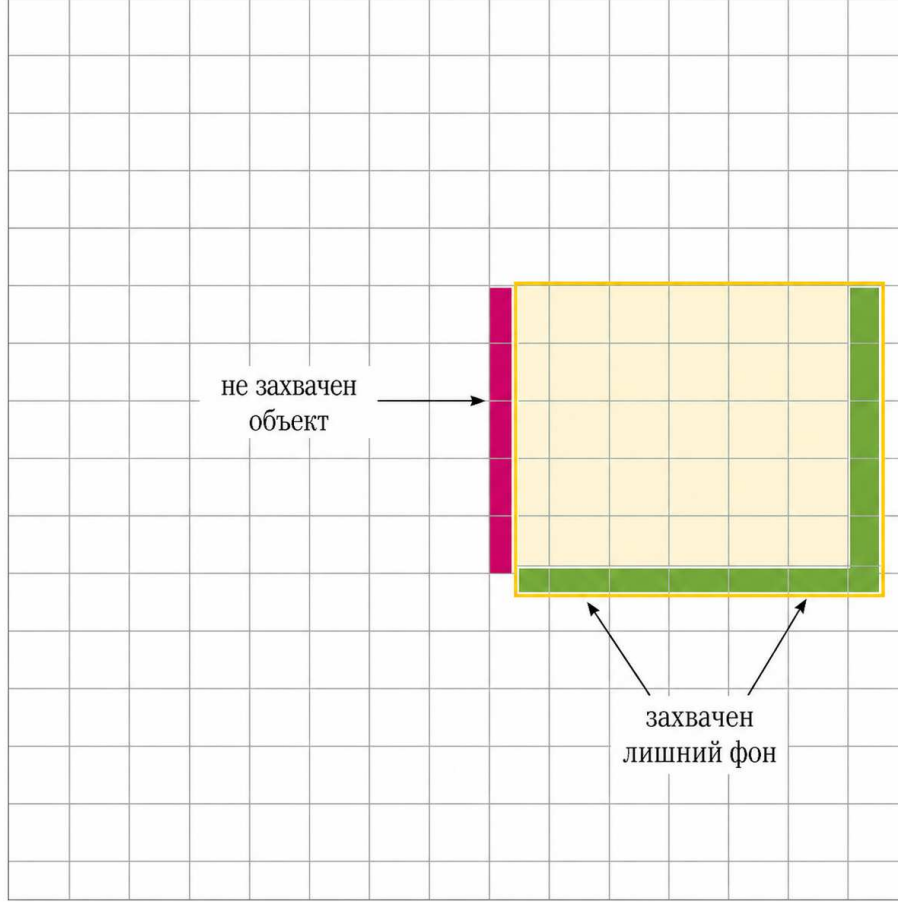


Рисунок 2.2 – Ошибки округления при проецировании ограничивающего прямоугольника на карту признаков.

Определение. Операция RoI Align принимает карту признаков $\mathbf{F}^{(l)}$, вещественные координаты рамки $(\tilde{x}_1^{(l)}, \tilde{y}_1^{(l)}, \tilde{x}_2^{(l)}, \tilde{y}_2^{(l)})$ и целевое разрешение $r_h \times r_w$, после чего возвращает тензор $\mathbf{E}^{(l)} \in \mathbb{R}^{r_h \times r_w \times C_l}$, не производя ни одного округления координат рамки.

Рамка делится на $r_h \times r_w$ равных ячеек; внутри каждой ячейки размещаются $k \times k$ равноудалённых опорных точек ($k = 2$ по умолчанию [43]). Значение в каждой опорной точке с непрерывными координатами (x, y) вычисляется билинейной интерполяцией по четырём ближайшим узлам сетки карты признаков:

$$v(x, y) = \sum_{i \in \{0,1\}} \sum_{j \in \{0,1\}} \mathbf{F}^{(l)}[\lfloor y \rfloor + i, \lfloor x \rfloor + j, :] \cdot (1 - |x - \lfloor x \rfloor - j|) \cdot (1 - |y - \lfloor y \rfloor - i|). \quad (2.2)$$

Итоговое значение ячейки получается усреднением по k^2 опорным точкам. Таким образом, RoI Align реализует точное пространственное выравнивание в единой операции, не вводя округления ни на одном из шагов (рисунок 2.3).

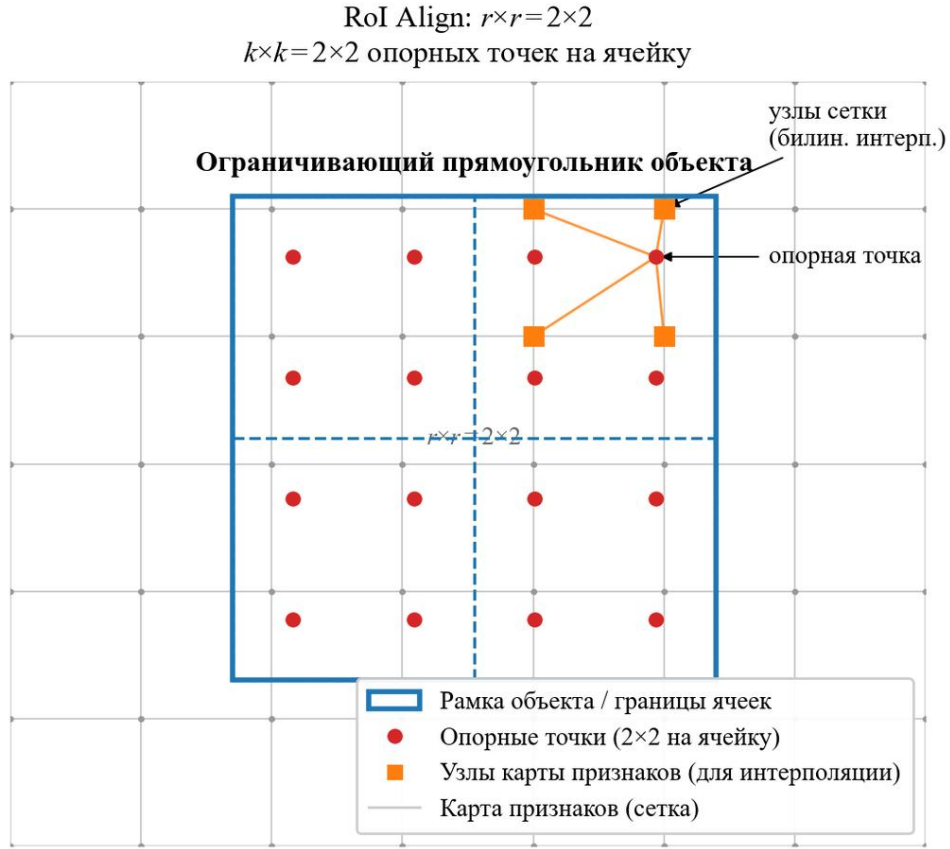


Рисунок 2.3 – Схема операции RoI Align при целевом разрешении 2×2 [43].

В отличие от Region of Interest Pooling (RoI Pooling) [32], применявшегося в двухстадийных детекторах и выполнявшего двойное округление, сначала проецируя рамку на целочисленную сетку карты, затем округляя границы ячеек пулинга, RoI Align сохраняет непрерывные координаты на обоих этапах. В работе Mask R-CNN [43] оба подхода сравнивались напрямую: замена RoI Pooling на RoI Align при прочих равных условиях увеличила точность предсказания масок на 3 п.п. по мере качества Average Precision (AP) и 5 п.п. по AP_{75} . Авторы объясняют этот прирост именно устранением накопленной ошибки округления, особенно значимой для точных задач локализации.

В методе AREA RoI Align применяется к каждому из L уровней пирамиды независимо. Для объекта с рамкой $\mathbf{b}$ операция на уровне l возвращает тензор $\mathbf{E}^{(l)} \in \mathbb{R}^{r_h \times r_w \times C_l}$, который разворачивается в вектор $\mathbf{e}^{(l)} \in \mathbb{R}^{r_h \cdot r_w \cdot C_l}$. Векторы со всех уровней конкатенируются вдоль оси каналов:

$$\mathbf{e} = [\mathbf{e}^{(1)} \parallel \mathbf{e}^{(2)} \parallel \dots \parallel \mathbf{e}^{(L)}] \in \mathbb{R}^{r_h \cdot r_w \cdot \sum_{l=1}^L C_l}. \quad (2.3)$$

Нижние уровни пирамиды (высокое разрешение, мелкие детали) вносят в e информацию о форме и текстуре объекта, верхние уровни (низкое разрешение, высокая семантика) – контекстно-зависимое представление о его смысловом содержании. За счёт фиксированного целевого разрешения $r_h \times r_w$ вектор e имеет одинаковую размерность для объектов любого исходного размера и любого уровня пирамиды. Схема извлечения такого эмбединга показана на рисунке 2.4.

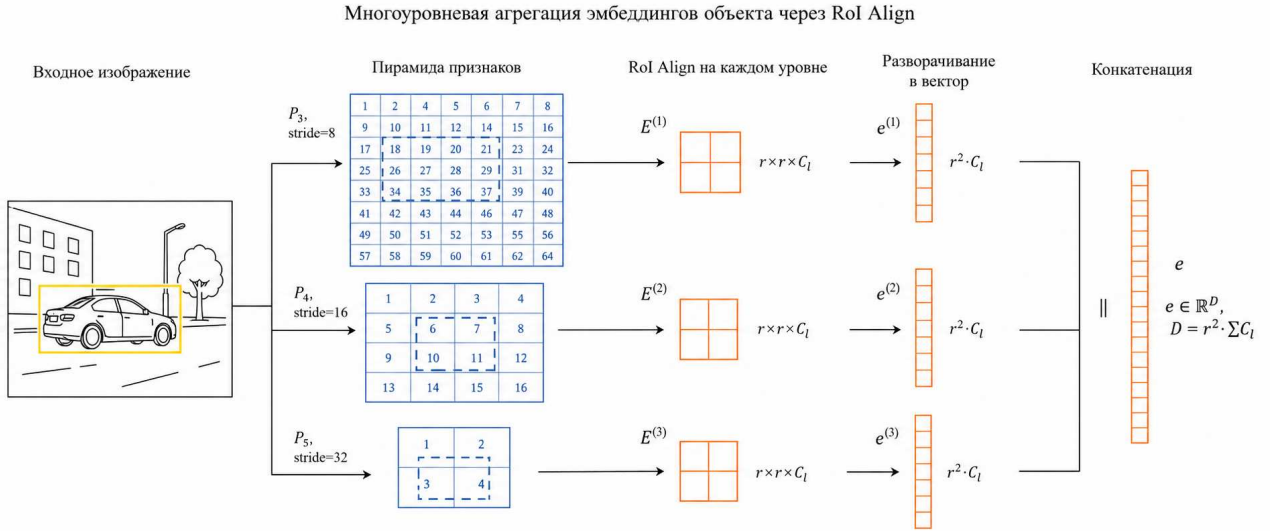


Рисунок 2.4 – Схема извлечения многоуровневого локального эмбединга объекта.

Таким образом, за счёт применения RoI Align одновременно решаются две проблемы: потеря информации из-за округления координат при наложении рамки на карту активаций и приведение векторов с разных карт активаций к одной и той же размерности. Каждому предсказанному детектором объекту ставится в соответствие единый вектор e – *локальный эмбединг объекта*. Совокупность таких векторов для всех аннотированных изображений образует пространство представлений объектов, в котором далее формулируется критерий активного обучения AREA.

2.2 Предобработка эмбедингов и критерий отбора AREA

Локальный эмбединг объекта e , построенный в разделе 2.1, формально имеет размерность $r_h \cdot r_w \cdot \sum_{l=1}^L C_l$, где C_l – число каналов l -го уровня пирамиды признаков. При применении к промежуточным картам активации YOLOv8s

($C_1 = 128$, $C_2 = 256$, $C_3 = 512$) при разрешении $r_h = 12$, $r_w = 6$ размерность составила бы $12 \cdot 6 \cdot 896 = 64\,512$, что слишком много для эффективного поиска ближайших соседей.

Поэтому в методе AREA [61] используются не промежуточные карты активации перед выходным модулем, а *выходы* выходного модуля. Каждый выход содержит вектор из $(C + 4)$ значений, где C – число классов набора данных, а 4 – координаты ограничивающего прямоугольника. При RoI Align с разрешением $r_h = 12$, $r_w = 6$ и $L = 3$ уровнями пирамиды для набора данных COCO ($C = 80$) итоговая размерность составляет:

$$D = r_h \cdot r_w \cdot (C + 4) \cdot L = 12 \cdot 6 \cdot 84 \cdot 3 = 18\,144. \quad (2.4)$$

Это существенно меньше, чем при использовании промежуточных карт, однако выходное пространство детектора содержит меньше визуальной информации, чем скрытые представления промежуточных слоёв.

Работа непосредственно в пространстве размерности $D = 18\,144$ по-прежнему сопряжена с двумя практическими трудностями.

Первая трудность состоит в вычислительных затратах на поиск ближайших соседей. При числе аннотированных объектов в размеченном пуле $|E_{L_{t-1}}|$ поиск k ближайших соседей для каждого кандидата требует $O(|E_{L_{t-1}}| \cdot D)$ операций, где $D = 18\,144$; с ростом аннотированного пула эта стоимость становится существенной. Вторая трудность состоит в явлении *проклятия размерности*: при больших D косинусные расстояния между случайными векторами концентрируются вблизи одного значения, что затрудняет различение объектов по оценке новизны. Для устранения обеих трудностей в методе AREA применяется предварительное снижение размерности методом главных компонент (Principal Component Analysis, PCA) [36; 61]. Перед каждым шагом активного обучения строится проекция $\Phi: \mathbb{R}^D \rightarrow \mathbb{R}^d$, где $d \ll D$ – заданное число главных компонент. Проекция обучается на объединённом множестве эмбедингов объектов из размеченного пула L_{t-1} и множества кандидатов U_{t-1} :

$$\Phi = \text{PCA}(\{e(o) \mid o \in \mathcal{O}(L_{t-1}) \cup \mathcal{O}(U_{t-1})\}, d), \quad (2.5)$$

где $\mathcal{O}(S)$ обозначает совокупность всех объектов, обнаруженных детектором f_θ на изображениях из множества S . Использование объединённого множества гарантирует, что проекция учитывает как уже аннотированные образцы, так и структуру пространства кандидатов.

Сжатый эмбединг объекта обозначается

$$\hat{e} = \Phi(e) \in \mathbb{R}^d. \quad (2.6)$$

Размерность d является гиперпараметром метода; в методе AREA используется значение 512, которое обеспечивает достаточную информативность при приемлемых вычислительных затратах.

На основе сжатых эмбедингов формулируется мера новизны для каждого объекта-кандидата. Множество сжатых эмбедингов аннотированных объектов обозначим $\Phi(E_{L_{t-1}}) = \{\Phi(e(o)) \mid o \in \mathcal{O}(L_{t-1})\}$. Для объекта o с эмбедингом $\hat{e} = \Phi(e(o))$ мера расстояния определяется как минимальное косинусное расстояние до ближайшего объекта из размеченного пула:

$$a(o) = \min_{z \in \Phi(E_{L_{t-1}})} d_{\cos}(\hat{e}, z), \quad (2.7)$$

где косинусное расстояние задаётся формулой

$$d_{\cos}(u, v) = 1 - \frac{u \cdot v}{\|u\| \cdot \|v\|}. \quad (2.8)$$

Выбор косинусного расстояния обусловлен его свойствами применительно к нейросетевым представлениям. Косинусное расстояние измеряет угол между векторами, не зависящий от их длины: два вектора одинакового направления получают нулевое расстояние вне зависимости от того, в каких масштабах они представлены [42]. Это важно для детекционных эмбедингов: абсолютные значения активаций определяются интенсивностью признаков и могут существенно различаться между объектами разных размеров, тогда как направление вектора несёт информацию о визуальном содержимом. Кроме того, при высокой размерности d евклидовы расстояния склонны к концентрации (все точки становятся приблизительно равноудалёнными), тогда как косинусное расстояние сохраняет способность различать объекты [12].

Поскольку одно изображение может содержать несколько объектов, необходима агрегация оценок для объектов на уровне изображения. Одно изображение x содержит несколько обнаруженных объектов $\mathcal{O}(x) = \{o_1, \dots, o_m\}$. Ценность изображения как кандидата для разметки определяется максимальной удалённостью в пространстве эмбедингов среди всех его объектов:

$$a_{\text{AREA}}(x, f_{\theta}) = \max_{o \in \mathcal{O}(x)} \min_{z \in \Phi(E_{L_{t-1}})} d_{\cos}(\Phi(e(o)), z). \quad (2.9)$$

Агрегация через максимум отражает следующую логику: кадр является ценным для разметки, если он содержит хотя бы один объект, визуально непохожий ни на один из уже аннотированных; наличие нескольких типичных объектов не снижает ценность такого кадра, но и не повышает её сверх меры наиболее удалённого в пространстве эмбедингов. Тем самым критерий сосредоточен на покрытии пространства визуальных представлений объектов, а не на агрегированных статистиках изображения в целом.

Перейдём к формальному определению пакетного критерия отбора. На каждом шаге активного обучения требуется отобрать пакет из b изображений для разметки. В критерии AREA эта задача формулируется как выбор подмножества S с наибольшей суммарной удалённостью от размеченного пула:

$$A_{\text{AREA}}(L_{t-1}, U_{t-1}, f_{\theta}) = \arg \max_{S \subset U_{t-1}, |S|=b} \sum_{x \in S} a_{\text{AREA}}(x, f_{\theta}). \quad (2.10)$$

Поскольку оценки $a_{\text{AREA}}(x, f_{\theta})$ для разных кандидатов вычисляются независимо друг от друга, задача (2.10) решается жадным образом – выбираются b изображений с наибольшими индивидуальными оценками. Весь отбор выполняется за один проход по множеству кандидатов, что обеспечивает линейную вычислительную сложность относительно $|U_{t-1}|$.

Следует, однако, отметить ограничение жадной стратегии. Поскольку изображения оцениваются независимо и отбираются по убыванию индивидуальной оценки, в одном пакете могут оказаться несколько кадров, содержащих похожие объекты из одной визуальной подгруппы. Отобранные изображения могут существенно отличаться от размеченного множества L_{t-1} , но быть похожими друг на друга. Поэтому независимый отбор не гарантирует разнообразия внутри пакета. Устранение этой избыточности потребовало бы учёта попарного сходства внутри пакета, что существенно увеличило бы вычислительную стоимость отбора. В методе AREA данный компромисс сознательно принят в пользу скорости.

Для сравнения в экспериментах наряду с критерием расстояния рассматривается *критерий по плотности* [61]. Вместо расстояния до ближайшего соседа он оценивает локальную плотность аннотированных объектов в окрестности кандидата:

$$\rho(o) = \sum_{i=1}^k \frac{1}{d_i^2(o)}, \quad (2.11)$$

где $d_i(o)$ – косинусное расстояние от эмбединга объекта o до i -го ближайшего аннотированного объекта, $k = 5$ по умолчанию [61]. Большое значение $\rho(o)$ соответствует плотному окружению из известных объектов, малое – разреженной, слабо покрытой области пространства. Оценка изображения по плотности строится аналогично критерию по расстоянию через минимум по объектам:

$$a_{\text{dens}}(x, f_{\theta}) = \min_{o \in \mathcal{O}(x)} \rho(o). \quad (2.12)$$

Чем меньше значение a_{dens} , тем в менее покрытой области находится наиболее редкий объект изображения; при отборе изображения сортируются по возрастанию a_{dens} .

Критерий плотности учитывает k соседей вместо одного, что делает его несколько более устойчивым к случайным выбросам вблизи границ кластеров. Тем не менее, как и критерий по расстоянию, он склонен выбирать изолированные выбросы: точки, у которых все k ближайших соседей далеко, получают минимальный ρ . Это является общим ограничением обоих критериев: одиночные примеры на периферии пространства эмбедингов получают приоритет вне зависимости от того, насколько они характерны для целевого домена. На рисунке 2.5 кандидаты P и Q одинаково удалены от разметки, поэтому критерий по расстоянию не различает их. Критерий по плотности выбирает P, поскольку вблизи него меньше аннотированных соседей ($\rho_P < \rho_Q$).

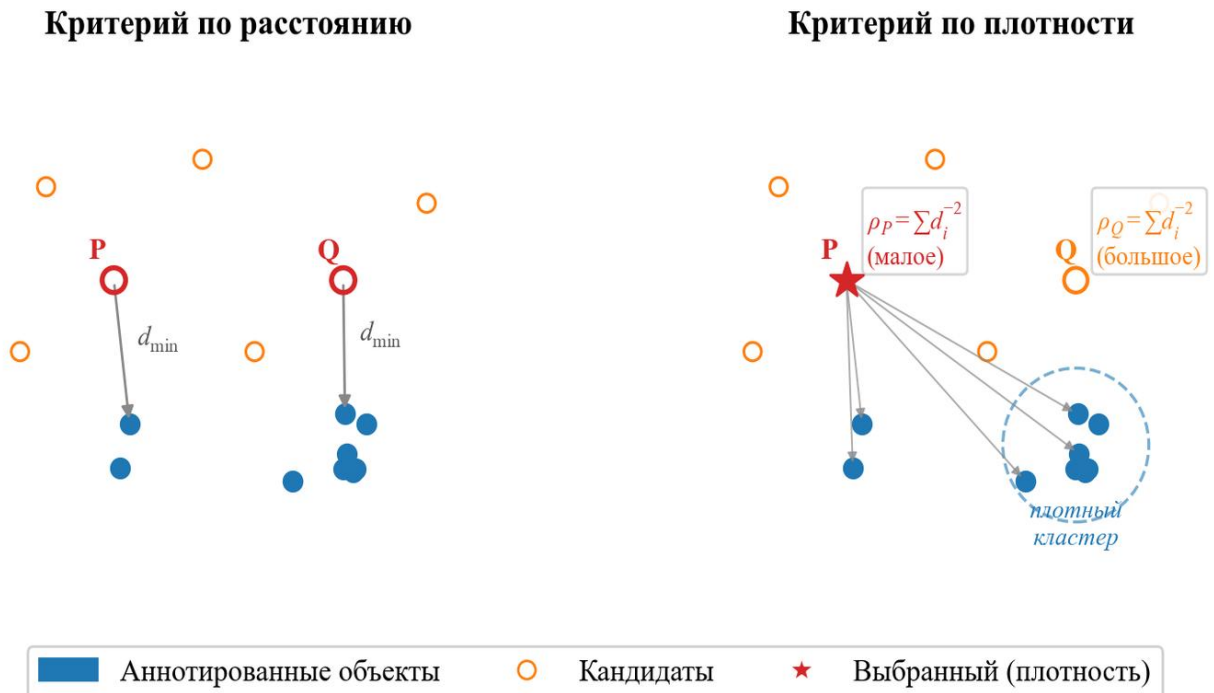


Рисунок 2.5 – Сравнение критериев по расстоянию и по плотности (2.12).

