

Научная статья

УДК 004.93

URL: <https://trudymai.ru/published.php?ID=188283>

EDN: <https://www.elibrary.ru/FACXRB>

КОМПЛЕКСНЫЙ ПОДХОД К ОБНАРУЖЕНИЮ И ОТСЛЕЖИВАНИЮ БЕСПИЛОТНЫХ ЛЕТАТЕЛЬНЫХ АППАРАТОВ НА ОСНОВЕ КОМПЬЮТЕРНОГО ЗРЕНИЯ

А.С. Голубев[✉], В. Иламанова, М.З. Нгуен, Д.М. Сошников, А.А. Хиленко

Владимирский государственный университет имени Александра Григорьевича и

Николая Григорьевича Столетовых, Владимир, Россия

✉ andrey.golubev@vlsu.ru

Цитирование: Голубев А.С., Иламанова В., Нгуен М.З., Сошников Д.М., Хиленко А.А. Комплексный подход к обнаружению и отслеживанию беспилотных летательных аппаратов на основе компьютерного зрения // Труды МАИ. 2026. № 148. URL: <https://trudymai.ru/published.php?ID=188283>

Аннотация. Предложен комплексный подход к противодействию беспилотным летательным аппаратам, охватывающий все этапы обработки: обнаружение, классификацию, отслеживание и прогнозирование траектории. На этапе обнаружения проведено сравнительное исследование современных архитектур нейронных сетей (YOLOv8, YOLOX, SSD), где модель YOLOX показала наивысшую точность с mAP50=0,964. Для классификации типов БПЛА использованы сверточные нейронные сети, обеспечившие точность до 91,35%. На этапе отслеживания оптимальным признан алгоритм CSRT, демонстрирующий устойчивость к перекрытиям и изменению освещения. Для прогнозирования траектории наилучшие результаты показала полиномиальная регрессия со средней ошибкой 1,27%. Созданные тестовые датасеты и выполнено дообучение моделей с использованием аугментации данных. Результаты могут быть применены в системах мониторинга и защиты воздушного пространства. В работе рассмотрены традиционные и современные методы обнаружения БПЛА, включая радиочастотные, акустические и оптические подходы. Проведён анализ существующих решений и их ограничений. Разработана методология исследования, включающая создание специализированных датасетов, обучение и

тестирование моделей. Эксперименты выполнены в среде Google Colab с использованием GPU NVIDIA Tesla T4. Программная реализация осуществлена на Python 3.9 с библиотеками PyTorch и OpenCV 4.8.0. Полученные результаты подтверждают эффективность предложенного подхода в реальных условиях.

Ключевые слова: беспилотные летательные аппараты; компьютерное зрение; распознавание объектов; отслеживание; прогнозирование траектории движения; YOLO; CSRT; OpenCV.

Финансирование: работа выполнена в рамках государственного задания в сфере научной деятельности Министерства науки и высшего образования Российской Федерации (тема FZUN-2024-0020, госзадание ВлГУ).

A COMPREHENSIVE APPROACH TO DETECTION AND TRACKING OF UNMANNED AERIAL VEHICLES BASED ON COMPUTER VISION

A.S. Golubev[✉], V. Ilamanova, M.Z. Nguyen, D.M. Soshnikov, A.A. Khilenko

Vladimir State University named after Alexander and Nikolai Stoletovs,
Vladimir, Russia

✉ andrey.golubev@vlsu.ru

Citation: Golubev A.S., Ilamanova V., Nguyen M.Z., Soshnikov D.M., Khilenko A.A. A comprehensive approach to detection and tracking of unmanned aerial vehicles based on computer vision // Trudy MAI. 2026. No. 148. (In Russ.). URL: <https://trudymai.ru/published.php?ID=188283>

Abstract. A comprehensive approach to countering unmanned aerial vehicles is proposed, covering all processing stages: detection, classification, tracking, and trajectory prediction. At the detection stage, a comparative study of modern neural network architectures (YOLOv8, YOLOX, SSD) was conducted, where the YOLOX model showed the highest accuracy with mAP50=0.964. Convolutional neural networks were used for UAV type classification, providing accuracy up to 91.35%. At the tracking stage, the CSRT algorithm was recognized as optimal, demonstrating robustness to occlusions and lighting changes. For trajectory prediction, polynomial regression showed the best results with an average error of 1.27%. Created test datasets and fine-tuning of models using data augmentation were performed. The results can be applied in airspace monitoring and protection systems. The work considers traditional and modern UAV detection methods, including radio frequency, acoustic, and optical approaches. An

analysis of existing solutions and their limitations is carried out. A research methodology is developed, including the creation of specialized datasets, training, and testing of models. Experiments were performed in the Google Colab environment using NVIDIA Tesla T4 GPU. The software implementation was carried out in Python 3.9 with PyTorch and OpenCV 4.8.0 libraries. The obtained results confirm the effectiveness of the proposed approach in real conditions.

