

Система интеллектуального мониторинга дефектов картонной упаковки на основе компьютерного зрения

К.А. Полянцева, А.Е. Сергеев

Ордена Трудового Красного Знамени федеральное государственное бюджетное образовательное учреждение высшего образования «Московский технический университет связи и информатики»

Аннотация: В статье рассматривается разработка системы мониторинга для автоматического обнаружения и классификации дефектов картонной упаковки на складе. Предлагаемое решение представляет собой веб-приложение, использующее нейросетевую модель YOLOv8 для детекции дефектов в реальном времени. Модель обучена на кастомном датасете изображений дефектных и исправных коробок. Выполнена интеграция модели в приложение на Flask и добавлен аналитический модуль (Apache Superset) для визуализации статистики. Испытания системы в различных сценариях подтвердили ее высокую точность и эффективность в улучшении контроля качества упаковки.

Ключевые слова: компьютерное зрение, обнаружение дефектов, картонная упаковка, YOLOv8, глубокое обучение, система мониторинга, видеоаналитика.

Введение

В современном складском хозяйстве качество упаковки товаров является критически важным фактором, влияющим на сохранность продукции и удовлетворенность клиентов. Некачественная или повреждённая картонная тара повышает риск порчи товара при транспортировке и хранении, что напрямую ведёт к финансовым потерям для компаний и снижению доверия потребителей.

Особенно актуальна эта проблема в условиях стремительного роста электронной коммерции, когда объёмы посылок возрастают, а требования к оперативности и надёжности доставки ужесточаются. Автоматизация контроля качества упаковки с помощью технологий компьютерного зрения и глубокого обучения способна существенно снизить человеческий фактор и повысить эффективность логистических процессов. В последние годы появляются решения по визуальному контролю на производстве и складах, применяющие сверточные нейросети для обнаружения дефектов продукции в

реальном времени [1-3]. Известны разработки промышленных систем, например, на основе моделей семейства YOLO, показывающие высокую точность детекции дефектов упаковки на конвейерных линиях [4]. Эффективность применения нейронных сетей в задачах компьютерного зрения подтверждается современными исследованиями [5].

Цель исследования – разработать автоматизированную систему мониторинга, способную обнаруживать дефекты картонных коробок и классифицировать их состояние (дефектная или нормальная упаковка) в режиме реального времени на складе. Для достижения этой цели в работе реализованы следующие этапы:

1. Анализ предметной области и существующих решений в области автоматического контроля упаковки;
2. Разработка модели детекции дефектов на базе современной архитектуры YOLOv8, включая сбор и разметку данных, обучение и оценку качества модели;
3. Разработка веб-системы мониторинга: интеграция обученной модели в серверное приложение (Flask), организация приема видеопотока с камер наблюдения, отображение результатов детекции и сбор данных;
4. Добавление модуля аналитики на базе BI-платформы (Apache Superset) для визуализации статистики обнаруженных дефектов и нормальных упаковок;
5. Тестирование и апробация системы в различных условиях, анализ результатов и формулировка выводов.

Выбор нейросетевой модели

Для решения задачи детектирования дефектов был выбран подход на основе свёрточных нейронных сетей, обеспечивающий одновременно высокую точность и быстродействие. В частности, использована модель YOLOv8 – новейшая архитектура из семейства You Only Look Once,

представленная компанией Ultralytics в 2023 году [6, 7]. Семейство YOLO зарекомендовало себя как эффективное решение для детекции объектов в реальном времени благодаря объединению этапов обнаружения и классификации в единой нейросетевой модели. Последняя версия YOLOv8 включает усовершенствования, повышающие точность и скорость по сравнению с предшественниками (YOLOv5/YOLOv7 и др.). Архитектура YOLOv8 отличается использованием модифицированного CSP-подхода и расширенного механизма прогнозирующей головы, что улучшает детекцию мелких объектов и обобщающую способность модели. Кроме того, YOLOv8 поддерживает различные масштабные версии модели (n, s, m, l, x) с разным числом параметров, позволяя подобрать баланс между точностью и производительностью. Модели семейства YOLO продолжают оставаться одними из лидеров по сочетанию скорости и качества среди алгоритмов компьютерного зрения. Учитывая эти преимущества, YOLOv8 была выбрана в качестве основы для нашей системы мониторинга.