Таким образом, критерий AREA напрямую отвечает требованиям, сформулированные в разделе 1.2. Требование 1 (работа на уровне объектов) выполнено: оценка (2.9) строится по эмбедингам объектов, а не по глобальному эмбедингу изображения. Требование 2 (различение подгрупп внутри класса) обеспечивается использованием пространства признаков детектора: объекты одного класса, но разных визуальных условий, занимают различные области пространства $\mathbb{R}^d$, и их косинусные расстояния до аннотированных образцов различаются. Требование 3 (мультимасштабность) реализовано через конкатенацию эмбедингов со всех уровней пирамиды признаков в формуле (2.3). Требование 4 (новизна относительно размеченного пула) удовлетворяется явно через $\min_{z \in \Phi(E_{L_{t-1}})}$ в (2.7): мера новизны каждого кандидата вычисляется не как расстояние до других кандидатов, а как расстояние до уже аннотированных объектов.

2.3 Схема экспериментов

Протокол экспериментов, описанный в данном разделе, является общим для всех методов, рассматриваемых в диссертации. Все сравниваемые методы активного обучения тестировались в режиме псевдоразметки, т.е. при оценке объектов неразмеченного пула использовались предсказания текущего детектора.

Наборы данных. В экспериментах используются четыре набора данных (подробное описание в разделе 2.4): Pascal VOC [68], COCO 2017 [48], CrowdHuman [20] и ind_safety. Первые два обеспечивают сравнимость с опубликованными методами; CrowdHuman проверяет работу на плотных одноклассовых сценах; ind_safety соответствует целевому прикладному домену – промышленной видеоаналитике.

Модель детектора. Во всех экспериментах кроме эксперимента по сравнению критериев отбора по плотности и по расстоянию используется YOLOv8s [34] – малая конфигурация архитектуры YOLOv8 с числом параметров около 11 млн. Модель инициализируется весами, предобученными на наборе Open Images v7. Обучение выполняется с нуля на каждой новой порции разметки: 65 эпох, размер входного изображения 640×640 пикселей, размер пакета 48, стандартные гиперпараметры оптимизатора Ultralytics YOLOv8. Вычисления проводились на видеокартах NVIDIA RTX 3090 и NVIDIA H100.

Протокол цепочки активного обучения. Общий объём размеченных данных делится на доли $\alpha \in \{0.2, 0.3, 0.4, 0.5, 0.6, 0.7\}$ от обучающей выборки. Наборы строятся *вложенно*: набор доли α содержит в себе весь набор доли $\alpha - 0.1$. Стартовой точкой для всех методов служит модель, обученная на случайно выбранной доле 0.2. На каждом шаге AL текущая модель запускается на неразмеченном пуле, критерий отбора ранжирует кандидатов, и следующая порция 0.1 изображений добавляется к обучающему множеству. Для оценки модели используется мера качества mAP50–95 на валидационном множестве.

Базовые методы. Метод AREA сравнивается со случайным отбором (RANDOM) – равномерной выборкой из пула без учёта состояния модели. Превышение качества случайного отбора при том же объёме разметки является минимальным критерием полезности метода активного обучения. Кроме того, проведено сравнение с SOTA-решением – PPAL, а именно второй его стадией, осуществляющей отбор по разнообразию. В разделе 2.5 дополнительно сравниваются AREA с критерием по расстоянию и AREA с критерием по плотности.

Ход эксперимента на каждом шаге цепочки активного обучения. На каждом шаге цепочки активного обучения выполняется следующая последовательность действий [62].

1. Детектор YOLOv8s обучается с нуля на текущем размеченном множестве L_{t-1} (65 эпох, 640×640 , размер пакета – 48, стандартные гиперпараметры).
2. Обученная модель применяется к неразмеченному пулу U_{t-1} ; для каждого найденного объекта извлекаются эмбединги и вычисляется оценка a_{AREA} по формуле (2.9).
3. Отбираются b изображений с наибольшими оценками (2.10). Выбранные изображения передаются оракулу и добавляются в L_t .
4. Шаги 1–3 повторяются для долей $0.2 \rightarrow 0.3 \rightarrow \dots \rightarrow 0.7$.

Начальная точка – модель, обученная на случайно выбранной доле 0.2. Все переходы с доли $\alpha - 0.1$ на долю α используют веса, сохранённые после обучения на предыдущей доле данного метода. Для оценки воспроизводимости используются 4 независимые инициализации генератора псевдослучайных чисел; в таблицах ниже приводится среднее и 95 % доверительный интервал.

2.4 Наборы данных

Pascal VOC [68]. Набор общего назначения с изображениями 20 категорий объектов: транспортные средства, животные, мебель, бытовые предметы, люди и др. Обучающая выборка содержит 16 551 изображение, валидационная – 4 952. Среднее число объектов на изображении невелико (2–3), что делает набор удобным для сравнения методов AL при умеренной сложности. Набор хорошо изучен и используется как основной контрольный ориентир. Примеры изображений Pascal VOC приведены на рисунке 2.6.

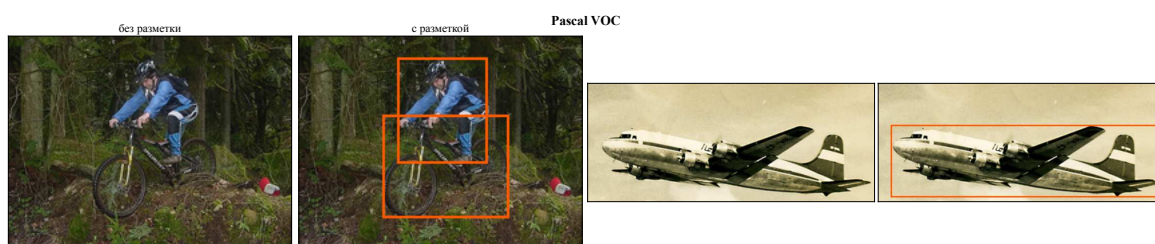


Рисунок 2.6 – Примеры изображений Pascal VOC [68]: без разметки (слева) и с ограничивающими прямоугольниками (справа).

COCO 2017 [48]. Крупный набор общего назначения с 80 категориями объектов и сценами высокой сложности. Обучающая выборка насчитывает 118 287 изображений, валидационная – 5 000. На изображении в среднем около 7 объектов; отдельные сцены содержат значительно больше экземпляров разных классов. Большой объём пула кандидатов позволяет оценить масштабируемость метода отбора. Примеры изображений COCO 2017 приведены на рисунке 2.7.

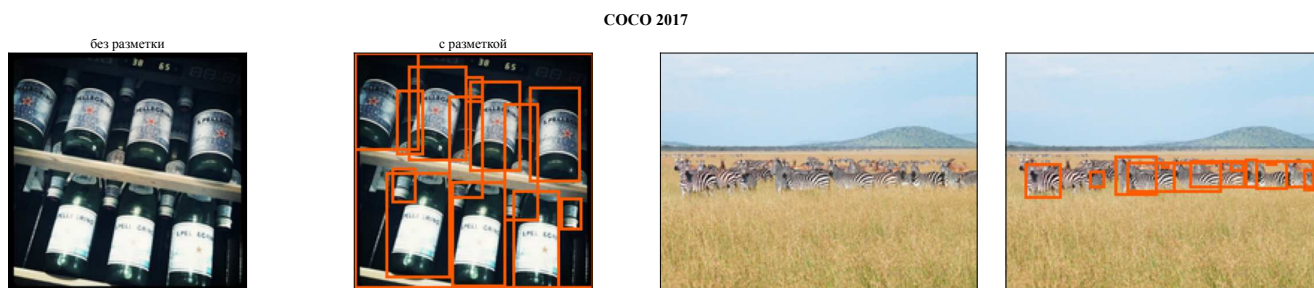


Рисунок 2.7 – Примеры изображений COCO 2017 [48].

CrowdHuman [20]. Набор для детекции людей в плотных сценах с сильными взаимными перекрытиями. Содержит 15 000 обучающих и 4 370 валидационных изображений с одним классом *человек*. Особенность набора – средняя плотность 22 человека на изображение, при этом значительная часть экземпляров

частично перекрыта. Одноклассовая задача позволяет изолировать способность метода AL различать внутриклассовые визуальные подгруппы без влияния классового баланса. Примеры изображений CrowdHuman приведены на рисунке 2.8.

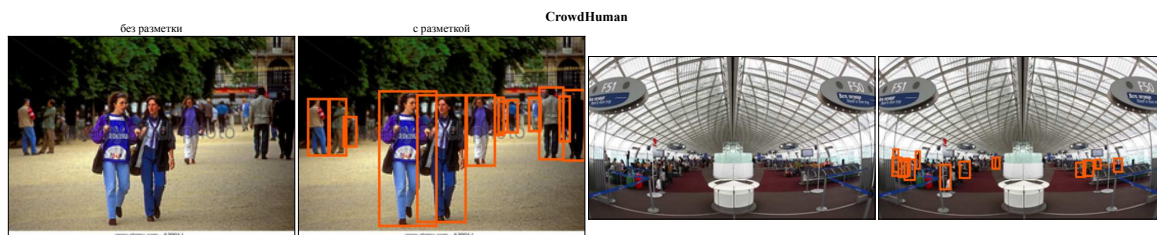


Рисунок 2.8 – Примеры изображений CrowdHuman [20]: плотные сцены с перекрывающимися людьми.

ind_safety. Промышленный набор для задачи контроля соблюдения требований безопасности при проведении работ на нефтегазовом оборудовании. Содержит три класса объектов, связанных со страховочным снаряжением. Обучающая выборка содержит 69 925 изображений, валидационная – 7 029. Съёмка ведётся стационарными промышленными камерами, что определяет специфичные ракурсы и условия освещения. Набор характеризуется значительной внутриклассовой вариативностью: один и тот же тип объекта выглядит принципиально по-разному при дневном свете, в условиях задымления или ночного освещения. Кроме того, классы визуально схожи друг с другом, тем самым их классификация дополнительно усложнена. Именно на этом наборе проверяется способность метода работать в промышленных условиях.

2.5 Результаты AREA

Сравнение критериев по расстоянию и по плотности.

В разделе 2.2 сформулированы два варианта критерия AREA. Критерий по расстоянию (2.7) оценивает каждый объект-кандидат через минимальное косинусное расстояние до ближайшего аннотированного объекта. Критерий по плотности (2.11) суммирует обратные квадраты расстояний до $k = 5$ ближайших соседей и тем самым учитывает более широкую окрестность точки. Сравнительные эксперименты на Pascal VOC и COCO 2017 [61] показали, что оба критерия дают схожие результаты: преимущество ни одного из них не является систематическим

и не выходит за пределы погрешности, обусловленной случайной инициализацией. На COCO 2017 критерий по плотности незначительно превосходит критерий по расстоянию при больших долях разметки ($\alpha \geq 0.5$); на Pascal VOC критерий по расстоянию в целом несколько предпочтительнее при малых долях, однако разница вновь не является статистически значимой. Поскольку критерий по расстоянию вычислительно дешевле – требует поиска одного ближайшего соседа против пяти у критерия по плотности – и при этом показывает сопоставимое качество, в основных экспериментах ниже используется критерий по расстоянию. Сравнение двух вариантов критерия приведено на рисунке 2.9.

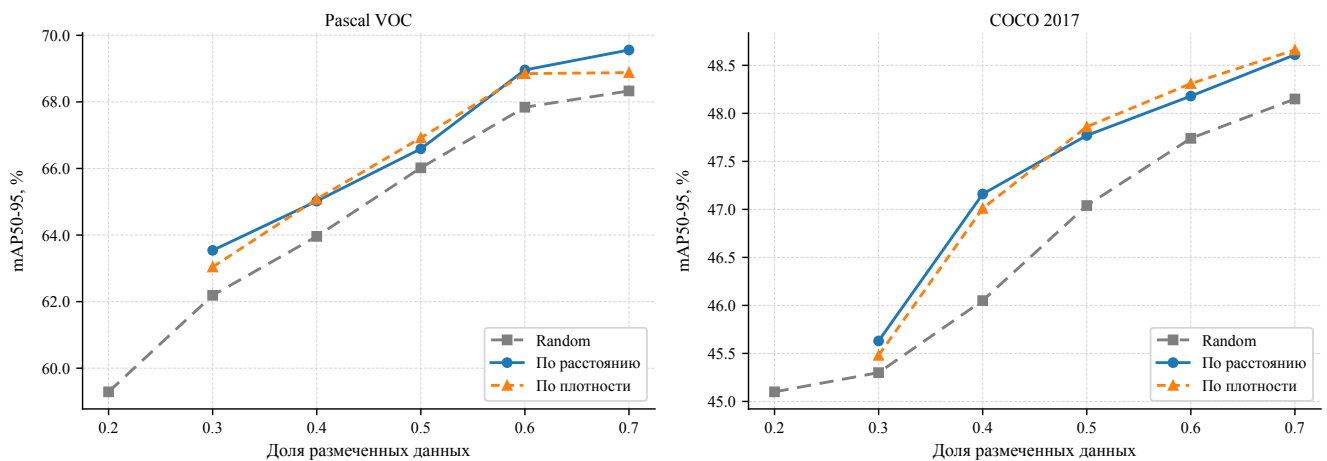


Рисунок 2.9 – Сравнение критериев по расстоянию и по плотности на Pascal VOC и COCO 2017. YOLOv8m, инициализация весами COCO [61].

Вычислительная сложность AREA.

В методе AREA реализуется одностадийный независимый отбор: оценка каждого кандидата $u \in U_{t-1}$ вычисляется относительно неизменного множества L_{t-1} за один проход [62]. Это принципиально отличает его от PPAL, где на шаге разнообразия требуется k итераций жадного выбора с пересчётом расстояний на каждом шаге ($O(k \cdot |U_{t-1}| \cdot D)$). В AREA весь отбор сводится к поиску ближайших соседей в статическом индексе и выбору b изображений с наибольшими оценками за линейное время $O(|U_{t-1}|)$.

Поскольку множество L_{t-1} не меняется в ходе раунда, для поиска соседей строится единственный статический индекс ANN (Approximate Nearest Neighbor) (в настоящей работе – Annoy [63]). Независимость оценок кандидатов позволяет распараллелить вычисление расстояний по пакетам, что дополнительно ускоряет отбор. В отличие от PPAL, который вынужден применять точный попарный поиск из-за динамического пополнения множества центров, AREA полностью совместим с приближёнными методами поиска.

Основные результаты AREA на четырёх наборах данных приведены по экспериментам из [62].

Результаты на Pascal VOC.

При обучении на полном обучающем множестве VOC достигается 66.14 % mAP50–95. График цепочки активного обучения приведён на рисунке 2.10, численные значения – в таблице 2.

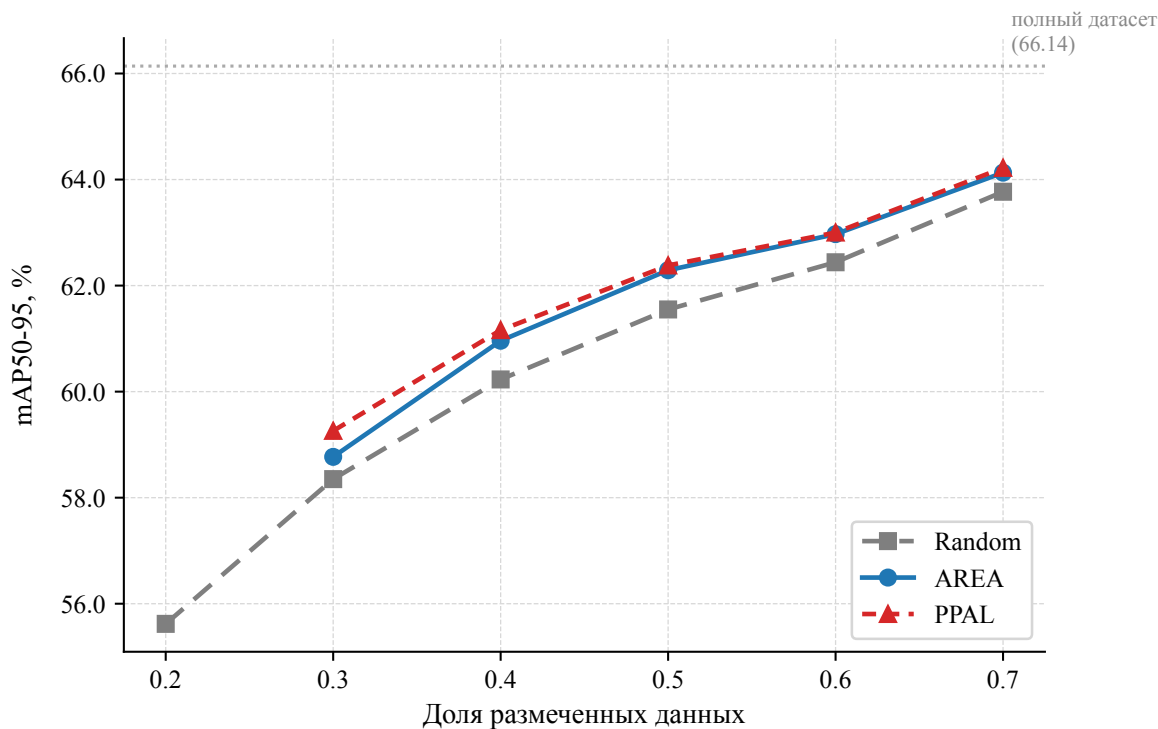


Рисунок 2.10 – Цепочки активного обучения на Pascal VOC [62]. Пунктир – качество на полном наборе данных (66.14 %).

Таблица 2 – mAP50–95, %, на Pascal VOC (среднее $\pm$ 95 % CI (Confidence Interval), 4 инициализации) [62]

Доля	RANDOM	AREA	PPAL
0.2	55.62 $\pm$ 0.80	—	—
0.3	58.35 $\pm$ 0.32	58.77 $\pm$ 0.78	59.26 $\pm$ 0.62
0.4	60.23 $\pm$ 0.33	60.96 $\pm$ 0.69	61.16 $\pm$ 0.38
0.5	61.55 $\pm$ 0.26	62.29 $\pm$ 0.32	62.38 $\pm$ 0.42
0.6	62.44 $\pm$ 0.25	62.97 $\pm$ 0.28	63.00 $\pm$ 0.28
0.7	63.77 $\pm$ 0.40	64.13 $\pm$ 0.28	64.22 $\pm$ 0.29

AREA устойчиво превосходит RANDOM на Pascal VOC начиная с доли 0.3. При доле 0.3 для AREA значение mAP50–95 выше RANDOM на +0.42 п.п., а PPAL – на +0.91 п.п. По мере роста разметки разрыв сохраняется: при доле 0.7

AREA впереди RANDOM на $+0.36$ п.п. mAP50–95. Для PPAL и AREA получены сопоставимые результаты, причём при малых долях (0.3–0.5) наблюдается небольшое преимущество PPAL, а при больших (0.6–0.7) разница нивелируется. Это говорит о том, что на общем публичном наборе данных оба метода эффективно покрывают пространство представлений объектов.

Результаты на CrowdHuman.

При обучении на полном обучающем множестве CrowdHuman достигается 56.58 % mAP50–95. График цепочки активного обучения приведён на рисунке 2.11, численные значения – в таблице 3.

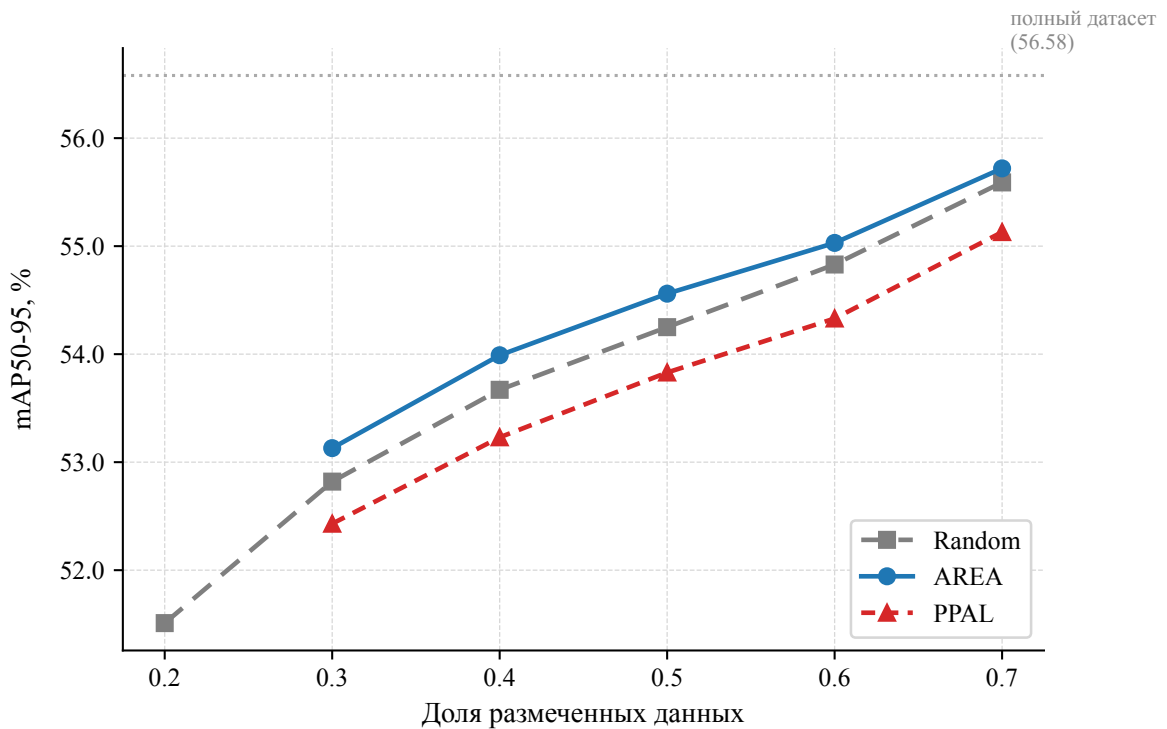


Рисунок 2.11 – Цепочки активного обучения на CrowdHuman [62]. Пунктир – качество на полном наборе данных (56.58 %).