Keywords: unmanned aerial vehicles; computer vision; object detection; tracking; trajectory prediction; YOLO; CSRT; OpenCV.

Funding: the work was carried out as part of the state assignment in the field of scientific activity of the Ministry of Science and Higher Education of the Russian Federation (topic FZUN-2024-0020, state assignment of Vladimir State University).

Введение

Беспилотные летательные аппараты (БПЛА) стали неотъемлемой частью современных технологий, находя применение в доставке грузов, аэрофотосъемке, сельском хозяйстве, а также в военных и охранных задачах. Их компактные размеры, высокая маневренность и способность функционировать в сложных условиях делают их универсальным инструментом. Однако широкое распространение БПЛА порождает новые проблемы, связанные с безопасностью воздушного пространства.

Традиционные методы обнаружения, такие как радиолокация, часто оказываются неэффективными для идентификации маленьких дронов, особенно в условиях городской застройки или природных ландшафтов. При этом проблемы связаны не только с их малыми размерами, но и сходством с другими объектами, например, птицами. Это создает потребность в системах, способных надежно обнаруживать и отслеживать БПЛА в реальном времени. Компьютерное зрение, основанное на алгоритмах глубокого обучения, предлагает перспективное решение, сочетающее высокую точность с возможностью обработки больших объемов визуальных данных.

Задача противодействия БПЛА логически декомпозируется на четыре взаимосвязанных этапа: обнаружение, классификация, отслеживание и

предсказание траектории. Этап обнаружения предполагает идентификацию наличия БПЛА в поле зрения системы и является базовым для последующей обработки. Классификация позволяет определить тип БПЛА (мультикоптер, самолетного типа и др.), что критически важно для оценки потенциальной угрозы и выбора адекватных мер противодействия. Отслеживание обеспечивает непрерывное определение пространственного положения БПЛА в последовательности видеок кадров, формируя траекторию движения. Предсказание траектории на основе накопленных данных о движении позволяет осуществлять прогноз будущего местоположения БПЛА, что необходимо для упреждающего реагирования и планирования контрмер. Каждый из этих этапов представляет собой самостоятельную научно-техническую задачу, требующую специализированных алгоритмов и методов обработки данных.

В области обнаружения объектов наиболее эффективными показали себя сверточные нейронные сети, в частности архитектуры YOLO (You Only Look Once), SSD (Single Shot MultiBox Detector) и Faster R-CNN (Region-based Convolutional Neural Networks). Каждая из этих архитектур имеет свои преимущества и недостатки в контексте задачи обнаружения БПЛА. YOLO демонстрирует высокую скорость обработки, что критически важно для систем реального времени, в то время как Faster R-CNN обеспечивает более высокую точность ценой вычислительной сложности.

Для задач отслеживания объектов на видео часто используются алгоритмы трекинга, такие как CSRT (Channel and Spatial Reliability Tracking), KCF (Kernelized Correlation Filters), MOSSE (Minimum Output Sum of Squared Error) и MIL (Multiple Instance Learning). Эти алгоритмы различаются по скорости работы, точности и устойчивости к различным условиям съемки.

В данной работе предлагается комплексный подход к обнаружению и отслеживанию БПЛА, сочетающий современные методы детекции на основе глубокого обучения с эффективными алгоритмами трекинга. Проведено сравнительное исследование различных архитектур нейронных сетей и алгоритмов трекинга, разработаны методы улучшения качества работы системы за счет аугментации данных и дообучения моделей.

Обзор существующих решений

Существующие решения для обнаружения и отслеживания БПЛА можно систематизировать по используемым физическим принципам и методам обработки данных.

Радиочастотные и радиолокационные системы играют ключевую роль в обнаружении БПЛА. Радары (РЛС) эффективны для обнаружения летящих объектов по отраженному радиосигналу на больших расстояниях и в различных погодных условиях [1]. Однако их эффективность снижается при работе с малогабаритными БПЛА на предельно малых высотах, а сами системы отличаются высокой стоимостью и сложностью эксплуатации [1, 2]. Пассивной альтернативой являются системы радиоперехвата, которые обнаруживают дроны по анализу сигналов управления и передачи видео. Их ключевое преимущество — возможность определения местоположения оператора БПЛА, хотя они неэффективны против автономных аппаратов и дронов с зашифрованными каналами связи [1]. Современные исследования также ориентированы на использование систем связи, таких как LTE, в качестве пассивных радаров (пассивное радиолокационное зондирование) и применение технологий совместного использования связи и зондирования (Joint Communications and Sensing, JC&S) [2].