Датасет и аннотирование данных

Для обучения модели потребовался набор данных изображений картонных коробок с различными дефектами, а также без дефектов (для распознавания нормальной упаковки). В рамках работы был сформирован кастомный датасет из фотографий, полученных в складских условиях (камеры видеонаблюдения), а также изображений из открытых источников (публичные датасеты) с дополнительной разметкой [8, 9]. Все изображения были размечены вручную: на дефектных коробках выделены области повреждений или деформаций, после чего каждой коробке присвоен класс «дефектная» или «не дефектная». Разметка выполнялась с помощью специализированных инструментов (например, Roboflow) для удобства и единообразия формата данных. Полученный датасет содержал изображения коробок в различных условиях: при разном освещении, с разных ракурсов,

включая как отдельные коробки, так и сцены с несколькими объектами (например, паллеты с коробками на складе). Наличие разнообразных сцен обеспечило модели возможность обучиться распознавать дефекты в сложной обстановке реального склада.

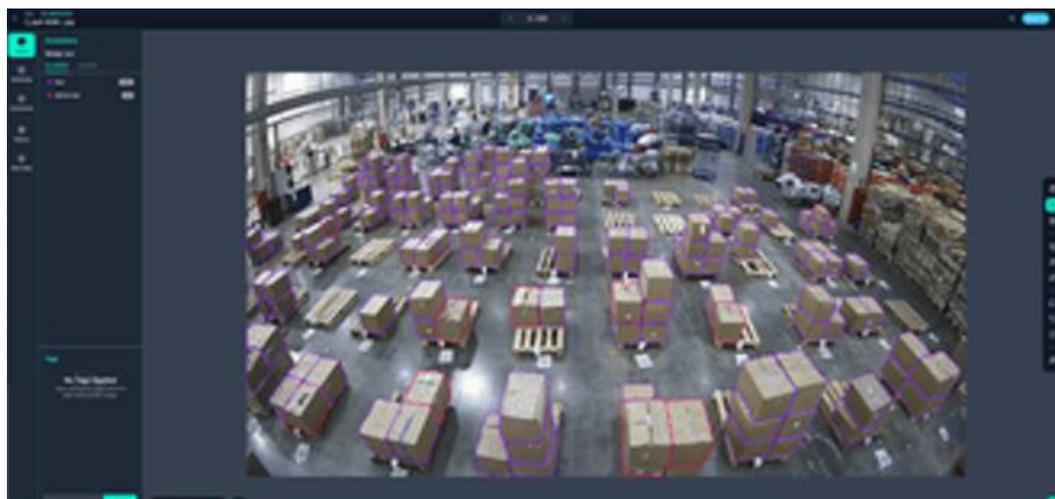


Рис. 1. – Аннотация изображений

Чтобы повысить устойчивость модели к вариациям условий съёмки, к исходным изображениям были применены методы аугментации данных [10]. В процессе подготовки набора данных реализованы следующие преобразования:

1. Изменение ориентации (Auto-Orient): автоматическое выравнивание угла наклона изображений для унификации ориентации объектов.
2. Масштабирование (Resize): приведение разрешения всех изображений к единому размеру (640×640 пикселей), что соответствует входному формату модели.
3. Конвертация в градации серого (Grayscale): случайное преобразование ~15% изображений в черно-белый формат для повышения устойчивости модели к изменениям цветового тона сцены.

4. Изменение яркости/контрастности: варьирование экспозиции некоторых изображений, имитируя различное освещение на складе.

5. Добавление шумов и искажение: синтетическое добавление гауссовского шума, небольших случайных смещений и поворотов, чтобы расширить разнообразие входных данных.

Применение комплекса аугментаций позволило существенно увеличить и разнообразить обучающий набор без привлечения новых размеченных данных. Это, в свою очередь, улучшает обобщающую способность модели – согласно результатам, модель успешно детектирует дефекты даже в условиях, несколько отличных от тех, что присутствовали в изначальном датасете (например, при ином освещении или фоновых объектах).

Обучение модели

Для обучения модели использовалась библиотека Ultralytics YOLO на языке Python. На этапе экспериментов были опробованы две конфигурации модели – YOLOv8m и более крупная YOLOv8l (medium и large версии). Обе модели инициализировались предобученными весами (на наборе COCO), после чего были дообучены на нашем специализированном датасете дефектов упаковки. Обучение проводилось на графическом ускорителе NVIDIA (в работе – GPU типа NVIDIA TITAN RTX) в течение 100 эпох для каждой модели. Параметры: `yolo task=detect mode=train model=yolov8m/l-ov7.pt imgsz=640 data=data.yaml epoch=100 batch=-1 name=yolov8m/l_custom`.