Таблица 3 – mAP50–95, %, на CrowdHuman (среднее $\pm$ 95 % CI, 4 инициализации) [62]

Доля	RANDOM	AREA	PPAL
0.2	51.51 $\pm$ 0.11	–	–
0.3	52.82 $\pm$ 0.05	53.13 $\pm$ 0.08	52.43 $\pm$ 0.09
0.4	53.67 $\pm$ 0.06	53.99 $\pm$ 0.13	53.23 $\pm$ 0.24
0.5	54.25 $\pm$ 0.08	54.56 $\pm$ 0.05	53.83 $\pm$ 0.10
0.6	54.83 $\pm$ 0.05	55.03 $\pm$ 0.07	54.33 $\pm$ 0.15
0.7	55.59 $\pm$ 0.03	55.72 $\pm$ 0.05	55.13 $\pm$ 0.11

На CrowdHuman для AREA получены результаты выше обоих базовых методов. При доле 0.3 значение mAP50–95 для AREA выше RANDOM на +0.31 п.п. и PPAL на +0.70 п.п. Наибольшее преимущество AREA перед PPAL наблюдается при доле 0.4 и составляет 0.76 п.п. mAP50–95; при доле 0.7 разница равна 0.59 п.п. Возможно объяснение связано с тем, что в методе PPAL максимальные расстояния между объектами усредняются, и сигналы о полезности отдельных объектов теряются после усреднения. Датасет CrowdHuman характеризуется высоким количеством объектов на изображении, из-за чего вышеописанный эффект усиливается, что приводит к тому, что при отборе с помощью PPAL отбирается меньше полезных изображений.

Результаты на COCO 2017.

При обучении на полном обучающем множестве COCO 2017 достигается 43.61 % mAP50–95. График цепочки активного обучения приведён на рисунке 2.12, численные значения – в таблице 4.

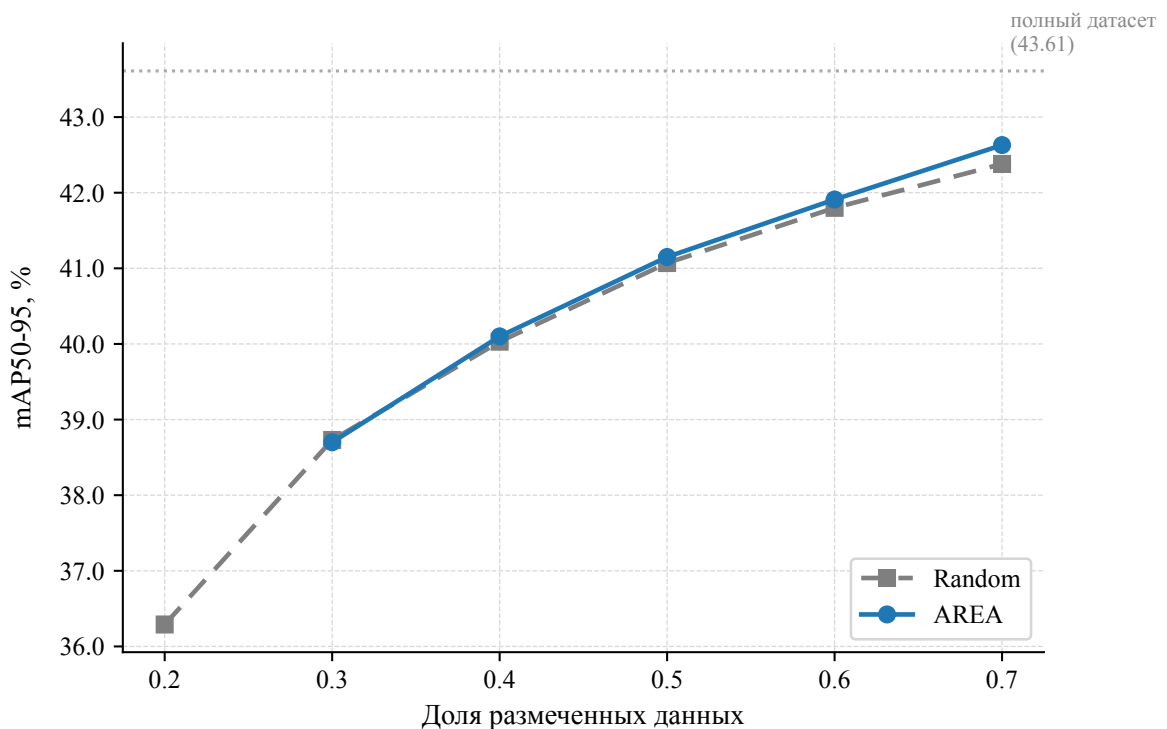


Рисунок 2.12 – Цепочки активного обучения на COCO 2017 [62]. Пунктир – качество на полном наборе данных (43.61 %).

На COCO 2017 различия между AREA и RANDOM невелики: при доле 0.3 разница составляет лишь -0.03 п.п. (AREA незначительно уступает). Начиная с доли 0.4 AREA стабильно опережает RANDOM ($+0.07$ – $+0.25$ п.п.), достигая максимального преимущества $+0.25$ п.п. при доле 0.7. Возможное объяснение связано с разнообразием 80 классов и сцен: пространство эмбедингов очень разно-

Таблица 4 – mAP50–95, %, на COCO 2017 (среднее $\pm$ 95 % CI, 4 инициализации) [62]

Доля	RANDOM	AREA
0.2	36.29 ± 0.23	–
0.3	38.73 ± 0.13	38.70 ± 0.38
0.4	40.03 ± 0.28	40.10 ± 0.37
0.5	41.07 ± 0.10	41.15 ± 0.14
0.6	41.80 ± 0.27	41.91 ± 0.09
0.7	42.38 ± 0.11	42.63 ± 0.07

родно, и локальный критерий новизны работает менее агрессивно, чем на более специализированных наборах. Тем не менее устойчивое преимущество при больших долях подтверждает обоснованность критерия на уровне объектов даже в масштабных многоклассовых сценариях.

Результаты на ind_safety.

При обучении на полном обучающем множестве ind_safety достигается 66.85 % mAP50–95. График цепочки активного обучения приведён на рисунке 2.13, численные значения – в таблице 5.

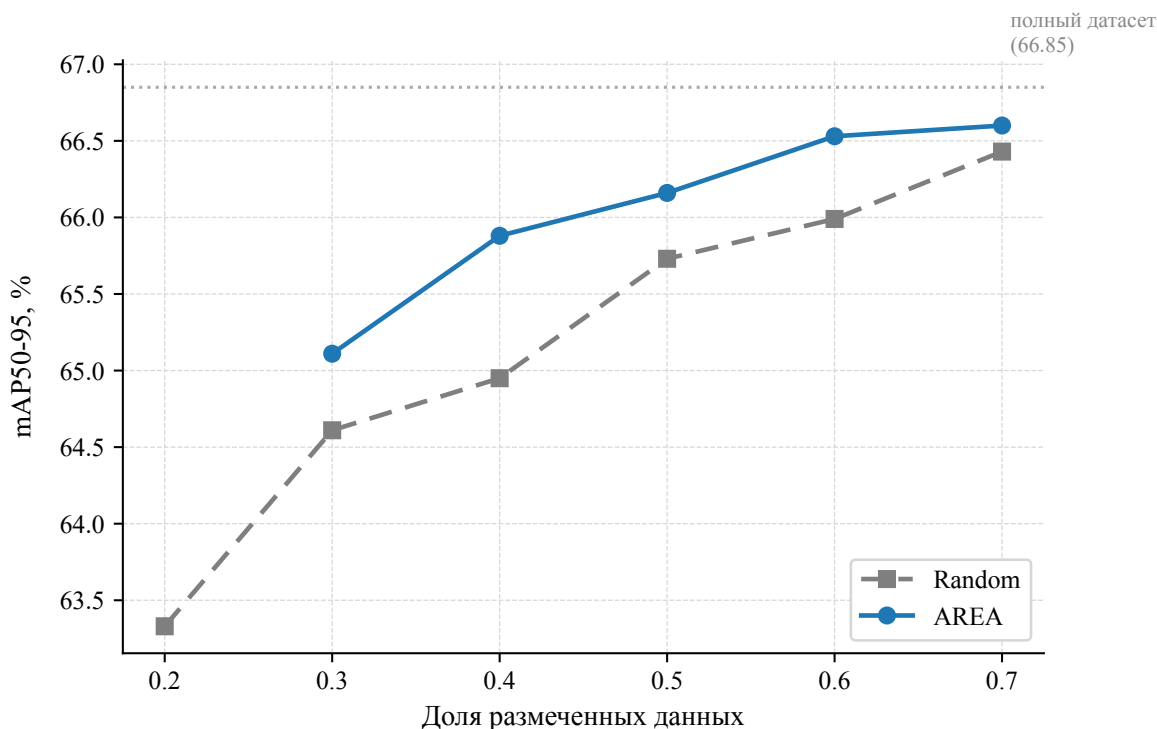


Рисунок 2.13 – Цепочки активного обучения на ind_safety [62]. Пунктир – качество на полном наборе данных (66.85 %).

На промышленном наборе ind_safety для AREA наблюдается наиболее выраженное преимущество при малых долях разметки. При доле 0.3 прирост отно-

Таблица 5 – mAP50–95, %, на ind_safety (среднее $\pm$ 95 % CI, 4 инициализации) [62]

Доля	RANDOM	AREA
0.2	63.33 $\pm$ 0.69	–
0.3	64.61 $\pm$ 0.60	65.11 $\pm$ 0.18
0.4	64.95 $\pm$ 0.22	65.88 $\pm$ 0.14
0.5	65.73 $\pm$ 0.36	66.16 $\pm$ 0.35
0.6	65.99 $\pm$ 0.35	66.53 $\pm$ 0.64
0.7	66.43 $\pm$ 0.19	66.60 $\pm$ 0.36

сительно RANDOM составляет +0.50 п.п., при доле 0.4 – +0.93 п.п., при доле 0.5 – +0.43 п.п. Это подтверждает центральную гипотезу работы: в промышленных доменах видеоаналитики с высокой внутриклассовой вариативностью критерий новизны объектов позволяет эффективнее покрывать редкие визуальные подгруппы при ограниченном бюджете разметки. При доле 0.7 преимущество AREA (+0.17 п.п.) сохраняется, но становится меньше – что типично для любого метода AL при приближении к полному набору данных.

Скорость работы.

Благодаря линейной сложности и использованию ANN-индекса, AREA существенно быстрее PPAL при одинаковом объёме пула [62]. На наборе CrowdHuman полный цикл подготовки разбиения для AREA составляет 730 секунд против 1029 секунд для PPAL. На наборах COCO 2017 и ind_safety: AREA – 2712 секунд, что свидетельствует о линейном масштабировании с объёмом пула. Основная часть времени AREA приходится на этап извлечения эмбедингов (прогон детектора по всему пулу) и PCA-предобработку; сам этап отбора по расстояниям занимает незначительное время благодаря ANN-поиску.

Для COCO 2017 и ind_safety запуск PPAL в реалистичной постановке (добор от доли 0.2 до доли 0.3) не удалось завершить: один шаг отбора выполнялся более 12 часов без получения результата. Это объясняется квадратичной сложностью этапа отбора по разнообразию в PPAL по числу кандидатов: при больших пулах стоимость попарных сравнений становится недопустимо высокой даже при предварительном отсеке множества кандидатов. По этой причине сравнение с PPAL приводится только для Pascal VOC и CrowdHuman (где эксперимент завершился в разумное время).

2.6 Выводы по главе 2

В главе 2 описан метод AREA – первый предложенный в диссертации метод активного обучения детекторов объектов на основе локальных эмбеддингов. Основные результаты главы состоят в следующем.

1. Разработан способ извлечения локальных эмбеддингов из мультимасштабных карт признаков детектора YOLOv8s. Операция RoI Align применяется к каждому уровню пирамиды признаков независимо, результаты конкатенируются в единый вектор фиксированной размерности (2.3). При таком подходе устраняется погрешность округления координат, характерная для наивного взятия среза тензора, и учитываются объекты всех масштабов через многоуровневую агрегацию.
2. Предложен критерий отбора данных по новизне объектов относительно уже размеченного множества: изображению присваивается максимальная оценка среди всех его объектов (2.9). Показано, что вычислительная сложность отбора составляет $O(|U_{t-1}|)$ – линейна по объёму неразмеченного пула, что обеспечивает практическое преимущество по скорости перед PPAL.
3. Для метода AREA получены устойчиво более высокие результаты, чем для случайного отбора, на всех четырёх наборах данных: Pascal VOC, COCO 2017, CrowdHuman и ind_safety. Наибольший относительный выигрыш достигается на промышленном наборе ind_safety при малых долях разметки ($\alpha = 0.3\text{--}0.4$), что соответствует целевому сценарию видеоаналитики с ограниченным бюджетом разметки. На CrowdHuman для AREA получены результаты выше не только RANDOM, но и PPAL,
4. Проведённое сравнение критерия по расстоянию и критерия по плотности не выявило систематического преимущества ни одного из них. Поскольку критерий по расстоянию вычислительно дешевле, он принят как основной критерий AREA [61].

Ограничения AREA и требования к следующему методу. У метода есть два структурных ограничения.

Первое ограничение: применение PCA для снижения размерности с $D = 18\,144$ до $d = 512$. Этот шаг добавляет вычислительные затраты, занимает зна-

чительную долю полного времени отбора и потенциально искажает геометрию пространства эмбедингов в областях с малым числом обучающих примеров.

Второе ограничение: источник представлений. В AREA используются предсказания выходного модуля YOLOv8 (пространство вероятностей классов и координат рамок, размерность $r_h r_w (C+4)L$ после RoI Align), а не промежуточные карты активации перед ним. RoI Align сохраняет пространственное расположение значений на сетке 12×6 внутри рамки. Ограничение связано с содержанием выходных каналов, а не с отсутствием пространственной структуры: они несут меньше визуальной информации, чем скрытые карты признаков [16; 28].

Для преодоления обоих ограничений необходим детектор, который формирует семейство согласованных вложенных представлений непосредственно из промежуточных карт признаков. Такое семейство позволило бы: отказаться от PCA, заменив его каскадным поиском по подвекторам меньшей размерности; работать с информативными скрытыми признаками вместо компактных выходных; поддерживать произвольный уровень детализации. В главе 3 описывается архитектурная модификация выходного модуля YOLOv8, реализующая эти свойства.

Глава 3. Модификация выходного модуля YOLOv8 по принципу Matryoshka Representation Learning

В настоящей главе описывается архитектурная модификация выходного модуля YOLOv8, в рамках которой обучается семейство вложенных представлений разной размерности без введения дополнительных обучаемых параметров [4]. В разделе 3.1 обосновывается актуальность вложенных представлений для задач детекции. В разделе 3.2 приводится обзор смежных работ. В разделе 3.3 вводится формальное описание модификации. В разделе 3.4 рассматриваются механизмы стабилизации обучения и извлечение вложенных эмбеддингов объектов. В разделе 3.5 описывается каскадный поиск. Экспериментальная проверка приведена в разделе 3.6.

3.1 Вложенные представления объектов

Локальный эмбеддинг объекта, извлечённый из нейросетевого детектора, описывает визуальное содержимое объекта в виде вектора фиксированной размерности. Такие векторы применяются в широком круге задач: поиске похожих объектов в базе данных детекций, активном обучении детекторов (как в методе AREA, описанном в главе 2), дообучении моделей на новые домены, и объяснении предсказаний через поиск ближайших ранее наблюдавшихся примеров.

Во всех этих задачах возникает один и тот же компромисс: вектор большой размерности точнее описывает объект, но дороже в хранении и поиске; вектор малой размерности быстрее сравнивать, но он теряет часть информации. Стандартный путь снижения размерности – PCA или дополнительный проекционный слой – требует отдельного шага после обучения детектора и не гарантирует, что укороченное представление останется согласованным с полным: проекция может не сохранять значимые для сравнения объектов направления признакового пространства.

Принцип *Matryoshka Representation Learning* (MRL) [47] предлагает принципиально иной подход: первые m координат обученного вектора образуют самостоятельное и информативное представление меньшей размерности, а после-

дующие координаты лишь уточняют его. Одно и то же обученное представление обеспечивает семейство вложенных подвекторов, каждый из которых является полноценным эмбедингом. В дальнейшем такое представление рассматривается как масштабируемое по размерности: при переходе от $e_{1/8}$ к $e_{1/4}$ и e_1 меняется объём используемых признаков, но не требуется обучать отдельную модель или строить отдельную проекцию.

Такое семейство открывает возможность каскадного поиска (coarse-to-fine): на первом этапе сравнение выполняется по короткому подвектору, что не требует серьезных вычислительных ресурсов и занимает малое количество времени; на втором этапе небольшое число отобранных кандидатов уточняется по более длинному, информативному подвектору. Итоговый результат близок к полному перебору в высокой размерности, но достигается в несколько раз быстрее. Принцип каскадного поиска иллюстрирует рисунок 3.1.

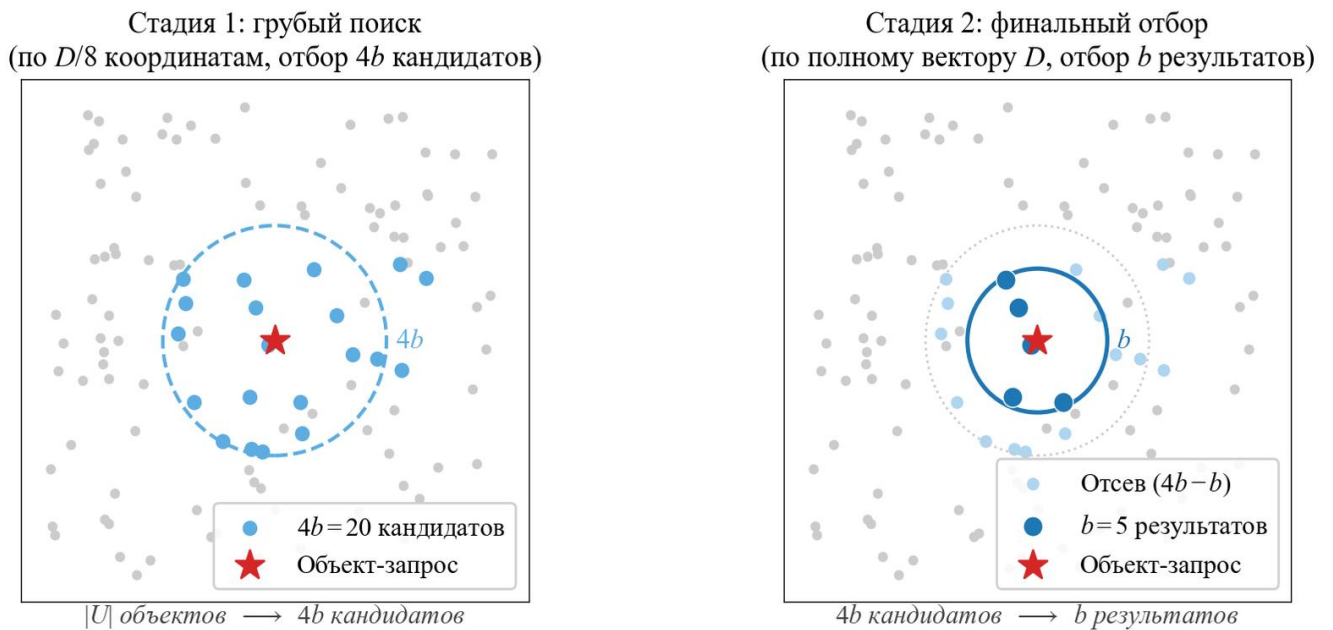


Рисунок 3.1 – Схема каскадного поиска по вложенным эмбедингам [4; 47].

В задаче активного обучения семейство вложенных эмбедингов снимает оба структурных ограничения метода AREA. Первое ограничение, необходимость в PCA, устраняется, т.к. короткий подвектор уже содержит основную структуру признакового пространства, поэтому отдельная проекция не нужна. Второе ограничение, ограниченная выразительность выходных представлений, устраняется, поскольку эмбединги извлекаются из промежуточных карт признаков с помощью RoI Align, а не из выходного пространства вероятностей.

Перенос принципа MRL на выходной модуль одностадийного детектора нетривиален: в выходном модуле одновременно решаются задачи классификации, локализации и согласования предсказаний с нескольких уровней пирамиды признаков. Описанию этой модификации посвящена настоящая глава.

3.2 Обзор литературы

Matryoshka Representation Learning. В работе [47] предложено обучать векторные представления так, чтобы подвектор любой заданной длины сам по себе давал хорошее описание объекта. Стандартная схема обучения дополняется набором целевых функций на каждом из заранее выбранных срезов эмбединга, с убывающими весами. Результат – не один вектор, а иерархия вложенных представлений: короткий подвектор несёт основную семантику, а дополнительные координаты последовательно добавляют детали. В [47] показано, что такая многоуровневая цель не ухудшает качество модели относительно независимо обученных представлений фиксированной размерности, зато открывает возможность многоступенчатого поиска.

Каскадная стратегия поиска [47] позволяет значительно сократить число полных сравнений. Сначала по короткому подвектору выполняется быстрый предварительный отбор небольшого множества кандидатов из всей базы. Затем расстояния до тех же кандидатов пересчитываются по более длинному или полному вектору, после чего выбирается финальный результат. Так полная информация привлекается лишь там, где это необходимо, а не при переборе всех хранящихся объектов. Экспериментально в [47] достигается ускорение поиска в несколько раз при снижении полноты не более чем на несколько процентов.

Принцип матрёшки нашёл применение в ряде последующих работ, охватывающих разные аспекты масштабируемости модели. В [44] гибкость по размерности вводится в скрытые слои трансформера: одна модель поддерживает разное число каналов в полносвязных блоках прямого распространения (FFN) без повторного обучения для каждого ограничения на вычислительные затраты. В работе [8] эта идея распространяется сразу на два измерения – глубину модели и размерность выходного представления, что позволяет одновременно адаптировать и стоимость вычислений, и информативность итогового вектора. В [45; 46] вло-

женность применяется к числу мультимодальных токенов в задачах визуально-языкового понимания. Во всех упомянутых работах объектом модификации служит основной модуль или текстовый кодировщик; выходной модуль одностадийного детектора, отвечающий одновременно за классификацию, локализацию и назначение предсказаний с нескольких уровней пирамиды, остаётся вне поля зрения. Перенос вложенности на выходной модуль детектора нетривиален именно потому, что в процедуре обучения выходного модуля используется совместное назначение предсказаний объектам, одни и те же карты признаков используются как для классификации, так и для детекции без разделения каналов.

Slimmable Networks. Архитектуры с переменной шириной каналов впервые систематически исследованы в [60]: авторы обучили единую сеть, способную в режиме вывода работать при нескольких заранее заданных значениях ширины. Ключевое техническое решение состоит в совместном обучении всех режимов с общими весами при параллельном пересчёте статистик пакетной нормализации (Batch Normalization, BatchNorm) для каждой ширины. В [79] ограничение на дискретный набор значений ширины было снято: модель стала поддерживать произвольную долю ширины, что исключило необходимость перечислять все варианты заранее. В работе [24] применимость динамического изменения ширины проверена также в задаче детекции объектов на Pascal VOC. Тем не менее во всех этих работах целью является адаптивная вычислительная стоимость при выводе, а не построение согласованных вложенных представлений. Подвектор, получаемый при малой ширине, соответствует лишь начальным каналам тензоров весов и не обязан содержать полную семантику объекта в смысле, необходимом для последующего поиска.

