

# TECHNOLOGICAL DEVELOPMENTS

UDC 339.54.012

## Gilfanova D.R., Grigoryan K.A. Development of an algorithm for ball detection and tracking in video for automated judging in rhythmic gymnastics

Разработка алгоритма детекции и отслеживания мяча на видео для задач автоматизированного судейства в художественной гимнастике

**Gilfanova Diana Rafaelevna,**

Master's candidate in Digital Product Analytics and Development Management at the ITIS KFU, Kazan

**Grigoryan Karen Albertovich,**

Candidate of Economics, Associate Professor, Kazan (Volga region) Federal University, Kazan

Гильфанова Диана Рафаэлевна,  
магистрант направления "Аналитика и управление разработкой цифровых продуктов"

института ИТИС КФУ, г. Казань

Григорян Карен Альбертович,

кандидат экономических наук, доцент, Казанский Приволжский Федеральный университет, г. Казань

**Abstract.** Rhythmic gymnastics is a sport that is challenging for objective judging, where the accuracy of evaluating elements, especially dynamic elements involving a ball, depends significantly on subjective factors and judges' fatigue. In this paper, an algorithm for automatic ball detection in video is proposed, based on a combination of a fine-tuned YOLOv8 detection model and the ByteTrack tracking algorithm. The model was fine-tuned on a custom dataset that accounts for domain-specific conditions, including small object size, motion blur, occlusions, and complex backgrounds. Experimental results demonstrate that the proposed algorithm achieves high detection accuracy ( $mAP@0.5 \approx 0.99$ ), as well as robustness to occlusions and stable object tracking, which makes it suitable as a foundation for developing automated evaluation systems in rhythmic gymnastics.

**Keywords:** rhythmic gymnastics, computer vision, object detection, object tracking, YOLOv8, ByteTrack, deep learning, sports analytics.

**Аннотация.** Художественная гимнастика представляет собой сложный для объективного судейства вид спорта, где точность оценки элементов, особенно динамических с использованием мяча, зависит от субъективных факторов и усталости судей. В данной статье предложен алгоритм автоматической детекции мяча на видео, основанный на комбинации дообученной модели детекции YOLOv8 и алгоритма трекинга ByteTrack. Модель дообучалась на кастомном датасете с учетом специфических условий: малого размера объекта, размытий, окклюзий и сложного фона. В результате проведенных экспериментов разработанный алгоритм продемонстрировал высокую точность детекции ( $mAP@0.5 \approx 0.99$ ), а также устойчивость к окклюзиям и стабильность отслеживания объекта, что позволяет рассматривать его в качестве основы для построения систем автоматизированной оценки в художественной гимнастике.

**Ключевые слова:** художественная гимнастика, компьютерное зрение, детекция объектов, отслеживание объектов, YOLOv8, ByteTrack, глубокое обучение, спортивная аналитика.

**Рецензент:** Торопцев Василий Владимирович - кандидат технических наук, доцент.  
ФГБОУ ВО «РГАУ-МСХА им. К.А. Тимирязева»

## ВВЕДЕНИЕ

Художественная гимнастика (ХГ) представляет собой один из наиболее зрелищных и субъективно сложных видов спорта, в котором исполнение элементов требует от спортсменок не только исключительной физической подготовки, но и предельной точности координации движений. Оценка таких элементов осуществляется вручную квалифицированными судьями в соответствии с действующими правилами Международной федерации гимнастики (FIG), однако они испытывают значительные психофизические нагрузки в течение длительных соревнований, что неизбежно приводит к усталости, снижению концентрации и, как следствие, ошибкам и пропуску мелких, но критичных нарушений [1, 2].

Особую сложность для судейства представляют динамические элементы с мячом, характеризующиеся высокой скоростью, сложной траекторией и необходимостью синхронной оценки движений гимнастки и предмета. Для полноценной реализации задачи автоматизации оценки таких элементов первоначально необходимо научиться с высокой точностью обнаруживать координаты мяча на видео для последующей обработки.

## ОБЗОР ПРЕДМЕТНОЙ ОБЛАСТИ

Задачи детекции и отслеживания мяча представляют собой классическую проблему компьютерного зрения, связанную с обнаружением малогабаритных быстро движущихся объектов в условиях сложного фона, частичных окклюзий и изменяющегося освещения. В контексте художественной гимнастики эта задача усугубляется следующими факторами:

- Малый объект в кадре. При удалении камеры мяч может занимать в кадре мало пикселей.
- Быстрое движение и размытие. Высокие броски дают сильное смазывание на кадрах.
- Сильные окклюзии. Мяч часто перекрывается частями тела гимнастки на короткие интервалы.
- Ложные детекции. Соревновательная сцена и одежда гимнасток создают фоновые шумы, требующие робастной фильтрации.
- Высокие требования к точности для последующего анализа техники.
- Однокамерная система. В рамках данного исследования нет возможности работы с несколькими камерами.

Вместе с тем имеются и некоторые упрощения:

- Необходимость детекции только одного мяча и одной гимнастки.
- Невысокая скорость детектируемого объекта.

## ОБЗОР ЛИТЕРАТУРЫ

В рамках решения задачи был проведен анализ современных компьютерных методов из источников, связанных преимущественно с детекцией мяча в спорте. Особое внимание уделялось подходам, обеспечивающим высокую точность и устойчивость в сложных условиях: при высокой скорости объекта, частичных перекрытиях, изменении освещения и сходстве с фоном.

Исторически первые решения основывались на классических алгоритмах компьютерного зрения, представленных на рисунке 1. Эти методы отличаются вычислительной простотой, однако обладают существенными ограничениями, принцип работы и ключевые ограничения некоторых из них отображены в таблице 1. Их эффективность критически снижается в динамичных сценах со сложным переменным освещением, наличием объектов-помех схожей цветовой гаммы и при сильном размытии. В частности, фильтры (Калмана, частиц), эффективные для линейного и предсказуемого движения, плохо адаптируются к резким нелинейным траекториям и изменению внешнего вида быстро движущегося мяча [3].