В ходе обучения отслеживались ключевые метрики качества: точность (precision), полнота (recall), средняя точность (mAP) и функция полезности (fitness), рассчитываемые на проверочном подмножестве данных после каждой эпохи. Контроль процесса обучения осуществлялся с помощью системы ClearML, которая позволила мониторить показатели качества модели и нагрузку на вычислительные ресурсы в реальном времени.



Рис. 2. – Нагрузка на вычислительные ресурсы

По завершении обучения обе конфигурации модели продемонстрировали высокие результаты: например, достигнута mAP@50 около 0.98 и mAP@50-95 порядка 0.95–0.96 на тестовых данных, что свидетельствует о способности модели практически безошибочно распознавать дефекты упаковки.

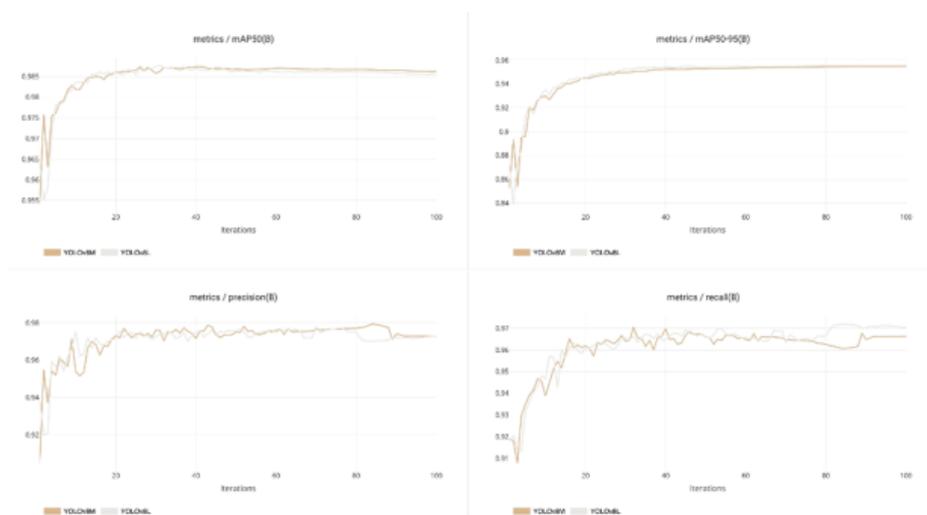


Рис. 3. – График обучения

Для сравнения, более лёгкая модель YOLOv8m обучалась быстрее (8 часов против 10 часов у YOLOv8l) и показала близкие значения точности. Разница между метриками двух моделей оказалась незначительной (например, mAP@50 98.63% vs 98.55% у большей модели), поэтому для дальнейшей интеграции была выбрана модель YOLOv8m как обеспечивающая лучший компромисс между скоростью и качеством.

Таблица № 1

Сравнительная таблица обученных моделей

Название модели	YOLOv8L	YOLOv8M
Время обучения (часы)	10	8
Метрика точности	0.9729	0.9728
Метрика полноты	0.9704	0.9662
Средняя точность (mAP50)	0.9855	0.9863
Средняя точность (mAP50-95)	0.9557	0.9547
Показатель эффективности (fitness)	0.9587	0.9579
Количество параметров модели	43,631,382	25,857,478
Вычислительная сложность (GFLOPs):	165.407	79.069
Скорость обработки на PyTorch (мс):	3.6	3.519

Итоговые веса модели были сохранены и экспортированы для дальнейшего развёртывания. Отметим, что благодаря возможностям фреймворка Ultralytics, обученную модель при необходимости можно экспортировать в оптимизированные форматы (ONNX, TensorRT и др.) для ускорения инференса на целевой платформе.