Связь между направлениями. Оба направления – MRL и *slimmable networks* – решают задачу получения нескольких режимов работы из одной обученной модели, но с разными приоритетами. MRL добивается семантической согласованности представлений разной длины за счёт оптимизации многоуровневой функции потерь, при этом вопрос вычислительных затрат при применении обученной модели отходит на второй план. *Slimmable networks* нацелены на экономию вычислений при выводе, не ставя задачи по получению содержательных промежуточных представлений. Предлагаемая модификация выходного модуля YOLOv8 объединяет эти подходы: многоуровневая функция потерь по схеме матрёшки (из MRL) при совместном обучении с общими весами и пересчётом статистик BatchNorm (из *slimmable networks*).

3.3 Модификация выходного модуля YOLOv8 по принципу Matryoshka Representation Learning

Выходной модуль YOLOv8 [34] включает регрессионную ветвь `cv2` и классификационную ветвь `cv3`. Обе ветви независимо применяются к каждому уровню пирамиды признаков и для каждой позиции карты выдают координаты рамки и вероятности классов. Первый свёрточный блок каждой ветви принимает входную карту уровня.

Карту признаков l -го уровня на входе выходного модуля обозначим через

$$X_l \in \mathbb{R}^{B \times C_l \times H_l \times W_l}, \quad (3.1)$$

где B – размер пакета, C_l – число каналов, $H_l \times W_l$ – пространственные размеры, $l \in \{1, 2, 3\}$. В исходной YOLOv8 весь тензор X_l без изменений передаётся в обе ветви.

Модификация добавляет вложенность вдоль оси каналов тензора X_l . Задаётся набор делителей $D = \{d_1, \dots, d_K\}$, $d_1 > \dots > d_K = 1$. Для каждого сочетания уровня l и режима k вычисляется ширина среза

$$g_{l,k} = \max\left(1, \left\lfloor \frac{C_l}{d_k} \right\rfloor\right) \quad (3.2)$$

и соответствующий срез по каналам:

$$X_{l,k} = X_l[:, :, g_{l,k} :, :] \in \mathbb{R}^{B \times g_{l,k} \times H_l \times W_l}. \quad (3.3)$$

Пространственные размеры и размер пакета при этом не меняются. Под шириной представления здесь понимается число используемых каналов $g_{l,k}$. Длина подвектора и глубина вложенного эмбединга далее используются как синонимы и обозначают число используемых координат вектора после RoI Align и переплетения. Число слоёв сети при изменении ширины остаётся постоянным. Рабочая конфигурация – делители $D = \{8, 4, 1\}$: доли каналов $\frac{1}{8}, \frac{1}{4}, 1$.

Изменение затрагивает только первый блок каждой ветви: вместо X_l он получает $X_{l,k}$, а его тензор весов срезается по входным каналам аналогичным образом. Остальная часть графа – выходные размеры первого блока, все последующие слои – остаётся без изменений. Никаких новых параметров не добавляется: используются срезы существующих тензоров `cv2` и `cv3`.

Пусть $h_{l,k}$ – результат работы выходного модуля на уровне пирамиды l при ширине $g_{l,k}$. Совокупность выходов при фиксированном k образует набор

$$H_k = \{h_{l,k}\}_{l=1}^L, \quad k = 1, \dots, K. \quad (3.4)$$

Для каждого H_k вычисляется та же потеря $\mathcal{L}_{\text{det}}$, что используется в стандартной YOLOv8. Совокупная функция потерь по схеме матрёшки – взвешенная сумма по всем режимам:

$$\mathcal{L}_{\text{mat}} = \sum_{k=1}^K \alpha_k \mathcal{L}_{\text{det}}(H_k), \quad (3.5)$$

где α_k – коэффициент режима k . Режим $d_K = 1$ (полная ширина) является основным и получает наибольший коэффициент. Для $D = \{8, 4, 1\}$ принято $[\alpha_1, \alpha_2, \alpha_3] = [0.2, 0.4, 1.0]$.

Такая модификация обладает двумя важными свойствами: Во-первых, число параметров не растёт: модель, обученная с потерей матрёшки, отличается от стандартной YOLOv8 лишь значениями весов. Во-вторых, при работе в режиме полной ширины её архитектура идентична исходному детектору. Таким образом, описанная модификация расширяет функциональность YOLOv8, не заменяя её.

Схема показана на рисунке 3.2. Для каждого уровня пирамиды задаются три значения ширины $g_{l,1} \leq g_{l,2} \leq g_{l,3} = C_l$; усекается только вход первого свёрточного слоя каждой ветви, а итоговая функция потерь (3.5) является взвешенной суммой стандартных детекционных потерь YOLOv8 при каждой ширине [4].

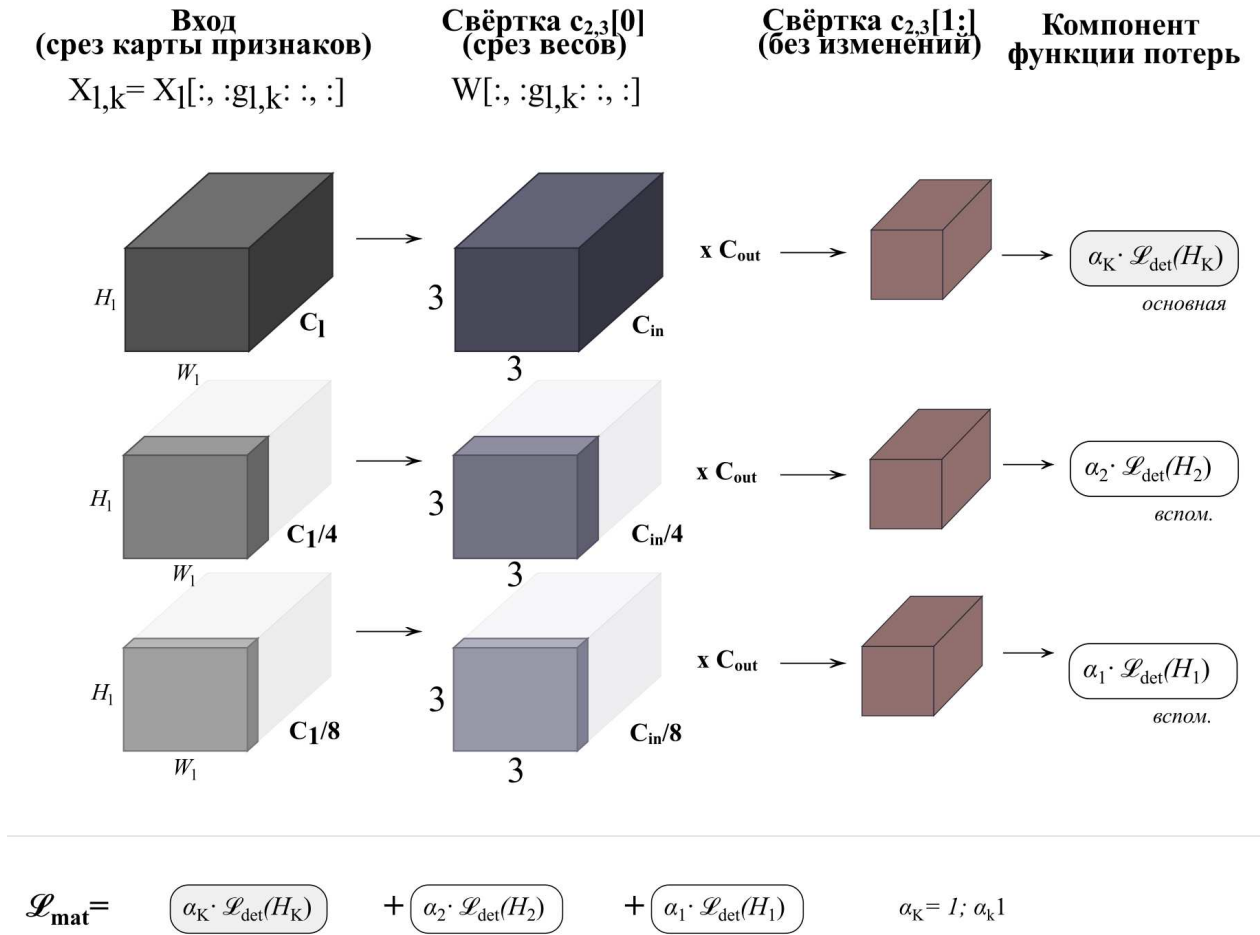


Рисунок 3.2 – Схема модификации выходного модуля YOLOv8 по принципу Matryoshka Representation Learning [4].

3.4 Стабилизация обучения и извлечение вложенных эмбеддингов

Прямолинейное применение схемы матрёшки к выходному модулю приводит к нестабильному обучению, поскольку несколько режимов разной ширины формируют конкурирующие обучающие сигналы. Ключевая сложность связана с процедурой назначения предсказаний объектам (assignment): в YOLOv8 она выполняется динамически.

В выходном модуле предсказания формируются в фиксированном наборе пространственных позиций на картах признаков всех трёх уровней пирамиды. Для каждого истинного объекта рассматриваются позиции, попадающие внутрь его ограничивающего прямоугольника. Для каждой такой позиции вычисляется мера согласованности, объединяющая предсказанную вероятность класса объекта и качество локализации (IoU между предсказанным и истинным прямоугольниками). Позиции с наибольшей мерой согласованности получают статус *положи-*

тельных и используются для вычисления регрессионного и классификационного компонентов функции потерь; остальные позиции считаются фоновыми.

Если проводить назначение независимо для каждой ширины, возникает противоречие: режимы с меньшей шириной имеют другие предсказанные вероятности и координаты рамок, поэтому они выбирают другие положительные позиции для одних и тех же целевых объектов. В результате вспомогательные режимы обучаются не только на иных признаках, но и по иной структуре задачи – с другими масками положительных позиций, что создаёт конкурирующие и противоречивые обучающие сигналы.

Общее назначение. Для устранения этого конфликта назначение предсказаний объектам выполняется один раз – по выходам режима полной ширины. Полученные маски положительных позиций, целевые прямоугольники и классовые веса затем используются без изменений для всех вспомогательных режимов [4]. По смыслу это напоминает дистилляцию, однако режим полной ширины передаёт вспомогательным не свои вероятности классов, а результат процедуры назначения. Схема общего назначения показана на рисунке 3.3.

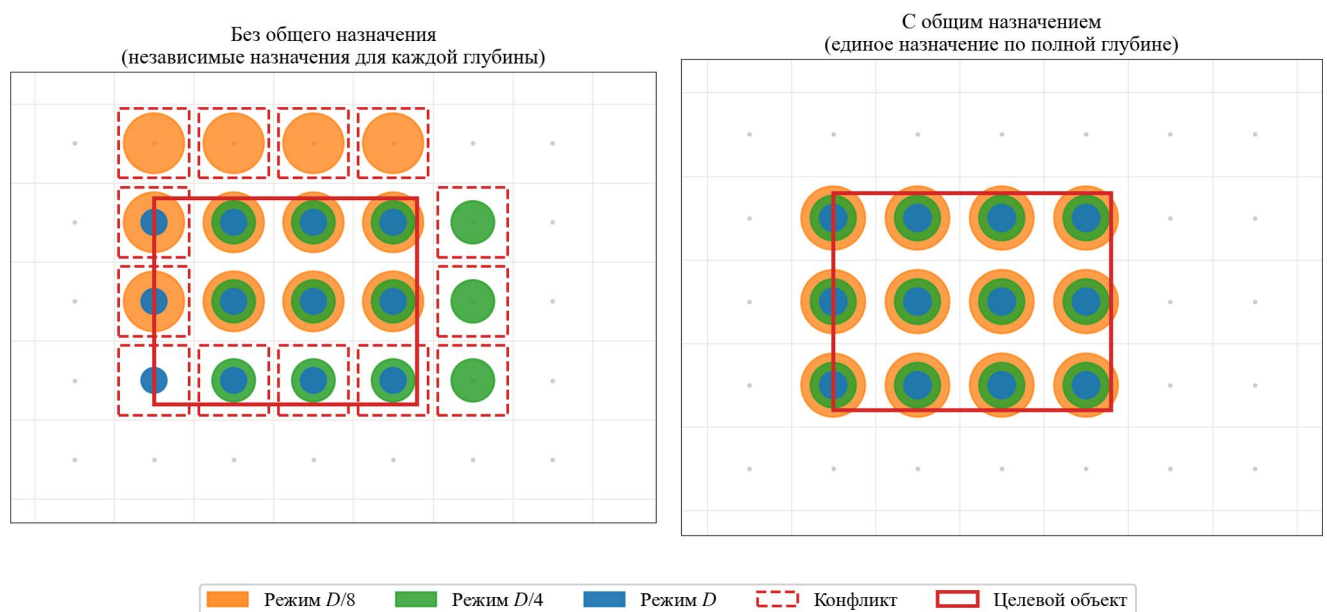


Рисунок 3.3 – Общее назначение (shared assignment) при обучении по схеме матрички [4].

Статистики BatchNorm. Представления меньшей ширины имеют иное распределение входных активаций по сравнению с полным режимом – из-за отсутствующих каналов. Поэтому во вспомогательных проходах накопленные скользящие оценки среднего и дисперсии BatchNorm фиксируются: слой нормализации использует статистики текущего пакета, но не обновляет накоплен-

ные оценки [4]. Основной полный режим сохраняет стандартное поведение BatchNorm.

Постепенное увеличение коэффициентов вспомогательных потерь. На ранних эпохах сильный обучающий сигнал от вспомогательных режимов может дестабилизировать основной режим. Поэтому коэффициенты α_k вспомогательных потерь при $k < K$ постепенно увеличиваются от нуля до заданных значений в течение первых нескольких эпох после включения режима матрёшки. Коэффициент основного режима α_K при этом не изменяется. Кроме того, сам режим матрёшки включается не с первой эпохи, а после завершения начального обучения основной модели; до этого момента используется стандартный режим выходного модуля YOLOv8 [4].

Стохастический выбор режимов и отложенный старт. Прохождение через все K режимов на каждом шаге увеличивает время эпохи пропорционально числу режимов. Для снижения затрат на каждом шаге случайным образом выбирается подмножество из нескольких режимов (параметр `sample_n`, в экспериментах – 2), при этом полный режим всегда включается в это подмножество [4].

Сам режим матрёшки включается не с первой эпохи, а после завершения начального обучения основной модели (параметр `start_epoch`, в экспериментах – 30). До этого момента используется стандартный режим выходного модуля YOLOv8, что позволяет основным весам сойтись прежде, чем будет добавлен вспомогательный обучающий сигнал.

Совместное применение описанных мер (`sample_n=2`, `start_epoch=30`, постепенное увеличение коэффициентов вспомогательных потерь) снижает замедление обучения до 6% по сравнению со стандартной YOLOv8, при этом итоговое качество детектора остаётся в пределах доверительного интервала стандартной конфигурации.

Извлечение эмбеддингов объектов с помощью RoI Align. После обучения модифицированного выходного модуля источником эмбеддингов служат промежуточные карты признаков на его входе. В отличие от метода AREA, где эмбеддинги брались из выходного пространства вероятностей (глава 2), промежуточные карты содержат семантически богатые и пространственно детализированные признаки до сжатия до размерности классификационного или регрессионного выхода.

Для каждого детектированного объекта ограничивающий прямоугольник проецируется на карты каждого уровня пирамиды; операция RoI Align [43] разме-

ром $r \times r$ ($r = 4$ в основной конфигурации) извлекает локальное представление без округления координат, сохраняя огрублённое пространственное расположение признаков на сетке 4×4 . В экспериментах используются карты со всех трёх уровней пирамиды (`/model.15/cv2/act/Mul`, `/model.18/cv2/act/Mul`, `/model.21/cv2/act/Mul` для YOLOv8s). Это выходы блоков C2f переходного модуля, поступающие на вход выходного модуля `model.22` [34]. Обозначение `cv2` в этих путях относится к свёртке внутри C2f и не обозначает одноимённую регрессионную ветвь выходного модуля. После применения RoI Align тензоры с разных уровней проходят процедуру переплетения, описанную ниже.

Переплетение признаков уровней пирамиды. Простая конкатенация признаков разных уровней по оси каналов нарушает свойство матришки: первая часть итогового вектора будет содержать только признаки одного уровня, тогда как все остальные уровни попадут лишь в хвост. Для устранения этого предлагается операция переплетения [4]. Представление каждого уровня l делится на d блоков одинакового размера по оси каналов:

$$X_l^{(\text{roi})} = [B_{l,1} \parallel B_{l,2} \parallel \dots \parallel B_{l,d}]. \quad (3.6)$$

Итоговый вектор формируется как чередование блоков с одинаковыми номерами по всем уровням:

$$e(o) = \left[\underbrace{B_{1,1} \parallel \dots \parallel B_{L,1}}_{\text{блок 1}} \parallel \underbrace{B_{1,2} \parallel \dots \parallel B_{L,2}}_{\text{блок 2}} \parallel \dots \parallel \underbrace{B_{1,d} \parallel \dots \parallel B_{L,d}}_{\text{блок } d} \right]. \quad (3.7)$$

Благодаря такому порядку начальная доля $1/d$ координат итогового вектора содержит соответствующую долю признаков со всех уровней пирамиды. Поэтому укороченный подвектор не сводится к признакам одного уровня FPN, а остаётся многомасштабным описанием объекта. Масштабируемость итогового эмбединга задаётся одновременно длиной используемого подвектора и сохранением признаков разных пространственных масштабов внутри этого подвектора. При конфигурации $D = \{8, 4, 1\}$ это даёт три длины подвектора (схема показана для $D = \{4, 2, 1\}$ на рисунке 3.4, в экспериментах используется $D = \{8, 4, 1\}$):

$$\mathcal{E}(o) = \{e_{1/8}(o), e_{1/4}(o), e_1(o)\}, \quad (3.8)$$

где $e_{1/8}$ и $e_{1/4}$ – подвекторы меньшей размерности, а e_1 – полное представление. Перед использованием выполняется L2-нормализация. Схема переплетения показана на рисунке 3.4. Каждый уровень разбивается на блоки; блоки с одинаковыми

номераи со всех уровней образуют соответствующую часть итогового вложенного вектора.

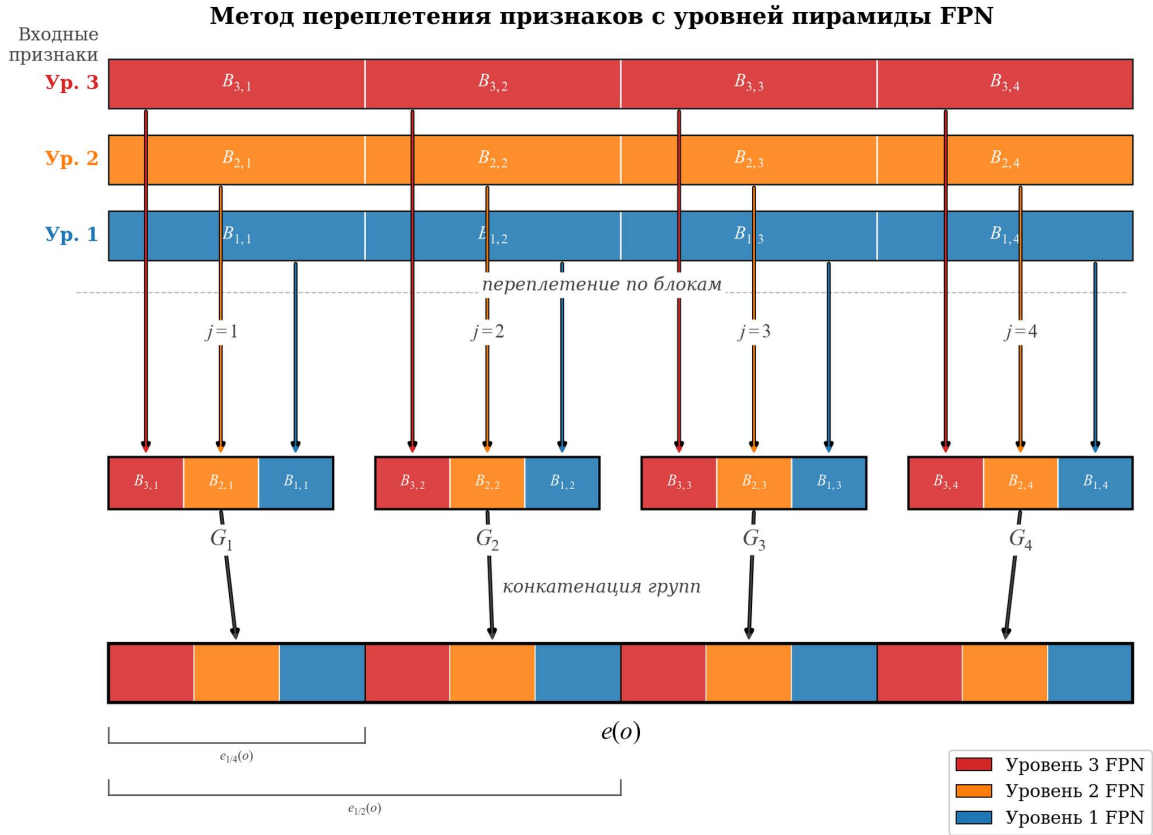


Рисунок 3.4 – Схема переплетения признаков с трёх уровней пирамиды в единый вложенный вектор [4].

3.5 Каскадный поиск по вложенным эмбедингам

Семейство вложенных эмбедингов (3.8) позволяет строить многоэтапный поиск, в котором каждый следующий этап уточняет результат предыдущего, используя более информативное представление. Принцип описан в разделе 3.1; здесь даётся формальное описание алгоритма применительно к задаче поиска похожих объектов [4; 47].

Алгоритм. Пусть требуется найти K объектов, ближайших к объекту-запросу q в галерее $\mathcal{G}$ размером $|\mathcal{G}|$. Поиск состоит из двух этапов.

На первом этапе по подвектору $e_{1/8}(q)$ из всей галереи отбирается предварительный список из $N_1 \gg K$ кандидатов. Для ускорения поиска используется

HNSW (Hierarchical Navigable Small World graphs) [41], построенный на коротких подвекторах всей галереи – это позволяет выполнять приближённый поиск ближайших соседей за логарифмическое время вместо линейного перебора.

На втором этапе только N_1 кандидатов пересчитываются по полному вектору $e_1(q)$, заново сортируются по косинусному расстоянию, и финальный список из K объектов извлекается из отсортированных кандидатов. Полные сравнения выполняются лишь для небольшой доли галереи: $N_1 \ll |\mathcal{G}|$.