Акустические системы предлагают альтернативный подход, основанный на регистрации и анализе звуков, создаваемых двигателями и винтами БПЛА. Данный метод эффективен в условиях плохой видимости и не требует прямой видимости цели. Как показали российские разработки, современные системы, использующие аугментацию данных и трансферное обучение, способны точно распознавать тип БПЛА даже в условиях городского шума. Ограничения метода включают сравнительно небольшую дальность обнаружения (0,5–1 км) и чувствительность к погодным условиям, таким как ветер [3].

Оптический метод обнаружения и отслеживания БПЛА включает традиционные алгоритмы обработки изображений, глубокие нейронные сети и мультисенсорные подходы. Традиционные методы, такие как анализ контуров, детекция движения и фильтрация, ограничены низкой точностью в сложных

условиях, например, при наличии фонового шума или погодных помех [8]. Глубокие нейронные сети демонстрируют высокую эффективность благодаря способности обрабатывать сложные визуальные данные. Например, YOLO выделяется скоростью и простотой интеграции [4, 5, 7-10], тогда как Faster R-CNN обеспечивает более высокую точность, но требует больше вычислительных ресурсов [8]. SSD, хотя и легковесна, уступает по точности и сложна в настройке [9]. С другой стороны, эффективность нейросетевых моделей критически зависит от качества выполненного процесса обучения, в том числе от качества и количества обучающего материала.

Современным трендом становятся комплексные и гибридные системы, в которых объединение разнородных сенсоров (радаров, оптико-электронных систем, радиоперехвата и акустических датчиков) в единый контур позволяет компенсировать недостатки каждого отдельного метода. В то же время, подобный подход существенно увеличивает сложность и стоимость конечной системы [9].

Таким образом, современный арсенал средств противодействия БПЛА характеризуется разнообразием и тенденцией к комплексной интеграции. Методы компьютерного зрения, обсуждаемые далее, на текущий момент являются наиболее сбалансированным способом обеспечения требуемых характеристик точности и надежности обнаружения и отслеживания таких объектов.

Методология

Задача обнаружения БПЛА.

На основе проведенного обзора была составлена таблица сравнительной производительности и эффективности наиболее распространенных моделей для решения задачи обнаружения объектов на изображении в реальном времени (таблица 1).

Таблица 1

Сравнительные характеристики моделей обнаружения

Модель	Размер модели, Mb	mAP@50 (COCO)	FPS (GPU)	Лицензия	Сложность настройки
YOLOv8	-26	-0.54	-500+	AGPL-3.0 / Commercial	Низкая
YOLOX	-47	-0.52	-300+	Apache 2.0	Средняя
SSD	-90	-0.52	-200	MIT / Apache 2.0	Средняя / Высокая
Faster R-CNN	-180	-0.60	-20	Apache 2.0	Высокая

Учитывая эти данные, а также принимая во внимание специфику видеоизображений беспилотных летательных аппаратов, для сравнительного исследования были выбраны следующие архитектуры нейронных сетей: YOLOv8, YOLOX и SSD.

YOLOv8 использует современные сверточные блоки (в частности, C2f-модули на основе CSP-архитектуры) и эффективную структуру, обеспечивая высокую производительность при относительно небольшом количестве параметров. Модель реализует единый сквозной pipeline для детекции, сегментации и оценки позы, оптимизированный для инференса на устройствах с ограниченными ресурсами.

YOLOX внедряет anchor-free подход, упрощающий архитектуру и уменьшающий число гиперпараметров. Другими ключевыми особенностями являются механизм decoupled head, который позволяет избегать конфликта между задачами классификации и локализации, и динамическое назначение меток SimOTA. Благодаря этому YOLOX достигает высокой точности даже в компактных конфигурациях, таких как YOLOX-nano.

SSD использует одноэтапный подход с многоуровневыми картами признаков, что делает её менее ресурсоёмкой и пригодной для реального времени. На каждом уровне пирамиды модель применяет набор заранее заданных якорных рамок (priors) различных масштабов и соотношений сторон для предсказания объектов. Несмотря на простоту архитектуры, эффективность SSD сильно зависит от тщательного подбора параметров якорей и аугментации данных. Это делает её чувствительной к домену, но всё ещё актуальной для встраиваемых систем.

Обучение всех моделей проводилось на двух датасетах, целью которых была оценка влияния объема обучающей выборки на результат. Первый содержал 7000 изображений дронов, второй — 2000 изображений. Дополнительно был создан датасет из 600 изображений (200 уникальных кадров с тремя аугментациями). Аугментация данных включала: изменение яркости и контраста, повороты и масштабирование, добавление гауссова шума, горизонтальное отражение.

Задача классификации типов БПЛА.

Задачи обнаружения и классификации объектов тесно связаны: обе предполагают распознавание объектов из заранее заданного конечного множества классов. Задача обнаружения сводится к локализации и идентификации любого дрона как единого класса («БПЛА»), в то время как задача классификации требует различения между несколькими подклассами — например, по типу или модели конкретного аппарата.