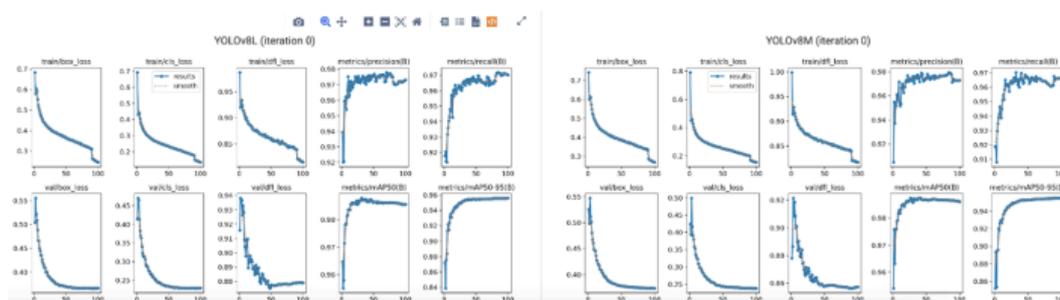


Рис. 4. – Метрики обучения моделей

Разработка системы мониторинга

Разработанная система мониторинга предназначена для приёма видеопотока с камер, установленных на складе, детектирования на каждом кадре дефектных коробок и отображения результатов в удобном для персонала виде. Архитектура решения включает несколько основных компонентов:

1. Камеры наблюдения – источники видео с зон складского хранения (например, зона приёмки/кросс-док, участок комплектации заказов и т.д.). Камеры передают потоковое видео (RTSP-поток) на сервер системы.

2. Серверное приложение (backend) – реализовано с помощью фреймворка Flask (Python). Сервер отвечает за прием видеопотока, разбивку его на кадры и проведение инференса нейросетевой модели на каждом кадре. В сервер встроена загруженная модель YOLOv8, которая выполняет обнаружение дефектных коробок на изображении и возвращает координаты обнаруженных объектов и класс («дефектная» / «нормальная» упаковка). Для работы с видеоизображениями используется библиотека OpenCV.

3. База данных – для хранения результатов детекции применяется СУБД PostgreSQL. В базу в режиме реального времени сохраняются записи о выявленных дефектах: время обнаружения, идентификатор камеры/зоны, степень уверенности модели, координаты обнаруженного объекта и др.

Также могут сохраняться кадры (или ссылки на них) с зафиксированными дефектами для последующего анализа.

4. Веб-интерфейс (frontend) – клиентская часть, через которую пользователи (сотрудники склада или менеджеры) взаимодействуют с системой. Интерфейс реализован как веб-страница, обслуживаемая Flask [12, 13]. В реальном времени на странице отображается видео с пометкой обнаруженных дефектов (рамки вокруг дефектных коробок, метки классов и процентов уверенности). Также интерфейс содержит элементы управления (например, выбор камеры или зоны наблюдения, запуск/остановка анализа) и информационные панели.

5. Модуль аналитики – подсистема на базе платформы Apache Superset, подключенная к основной базе данных. Она предназначена для построения дашбордов и графиков по накопленным данным о дефектах. Данный модуль не участвует в процессе детекции напрямую, но использует сохранённую статистику для генерации отчетов (например, график количества выявленных дефектов по дням/часам, доля дефектных упаковок в общем потоке и т.п.). Это дает возможность руководству получать сводную информацию о качестве упаковки на складе.

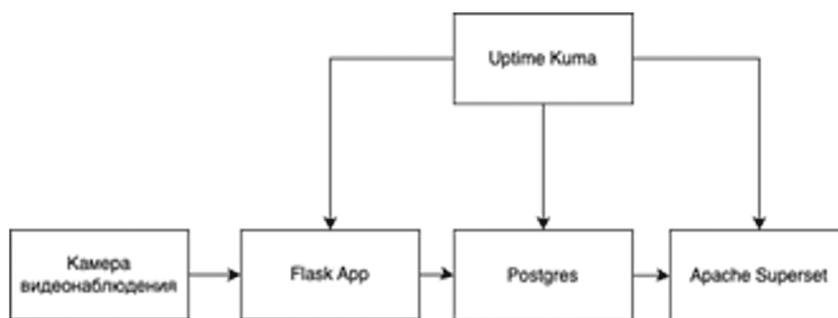


Рис. 5. – Архитектура системы

На Рис. 5 представлена упрощённая архитектура сервисов системы мониторинга: камера передает данные на сервер Flask, где работает модель YOLOv8; результаты записываются в базу, а также отправляются на

фронтенд для отображения, и параллельно могут визуализироваться в Superset-дашборде.