Меры качества поиска. В постановке поиска по изображениям используются меры качества Recall@K и mAP. Пусть Q – множество запросов, $r(q)$ – ранг первого релевантного объекта для запроса q , $\text{rel}_q(k) \in \{0,1\}$ – индикатор релевантности объекта на позиции k , а $P_q(k)$ – точность на первых k позициях списка результатов. Тогда

$$\begin{aligned} \text{Recall@K} &= \frac{1}{|Q|} \sum_{q \in Q} \mathbb{I}[r(q) \leq K], \\ AP(q) &= \frac{1}{R_q} \sum_{k=1}^N P_q(k) \text{rel}_q(k), \\ \text{mAP} &= \frac{1}{|Q|} \sum_{q \in Q} AP(q). \end{aligned} \tag{3.9}$$

В задаче поиска, рассматриваемой в данной главе, для каждого запроса существует один релевантный объект, поэтому Recall@K показывает долю запросов, для которых искомый объект попал в первые K результатов. Здесь R_q обозначает число релевантных объектов для запроса q ; в рассматриваемой постановке поиска $R_q = 1$.

Результаты экспериментов. На Pascal VOC одноэтапный поиск по полному вектору даёт Recall@10 = 0.9692 при среднем времени 20.5 мс на запрос. Двухэтапный каскад $e_{1/8} \rightarrow e_1$ достигает Recall@10 = 0.9695 при времени 6.7 мс на запрос – ускорение в 3.1 раза без потери качества [4]. На меньших пулах (4096 объектов) ускорение составляет около 2 раз при снижении Recall@10 на 0.78–6.91 процентного пункта (1.68–10.55 % относительно полного поиска) в зависимости от набора данных.

Две стадии каскадного поиска рассматриваются на примере шести классов Pascal VOC. Для каждого объекта-запроса (синяя рамка) показаны девять кандидатов первой стадии поиска по подвектору $e_{1/8}$ и три результата второй стадии поиска по полному вектору e_1 . Зелёная рамка обозначает совпадение класса с запросом, красная – другой класс; фиолетовая рамка обозначает переход кандидата

во вторую стадию. Числа в подписях на рисунке 3.5 указывают количество отображённых примеров.

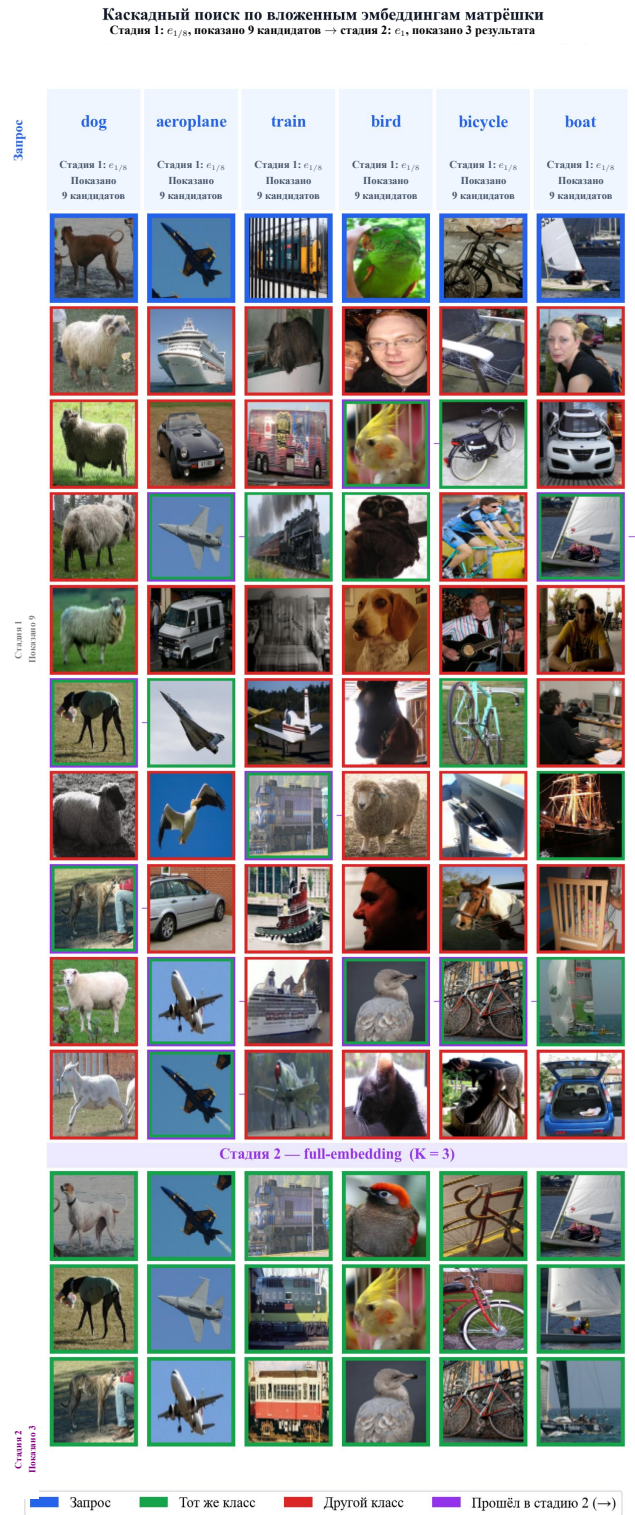


Рисунок 3.5 — Пример каскадного поиска по вложенным эмбедингам на Pascal VOC [4].

3.6 Экспериментальная проверка модели с вложенными представлениями

Сохранение качества детекции. На каждом из четырёх наборов данных сравниваются стандартная YOLOv8s (`plain`) и модифицированная по схеме матрёшки (`matr`). Каждая конфигурация обучается на вложенных долях $0.2, \dots, 0.7$ в четырёх независимых запусках; в ячейках – среднее mAP50–95 и 95 % доверительный интервал [4]. Численные результаты приведены в таблице 6, а графическое сравнение – на рисунке 3.6.

Таблица 6 – Качество детекции mAP50–95, %: стандартная YOLOv8s (`plain`) и модифицированная по схеме матрёшки (`matr`). Среднее $\pm$ 95 % CI, 4 инициализации [4].

Доля	Pascal VOC		CrowdHuman	
	plain	matr	plain	matr
0.2	55.62 $\pm$ 0.80	55.85 $\pm$ 0.39	51.51 $\pm$ 0.11	51.45 $\pm$ 0.11
0.3	58.35 $\pm$ 0.32	58.47 $\pm$ 0.13	52.82 $\pm$ 0.05	52.71 $\pm$ 0.08
0.4	60.23 $\pm$ 0.33	60.01 $\pm$ 0.37	53.67 $\pm$ 0.06	53.61 $\pm$ 0.09
0.5	61.55 $\pm$ 0.26	61.48 $\pm$ 0.25	54.25 $\pm$ 0.08	54.21 $\pm$ 0.05
0.6	62.44 $\pm$ 0.25	62.60 $\pm$ 0.24	54.83 $\pm$ 0.05	54.81 $\pm$ 0.06
0.7	63.77 $\pm$ 0.40	63.59 $\pm$ 0.17	55.59 $\pm$ 0.03	55.53 $\pm$ 0.07
Доля	COCO 2017		ind_safety	
	plain	matr	plain	matr
0.2	36.29 $\pm$ 0.23	36.34 $\pm$ 0.05	63.33 $\pm$ 0.69	63.43 $\pm$ 0.48
0.3	38.73 $\pm$ 0.13	38.48 $\pm$ 0.32	64.61 $\pm$ 0.60	64.59 $\pm$ 0.39
0.4	40.03 $\pm$ 0.28	39.90 $\pm$ 0.29	64.95 $\pm$ 0.22	65.15 $\pm$ 0.24
0.5	41.07 $\pm$ 0.10	40.90 $\pm$ 0.18	65.73 $\pm$ 0.36	65.51 $\pm$ 0.49
0.6	41.80 $\pm$ 0.27	41.51 $\pm$ 0.25	65.99 $\pm$ 0.35	66.15 $\pm$ 0.21
0.7	42.38 $\pm$ 0.11	42.28 $\pm$ 0.05	66.43 $\pm$ 0.19	66.40 $\pm$ 0.47

Модификация не ухудшает качество основного режима детектора: добавление вложенных представлений является функциональным расширением YOLOv8.

Меры качества детекции по длинам подвекторов. Ключевым преимуществом схемы матрёшки является сохранение качества детекции при использовании усечённых подвекторов. В таблице 7 сравниваются: `plain` – наивное усечение каналов стандартной модели; `matr` – модель, обученная по схеме матрёшки.

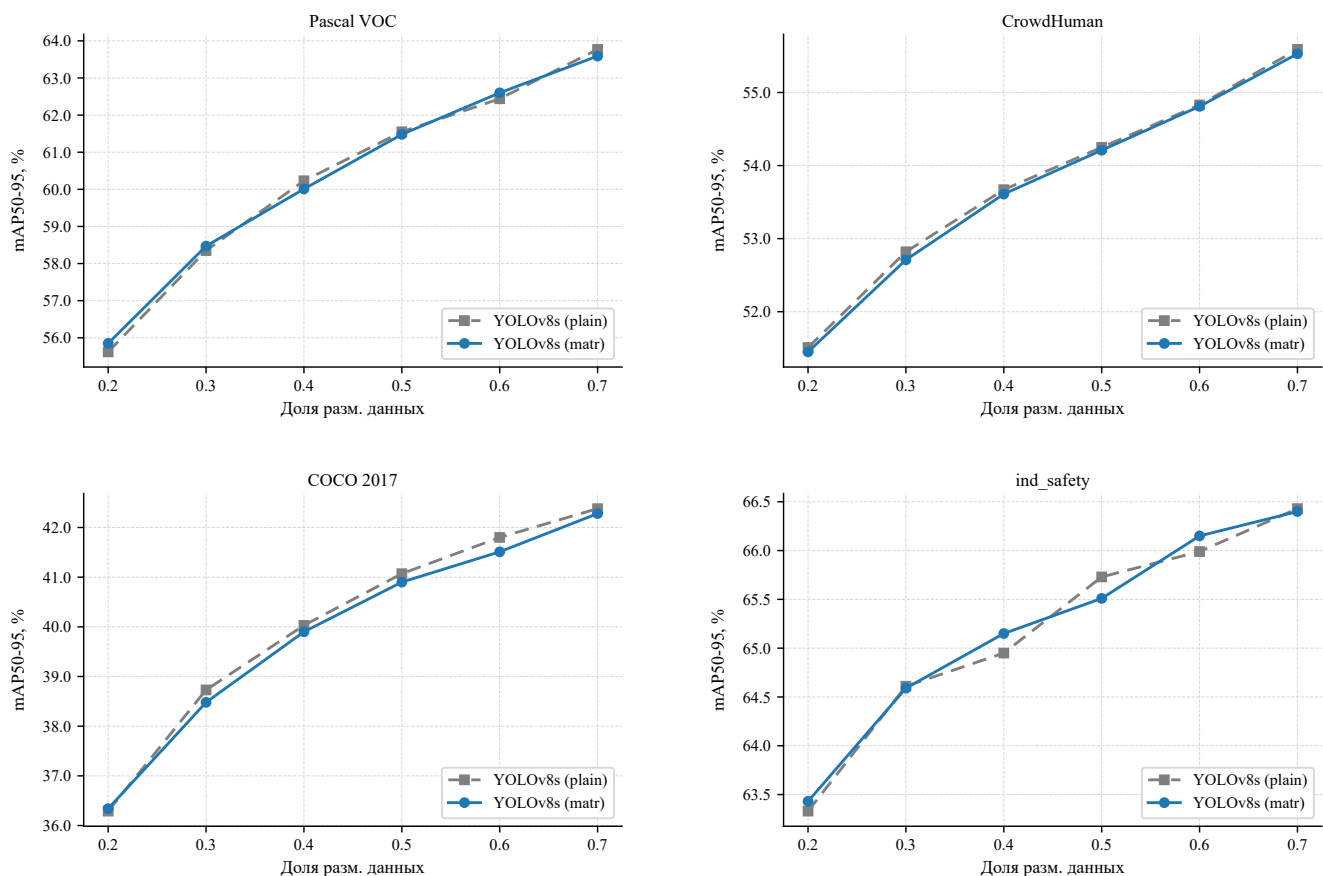


Рисунок 3.6 – Качество детекции YOLOv8s для стандартной конфигурации и модификации по схеме матрички [4].

Столбец Сохр. показывает долю mAP50 относительно полного вектора. Обозначение D в заголовке этой таблицы означает полное число входных каналов соответствующего уровня; $D/8$ и $D/4$ задают используемые доли каналов.

Таблица 7 – Качество детекции (mAP50, %) при использовании подвекторов разной длины. plain: наивное усечение каналов; matr: обучено по схеме матрички. Модели обучены на доле 0.7 с инициализацией OIV7 (Open Images V7) [4].

Набор данных	Модель	$D/8$		$D/4$		D (полная)
		mAP50, %	Сохр.	mAP50, %	Сохр.	mAP50, %
Pascal VOC	plain	20.1	24%	43.6	52%	83.1
	matr	77.1	93%	81.0	97%	83.3
CrowdHuman	plain	5.6	7%	27.6	32%	85.3
	matr	77.5	91%	80.7	95 %	85.2
COCO 2017	plain	6.4	11%	16.8	29%	58.9
	matr	47.2	80%	52.0	89%	58.7
ind_safety	plain	15.3	16%	57.6	60%	95.9
	matr	92.4	96%	94.2	98%	96.1

При наивном усечении стандартной модели до $D/8$ качество детекции сильно падает (7–24% от полного). Модель `matr`, напротив, сохраняет 80–96% качества при $D/8$ и 89–98% при $D/4$. При использовании полного вектора обе модели показывают сопоставимые результаты – разница находится в пределах 0.2 процентного пункта mAP50. Таким образом, модификация по схеме матрёшки является работоспособным детектором при любой из трёх поддерживаемых ширин без дополнительного обучения.

Поиск похожих объектов и каскадное ускорение. Вложенные эмбединги проверяются в постановке поиска похожих объектов на пуле из 4096 объектов валидационной части каждого набора данных. Запрос – эмбединг объекта из исходного изображения; цель – тот же объект после горизонтального отражения среди остальных. Дополнительно измеряется ускорение каскадного поиска $e_{1/8} \rightarrow e_1$ на полном валидационном множестве Pascal VOC [4]. Результаты поиска приведены в таблице 8, а зависимость Recall@10 от длины подвектора вложенного вектора показана на рисунке 3.7.

Таблица 8 – Поиск похожих объектов: Recall@10, mAP и время на запрос для вложенных эмбедингов [4]. Пул – 4096 объектов валидационной части.

Набор данных	Цепочка	R@10	mAP, %	мс/запрос
Pascal VOC	Полный поиск	0.7935	60.32	0.004
Pascal VOC	$e_{1/8} \rightarrow e_1$	0.7778	58.95	0.002
CrowdHuman	Полный поиск	0.6550	46.73	0.004
CrowdHuman	$e_{1/8} \rightarrow e_1$	0.5859	44.16	0.002
COCO 2017	Полный поиск	0.9189	76.64	0.004
COCO 2017	$e_{1/8} \rightarrow e_1$	0.8879	75.37	0.002
ind_safety	Полный поиск	0.4648	28.99	0.004
ind_safety	$e_{1/8} \rightarrow e_1$	0.4570	28.53	0.002
<i>Крупная постановка (Pascal VOC, полное проверочное множество):</i>				
Pascal VOC	Полный поиск	0.9692	–	20.5
Pascal VOC	$e_{1/8} \rightarrow e_1$	0.9695	–	6.7

Каскад $e_{1/8} \rightarrow e_1$ сокращает время поиска примерно вдвое на малых пулах при снижении Recall@10 на 0.78–6.91 процентного пункта (1.68–10.55 % относительно полного поиска). На полном проверочном множестве Pascal VOC время поиска сокращается с 20.5 до 6.7 мс на запрос (примерно в 3.1 раза), а Recall@10 составляет 0.9695 (96.95 %) против 0.9692 (96.92%) у полного поиска.

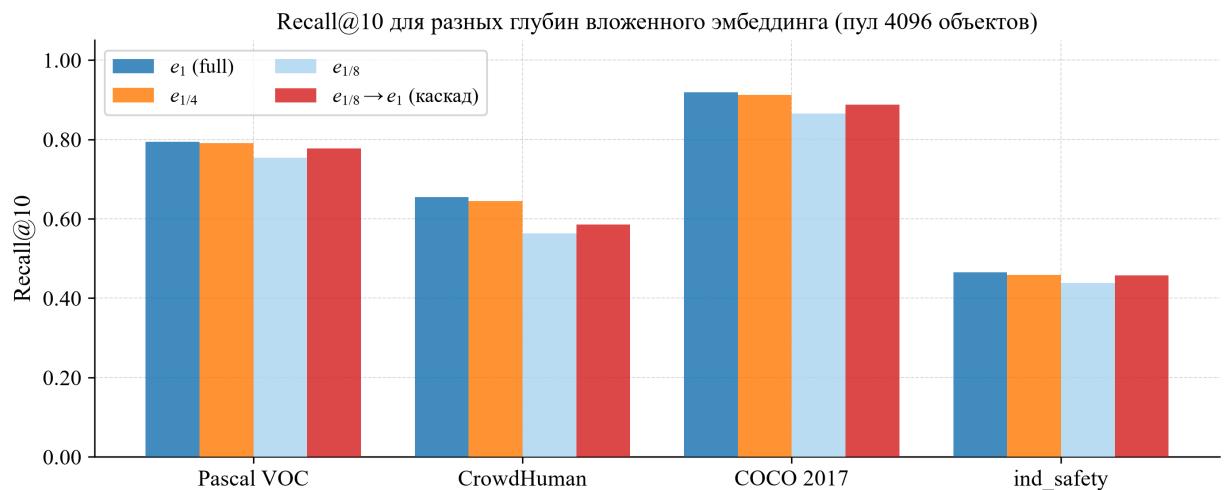


Рисунок 3.7 – Recall@10 при поиске похожих объектов для разных длин подвектора вложенного вектора [4].

3.7 Выводы по главе 3

В главе 3 описана архитектурная модификация выходного модуля YOLOv8, позволяющая получать семейство вложенных локальных эмбедингов объектов без введения дополнительных параметров. Основные результаты главы следующие.

1. Предложена модификация выходного модуля по принципу MRL: первый свёрточный слой каждой ветви получает срез входного тензора по числу каналов, а функция потерь является взвешенной суммой стандартных детекционных потерь при каждой ширине. Дополнительных параметров модификация не вводит.
2. Разработан набор механизмов стабилизации: общее назначение предсказаний объектам (shared assignment), фиксация накопленных статистик BatchNorm во вспомогательных проходах и постепенное увеличение коэффициентов вспомогательных потерь. При конфигурации `sample_n=2` и `start_epoch=30` замедление обучения составляет не более 6%.
3. Предложено переплетение признаков с нескольких уровней FPN-пирамиды, обеспечивающее свойство матришки в итоговом вложенном векторе: начальная доля $1/d$ координат вектора содержит соответствующую долю признаков со всех уровней пирамиды. Тем самым вложенный

эмбединг масштабируется по размерности без потери связи с разными пространственными масштабами объекта.

4. Эксперименты на Pascal VOC, CrowdHuman, COCO 2017 и ind_safety подтверждают, что качество детекции в основном режиме не уступает стандартной YOLOv8. Вложенные эмбединги пригодны для поиска похожих объектов. На полном проверочном множестве Pascal VOC каскадный поиск сокращает время запроса с 20.5 до 6.7 мс (примерно в 3.1 раза), а Recall@10 составляет 0.9695 против 0.9692 у полного поиска.

Полученная модель является источником вложенных эмбедингов для метода MAREА, описанного в главе 4. В отличие от AREА, в MAREА используются промежуточные карты признаков вместо выходного пространства вероятностей, что устраняет оба ограничения AREА: необходимость в PCA и ограниченную выразительность представлений.

Глава 4. MAREA: активное обучение на основе вложенных эмбедингов объектов

В MAREA метод AREА, описанный в главе 2, расширяется за счёт двух изменений. Во-первых, источником эмбедингов становятся карты признаков переходного модуля на входе выходного модуля детектора, обученного по схеме матрёшки (глава 3), а не выходное пространство вероятностей детектора. Во-вторых, поиск ближайших объектов выполняется каскадно по вложенным подвекторам [62], что устраняет необходимость в PCA-предобработке и ускоряет этап отбора. Критерий отбора сохраняет смысл AREА: приоритет получают изображения, объекты которых наиболее удалены от уже размеченных объектов в пространстве признаков.

4.1 Метод MAREA

Источник эмбедингов. В качестве источника вложенных представлений используется YOLOv8s, модифицированная по схеме матрёшки [4]. Для каждого детектированного объекта по его ограничивающему прямоугольнику применяется операция RoI Align ($r = 4$) к картам признаков переходного модуля на трёх уровнях пирамиды (`/model.15/cv2/act/Mul`, `/model.18/cv2/act/Mul`, `/model.21/cv2/act/Mul`). Эти узлы относятся к блокам C2f переходного модуля, выходы которых поступают в выходной модуль `model.22`. Результирующие тензоры переплетаются по схеме из раздела 3.4, и для объекта o формируется семейство вложенных эмбедингов

$$e(o) = \{e_{1/8}(o), e_{1/4}(o), e_1(o)\}. \quad (4.1)$$

Перед поиском выполняется L2-нормализация; PCA не применяется.

Критерий отбора. Оценка изображения строится по тому же принципу, что и в AREА: максимальное косинусное расстояние от объектов изображения до ближайших объектов размеченного множества. В режиме полного поиска:

$$a_{\text{MAREA}}(x, f_\theta) = \max_{e \in E(x)} \min_{z \in E_{L_{t-1}}} d_{\cos}(e_1, z_1), \quad (4.2)$$

где $E_{L_{t-1}}$ – множество эмбеддингов полной размерности для всех объектов уже размеченного множества, поиск ближайшего соседа выполняется по HNSW-индексу [41].

Каскадный поиск. В основной конфигурации используется трёхэтапный каскад: грубый поиск по $e_{1/8}$ формирует $4b$ кандидатов, промежуточное уточнение по $e_{1/4}$ оставляет $2b$, финальный отбор по e_1 выбирает пакет размера b . Это позволяет вычислять расстояния в пространстве полного вектора только для небольшой доли объектов пула.

Пакетный критерий. Набор из b изображений выбирается жадно по сумме оценок:

$$S_t^* = \arg \max_{S \subset U_{t-1}, |S|=b} \sum_{x \in S} a_{\text{MAREA}}(x, f_\theta). \quad (4.3)$$

Контрольный вариант RANDOM-MAT. Для отделения эффекта критерия отбора от эффекта самой модификации выходного модуля вводится контрольный метод RANDOM-MAT: он использует ту же модель YOLOv8 с выходным модулем по схеме матрёшки, но выбирает изображения случайно. Сравнение RANDOM и RANDOM-MAT позволяет проверить, что модификация выходного модуля сама по себе не меняет качество детектора на случайных подмножествах [62].

4.2 Схема экспериментов

Протокол цепочек активного обучения, наборы данных и базовые методы описаны в разделе 2.3 главы 2. Здесь перечислены отличия.