При определённых условиях, обе задачи могут быть решены совместно с помощью одной многоцелевой модели — «многоклассового» детектора. Действительно, такие архитектуры, как YOLOv8, YOLOX и SSD, изначально поддерживают одновременное обнаружение и классификацию множества объектов. Тем не менее, в прикладных системах противодействия БПЛА такое объединение оказывается нецелесообразным по практическим соображениям. Обнаружение дрона является критически важной задачей и должно выполняться с минимальной задержкой, тогда как классификация, хотя и менее чувствительна к скорости, требует значительно более высокой точности из-за увеличенного числа классов и сходства визуальных признаков между моделями.

Исходя из этого разделения требований, для решения задачи классификации типов БПЛА были выбраны и исследованы три современные архитектуры сверточных нейронных сетей: ResNet, ResNeXt и EfficientNet. Эти модели демонстрируют высокую точность в задачах чистой классификации изображений при умеренных вычислительных затратах, что делает их предпочтительными для постобработки после этапа обнаружения.

ResNet [11] была разработана как решение проблемы затухания градиентов, характерной для глубоких нейронных сетей. Основная идея состоит в использовании остаточных (skip) связей, позволяющих передавать информацию и градиенты через большое количество слоёв, что значительно упрощает процесс обучения. Ключевые особенности:

- Остаточные блоки: выход операции свертки суммируется с входным тензором до применения нелинейности (ReLU). Это позволяет модели «пропускать» слои, не вносящие значимого вклада в улучшение результата.

- Глубина сети: возможность построения очень глубоких архитектур (ResNet-50, ResNet-101, ResNet-152), что обеспечивает извлечение сложных иерархических признаков.

- Устойчивое обучение: благодаря остаточным соединениям ResNet обучается быстрее и демонстрирует большую стабильность по сравнению с классическими CNN сопоставимой глубины.

ResNeXt [12] является развитием идей ResNet и основана на концепции увеличения кардинальности — числа параллельных путей в сверточном блоке. Ключевые особенности:

- Split-Transform-Merge: каждый блок разделяется на несколько параллельных «ветвей», результаты которых объединяются, что повышает выразительную способность модели.

- Кардинальность: увеличение количества параллельных путей в блоке даёт более заметный прирост качества, чем простое наращивание глубины или ширины сети.

- Эффективность: ResNeXt достигает более высокой точности при умеренном увеличении числа параметров по сравнению с ResNet.

EfficientNet [13] предлагает новый принцип масштабирования CNN, заключающийся в одновременном изменении глубины, ширины и разрешения входного изображения с помощью единого коэффициента. Ключевые особенности:

- Compound Scaling: сбалансированное масштабирование всех параметров модели (глубины, ширины, разрешения), что позволяет достичь высокой точности при оптимальном использовании ресурсов.

- Семейство моделей: от EfficientNet-B0 (базовая модель) до EfficientNet-B7, обеспечивающих последовательный рост качества и вычислительной сложности.

- Экономичность: при сопоставимой точности EfficientNet требует меньше параметров и вычислений (FLOPs), что делает её привлекательной для мобильных и встраиваемых систем.

Для обучения этих классификаторов был составлен датасет, состоящий из следующих классов: дельтовидные дроны (delta-wing), дроны самолетного типа (fixed-wing), гексакоптеры (hexacopter), квадрокоптеры (quadrocopter), посторонние объекты (not-drone). Для улучшения стабильности предсказаний реализован механизм пакетной обработки кадров с усреднением вероятностей.

Задача отслеживания БПЛА.

Для исследования задачи отслеживания дронов в реальном времени использовались алгоритмы, реализованные в библиотеке OpenCV. Их сравнительная характеристика [14] приведена в таблице 2.

Таблица 2

Сравнение алгоритмов трекинга OpenCV

Трекер	Скорость (FPS)	Точность	Устойчивость к перекрытиям	Сложность настройки
CSRT	20–30	Высокая	Высокая	Низкая
KCF	50–70	Средняя	Средняя	Низкая
MOSSE	100–150	Низкая	Низкая	Низкая
MIL	20–40	Средняя	Средняя	Средняя
GOTURN	20–30	Средняя	Низкая	Высокая

CSRT демонстрирует оптимальный баланс между точностью и скоростью, устойчив к перекрытиям и изменению освещения, что делает его подходящим для динамичных условий. В отличие от KCF, который быстрее, но менее точен, CSRT лучше справляется с изменяющимся фоном и временными потерями объекта из виду. MOSSE, хотя и быстрее, не подходит для сложных сценариев из-за низкой точности. MIL и GOTURN требуют больше ресурсов и сложнее в настройке, что делает их менее предпочтительными для задач реального времени. CSRT, напротив, использует параметры по умолчанию, которые хорошо работают в большинстве случаев, и не требует сложной настройки.