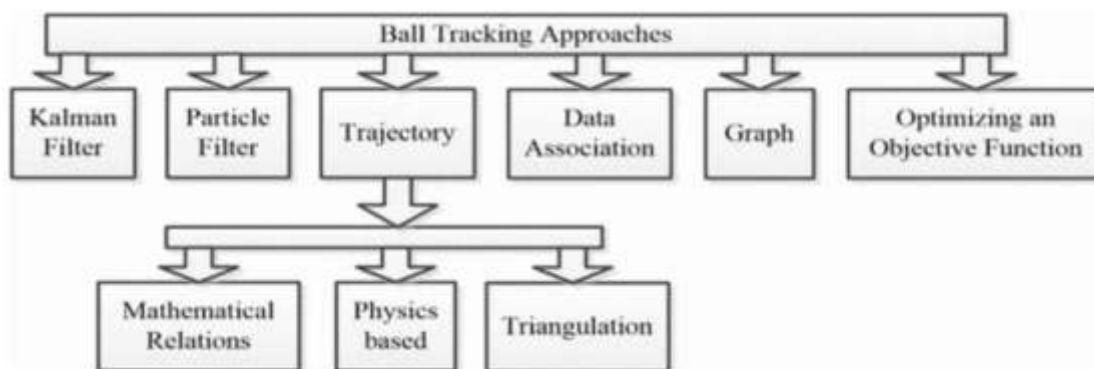


Рисунок 1. Классификация классических подходов к отслеживанию мяча [3, p.1698]

Таблица 1.

Принципы работы и ключевые ограничения основных классических подходов к  
отслеживанию объектов

Метод	Принцип работы	Ключевые ограничения для задачи детекции мяча
Цветовые фильтры	Сегментация по заданному цветовому диапазону	Неустойчивость к изменению освещения и схожему фону
KLT-трекер	Отслеживание угловых точек от кадра к кадру	Потеря трека при быстром движении и размытии
Фильтр Калмана	Оптимальная оценка состояния на основе предыдущих данных	Предполагает линейную динамику, что не всегда верно
Фильтр частиц	Оценка состояния на основе набора гипотез (частиц)	Деградация при малом размере цели и схожих помехах

Таким образом, классические методы не отвечают требованиям текущей задачи, где необходима робастность к нестабильным условиям. Вследствие этого акцент в исследованиях сместился в сторону методов на основе глубокого обучения.

### Методы на основе глубокого обучения

Современные алгоритмы на основе глубокого обучения можно разделить на две основные архитектурные парадигмы: двухэтапные и одноэтапные детекторы.

Двухэтапные детекторы (например, семейство R-CNN) разделяют процесс на генерацию регионов-кандидатов и их последующую классификацию с уточнением координат. Это обеспечивает высокую точность, но приводит к значительным вычислительным затратам. Одноэтапные детекторы (например, YOLO и SSD) исключают этап предложения регионов, предсказывая ограничивающие рамки (bounding boxes) и классы объектов напрямую за один проход по сети. Это обеспечивает высокую скорость работы, однако может сопровождаться некоторой потерей точности, особенно для малых объектов [4].

Для выбора оптимальной архитектуры был проведен анализ наиболее распространенных моделей.

**YOLO (You Only Look Once).** Архитектура YOLO относится к одностадийным детекторам, в которых задачи локализации и классификации объектов решаются единой сверточной нейронной сетью, что делает данный подход стандартом для задач

детекции в реальном времени. Архитектура модели YOLOv8 изображена на рисунке 2. По ней исходное изображение обрабатывается backbone-сетью для извлечения признаков, после чего с использованием писк-модуля (PAN/FPN) формируются многоуровневые карты признаков, обеспечивающие детекцию объектов различных масштабов. Detection head предсказывает для каждой ячейки сетки S×S параметры ограничивающей рамки (x,y,w,h), оценку наличия объекта и условные вероятности классов.

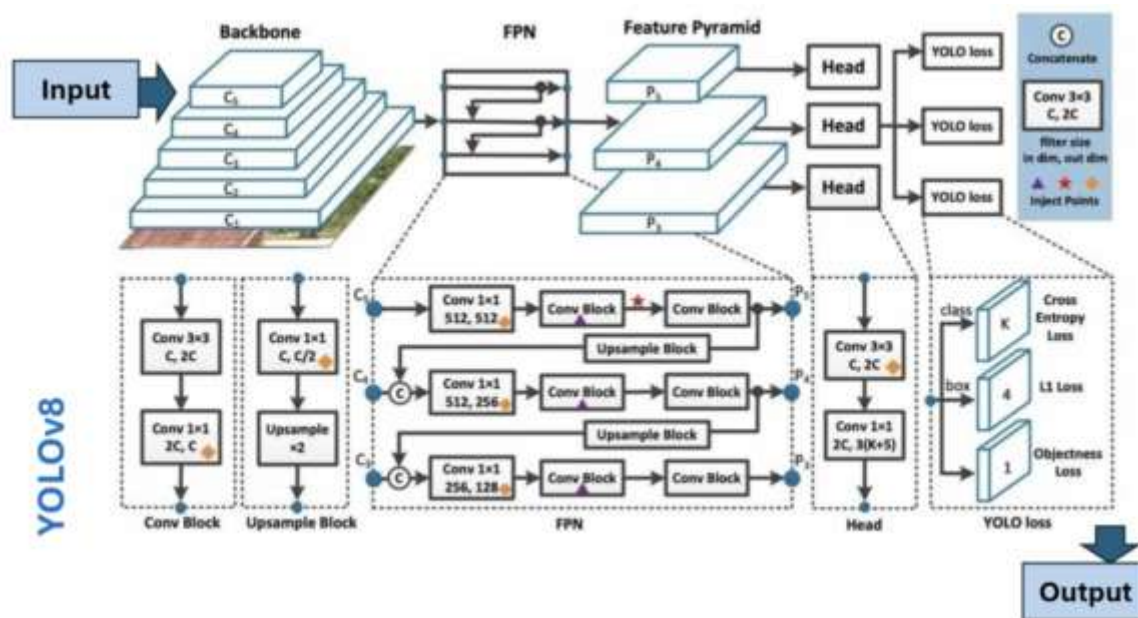


Рисунок 2. Архитектура YOLOv8 [5]

Оценка уверенности обнаружения объекта определяется как произведение вероятности наличия объекта в предсказанной области и меры пересечения предсказанной и истинной ограничивающих рамок (Intersection over Union, IoU) и вычисляется по формуле (1):

$$C = Pr(object) \times IoU_{pred}^{gt}, \quad (1)$$

где  $Pr(object)$  – вероятность наличия объекта,  $IoU_{pred}^{gt}$  – значение меры пересечения предсказанной и истинной ограничивающих рамок.