Интеграция модели и backend. [11] С целью обеспечения работы детектирующей модели в режиме реального времени в серверное приложение Flask был интегрирован модуль с нейронной сетью. Модель YOLOv8 (вариант yolov8m) загружается через интерфейс библиотеки Ultralytics (method YOLO(model_path)), после чего готова принимать на вход кадры изображений. При поступлении каждого нового кадра от камеры сервер выполняет предобработку (масштабирование до 640×640, нормализацию), подаёт изображение в модель и получает предсказания – список обнаруженных объектов с координатами bounding box и классами. Для ускорения обработки модель работает в режиме batch=1 на GPU, обеспечивая скорость инференса порядка 20–30 кадров/с, чего достаточно для типичных задач мониторинга. В реализации предусмотрены оптимизации, например: асинхронная обработка последовательных кадров (пока идёт распознавание на одном кадре, следующий кадр захватывается из потока), а также возможность задать порог уверенности, ниже которого детекции игнорируются (чтобы уменьшить ложные срабатывания). При обнаружении дефектной упаковки событие заносится в базу данных. Одновременно результаты передаются на фронтенд по протоколу WebSocket в формате JSON, что позволяет обновлять отметки на видео без перезагрузки страницы. Таким образом, достигается потоковая обработка: видео анализируется непрерывно с минимальной задержкой, и оператор видит маркировку дефектов практически в реальном времени. Подобный подход соответствует современным трендам «streaming perception», когда алгоритмы детекции оптимизируются для непрерывного видео, а не разрозненных изображений

Веб-интерфейс системы разработан с учётом удобства оператора складского контроля. При открытии веб-страницы пользователь может выбрать источник видео (определённую камеру или загруженный видеофайл для офлайн-анализа). После выбора камеры отображается текущий поток: кадры выводятся на экране, и при активированной модели сразу демонстрируются результирующие рамки вокруг обнаруженных дефектных коробок. Для каждой рамкой подписан класс (например, “defect”) и значение confidence (например, 95%), отражающее уверенность нейросети. В правой части интерфейса в табличной форме перечислены последние детекции с указанием времени и дополнительной информацией.

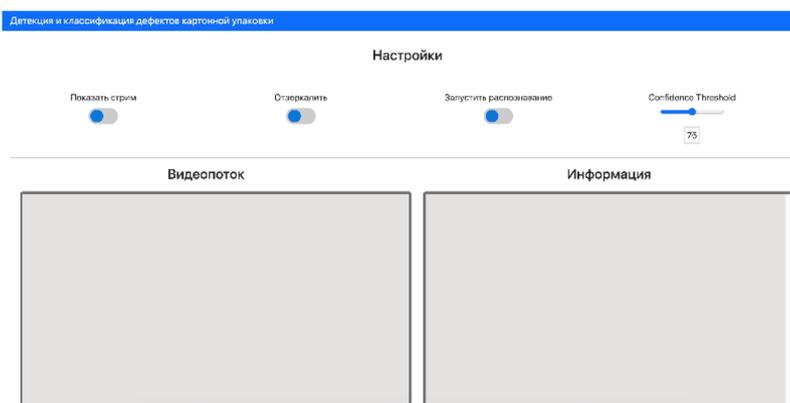


Рис. 6. – Веб-интерфейс

Предусмотрены элементы настройки: порог срабатывания модели по уверенности (ползунок), переключатель режима отображения (например, показывать или скрывать рамки, отзеркаливание видео потока при необходимости), кнопки запуска и остановки анализа.