Все алгоритмы, кроме GOTURN, используют собственные аналитические модели. Алгоритм GOTURN использует предварительно обученную модель на основе сверточной нейронной сети. Поскольку подготовка такой модели представляет отдельную задачу, данный алгоритм был исключён из дальнейшего исследования.

Задача прогнозирования траектории движения.

Прогнозирование траектории является классической задачей экстраполяции временного ряда из координат движущегося объекта. Спецификой рассматриваемой сферы применения является то, что прогноз должен быть сравнительно краткосрочным (до 1–2 с). Это связано с тем, что дрон всё время находится в области видимости камеры.

Целями прогнозирования являются 1) предсказание движения дрона при его кратковременной потере и 2) компенсация возможных задержек в обработке видеоданных и предсказание *истинного* положения дрона в текущий момент.

С учётом данных факторов, были выбраны и исследованы следующие методы.

- Полиномиальная регрессия. Используется полином степени n для аппроксимации траектории:

$$x(t) = \sum_{k=0}^n a_k t^k, \quad y(t) = \sum_{k=0}^n b_k t^k.$$

Коэффициенты a_k, b_k определяются методом наименьших квадратов.

- Фильтр Калмана. Рекурсивный алгоритм оценки состояния динамической системы:

$$\hat{x}_k = A\hat{x}_{k-1} + Bu_k, \quad P_k = AP_{k-1}A^T + Q,$$

где A — матрица перехода, B — матрица управления, Q — ковариационная матрица шума процесса.

- Фильтр частиц. Метод Монте-Карло для аппроксимации апостериорного распределения:

$$p(x_k | z_{1:k}) \approx \sum_{i=1}^N w_i^{(k)} \delta(x_k - x_i^{(k)}),$$

где $w_i^{(k)}$ — веса частиц, $x_i^{(k)}$ — состояния частиц.

Экспериментальная часть

Тестирование моделей обнаружения.

Обучение моделей YOLOv8, YOLOX и SSD проводилось в среде Google Colab с использованием GPU NVIDIA Tesla T4. Для оценки их эффективности в реальных

условиях выполнялись эксперименты на серии видеороликов. Программная реализация экспериментов была осуществлена с использованием Python 3.9, библиотек PyTorch и OpenCV 4.8.0.

Удобство использования.

YOLOv8 выделяется исключительной простотой внедрения благодаря официальной поддержке Ultralytics. Библиотека ultralytics устанавливается одной командой (`pip install ultralytics`), а обучение модели требует минимального кода с указанием пути к датасету и параметров. Хорошо документированные Python-модули и поддержка инструментов, таких как Roboflow, упрощают разметку данных и создание аннотаций. Активное сообщество и регулярные обновления минимизируют проблемы с совместимостью, что делает YOLOv8 идеальной для проектов, где важна скорость разработки.

SSD, напротив, значительно сложнее в интеграции, так как не имеет официального Python-модуля, подобного YOLOv8. В данном исследовании использовался исходный код из репозитория <https://github.com/dusty-nv/pytorch-ssd.git>. Несмотря на относительную актуальность, он потребовал значительных усилий и затрат времени на настройку и отладку. Неполная документация и зависимость от устаревших библиотек усложняли процесс, часто приводя к ошибкам формата данных или совместимости. Кроме того, потребовалась ручная конвертация датасета в необходимую VOC-структуру.

YOLOX обладает высокой производительностью, но его внедрение связано с рядом сложностей по сравнению с YOLOv8. Отсутствие официального Python-модуля, устанавливаемого через `pip`, требует работы с исходниками и ручной настройки окружения, что увеличивает начальные затраты времени. Документация менее подробна и не всегда охватывает тонкости подготовки датасета, например, для COCO-формата. Для запуска обучения и валидации требуется редактирование множества конфигурационных файлов, таких как файлы экспериментов (`exps/default/yolox*.py`). В отличие от YOLOv8, где процесс упрощён благодаря интуитивному API и поддержке сторонних инструментов (Roboflow), YOLOX требует значительных усилий на первоначальную настройку и отладку.

Результаты на датасетах.

На первом датасете (7000 изображений) YOLOv8 достигла показателей $mAP_{50} = 0,963$ и $mAP_{50-95} = 0,703$ после 50 эпох обучения. Время инференса составило 10–40 мс, что соответствует 25–100 FPS. YOLOX показала схожие результаты: $mAP_{50} = 0,975$ и $mAP_{50-95} = 0,68$ при времени инференса 50–80 мс (12–20 FPS).

На втором датасете (2000 изображений) результаты YOLOv8 были ниже: $mAP_{50} = 0,806$, $mAP_{50-95} = 0,486$, что указывает на необходимость большего объема данных для качественного обучения.