Итоговая вероятность принадлежности объекта классу k определяется как произведение оценки уверенности и условной вероятности класса и вычисляется по формуле (2):

$$Pr(class_k) = C \times p_k, \quad (2)$$

где  $p_k$  – условная вероятность принадлежности объекта классу k.

Ключевыми преимуществами YOLO являются высокая скорость инференса и возможность end-to-end обучения. Современные версии, включая YOLOv8, демонстрируют улучшенную точность при детекции малых объектов за счёт многоуровневых признаков и оптимизированных функций потерь, что особенно важно для задачи обнаружения мяча на сцене большого масштаба. В то же время эффективность метода может снижаться в условиях сильных окклюзий и размытия движения, что требует применения аугментации данных и тонкой настройки модели [5].

**SSD (Single Shot MultiBox Detector).** Архитектура SSD, изображенная на рисунке 3, относится к одностадийным детекторам и основывается на использовании сверточной нейронной сети *reduced VGG* в качестве базовой для извлечения признаков. На основе выходных слоёв backbone формируется набор многоуровневых карт признаков, каждая из которых используется для детекции объектов соответствующего масштаба. Для каждой карты признаков задается набор априорных ограничивающих рамок различных масштабов и аспектных отношений. Сеть одновременно предсказывает смещения координат рамок и вероятности принадлежности объекта определённому классу. Обучение SSD осуществляется путём минимизации суммарной функции потерь, включающей компоненты локализации и классификации, которая определяется выражением (3):

$$L = \left(\frac{1}{N}\right) \times (L_{loc} + \alpha \cdot L_{conf}), \quad (3)$$

где  $N$  – число априорных рамок, сопоставленных с объектами,  $L_{loc}$  – функция потерь локализации (Smooth L1),  $L_{conf}$  – функция потерь классификации (кросс-энтропия),  $\alpha$  – коэффициент балансировки.

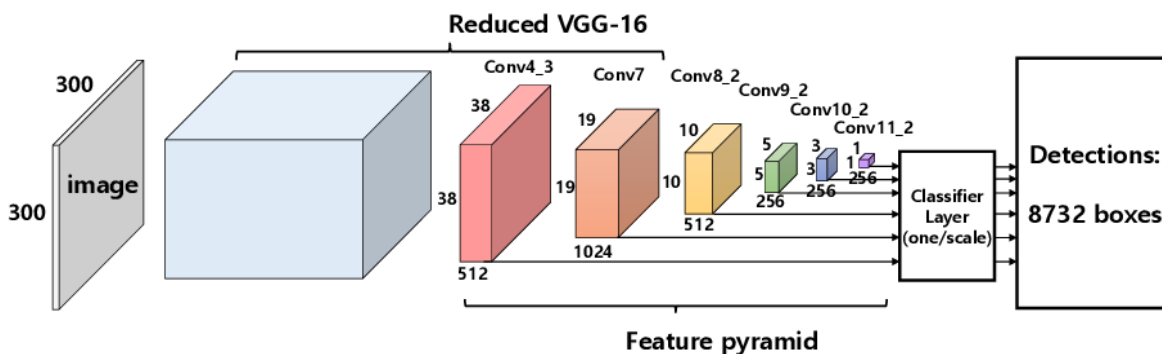


Рисунок 3. Архитектура SSD [6]

Использование многоуровневых карт признаков обеспечивает SSD приемлемый баланс между скоростью и точностью детекции. Однако отсутствие специализированных механизмов усиления признаков для малых объектов

ограничивает эффективность метода при обнаружении очень маленьких и быстро движущихся целей [7].

**Семейство R-CNN моделей.** Семейство моделей R-CNN относится к двухэтапным методам детекции, в которых процесс обнаружения объектов разделён на этап генерации кандидатов регионов и этап их классификации. В Faster R-CNN генерация регионов осуществляется с помощью сети предложений регионов (Region Proposal Network, RPN), встроенной в общую архитектуру и совместно обучаемой с классификатором. После этого для каждого предложенного региона выполняется операция RoI Align и последующая классификация и регрессия координат ограничивающей рамки. Функция потерь RPN определяется выражением (4):

$$L_{RPN} = L_{cls} + \alpha \cdot L_{reg}, \quad (4)$$

где  $L_{cls}$  – функция потерь классификации (объект / фон),  $L_{reg}$  – функция потерь регрессии координат рамок (Smooth L1),  $\alpha$  – коэффициент балансировки.

В модели Mask R-CNN архитектура дополнительно расширяется ветвью сегментации, что позволяет одновременно решать задачи детекции и попиксельного выделения объектов. Несмотря на высокую точность, особенно в сценах с перекрытиями и сложным фоном, двухэтапные модели характеризуются значительными вычислительными затратами и ограниченной скоростью инференса, что снижает их применимость для задач трекинга в реальном времени на стандартном оборудовании [8-10].