Детектор YOLOv8s с выходным модулем по схеме матрёшки [4]: обученный с параметрами: делители 8, 4, 1, веса [0.2, 0.4, 1.0], sample_n=2, start_epoch=30, warmup_epochs=10.

Сравниваются пять конфигураций: **RANDOM** – случайный отбор, основной нижний ориентир; **RANDOM-MAT** – случайный отбор на модели-матрёшке, контроль; **AREA** [61] – критерий по расстоянию из главы 2; **MAREA** [62] – предлагаемый метод; **PPAL** [74] – стадия отбора по разнообразию метода PPAL, применяемая ко всему неразмеченному пулу без предварительной стадии отбора по неопределённости. Сравнение с ней выполнено только на Pascal VOC и

CrowdHuman (на COCO 2017 и ind_safety один шаг отбора занимает более 12 часов).

4.3 Результаты

Ниже приведены цепочки активного обучения для всех четырёх наборов данных и числовые значения mAP50–95 с 95 % доверительными интервалами (4 инициализации). Сплошная горизонтальная линия на графиках соответствует модели, обученной на полном обучающем множестве (доля 1.0).

Pascal VOC (полный набор данных: 66.14 % mAP50–95). Цепочки активного обучения показаны на рисунке 4.1, численные результаты приведены в таблице 9.

Таблица 9 – Цепочки активного обучения на Pascal VOC (mAP50–95, %, среднее $\pm$ 95 % CI, 4 инициализации) [62]. Полный набор данных: 66.14 %.

Доля	RANDOM	RANDOM-MAT	AREA	MAREA	PPAL
0.2	55.62 $\pm$ 0.80	55.85 $\pm$ 0.39	–	–	–
0.3	58.35 $\pm$ 0.32	58.47 $\pm$ 0.13	58.77 $\pm$ 0.78	58.81 $\pm$ 0.54	59.26 $\pm$ 0.62
0.4	60.23 $\pm$ 0.33	60.01 $\pm$ 0.37	60.96 $\pm$ 0.69	60.99 $\pm$ 0.24	61.16 $\pm$ 0.38
0.5	61.55 $\pm$ 0.26	61.48 $\pm$ 0.25	62.29 $\pm$ 0.32	62.36 $\pm$ 0.16	62.38 $\pm$ 0.42
0.6	62.44 $\pm$ 0.25	62.60 $\pm$ 0.24	62.97 $\pm$ 0.28	63.10 $\pm$ 0.17	63.00 $\pm$ 0.28
0.7	63.77 $\pm$ 0.40	63.59 $\pm$ 0.17	64.13 $\pm$ 0.28	64.47 $\pm$ 0.39	64.22 $\pm$ 0.29

На Pascal VOC при долях 0.3–0.5 средние значения mAP50–95 у PPAL выше, чем у MAREA. При долях 0.6 и 0.7 средние значения MAREA составляют 63.10 % и 64.47 % против 63.00 % и 64.22 % у PPAL: разности равны +0.10 п.п. и +0.25 п.п. соответственно. Эти разности характеризуют средние значения. Кривые RANDOM и RANDOM-MAT близки, что подтверждает: наблюдаемый прирост MAREA обусловлен критерием отбора, а не самой модификацией выходного модуля.

CrowdHuman (полный набор данных: 56.58 % mAP50–95). На CrowdHuman для MAREA получены средние значения mAP50–95 выше результатов AREA и PPAL при всех исследованных долях разметки от 0.3 до 0.7. Средняя разность MAREA и PPAL по mAP50–95 составляет +0.94 п.п. на CrowdHuman и

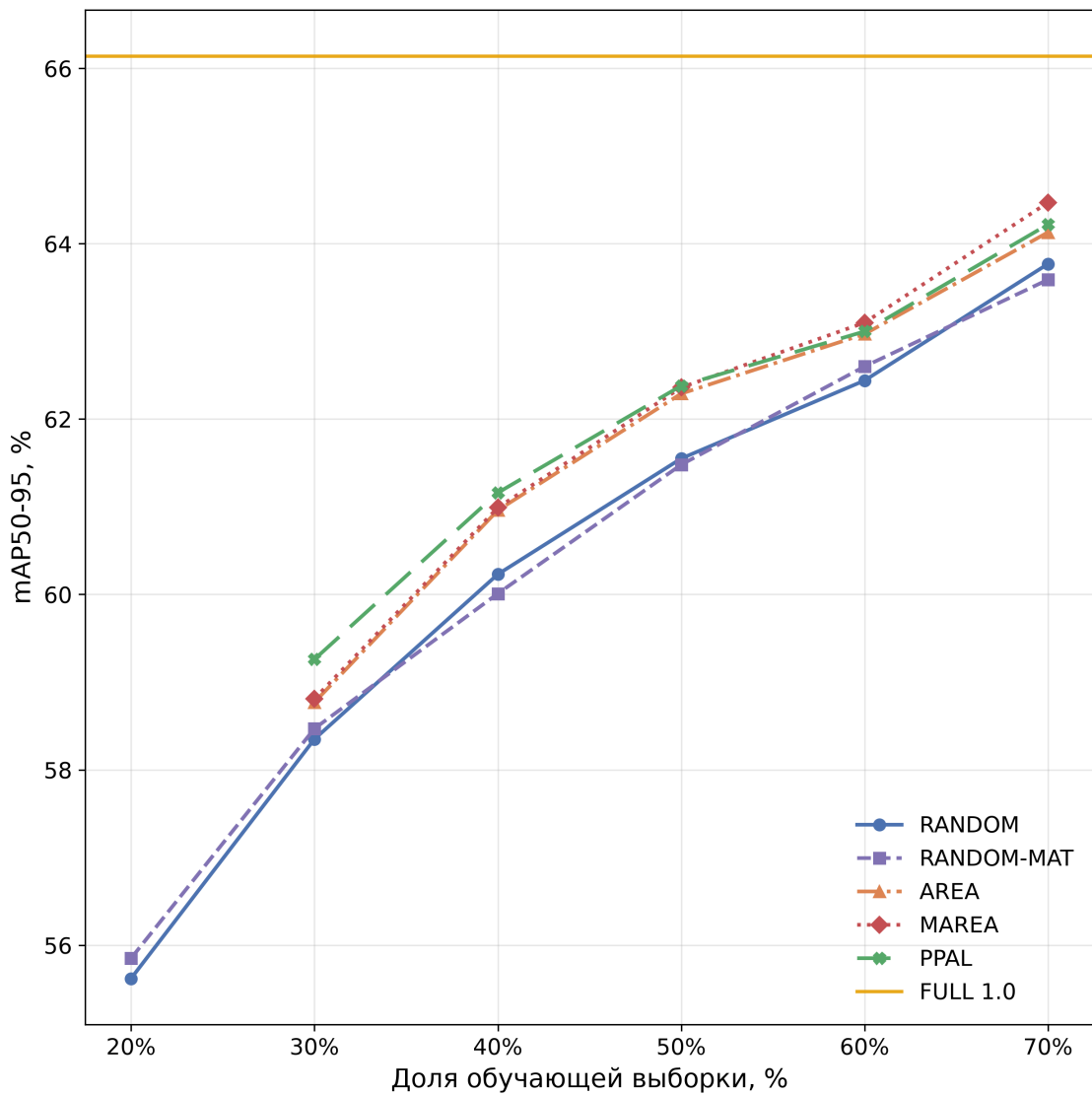


Рисунок 4.1 – Цепочки активного обучения на Pascal VOC [62]. Сплошная горизонтальная линия – качество на полном наборе данных (66.14 %).

–0.06 п.п. на Pascal VOC; для остальных наборов сравнение с PPAL не выполнено. Цепочки активного обучения показаны на рисунке 4.2, численные результаты приведены в таблице 10.

Таблица 10 – Цепочки активного обучения на CrowdHuman (mAP50–95, %) [62]. Полный набор данных: 56.58 %.

Доля	RANDOM	RANDOM-MAT	AREA	MAREA	PPAL
0.2	51.51 ± 0.11	51.45 ± 0.11	–	–	–
0.3	52.82 ± 0.05	52.71 ± 0.08	53.13 ± 0.08	53.42 ± 0.11	52.43 ± 0.09
0.4	53.67 ± 0.06	53.61 ± 0.09	53.99 ± 0.13	54.29 ± 0.17	53.23 ± 0.24
0.5	54.25 ± 0.08	54.21 ± 0.05	54.56 ± 0.05	54.83 ± 0.04	53.83 ± 0.10
0.6	54.83 ± 0.05	54.81 ± 0.06	55.03 ± 0.07	55.26 ± 0.02	54.33 ± 0.15
0.7	55.59 ± 0.03	55.53 ± 0.07	55.72 ± 0.05	55.88 ± 0.11	55.13 ± 0.11

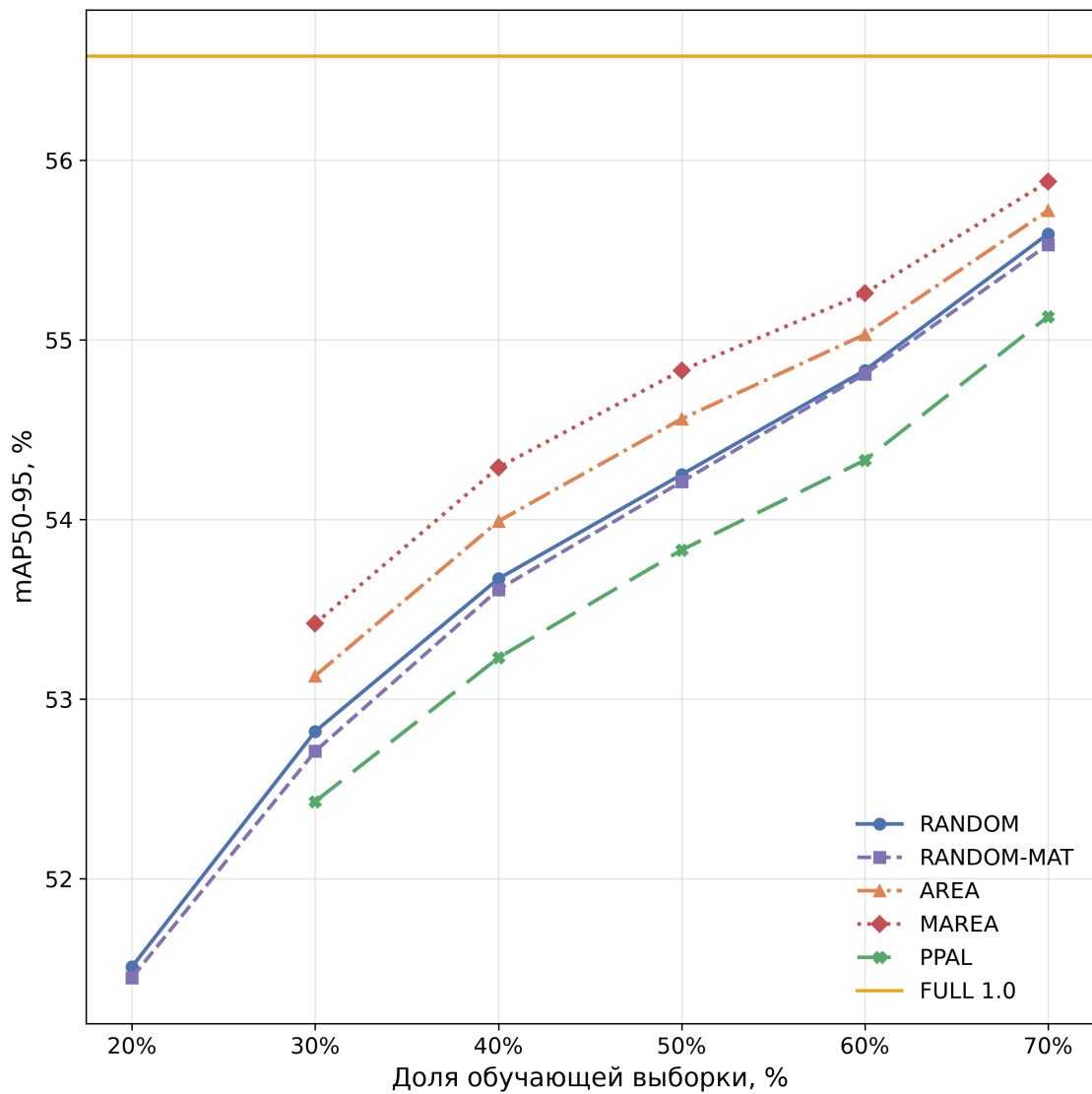


Рисунок 4.2 – Цепочки активного обучения на CrowdHuman [62]. Сплошная горизонтальная линия – качество на полном наборе данных (56.58 %).

COCO 2017 (полный набор данных: 43.61 % mAP50–95). Цепочки активного обучения показаны на рисунке 4.3, численные результаты приведены в таблице 11.

Таблица 11 – Цепочки активного обучения на COCO 2017 (mAP50–95, %) [62]. Полный набор данных: 43.61 %.

Доля	RANDOM	RANDOM-MAT	AREA	MAREA
0.2	36.29 ± 0.23	36.34 ± 0.05	—	—
0.3	38.73 ± 0.13	38.48 ± 0.32	38.70 ± 0.38	39.06 ± 0.16
0.4	40.03 ± 0.28	39.90 ± 0.29	40.10 ± 0.37	40.44 ± 0.14
0.5	41.07 ± 0.10	40.90 ± 0.18	41.15 ± 0.14	41.29 ± 0.34
0.6	41.80 ± 0.27	41.51 ± 0.25	41.91 ± 0.09	42.07 ± 0.27
0.7	42.38 ± 0.11	42.28 ± 0.05	42.63 ± 0.07	42.48 ± 0.18

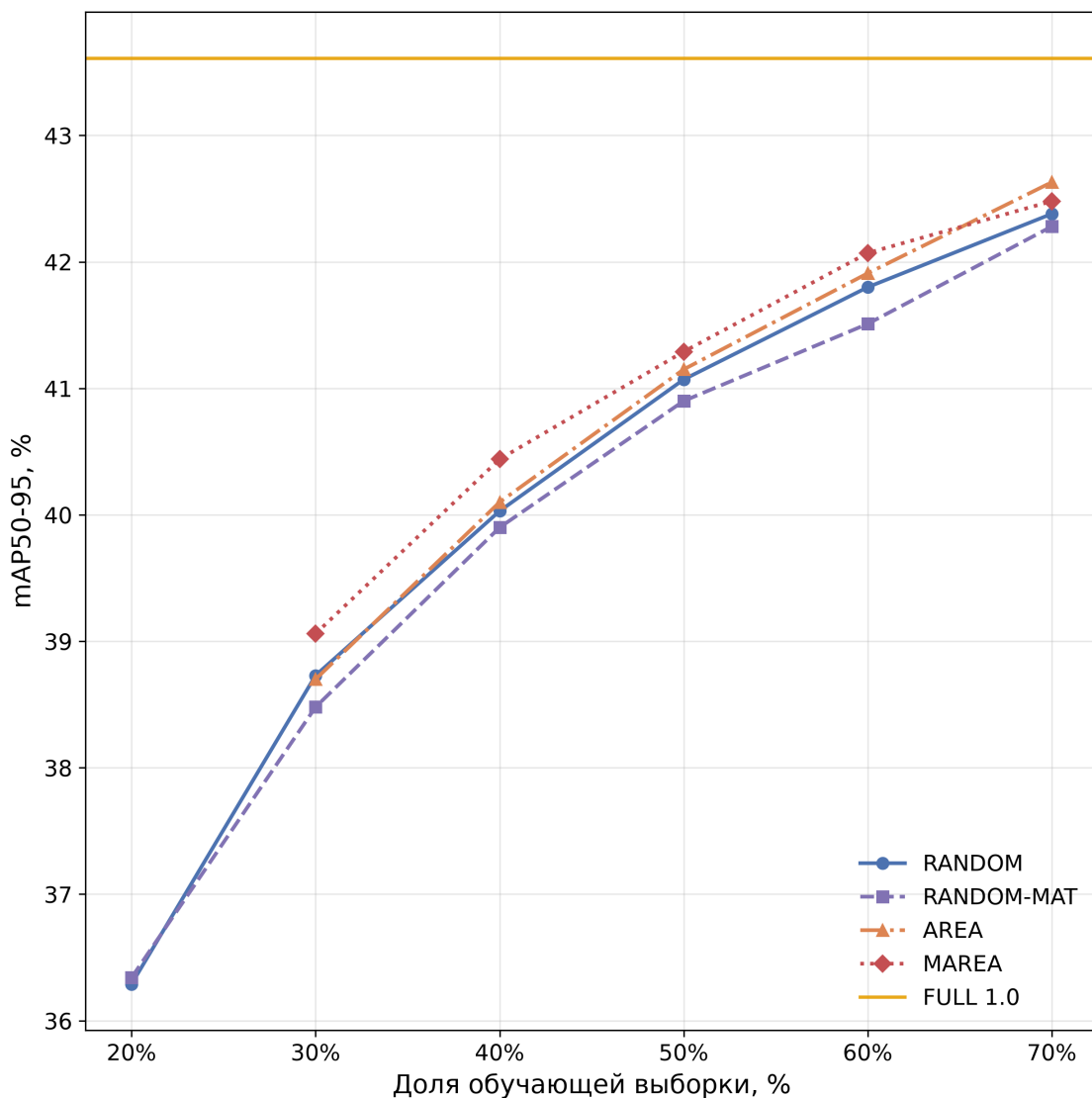


Рисунок 4.3 – Цепочки активного обучения на COCO 2017 [62]. Сплошная горизонтальная линия – качество на полном наборе данных (43.61 %).

На COCO 2017 для MAREA получены значения выше результатов AREA при долях 0.3–0.6 и незначительно ниже при 0.7 (–0.15 п.п.). Средний выигрыш над AREA составляет +0.17 п.п. Сравнение с PPAL не проводилось из-за вычислительных ограничений.

ind_safety (полный набор данных: 66.85 % mAP50–95). Цепочки активного обучения показаны на рисунке 4.4, численные результаты приведены в таблице 12.

На промышленном наборе данных для MAREA наблюдается наибольший абсолютный выигрыш над RANDOM (+0.75 п.п. в среднем); выигрыш над AREA составляет +0.23 п.п. При долях 0.6 и 0.7 средние значения MAREA составляют соответственно $(67.18 \pm 0.41) \%$ и $(66.98 \pm 0.07) \%$ против 66.85 % у модели, обученной на полном наборе данных. Численные разности равны +0.33 п.п. и +0.13 п.п. mAP50–95.

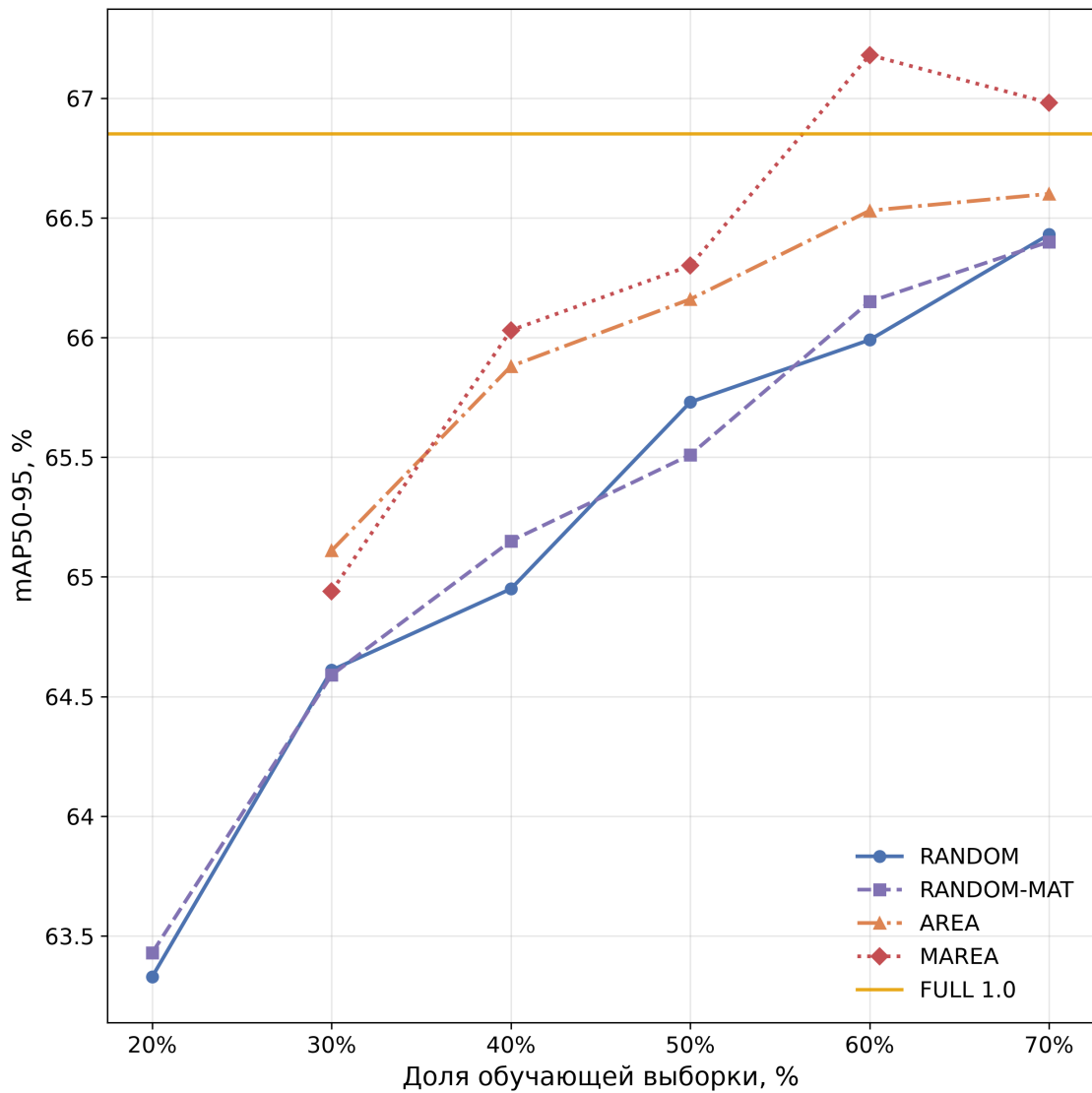


Рисунок 4.4 – Цепочки активного обучения на ind_safety [62]. Сплошная горизонтальная линия – качество на полном наборе данных (66.85 %).

Таблица 12 – Цепочки активного обучения на ind_safety (mAP50–95, %) [62]. Полный набор данных: **66.85 %**.