SSD показала наихудшие результаты с $mAP_{50} = 0,671$ при времени обучения более 5 часов до 55 эпох. Низкая точность и высокое количество ложных срабатываний ограничивают её применение в реальных условиях.

Сравнительные результаты моделей обнаружения приведены в таблице 3.

Таблица 3

Сравнительные результаты моделей обнаружения

Модель	mAP_{50}	mAP_{50-95}	Время инференса, мс	Размер датасета
YOLOv8	0,963	0,703	10–40	7000
YOLOv8	0,806	0,486	10–35	2000
YOLOX	0,975	0,680	50–80	7000
SSD	0,671	0,423	40–70	7000

Влияние размера входного изображения.

Исследовано влияние параметра `imgsz` на качество детекции:

- При `imgsz=1280` улучшено обнаружение дальних дронов, но увеличено количество ложных срабатываний на антеннах и деревьях.
- При `imgsz=896` достигнут баланс между точностью и количеством ошибок.
- При `imgsz=640` обеспечена стабильность работы в ущерб точности обнаружения мелких объектов.

Оптимальным признано использование `imgsz=896` для обучения и инференса.

Дообучение.

Дообучение YOLOv8 на датасете из 600 изображений (50, 100, 150 эпох) выявило важные закономерности. Все модели показали схожие mAP ($\sim 0,8$), но вариант на 150 эпохах демонстрировал признаки переобучения. Модели на 50 и 100 эпохах различались незначительно, но вторая показала меньшую уверенность на ложных срабатываниях и большую — на верных.

После этих выводов датасет был дополнен до 1200 изображений, включая изображения с аугментациями и валидационную выборку. Дообучение было произведено на 50 эпохах для YOLOv8 и на 35 эпохах для YOLOX. Показатели mAP получились следующими:

- YOLOv8: mAP50=0,863, mAP50-95=0,551
- YOLOX: mAP50=0,964, mAP50-95=0,634

Проверка на тестовых изображениях и видео показала значительное улучшение результатов. Модель YOLOX продемонстрировала точность 96,4% на изображениях, а проблемы на видео вообще не были отмечены. Модель YOLOv8 отработала неплохо, но заметно хуже YOLOX — на видео всё ещё были кадры с ложноположительными срабатываниями, а на изображениях модель достигла 80% точности.

Классификация типов БПЛА.

Результаты классификации типов БПЛА приведены в таблице 4.

Таблица 4

Результаты классификации типов БПЛА

Архитектура	Точность	Скорость, с
ResNet-50	0,9038	0,19
ResNeXt-50	0,9135	0,30
EfficientNet	0,871	0,17

Эксперименты по классификации проводились на датасете из 712 изображений, распределённых по шести классам. Используются архитектуры ResNet, ResNeXt и EfficientNet с transfer learning. По совокупности характеристик наиболее сбалансированным выбором будет архитектура ResNet-50. Наибольшие

трудности возникли при классификации классов hexacopter и fixed-wing_drone из-за ограниченного количества образцов данных классов в датасете.

Сравнительный анализ алгоритмов трекинга.

Проведено тестирование алгоритмов трекинга на шести сценариях: городская застройка, небо, лес, человек, ночные условия, морская поверхность.

Устойчивость (%) и производительность (FPS) алгоритмов трекинга в различных сценариях приведены в таблице 5.

Таблица 5

Устойчивость (%) и производительность (FPS) алгоритмов трекинга в различных сценариях

Сценарий	CSRT		KCF		MOSSE		MIL	
	%	FPS	%	FPS	%	FPS	%	FPS
Городская застройка	99,6	14,47	99,6	107,34	99,6	234,16	84,7	7,38
Небо	42,4	30,34	42,4	160,95	42,4	314,50	41,6	12,57
Лес	100,0	14,74	95,2	102,87	100,0	214,76	98,6	7,63
Ночь	100,0	13,27	100,0	73,15	100,0	246,05	100,0	7,40
Море	92,1	16,58	92,1	112,27	92,1	203,96	92,1	7,43

Анализ устойчивости трекеров показал, что CSRT рекомендуется для сложных условий, KCF и MOSSE — для систем с ограниченными вычислительными ресурсами.

Оценка методов прогнозирования траектории.

Проведено сравнение четырёх методов прогнозирования с использованием скользящего окна в 15 кадров (таблица 6).

Таблица 6

Прогнозирование траектории движения

Метод	Средняя ошибка, %	Скорость, FPS	Устойчивость к шуму
Полиномиальная регрессия	1,27	10,0	Средняя
Фильтр Кальмана	1,93	10,6	Высокая
Фильтр частиц	1,67	10,7	Очень высокая

Полиномиальная регрессия показала наилучшую точность со средней ошибкой 9,53 пикселя при времени обработки 100 мс на кадр.