**RetinaNet.** Это одностадийный детектор, использующий архитектуру Feature Pyramid Network (FPN), которая формирует иерархию карт признаков с различным пространственным разрешением на основе сверточной backbone-сети. Это позволяет эффективно детектировать объекты различных масштабов без необходимости отдельного этапа генерации регионов. Для каждой карты признаков применяются независимые подголовы классификации и регрессии ограничивающих рамок. Ключевой особенностью RetinaNet является использование фокальной функции потерь, определяемой выражением (5), направленной на решение проблемы сильного дисбаланса между фоновыми и объектными примерами.

$$FL(p_t) = -\alpha \cdot (1 - p_t)^\gamma \cdot \log(p_t), \quad (5)$$

где  $p_t$  – предсказанная вероятность истинного класса,  $\gamma$  – параметр фокусировки,  $\alpha$  – коэффициент балансировки.

Использование FPN в сочетании с фокальной функцией потерь позволяет RetinaNet достигать точности, сопоставимой с двухэтапными детекторами, при

сохранении высокой скорости инференса, что делает данный метод эффективным компромиссом между быстродействием и точностью [11].

Для решения задачи формирования непрерывных траекторий обнаруженного объекта применяется парадигма Tracking-by-Detection (TbD). Она предполагает двухэтапный процесс: независимую детекцию объекта на каждом кадре и последующую ассоциацию этих детекций во времени. Такой подход позволяет компенсировать возможные пропуски детектора и поддерживать идентификатор объекта, что может послужить хорошим улучшением системы.

Эволюция алгоритмов трекинга развивалась от простых и быстрых методов к более сложным, учитывающим особенности внешнего вида объекта:

- **SORT** – использует фильтр Калмана для предсказания положения и пересечение по объединению (IoU) для ассоциации, отличается высокой скоростью, но чувствителен к окклюзиям.
- **DeepSORT** – вводит эмбединги внешнего вида (ReID), что значительно повышает устойчивость к перекрытиям и схожим объектам на сцене.
- **ByteTrack** – ключевым нововведением является ассоциация детекций с низкой уверенностью, что резко повышает полноту треков (MOTA) [12].
- **BoT-SORT** – фокусируется на использовании эмбедингов и коррекции смещения камеры, обладает более стабильной ассоциацией при сложных сценах [13].
- **OC-SORT** – предлагает observation-centric подход к обновлению фильтра Калмана, демонстрируя хорошую устойчивость к нелинейному движению [14].

## ОПИСАНИЕ ВЫБРАННОГО РЕШЕНИЯ

На основе проведенного анализа сформирован конвейер обработки, состоящий из двух ключевых и взаимодополняющих модулей: детектора объектов для локализации мяча в каждом кадре и трекера для ассоциации обнаружений во времени.

В качестве основного детектора выбрана модель YOLOv8. Данное решение обосновано следующими факторами:

1. Оптимальное соотношение скорости и точности: архитектура YOLOv8 обеспечивает инференс в режиме, близком к реальному времени, что является обязательным требованием для обработки видеопотоков. При этом она демонстрирует достаточную точность (mAP), особенно после адаптации к конкретной задаче.
2. Доминирование в предметной области: согласно сводным данным, представленным в таблицах 2 и 3, из исследования [4], модели семейства YOLO демонстрируют наилучшие показатели для текущей задачи. Особенно показателен пример волейбола, где условия (высокая скорость малого объекта, частые окклюзии)

наиболее близки к задаче отслеживания мяча в художественной гимнастике и где YOLO-архитектуры стабильно показывают высокую эффективность. Это делает YOLOv8 логичным и обоснованным выбором для нашей методологии.

Таблица 2.

Лучшие алгоритмы детекции мяча для разных видов спорта [4, p.3107]

Sport	Best algorithm	Accuracy/mAP	Inference time	Strengths	Challenges
Football	YOLOv8 (Hybrid Enhancements)	High	Real-time	Handles occlusion, real-time tracking	Issues with distant ball tracking
Tennis	SSD-MobileNet	Medium-high	Fast	Lightweight, CPU-friendly	Sensitive to lighting
Table tennis	Optimized YOLOv3-Tiny	Very high	Very Fast	Great for small object detection	Needs high-res input
Basketball	YOLO-T2LSTM	Very high	Fast	Accurate action recognition	Complex model fusion
Golf	Faster R-CNN	Very high	Slow	High precision tracking	Not real-time capable
Handball	Mask R-CNN	Very high	Slow	High segmentation accuracy	High computational cost
Volleyball	YOLOv3 (Enhanced)	High	Fast	Improved detection of fast-moving objects	Training data limits generalizability
Boccia	Tiny-YOLO	High	Fast	Good color-based accuracy	Lighting sensitivity, narrow FOV
Rugby	YOLOv3	High	Fast	Compact architecture for embedded use	Lower resolution support
Padel	YOLO + Audio SED	High	Fast	Context-aware detection	Audio interference risks
Badminton	YOLO-HGNet	Very high	Fast	Enhanced classification and detection accuracy	Requires tuning and large datasets

Таблица 3.

Сравнение производительностей моделей детекции в спортивных статьях [4, p.3109]

**Table 7 Comparative performance metrics of object detection models used in sports applications.**

Model	Avg accuracy (%)	Avg inference time (ms)	Precision (%)	Recall (%)	F1-score (%)
YOLO (various versions)	85	20	84	82	83
SSD	78	25	76	75	75
Faster R-CNN	90	100	89	91	90
Mask R-CNN	92	120	91	92	91
RetinaNet	87	60	85	83	84
Traditional CV/ML	70	15	65	60	62

3. Удобство адаптации и дообучения: фреймворк Ultralytics предоставляет хорошо документированный и простой в использовании конвейер для настройки предобученных моделей на пользовательских данных. Это позволяет относительно легко адаптировать детектор к специфическому внешнему виду мяча для художественной гимнастики, вариациям освещения и ракурсов.

4. Масштабируемость: в рамках семейства YOLOv8 существует спектр моделей различного размера (от nano до extra-large). Для экспериментов выбраны версии YOLOv8l (large) и YOLOv8x (extra-large), так как они обеспечивают значительный прирост точности по сравнению с облегченными вариантами, сохраняя приемлемую для исследований скорость работы [15].