Доля	RANDOM	RANDOM-MAT	AREA	MAREA
0.2	63.33 ± 0.69	63.43 ± 0.48	–	–
0.3	64.61 ± 0.60	64.59 ± 0.39	65.11 ± 0.18	64.94 ± 0.34
0.4	64.95 ± 0.22	65.15 ± 0.24	65.88 ± 0.14	66.03 ± 0.50
0.5	65.73 ± 0.36	65.51 ± 0.49	66.16 ± 0.35	66.30 ± 0.58
0.6	65.99 ± 0.35	66.15 ± 0.21	66.53 ± 0.64	67.18 ± 0.41
0.7	66.43 ± 0.19	66.40 ± 0.47	66.60 ± 0.36	66.98 ± 0.07

4.4 Сводный анализ и скорость работы

Сводные средние выигрыши MAREА над базовыми методами приведены в таблице 13.

Таблица 13 – Средние выигрыши MAREА над базовыми методами, mAP50–95, п.п., усреднение по долям 0.3–0.7 [62].

Набор данных	MAREА – RANDOM	MAREА – AREA	MAREА – PPAL
Pascal VOC	+0.68	+0.12	–0.06
CrowdHuman	+0.50	+0.25	+0.94
COCO 2017	+0.27	+0.17	–
ind_safety	+0.75	+0.23	–

Средний выигрыш MAREА над AREA положителен на всех четырёх наборах данных. Наиболее существенное улучшение наблюдается на CrowdHuman (+0.25 п.п. над AREA, +0.94 п.п. над PPAL) и ind_safety (+0.23 п.п. над AREA). На Pascal VOC для MAREА получены значения выше PPAL при больших долях разметки, тогда как при малых (0.3–0.5) PPAL остаётся сильным конкурентом.

Скорость. На CrowdHuman полный цикл подготовки разбиения для MAREА составляет 516 секунд против 730 секунд для AREA (быстрее в 1.41 раза за счёт отказа от PCA) и против 1029 секунд для PPAL (быстрее в 1.99 раза) [62]. На COCO 2017 и ind_safety подготовка разбиения для MAREА занимает 2488 секунд против 2712 секунд для AREA (быстрее в 1.09 раза). Ускорение MAREА относительно AREA связано с тем, что PCA-предобработка, занимавшая значительную долю времени в AREA, заменена L2-нормализацией и HNSW-индексом, которые работают быстрее и без потери качества представлений.

4.5 Выводы по главе 4

В главе 4 описан метод MAREА и проведена его экспериментальная проверка на четырёх наборах данных. Основные результаты следующие.

1. В MAREА выходные представления AREA заменяются вложенными эмбедингами из карт переходного модуля на входе выходного модуля детектора, обученного по схеме матрёшки. PCA устраняется: каскадный

- поиск по подвекторам $e_{1/8}$, $e_{1/4}$, e_1 обеспечивает эффективный поиск без дополнительного шага проекции.
2. Для MAREА получены значения выше результатов AREА на всех четырёх наборах данных по среднему mAP50–95 при долях 0.3–0.7. На промышленном наборе данных ind_safety при долях 0.6 и 0.7 средние значения MAREА равны 67.18 % и 66.98 % против 66.85 % для полного набора данных.
 3. На CrowdHuman средние значения mAP50–95 у MAREА выше, чем у PPAL, при всех исследованных долях разметки от 0.3 до 0.7; средняя разность составляет +0.94 п.п.
 4. Сравнение RANDOM и RANDOM-MAT показывает, что прирост MAREА обусловлен критерием отбора, а не самой модификацией детектора по схеме матрёшки. Это изолирует вклад вложенных эмбедингов как источника дополнительной информации для активного обучения.
 5. MAREА быстрее AREА в 1.1–1.4 раза и быстрее PPAL в 2 раза по полному времени подготовки одного шага разбиения.

4.6 Ограничения и направления дальнейших исследований

Ограничения метода. В MAREА, как и в AREА, используется жадная стратегия пакетного отбора: каждое изображение оценивается независимо, поэтому в один пакет могут попасть несколько кадров с объектами из одной визуальной подгруппы. Такой пакет богат относительно уже размеченного множества, но избыточен внутри себя. Устранение этой избыточности потребовало бы учёта попарного сходства кандидатов, что существенно увеличило бы вычислительную стоимость, как это происходит при использовании PPAL. Эксперименты показали, что выбор в пользу скорости оправдан, поскольку метода MAREА сопоставим или превосходит метод PPAL в качестве работы.

На COCO 2017 при доле 0.7 AREА незначительно превосходит MAREА (+0.15 п.п.). Это указывает, что на крупных и разнородных наборах данных с 80 классами разница в источнике эмбедингов менее выражена, чем на специализированных доменах.

Для метода требуется детектор, обученный по схеме матрёшки: его развёртывание предполагает дополнительный шаг переобучения по сравнению со стандартной YOLOv8. При этом изменение не предполагает изменение количества параметров или архитектурные изменения, а остается легковесной модификаций способа прохождения данных через слои модели во время обучения.

Направления дальнейших исследований. Перспективным развитием MAREА является гибридный двухступенчатый алгоритм, использующий несогласованность вложенных эмбедингов разной размерности как дополнительный сигнал неопределённости [62]. Идея состоит в следующем: если объект хорошо известен модели, его представления разной размерности ($e_{1/8}$, $e_{1/4}$, e_1) указывают на одних и тех же ближайших соседей в размеченном множестве. Если же объект редок или визуально нов, подвекторы разной длины дают несогласованные ответы – этот сигнал несогласованности может служить мерой неопределённости, дополняющей критерий новизны. Такое расширение объединило бы преимущества подходов на основе неопределённости и разнообразия без привлечения дополнительных моделей или ансамблей.

Другим перспективным направлением является перенос MAREА на другие детекторы с многомасштабными картами признаков, в частности DINO и RT-DETR. Принцип извлечения вложенных эмбедингов через RoI Align не специфичен для YOLOv8 и применим к любой архитектуре, предоставляющей доступ к промежуточным картам признаков.

Заключение

В диссертации исследована задача активного обучения нейросетевых детекторов объектов для видеоаналитики. Разработаны методы отбора данных на основе локальных эмбедингов объектов и архитектурная модификация детектора для формирования вложенных представлений.

Проведённое исследование позволило получить следующие основные результаты.

1. Проведён анализ методов активного обучения и 36 архитектур детекторов. Сформулированы четыре требования к критериям отбора данных для промышленной видеоаналитики: работа на уровне объектов, различение внутриклассовых визуальных подгрупп, использование мультимасштабных карт признаков и оценка новизны относительно размеченного множества. Показано, что рассмотренные методы не удовлетворяют всем четырём требованиям одновременно.
2. Предложен метод AREA, использующий локальные эмбединги объектов, RoI Align и PCA для отбора изображений по новизне относительно размеченного множества. На четырёх наборах данных максимальные выигрыши AREA над случайным отбором по mAP50–95 составляют от 0.25 до 0.93 п.п. (таблицы 2–5).
3. Предложена модификация выходного модуля YOLOv8 по принципу Matryoshka Representation Learning, формирующая вложенные эмбединги без дополнительных обучаемых параметров. Разработаны механизмы стабилизации многорежимного обучения и операция переплетения признаков, сохраняющая информацию разных пространственных масштабов в коротких подвекторах. При использовании одной восьмой входных каналов выходного модуля на ind_safety получено mAP50, равное 92.4 %, против 15.3 % при наивном усечении стандартной модели (+77.1 п.п.; таблица 7). На Pascal VOC каскадный поиск сокращает время запроса с 20.5 до 6.7 мс, то есть в 3.1 раза, при Recall@10 96.95 % против 96.92 % у полного поиска (таблица 8).
4. Предложен метод MAREA, использующий вложенные эмбединги из карт признаков переходного модуля YOLOv8 и каскадный HNSW-поиск без PCA-предобработки. На ind_safety при использовании 60 % обучаю-

щих изображений получено mAP50–95, равное 67.18 %, против 66.85 % при обучении на полном наборе данных: качество выше на 0.33 п.п. при сокращении числа размеченных изображений на 40 % (таблица 12). На четырёх наборах данных максимальные выигрыши MAREА по mAP50–95 составляют от 0.41 до 1.19 п.п. над случайным отбором и от 0.30 до 0.65 п.п. над AREА. При пяти долях разметки от 0.3 до 0.7 MAREА превосходит случайный отбор во всех 20 сравнениях, а AREА — в 18 из 20 (таблицы 9–12). На CrowdHuman MAREА превосходит стадию отбора по разнообразию PPAL при всех пяти долях разметки; максимальный выигрыш составляет +1.06 п.п. при доле 0.4 (54.29 % против 53.23 %). На CrowdHuman подготовка разбиения занимает 516 секунд против 730 у AREА и 1029 у стадии отбора по разнообразию PPAL: ускорение в 1.41 и 1.99 раза соответственно.

Практическая значимость результатов состоит в повышении эффективности подготовки обучающих данных при ограниченном бюджете разметки. Вложенные эмбединги также пригодны для поиска похожих объектов и адаптации модели к требованиям по скорости и памяти без переобучения.

Дальнейшие исследования связаны с объединением новизны и несогласованности вложенных представлений в гибридном критерии отбора, а также с переносом MAREА на другие детекторы с многомасштабными картами признаков.

Список сокращений и условных обозначений

AL	Active Learning, активное обучение
ANN	Approximate Nearest Neighbor, приближённый поиск ближайших соседей
AP	Average Precision, средняя точность
AREA	Active learning with Region Embedding Alignment
BatchNorm	Batch Normalization, пакетная нормализация
BiFPN	Bidirectional Feature Pyramid Network
CI	Confidence Interval, доверительный интервал
COCO	Common Objects in Context
DCN	Deformable Convolutional Network
DETR	Detection Transformer
DINO	DETR with Improved deNoising anchor boxes
DLA	Deep Layer Aggregation
FCOS	Fully Convolutional One-Stage Object Detection
FFN	Feed-Forward Network, сеть прямого распространения
FPN	Feature Pyramid Network, пирамида признаков
GFL	Generalized Focal Loss
HNSW	Hierarchical Navigable Small World graphs
IoU	Intersection over Union
mAP	mean Average Precision
MAREA	Matryoshka Active learning with Region Embedding Alignment
MRL	Matryoshka Representation Learning
NMS	Non-Maximum Suppression
OIV7	Open Images V7, версия 7 набора данных Open Images
PANet	Path Aggregation Network
PCA	Principal Component Analysis, метод главных компонент
PPAL	Plug and Play Active Learning
R-CNN	Region-based Convolutional Neural Network
RoI Align	Region of Interest Align
RPN	Region Proposal Network
RT-DETR	Real-Time Detection Transformer

SSD	Single Shot MultiBox Detector
VOC	Visual Object Classes
YOLO	You Only Look Once

Словарь терминов

активное обучение (Active Learning, AL): Методология машинного обучения, в которой наиболее информативные примеры итеративно выбираются для разметки из пула неразмеченных данных.

алгоритм k -Center-Greedy: Жадный алгоритм выбора центров, на каждом шаге добавляющий объект, наиболее удалённый от уже выбранного множества. В активном обучении используется для покрытия пространства признаков.

алгоритм k -means++: Стохастическая процедура инициализации центров кластеров, в которой вероятность выбора точки пропорциональна квадрату расстояния до ближайшего уже выбранного центра.

архитектура DLA (Deep Layer Aggregation): Архитектура основного модуля, агрегирующая признаки разных глубин через иерархическую цепочку соединений. Обеспечивает многомасштабное слияние внутри основного модуля без явного переходного модуля FPN.

билинейная интерполяция: Метод интерполяции на прямоугольной сетке, вычисляющий значение в произвольной точке как взвешенную сумму значений в четырёх ближайших узлах.

блок CSP (Cross Stage Partial network): Блок основного или переходного модуля, разделяющий входной тензор на две части: одна часть проходит через серию свёрток, другая – через прямое соединение, после чего обе конкатенируются. Снижает число параметров и объём вычислений без потери точности.

вложенное представление (nested embedding): Векторное представление, первые m координат которого образуют самостоятельное, информативное описание объекта меньшей размерности.

выходной модуль детектора: Модуль нейросетевого детектора, принимающий карты признаков и формирующий предсказания ограничивающих прямоугольников и вероятностей классов.

двухстадийный детектор (two-stage detector): Детектор, в котором сначала формируются области-кандидаты, а затем для каждой такой области уточняются класс и положение объекта. К этому семейству относятся Faster R-CNN и Mask R-CNN.

детектор по ключевым точкам (keypoint-based detector): Семейство детекторов (CornerNet, ExtremeNet, CenterNet), в котором объекты представлены харак-

терными точками (углами, центрами, крайними точками), а не якорями. Основной модуль – Hourglass-сеть или DLA, выход – тепловые карты ключевых точек.

детекция объектов: Задача совместной классификации и локализации всех объектов на изображении путём предсказания ограничивающих прямоугольников с метками классов.

деформируемое внимание (Deformable Attention): Механизм внимания в трансформерах с обучаемыми смещениями точек выборки, применяемый в Deformable DETR и его производных. Позволяет эффективно обрабатывать мультимасштабные признаки за счёт ограниченного числа опорных точек.

деформируемые свёртки (Deformable Convolutions, DCN): Свёртки с обучаемыми смещениями позиций ядра, позволяющие адаптировать поле восприятия к форме объекта. Используются в InternImage и Deformable DETR.

дилатированная свёртка (Dilated / Atrous Convolution): Свёртка с вставленными нулями между элементами ядра. Коэффициент дилатации d определяет шаг между элементами ядра, что увеличивает поле восприятия без роста числа параметров. Используется в YOLOF для захвата объектов разных размеров на единственном уровне признаков.

доверительный интервал (confidence interval): Диапазон значений, в котором с заданной вероятностью (в данной работе – 95 %) находится оцениваемый параметр.

иерархический граф HNSW (Hierarchical Navigable Small World graphs): Структура данных для приближённого поиска ближайших соседей, обеспечивающая поиск за логарифмическое время [41].

карта признаков (feature map): Тензор промежуточных представлений нейросети с пространственными размерами и каналами признаков. В детекторах карты признаков разных уровней используются для обработки объектов разных масштабов.

каскадный поиск (coarse-to-fine search): Многоступенчатая стратегия поиска ближайших соседей: грубый поиск по короткому подвектору отбирает кандидатов, финальный отбор выполняется по полному вектору.

косинусное расстояние: Мера различия направлений векторов, равная $1 - \cos \theta$. Инвариантна к масштабу вектора; широко используется для сравнения эмбеддингов.

критерий неопределённости (uncertainty-based criterion): Критерий активного обучения, отдающий приоритет примерам, для которых текущая модель даёт наименее уверенное или наиболее неоднозначное предсказание.

критерий по плотности: Критерий активного обучения, оценивающий разреженность окрестности кандидата через сумму обратных квадратов расстояний до k ближайших аннотированных объектов.

критерий по расстоянию: Критерий активного обучения, отдающий приоритет изображениям, объекты которых наиболее удалены от ближайшего объекта в уже размеченном множестве.

критерий покрытия CoreSet: Семейство критериев активного обучения, в которых выбираемое подмножество должно покрывать пул так, чтобы максимальное расстояние от любой точки до ближайшего выбранного или уже размеченного объекта было минимальным.

критерий разнообразия (diversity-based criterion): Критерий активного обучения, ориентированный на покрытие пространства признаков и снижение избыточности выбираемых примеров.

локальный эмбединг объекта: Векторное представление, извлечённое из карт признаков детектора в области ограничивающего прямоугольника объекта с помощью операции RoI Align.

масштабируемое представление объекта: Эмбединг объекта, который можно использовать на разных уровнях детализации. В данной работе масштабируемость обеспечивается вложенной структурой вектора и переплетением признаков с нескольких уровней пирамиды, поэтому короткие подвекторы сохраняют информацию о разных пространственных масштабах объекта.

мера качества mAP (mean Average Precision, средняя точность): Средняя точность по всем классам и порогам IoU. Для детекции при фиксированном пороге IoU τ средняя точность класса c задаётся через кривую точности–полноты, а итоговая мера качества mAP50–95 вычисляется как

$$AP_c(\tau) = \int_0^1 p_c(r, \tau) dr, \quad mAP_{50-95} = \frac{1}{10} \sum_{\tau \in \{0.50, 0.55, \dots, 0.95\}} \frac{1}{C} \sum_{c=1}^C AP_c(\tau).$$

В данной работе используется mAP50–95 – среднее по порогам IoU от 0.50 до 0.95 с шагом 0.05.

мера качества поиска Recall@K (полнота): Доля запросов, для которых хотя бы один релевантный объект попал в первые K результатов:

$$\text{Recall@K} = \frac{1}{|Q|} \sum_{q \in Q} \mathbb{I}[r(q) \leq K],$$

где Q – множество запросов, а $r(q)$ – ранг первого релевантного объекта для запроса q . В задачах поиска с единственным релевантным объектом на запрос это выражение сводится к индикатору попадания объекта в первые K результатов.

метод активного обучения PPAL (Plug and Play Active Learning): Метод активного обучения для детекции объектов, сочетающий отбор по неопределённости и попарное разнообразие кандидатов через k -Center-Greedy и k -means++ [74].

метод главных компонент (Principal Component Analysis, PCA): Метод снижения размерности, проецирующий данные на направления наибольшей дисперсии [36].

мультимасштабная архитектура детектора: Архитектура, в которой предсказания объектов формируются на нескольких картах признаков разного пространственного разрешения. Это позволяет одной модели обрабатывать как мелкие, так и крупные объекты.

обучение вложенных представлений (Matryoshka Representation Learning, MRL): Метод обучения векторных представлений, при котором первые m координат вектора образуют самостоятельное представление меньшей размерности [47].

общее назначение (shared assignment): Приём стабилизации обучения по схеме матрёшки: маска положительных позиций вычисляется один раз по режиму полной ширины и используется для всех режимов.

ограничивающий прямоугольник (bounding box): Прямоугольная рамка, задающая положение объекта на изображении координатами двух углов или эквивалентным набором параметров.

одностадийный детектор (single-stage detector): Детектор, в котором классификация и локализация объектов выполняются за один прямой проход без отдельного этапа формирования областей-кандидатов. К этому семейству относятся YOLO, SSD и RetinaNet.

операция RoI Align (Region of Interest Align): Операция извлечения признаков из прямоугольной области карты признаков с помощью билинейной интерполяции, без округления координат [43].

основной модуль: Основная часть нейросетевого детектора, в которой последовательно применяются свёрточные или трансформерные блоки и формируется иерархия карт признаков убывающего пространственного разрешения.

пакетная нормализация (Batch Normalization, BatchNorm): Слой нормализации активаций по статистикам мини-пакета. В данной работе отдельно рассматривается поведение скользящих статистик BatchNorm при обучении нескольких вариантов ширины выходного модуля.

пакетное активное обучение с пулом (pool-based batch active learning): Постановка активного обучения, в которой все неразмеченные данные доступны заранее, а на каждом раунде для разметки выбирается пакет примеров.

переплетение (interleaving): Операция формирования вложенного эмбединга объекта из представлений нескольких уровней пирамиды признаков, при которой блоки каждого уровня чередуются так, что любой подвектор вектора содержит долю признаков со всех уровней.

пересечение над объединением (Intersection over Union, IoU): Мера совпадения двух прямоугольников: отношение площади их пересечения к площади их объединения. Используется для оценки точности локализации и в процедуре назначения предсказаний объектам.

переходный модуль: Модуль, объединяющий карты признаков нескольких уровней основного модуля для формирования мультимасштабных представлений, передаваемых выходному модулю.

пирамида BiFPN (Bidirectional Feature Pyramid Network): Двухнаправленная взвешенная пирамида признаков с обучаемыми весами слияния на каждом уровне; используется в EfficientDet. Выполняется несколько раз подряд, обеспечивая плотный обмен информацией между уровнями.

пирамида PANet (Path Aggregation Network): Двухнаправленная пирамида признаков, добавляющая к нисходящему проходу FPN восходящий путь снизу вверх. Такая схема объединения признаков используется, в частности, в YOLOv8.

пирамида признаков (Feature Pyramid Network, FPN): Архитектурный модуль детектора, объединяющий карты признаков нескольких уровней через нисходящий проход с боковыми соединениями для одновременной детекции объектов разных масштабов [28].

подавление немаксимумов (Non-Maximum Suppression, NMS): Постобработка предсказаний детектора: из группы перекрывающихся рамок для одного объекта оставляется та с наибольшей уверенностью, остальные удаляются.

подвектор эмбединга: Начальная часть вектора признаков. В схеме Matryoshka Representation Learning подвекторы разных длин рассматриваются как самостоятельные вложенные представления объекта.

пул (unlabeled pool): Множество доступных неразмеченных изображений, из которого алгоритм активного обучения выбирает кандидатов для разметки.

разметка: Процесс аннотирования изображений: для задачи детекции требуется построения ограничивающих прямоугольников и указания класса каждого объекта.

режим полной ширины: Режим работы модифицированного по схеме матрёшки выходного модуля детектора, в котором используются все каналы представления и сохраняется поведение стандартной YOLOv8.

семейство вложенных эмбедингов: Упорядоченный набор векторов одного объекта разной длины, в котором каждый более короткий является согласованным подвектором следующего.

сеть формирования областей-кандидатов (Region Proposal Network, RPN): Подсеть двухстадийного детектора, которая по заранее заданным якорям формирует области-кандидаты, предположительно содержащие объекты.

случайный отбор (Random baseline): Равномерный выбор изображений из пула без учёта состояния модели. Используется как нижний ориентир при оценке эффективности активного обучения.

трансформер (Transformer): Архитектура нейронной сети, основанная на механизме само-внимания (self-attention), позволяющем каждому элементу последовательности взаимодействовать со всеми остальными. В детекции применяется в семействе DETR.

шаг сетки: Расстояние между положениями соседних узлов карты признаков, выраженное в пикселях входного изображения. При шаге s переход к соседнему узлу по горизонтали или вертикали соответствует смещению на s пикселей по соответствующей оси входного изображения. Шаг сетки не определяет размер поля восприятия узла.

якорь (anchor): Прямоугольник фиксированного масштаба и соотношения сторон, используемый в качестве опорного предсказания в детекторах с якорями. Детекторы без якорей (anchor-free) предсказывают объекты напрямую без привязки к заранее заданным формам.