Заключение

В результате проведённого исследования разработан комплексный подход к обнаружению и отслеживанию БПЛА. Основные полученные результаты:

1. Модель YOLOX определена как наиболее эффективная с $mAP50=0,964$ после дообучения на расширенном датасете.
2. Алгоритм CSRT рекомендован для сложных условий, а KCF и MOSSE — для систем с ограниченными ресурсами.
3. Составлен датасет и проведён анализ моделей классификации типов БПЛА с точностью до 91,35%.
4. Полиномиальная регрессия показала преимущества для краткосрочного прогнозирования со средней ошибкой 1,27%.

Перспективы дальнейших исследований связаны с разработкой адаптивных алгоритмов и созданием специализированных датасетов.

Конфликт интересов

Авторы заявляют об отсутствии конфликта интересов.

Conflict of interest

The authors declare no conflict of interest.

Список источников

1. Основные методы обнаружения и глушения дронов (БПЛА) // АО «ТРАНЗАС Консалтинг». - Июнь 2025. URL: https://trancons.ru/information/stati_i_publicacii/drone-supression.
2. Khawaja W., Ezuma M., Semkin V., Erden F., Ozdemir O., Guvenc I. A Survey on Detection, Classification, and Tracking of UAVs Using Radar and Communications Systems // IEEE Communications Surveys & Tutorials. 2025. DOI: <https://doi.org/10.1109/COMST.2025.3554613>.
3. Разработана система для обнаружения БПЛА над городом по издаваемым звукам // CNews. 2025. 6 августа. URL: https://www.cnews.ru/news/top/2025-08-06_razrabotana_sistema_dlya.

4. Бахвалов И.М., Старцев С.В. Распознавание БПЛА различных классов средствами компьютерного зрения // НИТУ «МИСиС». Москва, 2025. URL: <https://www.researchgate.net/publication/387896752>.
5. Кульчицкий А.И., Макарьчук Н.С., Калашников Г.А. Использование компьютерного зрения для распознавания техники // Научный аспект. Т.18 2023. № 5. - С.2245-2253 URL: <https://na-journal.ru/5-2023-informacionnye-tehnologii/5535-ispolzovanie-kompyuternogo-zreniya-dlya-raspoznavaniya-tehniki>.
6. Liu, Z.; An, P.; Yang, Y.; Qiu, S.; Liu, Q.; Xu, X. Vision-Based Drone Detection in Complex Environments: A Survey. Drones 2024, 8, 643. DOI: <https://doi.org/10.3390/drones8110643>.
7. Alshaer, N., Abdelfatah, R., Ismail, T. and Mahmoud, H. (2025), Vision-Based UAV Detection and Tracking Using Deep Learning and Kalman Filter. Computational Intelligence, 41: e70026. DOI: <https://doi.org/10.1111/coin.70026>.
8. Ramachandran, Anitha Sangaiah, Arun. (2021). A Review on Object Detection in Unmanned Aerial Vehicle Surveillance. International Journal of Cognitive Computing in Engineering. 2. DOI: <https://doi.org/10.1016/j.ijcce.2021.11.005>.
9. Unlu E., Zenou E., Riviere N. et al. Deep learning-based strategies for the detection and tracking of drones using several cameras // IPSJ Transactions on Computer Vision and Applications. 2019. Vol. 11. Art. 7. DOI: <https://doi.org/10.1186/s41074-019-0059-x>.
10. Raed Abu Zitar, Mohammad Al-Betar, Mohamad Ryalat, Sofian Kassaymeh. (2023) A review of UAV Visual Detection and Tracking Methods. — arXiv:2306.05089 [cs.CV]. URL: <https://arxiv.org/pdf/2306.05089>.
11. K. He, X. Zhang, S. Ren and J. Sun, Deep Residual Learning for Image Recognition (2016) IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 770-778. DOI: <https://doi.org/10.1109/CVPR.2016.90>.
12. Xie S., Girshick R., Dollár P., Tu Z., He K. Aggregated Residual Transformations for Deep Neural Networks // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 1492-1500. DOI: <https://doi.org/10.1109/CVPR.2017.634>.

13. Tan, M. and Le, Q.V. (2019) Efficient Net: Rethinking Model Scaling for Convolutional Neural Networks. Proceedings of the 36th International Conference on Machine Learning, ICML 2019, Long Beach, 9-15 June 2019, pp. 6105-6114. URL: <http://proceedings.mlr.press/v97/tan19a.html>.

14. Yurushkin M. A Complete Review of the OpenCV Object Tracking Algorithms // Brouton Lab. URL: <https://broutonlab.com/blog/opencv-object-tracking>.

References

1. Main Methods of Detection and Jamming of Drones (UAVs), AO "TRANZAS Consulting", June 2025 (in Russian). URL: https://trancons.ru/information/stati_i_publicacii/drone-supression.