Алгоритмы, ориентированные на сцены с множеством объектов (BoT-SORT) или требующие стабильных наблюдений (OC-SORT), в данных условиях могут показать худшие результаты при длительных пропаданиях. ByteTrack, напротив, эффективно поддерживает целостность трека и корректно восстанавливает траекторию при кратковременных окклюзиях и пропаданиях, что случается довольно часто в контексте художественной гимнастики. Дополнительными практическими преимуществами являются его вычислительная эффективность и простота интеграции с YOLOv8. Поэтому для ассоциации объектов, обнаруженных дообученной YOLOv8, между кадрами выбран алгоритм ByteTrack, в качестве альтернативы для сравнения будет рассматриваться алгоритм OC-SORT.

## **РЕЗУЛЬТАТЫ**

### **Сбор и подготовка данных**

Первостепенной задачей стала разметка кастомного датасета для дообучения выбранной модели. В качестве вспомогательного инструмента интерактивной аннотации изображений была выбрана платформа Roboflow с расширением Roboflow Auto Label [16], сама разметка по большей части велась вручную.

Итого для обучения был создан датасет, состоящий из 2383 изображений по 20-ти видео с различных соревнований, с разными освещением, камерами, расстояниями до объекта. Также был выбран большой процент кадров с частичными окклюзиями (до 80%), размытиями и частичными потерями объекта. Сам датасет был разделен в соотношениях: 72 % – train, 14 % – validation, 14 % – test.

В процессе препроцессинга были применены такие приемы аугментации данных, как: случайное размытие, коррекция яркости, горизонтальное отражение и случайное кадрирование, так как исследования показали, что аугментация повышает способность модели обобщать данные и делают ее более устойчивой к шумам [17].

### **Дообучение моделей**

Для дообучения и последующего тестирования были выбраны две крупные модели семейства YOLOv8: YOLOv8l (large) и YOLOv8x (extra large). Обучение проводилось в среде Google Colab на GPU T4. Обучение для YOLOv8l проводилось на 20-ти эпохах с размером батча равным 20.

Критерии остановки: обучение модели было остановлено после 20-ти эпох, так как функция потерь на валидационной выборке перестала уменьшаться, а метрики качества улучшаться. Общее время обучения составило около 1.5 часов. Результаты представлены на рисунке 4.

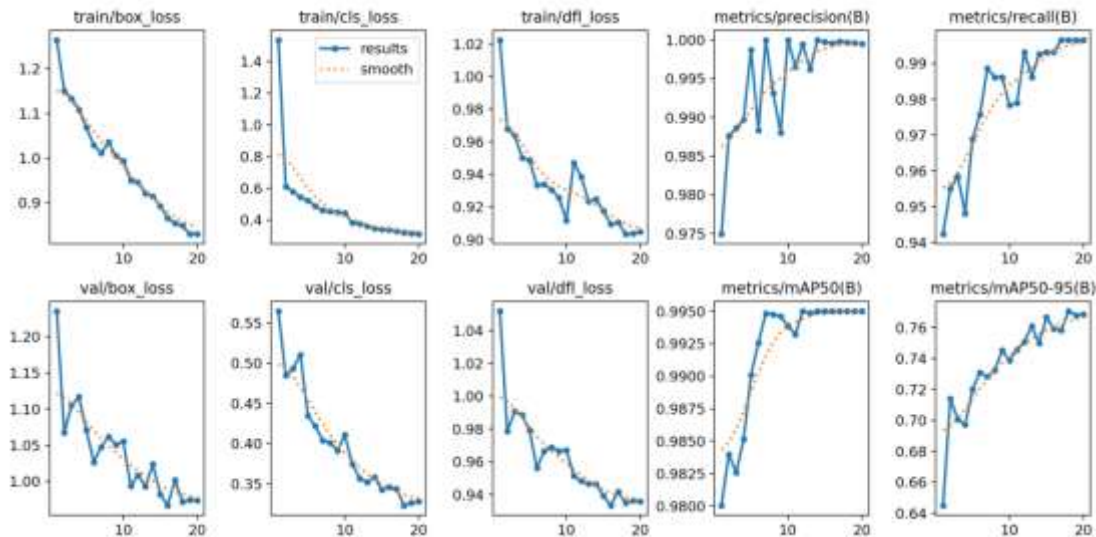


Рисунок 4. Изменение основных метрик во время дообучения YOLOv8l

На рисунке 5 представлено сравнение оригинальной и дообученной моделей. Заметны значительный улучшения по всем метрикам, главное – рост  $mAP_{50-95}$  с 0,46 до 0,77.

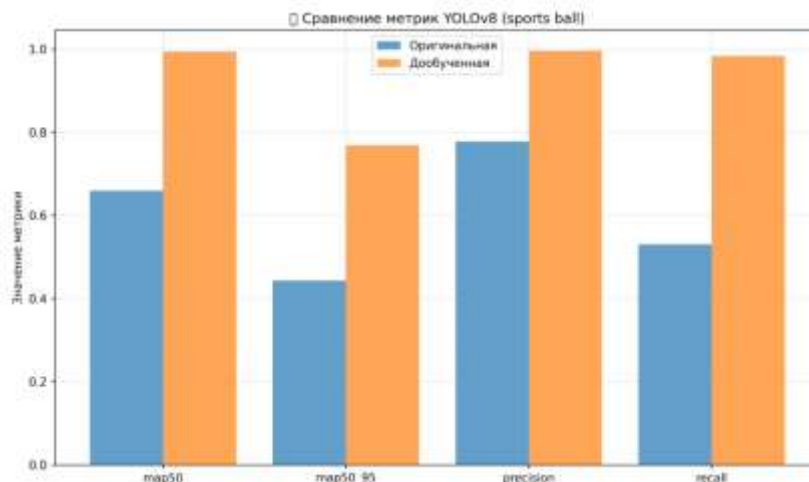


Рисунок 5. Сравнение метрик оригинальной и дообученной YOLOv8l

Параметры обучения для YOLOv8x были аналогичны, кроме количества эпох, их было 15. Обучение YOLOv8x заняло около 3 часов, что связано с большим размером модели.