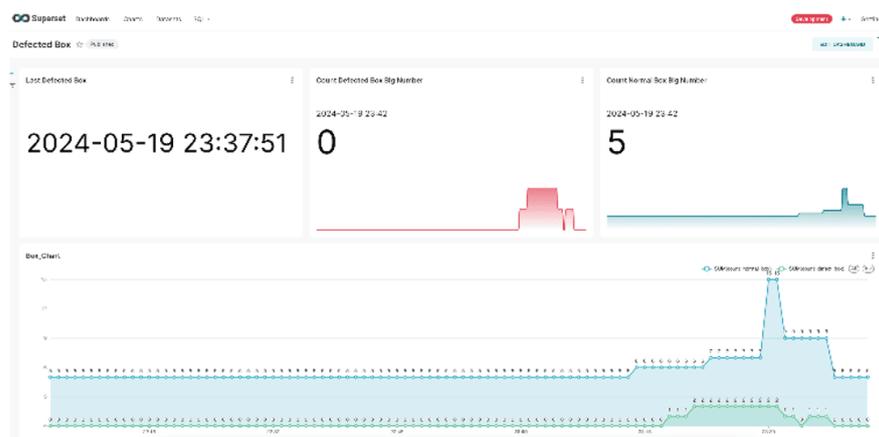


Рис. 8. – Superset дашборд

Совокупность этих визуализаций образует интерактивный дашборд качества упаковки, позволяющий ответственному персоналу быстро оценивать ситуацию. Например, с его помощью можно заметить, если в определённой партии товара доля дефектов превышает норму, и своевременно принять меры. Интеграция Superset выполняется через подключение к той же базе Postgres; обновление дашборда может производиться автоматически через заданные интервалы времени или вручную по запросу пользователя.

Тестирование системы

После разработки прототипа система мониторинга была подвергнута всестороннему тестированию в условиях, приближенных к реальным складским операциям. Тестирование проводилось на нескольких сценариях, охватывающих разные этапы логистического процесса и окружающие условия.

Сценарий 1. Складская зона хранения (кросс-док). Камера установлена на участке временного хранения поступающих грузов (кросс-докинг), где коробки перегружаются без распаковки. В этой зоне проверяется целостность упаковки при входном контроле. Система обрабатывала поток с камеры, охватывающей паллеты с коробками. В результате тестов модель успешно

идентифицировала дефекты типа вмятины, разрывы на углах коробок, даже когда несколько коробок находились в кадре одновременно. Ложные срабатывания практически отсутствовали: из тысяч проверенных коробок только единичные случаи нормальных упаковок были ошибочно помечены как дефектные (ошибка классификации $<1\%$), что свидетельствует о высокой специфичности. В то же время ни один явный дефект (например, большая пробоина) не остался пропущенным (чувствительность $\sim 100\%$ в данных испытаниях). Это подтверждает работоспособность модели в стеснённых условиях склада, при сложном фоне (полеты, техника) и разном освещении.



Рис. 9. – Тестирование модели «кросс-док»

Сценарий 2. Зона обработки заказов. Проверка осуществлялась на участке, где сотрудники комплектуют заказы и упаковывают товары. Здесь дефекты могут возникать при пересборке коробок или перегрузке товара. Камера контролировала конвейер или стол упаковки. Система корректно детектировала случаи неполного закрытия коробки лентой, сильные перегибы картона и другие нарушения. Важный результат – модель остаётся эффективной даже при частичном перекрытии коробок объектами (руками сотрудников, оборудованием): алгоритм уверенно распознавал видимые части дефекта. Это демонстрирует устойчивость детекции к окклюзиям. Кроме того, система работала в режиме реального времени без заметных

задержек, что позволило в некоторых случаях сразу же исправлять дефект (например, оператор повторно заклеивал коробку, узнав о проблеме из оповещения).



Рис. 10. – Тестирование модели «сортировка»

Сценарий 3. Курьерская доставка (выездные условия). Для проверки универсальности решение протестировано вне склада – на этапе доставки клиенту. Камера (например, смартфон курьера) снимала процесс передачи посылки. Модель успешно определяла поврежденные коробки перед вручением, что теоретически даёт возможность пометить отправление как проблемное ещё до передачи клиенту. В кадрах с участием человека система фокусировалась именно на упаковке, практически не выдавая ложных детекций на посторонние объекты (курьера, окружение). Это показывает, что модель обучилась отличать конкретно упаковки и их состояния.



Рис. 11. – Тестирование модели «курьерская доставка»

Пример работы модели в полевых условиях (этап доставки): на изображении система идентифицирует дефект (промятие) на коробке в руках курьера, выделяя область повреждения красной рамкой и классифицируя упаковку как дефектную. Благодаря высокой точности (уверенность ~94%) такой дефект не останется незамеченным.