Список литературы

1. owl.Base : свидетельство о государственной регистрации программы для ЭВМ № 2023689395 / М. В. Лазуков, Г. Е. Макаров, С. А. Марков, А. Д. Мершавка, Ю. А. Прохоров, В. А. Соколов, Л. М. Тепляков, А. А. Федоров, У. С. Цинпаев, И. А. Чайкин, Е. А. Шве́ц, А. В. Шошин ; Общество с ограниченной ответственностью «НОРД ВИЖЕН ИНТЕЛЛЕДЖЕНС РЕСЕРЧ». – № 2023689496 ; заявл. 29.12.2023 ; зарегистрировано в реестре программ для ЭВМ 29.12.2023.
2. owl.Drill : свидетельство о государственной регистрации программы для ЭВМ № 2023689397 / М. В. Лазуков, С. А. Марков, А. Д. Мершавка, Ю. А. Прохоров, В. А. Соколов, Л. М. Тепляков, А. А. Федоров, А. В. Фомин, И. А. Чайкин, Е. А. Шве́ц, А. В. Шошин, Д. И. Юрманов ; Общество с ограниченной ответственностью «НОРД ВИЖЕН ИНТЕЛЛЕДЖЕНС РЕСЕРЧ». – № 2023689498 ; заявл. 29.12.2023 ; зарегистрировано в реестре программ для ЭВМ 29.12.2023.
3. owl.Guard: версия 6.0 : свидетельство о государственной регистрации программы для ЭВМ № 2024617256 / М. В. Вербицкий, П. Г. Гуляев, Я. С. Гурьев, К. А. Ионкин, М. В. Лазуков, С. И. Литвинов, Г. Е. Макаров, С. А. Марков, А. Д. Мершавка, О. В. Пинарин, Ю. А. Прохоров, В. А. Соколов, Л. М. Тепляков, А. А. Федоров, А. В. Фомин, У. С. Цинпаев, Е. А. Шве́ц, А. В. Шошин, Д. И. Юрманов ; Общество с ограниченной ответственностью «НОРД ВИЖЕН ИНТЕЛЛЕДЖЕНС РЕСЕРЧ». – № 2024616277 ; заявл. 29.03.2024 ; зарегистрировано в реестре программ для ЭВМ 29.03.2024.
4. *Матвеев, И. А.* Модификация нейросетевого детектора для получения вложенных эмбе́ддингов объектов / И. А. Матвеев, В. А. Соколов // Известия Российской академии наук. Теория и системы управления. – 2026. – № 4. – С. 78–87. – URL: <https://doi.org/10.7868/S3034644426040076>.
5. *Соколов, В. А.* Использование эмбе́ддингов-матрёшек в критерии отбора для активного обучения свёрточных сетей / В. А. Соколов, И. А. Матвеев // Математические методы распознавания образов: тезисы докладов 22-й Всероссийской конференции. – Муром, Россия, 2025. – С. 83–85. – 22–26 сентября 2025 г.

6. *Соколов, В. А.* Выделение эмбедингов для активного обучения, соответствующих регионам интереса на изображении / В. А. Соколов, И. А. Матвеев, Л. М. Тепляков // 15-я Международная конференция «Интеллектуализация обработки информации» (ИОИ-2024): тезисы докладов. – Гродно, Беларусь, 2024. – С. 45. – URL: <https://mmro.ru/wp-content/uploads/2025/01/idp-2024.pdf>.
7. *Соколов, В. А.* Критерий активного обучения на основе анализа локальных эмбедингов / В. А. Соколов, И. А. Матвеев, Л. М. Тепляков // 67-я Всероссийская научная конференция Московского физико-технического института: тезисы докладов. Прикладная математика и информатика. – Долгопрудный, Россия, 2025. – С. 127–129. – URL: https://old.mipt.ru/upload/medialibrary/1e0/konferentsiya_5_fpmi.pdf.
8. 2D Matryoshka Sentence Embeddings / X. Li [et al.] // arXiv preprint arXiv:2402.14776. – 2024. – URL: <https://arxiv.org/abs/2402.14776>.
9. A Comparative Survey of Deep Active Learning / X. Zhan [et al.] // arXiv preprint arXiv:2203.13450. – 2022.
10. A Survey of Deep Active Learning / P. Ren [et al.] // ACM Computing Surveys. – 2021. – Vol. 54, no. 9. – P. 1–40.
11. A Survey on Deep Active Learning: Recent Advances and New Frontiers / D. Li [et al.] // IEEE Transactions on Neural Networks and Learning Systems. – 2025. – Vol. 36, no. 4. – P. 5879–5899. – URL: <https://ieeexplore.ieee.org/document/10537213/>.
12. *Aggarwal, C. C.* On the Surprising Behavior of Distance Metrics in High Dimensional Space / C. C. Aggarwal, A. Hinneburg, D. A. Keim // Proc. 8th Intern. Conf. Database Theory. – London, UK, 2001. – P. 420–434.
13. Anchor DETR: Query Design for Transformer-Based Object Detection / Y. Wang [et al.] // Proc. AAAI Conf. Artificial Intelligence. Vol. 36. – 2022. – P. 2567–2575.
14. *Arthur, D.* *k*-means++: The Advantages of Careful Seeding / D. Arthur, S. Vassilvitskii // Proc. 18th Annual ACM-SIAM Symp. Discrete Algorithms. – New Orleans, USA, 2007. – P. 1027–1035.

15. *Auer, P.* Finite-Time Analysis of the Multiarmed Bandit Problem / P. Auer, N. Cesa-Bianchi, P. Fischer // *Machine Learning*. – 2002. – Vol. 47, no. 2/3. – P. 235–256.
16. *Bengio, Y.* Representation Learning: A Review and New Perspectives / Y. Bengio, A. Courville, P. Vincent // *IEEE Trans. Pattern Analysis and Machine Intelligence*. – 2013. – Vol. 35, no. 8. – P. 1798–1828.
17. *Cai, Z.* Cascade R-CNN: Delving into High Quality Object Detection / Z. Cai, N. Vasconcelos // *Proc. IEEE Conf. Computer Vision and Pattern Recognition*. – Salt Lake City, USA, 2018. – P. 6154–6162.
18. Conditional DETR for Fast Training Convergence / D. Meng [et al.] // *Proc. IEEE/CVF Intern. Conf. Computer Vision*. – Montreal, Canada, 2021. – P. 3651–3660.
19. Consistency-Based Active Learning for Object Detection / W. Yu [et al.] // *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition Workshops*. – New Orleans, USA, 2022. – P. 3951–3960.
20. CrowdHuman: A Benchmark for Detecting Human in a Crowd / S. Shao [et al.] // *arXiv preprint arXiv:1805.00123*. – 2018.
21. Deformable DETR: Deformable Transformers for End-to-End Object Detection / X. Zhu [et al.] // *Proc. Intern. Conf. Learning Representations*. – Virtual, 2021.
22. DETRs Beat YOLOs on Real-Time Object Detection / W. Lv [et al.] // *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition*. – Seattle, USA, 2024. – P. 16965–16974.
23. DINO: DETR with Improved DeNoising Anchor Boxes for End-to-End Object Detection / H. Zhang [et al.] // *Proc. Intern. Conf. Learning Representations*. – Kigali, Rwanda, 2023.
24. Dynamic Slimmable Network / C. Li [et al.] // *Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition*. – 2021. – P. 8607–8617.
25. End-to-End Object Detection with Transformers / N. Carion [et al.] // *Proc. European Conf. Computer Vision*. – Glasgow, UK, 2020. – P. 213–229.
26. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks / S. Ren [et al.] // *Advances in Neural Information Processing Systems*. Vol. 28. – 2015. – P. 91–99.

27. FCOS: Fully Convolutional One-Stage Object Detection / Z. Tian [et al.] // Proc. IEEE/CVF Intern. Conf. Computer Vision. – Seoul, Korea, 2019. – P. 9627–9636.
28. Feature Pyramid Networks for Object Detection / T.-Y. Lin [et al.] // Proc. IEEE Conf. Computer Vision and Pattern Recognition. – Honolulu, USA, 2017. – P. 2117–2125.
29. Focal Loss for Dense Object Detection / T.-Y. Lin [et al.] // Proc. IEEE Intern. Conf. Computer Vision. – Venice, Italy, 2017. – P. 2980–2988.
30. *Gal, Y.* Deep Bayesian Active Learning with Image Data / Y. Gal, R. Islam, Z. Ghahramani // Proc. 34th Intern. Conf. Machine Learning. Vol. 70. – Sydney, Australia, 2017. – P. 1183–1192.
31. Generalized Focal Loss: Learning Qualified and Distributed Bounding Boxes for Dense Object Detection / X. Li [et al.] // Advances in Neural Information Processing Systems. Vol. 33. – 2020. – P. 21002–21012.
32. *Girshick, R.* Fast R-CNN / R. Girshick // Proc. IEEE Intern. Conf. Computer Vision. – 2015. – P. 1440–1448. – URL: https://openaccess.thecvf.com/content_iccv_2015/papers/Girshick_Fast_R-CNN_ICCV_2015_paper.pdf.
33. InternImage: Exploring Large-Scale Vision Foundation Models with Deformable Convolutions / W. Wang [et al.] // Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition. – Vancouver, Canada, 2023. – P. 14408–14419.
34. *Jocher, G.* Ultralytics YOLOv8 / G. Jocher, A. Chaurasia, J. Qiu. – 2023. – URL: <https://github.com/ultralytics/ultralytics> (дата обращения: 04.06.2026).
35. *Jocher, G.* Ultralytics YOLO11 / G. Jocher, J. Qiu. – 2024. – URL: <https://github.com/ultralytics/ultralytics> (дата обращения: 04.06.2026).
36. *Jolliffe, I. T.* Principal Component Analysis / I. T. Jolliffe. – 2nd ed. – New York : Springer, 2002. – 487 p.
37. *Joshi, A. J.* Multi-Class Active Learning for Image Classification / A. J. Joshi, F. Porikli, N. Papanikolopoulos // Proc. IEEE Conf. Computer Vision and Pattern Recognition. – Miami, USA, 2009. – P. 2372–2379.
38. *Law, H.* CornerNet: Detecting Objects as Paired Keypoints / H. Law, J. Deng // Proc. European Conf. Computer Vision. – Munich, Germany, 2018. – P. 765–781.
39. *LeCun, Y.* Deep Learning / Y. LeCun, Y. Bengio, G. Hinton // Nature. – 2015. – Vol. 521, no. 7553. – P. 436–444.

40. *Lewis, D. D.* A Sequential Algorithm for Training Text Classifiers / D. D. Lewis, W. A. Gale // Proc. 17th Annual Intern. ACM SIGIR Conf. Research and Development in Information Retrieval. – Dublin, Ireland, 1994. – P. 3–12.
41. *Malkov, Y. A.* Efficient and Robust Approximate Nearest Neighbor Search Using Hierarchical Navigable Small World Graphs / Y. A. Malkov, D. A. Yashunin // IEEE Trans. Pattern Analysis and Machine Intelligence. – 2020. – Vol. 42, no. 4. – P. 824–836.
42. *Manning, C. D.* Introduction to Information Retrieval / C. D. Manning, P. Raghavan, H. Schütze. – Cambridge : Cambridge University Press, 2008. – 482 p. – URL: <https://nlp.stanford.edu/IR-book/>.
43. Mask R-CNN / K. He [et al.] // Proc. IEEE Intern. Conf. Computer Vision. – Venice, Italy, 2017. – P. 2961–2969.
44. MatFormer: Nested Transformer for Elastic Inference / Devvrit [et al.] // Advances in Neural Information Processing Systems. Vol. 37. – 2024. – P. 140535–140564.
45. Matryoshka Multimodal Models / M. Cai [et al.] // Proc. Intern. Conf. Learning Representations. – 2025. – URL: <https://openreview.net/forum?id=Uhj5OxAz7I>.
46. Matryoshka Query Transformer for Large Vision-Language Models / W. Hu [et al.] // Advances in Neural Information Processing Systems. Vol. 37. – 2024.
47. Matryoshka Representation Learning / A. Kusupati [et al.] // Advances in Neural Information Processing Systems. Vol. 35. – 2022. – P. 30233–30249.
48. Microsoft COCO: Common Objects in Context / T.-Y. Lin [et al.] // Proc. European Conf. Computer Vision. – Zurich, Switzerland, 2014. – P. 740–755.
49. Multiple Instance Active Learning for Object Detection / T. Yuan [et al.] // Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition. – Nashville, USA, 2021. – P. 5330–5339.
50. *Nguyen, H. T.* Active Learning Using Pre-Clustering / H. T. Nguyen, A. Smeulders // Proc. 21st Intern. Conf. Machine Learning. – Banff, Canada, 2004. – P. 79–86.
51. Object Detection in 20 Years: A Survey / Z. Zou [et al.] // Proceedings of the IEEE. – 2023. – Vol. 111, no. 3. – P. 257–276.
52. PP-YOLOE: An Evolved Version of YOLO / S. Xu [et al.] // arXiv preprint arXiv:2203.16250. – 2022.

53. *Redmon, J.* YOLO9000: Better, Faster, Stronger / J. Redmon, A. Farhadi // Proc. IEEE Conf. Computer Vision and Pattern Recognition. – Honolulu, USA, 2017. – P. 7263–7271.
54. *Redmon, J.* YOLOv3: An Incremental Improvement / J. Redmon, A. Farhadi // arXiv preprint arXiv:1804.02767. – 2018.
55. RT-DETRv2: Improved Baseline with Bag-of-Freebies for Real-Time Detection Transformer / W. Lv [et al.] // arXiv preprint arXiv:2407.17140. – 2024.
56. *Sener, O.* Active Learning for Convolutional Neural Networks: A Core-Set Approach / O. Sener, S. Savarese // Proc. 6th Intern. Conf. Learning Representations. – Vancouver, Canada, 2018. – URL: <https://openreview.net/forum?id=H1aIuk-RW>.
57. *Settles, B.* Active Learning Literature Survey : tech. rep. / B. Settles ; University of Wisconsin–Madison. – 2009. – P. 1–67. – No. 1648.
58. *Seung, H. S.* Query by Committee / H. S. Seung, M. Oppor, H. Sompolinsky // Proc. 5th Annual Workshop Computational Learning Theory. – Pittsburgh, USA, 1992. – P. 287–294.
59. *Shannon, C. E.* A Mathematical Theory of Communication / C. E. Shannon // Bell System Technical Journal. – 1948. – Vol. 27, no. 3. – P. 379–423.
60. Slimmable Neural Networks / J. Yu [et al.] // Proc. Intern. Conf. Learning Representations. – 2019.
61. *Sokolov, V. A.* Extracting Embeddings for Active Learning Corresponding to Regions of Interest in an Image / V. A. Sokolov, I. A. Matveev, L. M. Teplyakov // Pattern Recognition and Image Analysis. – 2025. – Vol. 35. – P. 417–426.
62. *Sokolov, V. A.* MAREA: Active Learning for Object Detection Using Nested Local Embeddings / V. A. Sokolov, L. M. Teplyakov, I. A. Matveev // ICPR 2026 Workshop Papers: Image Mining. Theory and Applications (IMTAX-2026). – Lyon, France, 2026. – URL: https://icpr2026orgteam.github.io/WSpapers/pdfs/paper_0059.pdf (дата обращения: 13.09.2026).
63. *Spotify.* Annoy: Approximate Nearest Neighbors Oh Yeah / Spotify. – URL: <https://github.com/spotify/annoy> (дата обращения: 02.10.2026).
64. SSD: Single Shot MultiBox Detector / W. Liu [et al.] // Proc. European Conf. Computer Vision. – Amsterdam, Netherlands, 2016. – P. 21–37.

65. *Sutton, R. S. Reinforcement Learning: An Introduction / R. S. Sutton, A. G. Barto. – 2nd ed. – Cambridge, MA : MIT Press, 2018.*
66. *Tan, M. EfficientDet: Scalable and Efficient Object Detection / M. Tan, R. Pang, Q. V. Le // Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition. – Seattle, USA, 2020. – P. 10781–10790.*
67. *Ten Years of Active Learning Techniques and Object Detection: A Systematic Review / D. Garcia [et al.] // Applied Sciences. – 2023. – Vol. 13, no. 19. – P. 10667.*
68. *The Pascal Visual Object Classes (VOC) Challenge / M. Everingham [et al.] // International Journal of Computer Vision. Vol. 88. – 2010. – P. 303–338.*
69. *The Power of Ensembles for Active Learning in Image Classification / W. H. Beluch [et al.] // Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition. – Salt Lake City, USA, 2018. – P. 9368–9377.*
70. *Tian, Y. YOLOv12: Attention-Centric Real-Time Object Detectors / Y. Tian, Q. Ye, D. Doermann // Advances in Neural Information Processing Systems. Vol. 38. – 2025. – P. 78433–78457. – URL: https://proceedings.neurips.cc/paper_files/paper/2025/hash/7103444259031cc58051f8c9a4868533-Abstract-Conference.html.*
71. *Ultralytics YOLOv5 / G. Jocher [et al.]. – 2020. – URL: <https://github.com/ultralytics/yolov5> (дата обращения: 04.06.2026).*
72. *Wang, C.-Y. YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors / C.-Y. Wang, A. Bochkovskiy, H.-Y. M. Liao // Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition. – Vancouver, Canada, 2023. – P. 7464–7475.*
73. *Wang, C.-Y. YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information / C.-Y. Wang, I.-H. Yeh, H.-Y. M. Liao // Computer Vision – ECCV 2024. – 2024. – P. 1–21. – URL: https://doi.org/10.1007/978-3-031-72751-1_1.*
74. *Yang, C. Plug and Play Active Learning for Object Detection / C. Yang, L. Huang, E. J. Crowley // Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition. – Seattle, USA, 2024. – P. 17784–17793.*
75. *YOLOF: You Only Look One-Level Feature / Q. Chen [et al.] // Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition. – Nashville, USA, 2021. – P. 13034–13043.*

76. YOLOv10: Real-Time End-to-End Object Detection / A. Wang [et al.] // Advances in Neural Information Processing Systems. Vol. 37. – 2024. – P. 107984–108011. – URL: https://proceedings.neurips.cc/paper_files/paper/2024/hash/c34ddd05eb089991f06f3c5dc36836e0-Abstract-Conference.html.
77. YOLOX: Exceeding YOLO Series in 2021 / Z. Ge [et al.] // arXiv preprint arXiv:2107.08430. – 2021.
78. You Only Look Once: Unified, Real-Time Object Detection / J. Redmon [et al.] // Proc. IEEE Conf. Computer Vision and Pattern Recognition. – Las Vegas, USA, 2016. – P. 779–788.
79. *Yu, J.* Universally Slimmable Networks and Improved Training Techniques / J. Yu, T. S. Huang // Proc. IEEE/CVF Intern. Conf. Computer Vision. – 2019. – P. 1803–1811.
80. *Zhou, X.* Probabilistic Two-Stage Detection / X. Zhou, V. Koltun, P. Krähenbühl // arXiv preprint arXiv:2103.07461. – 2021.
81. *Zhou, X.* Objects as Points / X. Zhou, D. Wang, P. Krähenbühl // arXiv preprint arXiv:1904.07850. – 2019.
82. *Zhou, X.* Bottom-Up Object Detection by Grouping Extreme and Center Points / X. Zhou, J. Zhuo, P. Krähenbühl // Proc. IEEE/CVF Conf. Computer Vision and Pattern Recognition. – Long Beach, USA, 2019. – P. 850–859.
83. *Zong, Z.* DETRs with Collaborative Hybrid Assignments Training / Z. Zong, G. Song, Y. Liu // Proc. IEEE/CVF Intern. Conf. Computer Vision. – 2023. – P. 6748–6758. – URL: https://openaccess.thecvf.com/content/ICCV2023/html/Zong_DETRs_with_Collaborative_Hybrid_Assignments_Training_ICCV_2023_paper.html.

Список рисунков

1.1	Архитектура пирамиды признаков	16
1.2	Точность и скорость детекторов на COCO 2017	17
1.3	Цикл пакетного активного обучения с пулом.	21
1.4	Дилемма исследования и использования	23
2.1	Несовпадение рамки с сеткой карты признаков	31
2.2	Ошибки округления при проецировании рамки	32
2.3	Схема операции RoI Align	33
2.4	Извлечение многоуровневого локального эмбединга	34
2.5	Сравнение критериев по расстоянию и по плотности	38
2.6	Примеры изображений Pascal VOC [68]: без разметки (слева) и с ограничивающими прямоугольниками (справа).	41
2.7	Примеры изображений COCO 2017 [48].	41
2.8	Примеры изображений CrowdHuman [20]: плотные сцены с перекрывающимися людьми.	42
2.9	Сравнение критериев по расстоянию и по плотности на Pascal VOC и COCO 2017. YOLOv8m, инициализация весами COCO [61].	43
2.10	Цепочки активного обучения на Pascal VOC [62]. Пунктир – качество на полном наборе данных (66.14 %).	44
2.11	Цепочки активного обучения на CrowdHuman [62]. Пунктир – качество на полном наборе данных (56.58 %).	45
2.12	Цепочки активного обучения на COCO 2017 [62]. Пунктир – качество на полном наборе данных (43.61 %).	46
2.13	Цепочки активного обучения на ind_safety [62]. Пунктир – качество на полном наборе данных (66.85 %).	47
3.1	Схема каскадного поиска	52
3.2	Модификация выходного модуля YOLOv8 по принципу MRL	57
3.3	Общее назначение при обучении по схеме матришки	58
3.4	Переплетение признаков в единый вложенный вектор	61
3.5	Пример каскадного поиска на Pascal VOC	63
3.6	Качество детекции YOLOv8s	65
3.7	Recall@10 для разных длин подвектора	67

- 4.1 Цепочки активного обучения на Pascal VOC [62]. Сплошная
горизонтальная линия – качество на полном наборе данных (66.14 %). . 72
- 4.2 Цепочки активного обучения на CrowdHuman [62]. Сплошная
горизонтальная линия – качество на полном наборе данных (56.58 %). . 73
- 4.3 Цепочки активного обучения на COCO 2017 [62]. Сплошная
горизонтальная линия – качество на полном наборе данных (43.61 %). . 74
- 4.4 Цепочки активного обучения на ind_safety 75

Список таблиц

1	Детекторы объектов, представленные на рисунке 1.2	16
2	mAP50–95, %, на Pascal VOC (среднее $\pm$ 95 % CI (Confidence Interval), 4 инициализации) [62]	44
3	mAP50–95, %, на CrowdHuman (среднее $\pm$ 95 % CI, 4 инициализации) [62]	45
4	mAP50–95, %, на COCO 2017 (среднее $\pm$ 95 % CI, 4 инициализации) [62]	47
5	mAP50–95, %, на ind_safety (среднее $\pm$ 95 % CI, 4 инициализации) [62]	48
6	Качество детекции YOLOv8s и YOLOv8s-MRL	64
7	Качество детекции при разных длинах подвектора	65
8	Поиск похожих объектов для вложенных эмбедингов	66
9	Цепочки активного обучения на Pascal VOC (mAP50–95, %, среднее $\pm$ 95 % CI, 4 инициализации) [62]. Полный набор данных: 66.14 %.	71
10	Цепочки активного обучения на CrowdHuman (mAP50–95, %) [62]. Полный набор данных: 56.58 %.	72
11	Цепочки активного обучения на COCO 2017 (mAP50–95, %) [62]. Полный набор данных: 43.61 %.	73
12	Цепочки активного обучения на ind_safety (mAP50–95, %) [62]. Полный набор данных: 66.85 %.	75
13	Средние выигрыши MAREA над базовыми методами, mAP50–95, п.п., усреднение по долям 0.3–0.7 [62].	76