Получившиеся результаты сравнения всех четырех моделей по основным метрикам представлены на рисунках 6 и 7.

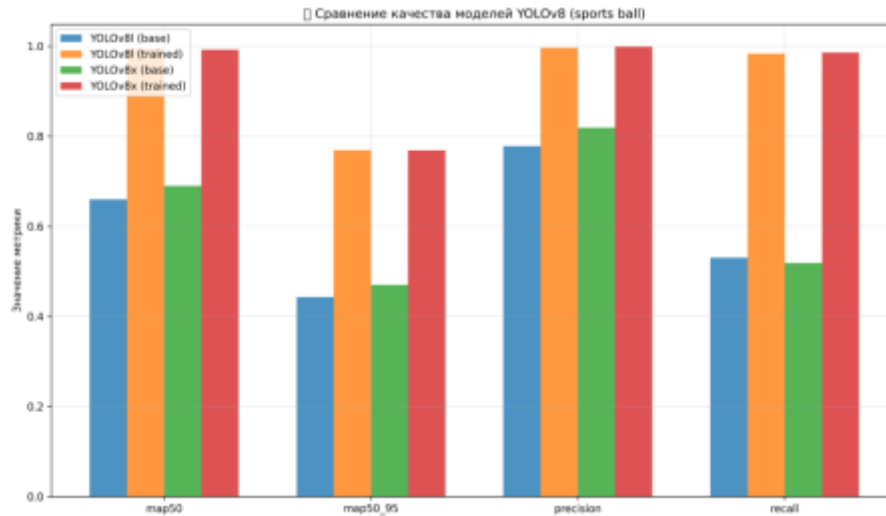


Рисунок 6. Диаграмма сравнения моделей по метрикам mAP, precision, recall

ИТОГОВОЕ СРАВНЕНИЕ МОДЕЛЕЙ:

Модель	mAP50	mAP50-95	Precision	Recall	Speed(ms)
YOLOv8l (base)	0.660	0.443	0.778	0.530	37.64
YOLOv8l (trained)	0.994	0.769	0.996	0.984	38.88
YOLOv8x (base)	0.690	0.470	0.819	0.519	62.30
YOLOv8x (trained)	0.992	0.768	0.999	0.986	60.41

ЛУЧШИЕ МОДЕЛИ ПО КРИТЕРИЯМ:

- mAP50 : YOLOv8l (trained) (0.994)
- mAP50\_95 : YOLOv8l (trained) (0.769)
- precision : YOLOv8x (trained) (0.999)
- recall : YOLOv8x (trained) (0.986)

Самая быстрая модель: YOLOv8l (base) (37.64 ms)

Рисунок 7. Численное сравнение моделей по ключевым метрикам

Метрики вычислялись автоматически по тестовой выборке на основе сравнения предсказанных и эталонных ограничивающих рамок. Предсказание считалось корректным, если значение меры пересечения IoU (отношение между предсказанной и эталонной рамками) превышало заданный порог. На основе IoU вычислялись основные метрики: precision – отношение числа корректных детекций к общему числу предсказаний, recall – отношение числа корректных детекций к числу объектов в тестовой выборке, average precision (AP) – площадь под кривой precision–recall при фиксированном пороге IoU, mAP@0.5 – среднее значение AP при IoU ≥ 0.5, mAP@0.5:0.95 - среднее значение AP при порогах IoU от 0.5 до 0.95 с шагом 0.05. Время инференса вычислялось как среднее время обработки одного изображения моделью на тестовой выборке.

Все метрики продемонстрировали значительный рост после дообучения. Модель научилась находить мяч в сложных условиях, где базовая модель его пропускала.

Сравнивая модели YOLOv8l и YOLOv8x, можно заметить, что после дообучения обе модели показали исключительно высокие и практически идентичные результаты по mAP@0.5 (~0.99). YOLOv8x демонстрирует относительно небольшое преимущество в метриках precision и recall (0.999 и 0.986 против 0.996 и 0.984 у YOLOv8l), что ожидаемо в связи с ее большей емкостью. Ключевым отличием является скорость работы. YOLOv8l обрабатывает изображение за 38.88 мс, что примерно на 35% быстрее, чем YOLOv8x (60.41 мс). Это критически важно для систем реального времени.

### **Качественный анализ и обоснование выбора модели**

На основе количественных метрик в качестве оптимальной была выбрана модель YOLOv8l:

1. Практическая эквивалентность точности: разница в mAP@0.5 между YOLOv8l и YOLOv8x составляет 0.002.
2. Преимущество в скорости: YOLOv8l значительно быстрее, что позволяет обрабатывать видео с более высоким FPS.
3. Эффективность по ресурсам: меньший размер модели делает ее более предпочтительной для развертывания в условиях ограниченных вычислительных ресурсов.

Качественные улучшения дообученной YOLOv8l:

- Устойчивость к окклюзиям: модель успешно детектирует мяч, даже когда он перекрыт телом гимнастки на 70-80% (рисунок 8).
- Работа с размытием: алгоритм стабильно находит мяч на кадрах с размытием, где базовая модель теряет объект.
- Стабильность трекинга: улучшенная метрика recall напрямую ведет к более стабильному трекингу, так как мяч реже теряется при выходе из кадра или кратковременном перекрытии.