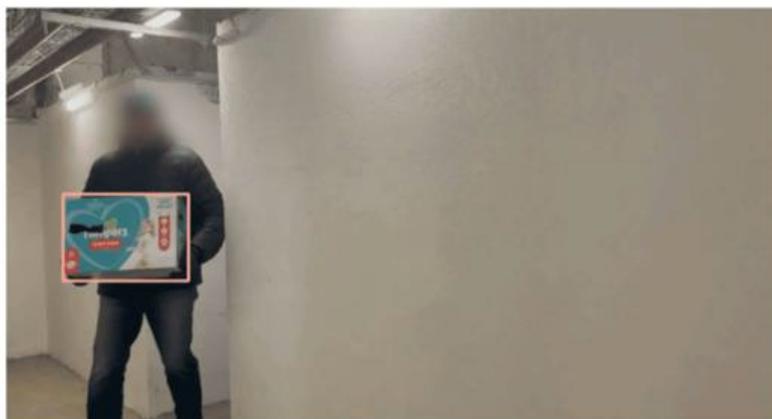


Рис. 12. – Тестирование модели «курьерская доставка в руках»

Обобщая результаты, можно сказать, что разработанная система продемонстрировала высокую эффективность: доля правильно распознанных дефектов превышает 95%, при этом скорость анализа (~25 FPS на одной видеокамере) достаточна для практического использования без пропуска коробок в потоке. Отдельно отметим удобство, которое дают возможности аналитического модуля: накопленные за неделю данные на тестовом складе позволили выявить, что подавляющее большинство дефектов происходит с коробками от определённого поставщика, что иллюстрирует практическую ценность системы для принятия управленческих решений.

Сравнение с альтернативными подходами показывает конкурентоспособность нашего решения. Традиционные системы машинного зрения с правилами (без нейросетей) часто не справляются с разнообразием видов дефектов или требуют тщательной настройки под конкретные условия. В то же время предложенная нейросетевая модель адаптируется к новым данным и способна обнаруживать даже ранее не встречавшиеся

разновидности повреждений за счёт обученной обобщённой структуры признаков. Известно также о разработках улучшенных версий YOLOv8 для схожих задач: например, Zhang и Chang предложили модификацию SFN-YOLOv8 для детекции дефектов почтовых упаковок, добившись повышения точности на неравномерных датасетах [9]. Появление подобных исследований подтверждает актуальность темы и избранного подхода. Наша реализация интегрирует компьютерное зрение с информационными системами склада (БД, веб-интерфейс, BI-аналитика), что способствует практическому внедрению технологий ИИ в складские процессы.

Тестирование системы

В данной работе представлен комплексный подход к автоматизации контроля качества картонной упаковки на складе с использованием современных методов машинного зрения. Разработана и исследована система мониторинга дефектов упаковки, включающая нейросетевой модуль детекции на базе YOLOv8 и веб-приложение для отображения результатов и аналитики. Модель успешно обучена на специализированном датасете и продемонстрировала высокую точность в выявлении различных типов дефектов (вмятины, разрывы, перекосы и пр.) в реальных условиях. Интеграция модели в инфраструктуру склада через Flask-сервис и подключение камер наблюдения позволяют осуществлять непрерывный контроль качества упаковки в режиме реального времени, минимизируя участие человека в первичном обнаружении брака.

Проведённые испытания в разных сценариях показали работоспособность и полезность системы: она своевременно обнаруживает проблемные коробки, что даёт возможность предпринять корректирующие действия (замена тары, дополнительная упаковка) ещё до того, как товар покинет склад или будет передан клиенту. Таким образом, внедрение подобной системы может привести к снижению убытков от повреждённого

товара и повысить эффективность логистических операций. Кроме того, модуль аналитики на базе собранных данных предоставляет ценную информацию для управления качеством – например, позволяет определить на каком этапе или у какого контрагента чаще возникают проблемы с упаковкой.

Перспективы развития системы включают расширение классификации на несколько категорий дефектов (например, различать типы повреждений для более точной оценки степени брака), масштабирование на большее число камер и использование распределённых вычислений для параллельной обработки потоков. Также интересным направлением является внедрение механизмов дообучения модели на месте: система могла бы периодически перепереобучаться на новых собранных данных с конкретного склада, повышая свою точность с учётом специфики объектов. Ещё одним шагом может стать интеграция с Warehouse Management System (WMS) предприятия – чтобы данные о выявленных дефектах автоматически фиксировались в общей системе учета и инициировали бизнес-процессы (создание задания на замену упаковки и т.п.). В целом, результаты работы подтверждают, что сочетание современных технологий глубокого обучения и веб-разработки позволяет создать эффективный инструмент для улучшения качества и безопасности логистических процессов.

Литература

1. Некрасов А. Г., Сеницына А. С. Транспортно-логистические системы нового поколения в цифровой экономике // СИЛА систем. 2017. № 3(4). С. 11–23.
2. BOX DEFECT DETECTION. URL: digital-services.research.konicaminolta.com/box-defect-detection-vss (дата обращения: 01.09.2025).