2. Khawaja W., Ezuma M., Semkin V., Erden F., Ozdemir O., Guvenc I. A Survey on Detection, Classification, and Tracking of UAVs Using Radar and Communications Systems // IEEE Communications Surveys & Tutorials. 2025. DOI: <https://doi.org/10.1109/COMST.2025.3554613>.

3. A system for detecting UAVs over the city by the sounds they emit has been developed, CNews, August 6, 2025 (in Russian). URL: https://www.cnews.ru/news/top/2025-08-06_razrabotana_sistema_dlya.

4. Bakhalov I. M., Startsev S. V. Recognition of UAVs of various classes by means of computer vision, NITU "MISIS", Moscow, 2025 (in Russian). URL: <https://www.researchgate.net/publication/387896752>.

5. Kul'chitskiy A.I., Makarchuk N.S., Kalashnikov G.A. Using computer vision for equipment recognition, Nauchnyi Al'manakh, 2023. Vol. 18, no. 5, pp. 2245-2253. (in Russian). URL: <https://na-journal.ru/5-2023-informacionnye-tehnologii/5535-ispolzovanie-kompyuternogo-zreniya-dlya-raspoznavaniya-tehniki>.

6. Liu, Z.; An, P.; Yang, Y.; Qiu, S.; Liu, Q.; Xu, X. Vision-Based Drone Detection in Complex Environments: A Survey. Drones 2024, 8, 643. DOI: <https://doi.org/10.3390/drones8110643>.

7. Alshaer, N., Abdelfatah, R., Ismail, T. and Mahmoud, H. (2025), Vision-Based UAV Detection and Tracking Using Deep Learning and Kalman Filter. Computational Intelligence, 41: e70026. DOI: <https://doi.org/10.1111/coin.70026>.

8. Ramachandran, Anitha Sangaiah, Arun. (2021). A Review on Object Detection in Unmanned Aerial Vehicle Surveillance. International Journal of Cognitive Computing in Engineering. 2. DOI: <https://doi.org/10.1016/j.ijcce.2021.11.005>.
9. Unlu E., Zenou E., Riviere N. et al. Deep learning-based strategies for the detection and tracking of drones using several cameras // IPSJ Transactions on Computer Vision and Applications. 2019. Vol. 11. Art. 7. DOI: <https://doi.org/10.1186/s41074-019-0059-x>.
10. Raed Abu Zitar, Mohammad Al-Betar, Mohamad Ryalat, Sofian Kassaymeh. (2023) A review of UAV Visual Detection and Tracking Methods. arXiv:2306.05089 [cs.CV]. URL: <https://arxiv.org/pdf/2306.05089>.
11. K. He, X. Zhang, S. Ren and J. Sun, Deep Residual Learning for Image Recognition (2016) IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 770-778. DOI: <https://doi.org/10.1109/CVPR.2016.90>.
12. Xie S., Girshick R., Dollár P., Tu Z., He K. Aggregated Residual Transformations for Deep Neural Networks // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017, pp. 1492-1500. DOI: <https://doi.org/10.1109/CVPR.2017.634>.
13. Tan, M. and Le, Q.V. (2019) Efficient Net: Rethinking Model Scaling for Convolutional Neural Networks. Proceedings of the 36th International Conference on Machine Learning, ICML 2019, Long Beach, 9-15 June 2019, pp. 6105-6114. URL: <http://proceedings.mlr.press/v97/tan19a.html>.
14. Yurushkin M. A Complete Review of the OpenCV Object Tracking Algorithms // Brouton Lab. URL: <https://broutonlab.com/blog/opencv-object-tracking>.

Информация об авторах

Андрей Сергеевич Голубев, доцент кафедры физики и прикладной математики ВлГУ, г. Владимир, Россия; e-mail: andrey.golubev@vlsu.ru

Виктория Иламанова, студент ВлГУ, г. Владимир, Россия

Максим Зунгович Нгуен студент ВлГУ, г. Владимир, Россия

Дмитрий Максимович Сошников студент ВлГУ, г. Владимир, Россия

Хиленко Алексей Антонович студент ВлГУ, г. Владимир, Россия

Information about the authors

Andrey S. Golubev, Associate Professor at the Department of Physics and Applied Mathematics, VLSU, Vladimir, Russia; e-mail: andrey.golubev@vlsu.ru

Victoria Ilamanova, student of VLSU, Vladimir, Russia

Maxim Zungovich Nguyen, student of VLSU, Vladimir, Russia

Dmitry M. Soshnikov, student of VLSU, Vladimir, Russia

Alexey A. Khilenko, student of VLSU, Vladimir, Russia

Получено 30 марта 2026 ● Принято к публикации 05 июня 2026 ● Опубликовано 25 июня 2026
Received 30 March 2026 ● Accepted 05 June 2026 ● Published 25 June 2026