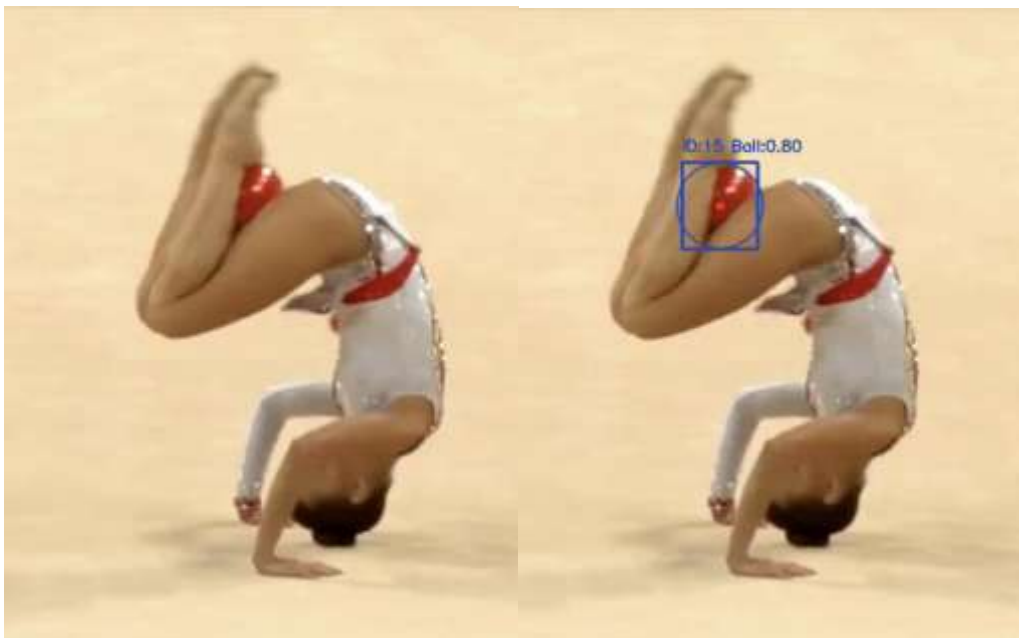


Рисунок 8. Результат детекции до и после дообучения YOLOv8l

### Улучшение с помощью Track-by-detection

Интеграция алгоритма ByteTrack с дообученной моделью YOLOv8 позволила достичь значительных улучшений в системе трекинга мяча для художественной гимнастики. Количественные результаты продемонстрировали снижение количества переключений идентификаторов (ID switches) на 85-90% – с 45 до 2-5 переключений на 100 кадров по 20-ти тестовым видео, что напрямую повышает стабильность отслеживания.

Также зафиксировано увеличение длины трека, что свидетельствует о способности системы сохранять идентификатор объекта при временных окклюзиях и выходах за пределы кадра. Стоит отметить, что при использовании OC-SORT данный показатель составил в среднем 12–18 переключений на 100 кадров, а средняя длина трека вдвое меньше, чем у ByteTrack.

Практическое сравнение результатов, представленное на рисунке 9, демонстрирует преимущества интеграции алгоритма ByteTrack. В первом ряду с изображениями отслеживания без использования ByteTrack наблюдаются частые потери объекта между последовательными кадрами с последующим ошибочным присвоением новых идентификаторов. В отличие от этого, результаты с применением ByteTrack (второй ряд изображений) характеризуются поддержанием идентификации объекта даже в условиях его временного исчезновения. Алгоритм демонстрирует способность корректной реидентификации при повторном появлении объекта,

сохраняя первоначальный идентификатор, что подтверждает его эффективность для задач непрерывного отслеживания в динамических сценах.



Рисунок 9. Результаты до и после использования алгоритма ByteTrack

### Итоги

Получившийся в результате работы алгоритм представлен на рисунке 10. Архитектура и принципы работы детектора YOLOv8 были рассмотрены выше, поэтому далее будет описана работа модуля ByteTrack. Для каждого активного трека применяется фильтр Калмана для прогнозирования положения на следующий кадр. Модель состояния включает координаты и скорости объекта, а предсказание осуществляется по линейной динамической модели. Далее детекции YOLOv8 разделяются на высоко и низко вероятные, и ассоциация выполняется в два этапа. На первом этапе высоко вероятные детекции ассоциируются с предсказанными треками по метрике IoU с использованием алгоритма венгерского назначения. На втором этапе оставшиеся несопоставленные треки пытаются ассоциировать с низкоуверенными детекциями. Успешно ассоциированные детекции обновляют состояние трека через фильтр Калмана. Треки, оставшиеся несопоставленными, увеличивают счетчик потерь, а при превышении определенного порога удаляются [12].

Эффективность предложенного подхода обусловлена сочетанием специализированного детектора и вероятностной модели движения. Дообучение YOLOv8 на кастомном датасете существенно увеличило полноту детекций и устойчивость к размытиям и перекрытиям, тем самым улучшая качество наблюдений.

ByteTrack, в свою очередь, реализует байесовскую временную фильтрацию, объединяя эти наблюдения с априорной моделью движения. Двухуровневая ассоциация детекций позволила восстановить траекторию даже при деградации уверенности детектора, а предсказание фильтра Калмана обеспечило непрерывность трека в условиях временного отсутствия мяча в кадре.

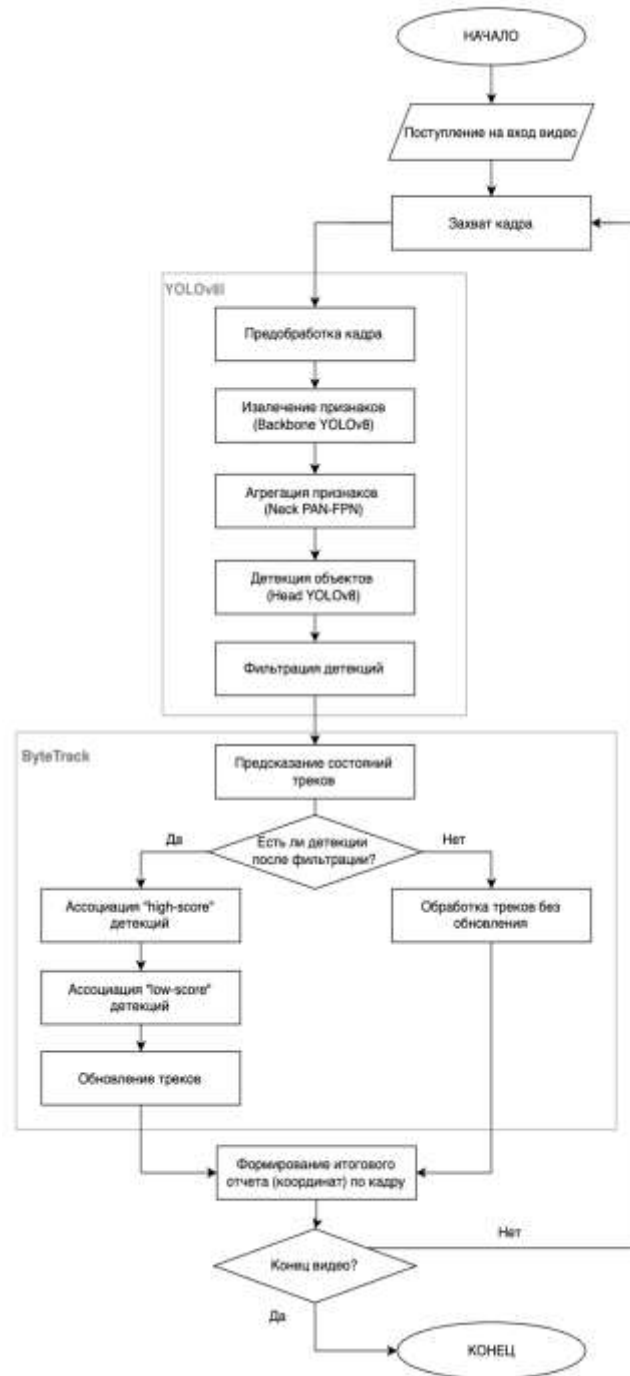


Рисунок 10. Блок-схема алгоритма детекции и отслеживания мяча на видео

## ЗАКЛЮЧЕНИЕ

Таким образом, был разработан и экспериментально исследован алгоритм автоматической детекции и отслеживания мяча на видео в художественной гимнастике, основанный на комбинации дообученной модели YOLOv8l и трекера ByteTrack. Проведенные эксперименты показали, что адаптация YOLOv8 к специализированному датасету позволила добиться высокой точности обнаружения мяча, устойчивости к размытию, окклюзиям и сложному фону. Использование ByteTrack обеспечило значительное улучшение стабильности трекинга и снижение числа переключений идентификаторов, что позволило формировать непрерывные и корректные траектории движения мяча.

Полученный алгоритм представляет собой надежную основу для построения систем автоматизированного анализа и оценки динамических элементов в художественной гимнастике. В частности, он может быть использован для вычисления параметров бросков и ловли мяча, анализа траекторий и синхронизации движений гимнастки и предмета, что открывает возможности для объективной поддержки судейства и спортивной аналитики.

## References

1. Boksem M. A. S., Meijman T. F., Lorist M. M. Effects of mental fatigue on attention: an ERP study //Cognitive brain research. – 2005. – Т. 25. – №. 1. – С. 107-116.
2. Flessas K. et al. Judging the judges' performance in rhythmic gymnastics //Medicine & Science in Sports & Exercise. – 2015. – Т. 47. – №. 3. – С. 640-648.
3. Kamble P. R., Keskar A. G., Bhurchandi K. M. Ball tracking in sports: a survey //Artificial Intelligence Review. – 2019. – V. 52. – N. 3. – P. 1655-1705.
4. Moreira C., Ferreira L., Coelho P. J. A comprehensive review of ball detection techniques in sports //PeerJ Computer Science. – 2025. – V. 11. – P. 3109.
5. Sapkota R., Karkee M. Ultralytics YOLO Evolution: An Overview of YOLO26, YOLO11, YOLOv8 and YOLOv5 Object Detectors for Computer Vision and Pattern Recognition //arXiv preprint arXiv:2510.09653. – 2025.
6. Jeong J., Park H., Kwak N. Enhancement of SSD by concatenating feature maps for object detection //arXiv preprint arXiv:1705.09587. – 2017.
7. Liu W. et al. Ssd: Single shot multibox detector //European conference on computer vision. – Cham : Springer International Publishing, 2016. – С. 21-37.
8. Ren S. et al. Faster r-cnn: Towards real-time object detection with region proposal networks //Advances in neural information processing systems. – 2015. – Т. 28.

9. He K. et al. Mask r-cnn //Proceedings of the IEEE international conference on computer vision. – 2017. – С. 2961-2969.
10. Buric M., Pobar M., Ivasic-Kos M. Ball detection using YOLO and Mask R-CNN //2018 International conference on computational science and computational intelligence (CSCI). – IEEE, 2018. – P. 319-323.
11. Lin T. Y. et al. Focal loss for dense object detection //Proceedings of the IEEE international conference on computer vision. – 2017. – P. 2980-2988.
12. Zhang Y. et al. Bytetrack: Multi-object tracking by associating every detection box //European conference on computer vision. – Cham : Springer Nature Switzerland, 2022. – С. 1-21.
13. Aharon N., Orfaig R., Bobrovsky B. Z. BoT-SORT: Robust associations multi-pedestrian tracking //arXiv preprint arXiv:2206.14651. – 2022.
14. Cao J. et al. Observation-centric sort: Rethinking sort for robust multi-object tracking //Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. – 2023. – С. 9686-9696.
15. Ultralytics YOLOv8 Documentation [Электронный ресурс] : [официальная документация] / Ultralytics. – 2023. – URL: <https://docs.ultralytics.com/ru/models/yolov8/#performance-metrics> (дата обращения: 06.01.2026).
16. Журавлёв А. К., Григорян К. А. Автоматическая разметка обучающих выборок в компьютерном зрении с использованием методов машинного обучения //Электронные библиотеки. – 2024. – Т. 27. – №. 5. – С. 718-729.
17. Shorten C., Khoshgoftaar T. M. A survey on image data augmentation for deep learning //Journal of big data. – 2019. – Т. 6. – №. 1. – С. 1-48.