3. Корчагин С.А. Алгоритмы компьютерного зрения для распознавания объектов в условиях ограниченной информативности изображений // Инженерный вестник Дона. 2024. № 10. URL: ivdon.ru/ru/magazine/archive/n10y2024/9577.
4. Maity M., Banerjee S., Sinha Chaudhuri S. Faster R-CNN and YOLO based Vehicle detection: A Survey // 2021 5th International Conference on Computing Methodologies and Communication (ICCMC). Erode, India, 2021. P. 1442–1447. DOI: 10.1109/ICCMC51019.2021.9418274.
5. Лебедев О.Б. Использование нейронных сетей для решения задач компьютерного зрения // Инженерный вестник Дона. 2025. № 2. URL: ivdon.ru/ru/magazine/archive/n2y2025/9870
6. YOLOv8 (You Only Look Once) от Ultralytics. URL: docs.ultralytics.com (дата обращения: 01.09.2025).
7. Terven J., Córdova-Esparza D. M., Romero-González J. A. A Comprehensive Review of YOLO Architectures in Computer Vision: From YOLOv1 to YOLOv8 and YOLO-NAS // Machine Learning and Knowledge Extraction. 2023. Vol. 5. No. 4. P. 1680–1716. DOI: 10.3390/make5040083.
8. Yang J., Wu S., Gou L., Yu H., Lin C., Wang J., Wang P., Li M., Li X. SCD: A Stacked Carton Dataset for Detection and Segmentation // Sensors. 2022. Vol. 22. No. 10. P. 3617. DOI: 10.3390/s22103617.
9. Liang B., Wang X., Zhao W., Wang X. High-Precision Carton Detection Based on Adaptive Image Augmentation for Unmanned Cargo Handling Tasks // Sensors. 2024. Vol. 24. No. 1. P. 12. DOI: 10.3390/s24010012.
10. Yang J., Liu S., Li Z., Li X., Sun J. Real-time Object Detection for Streaming Perception // 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). New Orleans, LA, USA, 2022. P. 5375–5385. DOI: 10.1109/CVPR52688.2022.00531.

11. Полянцева К. А. Высоконагруженная платформа для агрегации и анализа неструктурированных данных о состоянии дорожного полотна // Автоматизация в промышленности. 2022. № 5. С. 32–37. DOI: 10.25728/avtprom.2022.05.09.
12. Babu K. S., Rani D., Giri S. V., Teja N. S., Rao K. R., Vamsi K., Komali M. Objection Detection Web App Using Open Cv Tensor Flow Flask // Proceedings of the KILBY 100 7th International Conference on Computing Sciences 2023 (ICCS 2023). DOI: 10.2139/ssrn.4483835.
13. Солонько М. К., Краснов К. А. Docker-контейнеры // Телекоммуникации и информационные технологии. 2020. Т. 7. № 1. С. 65–69.

References

1. Nekrasov A.G., Sinitsyna A.S. 2017. No. 3(4). pp. 11–23.
2. Konica Minolta. Box defect detection. URL: digital-services.research.konicaminolta.com/box-defect-detection-vss (accessed: 01.09.2025).
3. Korchagin S.A. Inzhenernyj vestnik Dona, 2024. No. 10. URL: ivdon.ru/ru/magazine/archive/n10y2024/9577
4. Maity M., Banerjee S., Sinha Chaudhuri S. 2021. 5th International Conference on Computing Methodologies and Communication (ICCMC). Erode, 2021. pp. 1442–1447. DOI: 10.1109/ICCMC51019.2021.9418274.
5. Lebedev O.B. Inzhenernyj vestnik Dona, 2025. No. 2. URL: ivdon.ru/ru/magazine/archive/n2y2025/9870
6. Ultralytics. YOLOv8 (You Only Look Once). URL: docs.ultralytics.com (accessed: 01.09.2025).
7. Terven J., Córdova-Esparza D.M., Romero-González J.A. 2023. Vol. 5. No. 4. pp. 1680–1716. DOI: 10.3390/make5040083.



8. Yang J., Liu S., Li Z., Li X., Sun J. 2022. IEEE CVF Conference on Computer Vision and Pattern Recognition (CVPR). New Orleans, 2022. pp. 5375–5385. DOI: 10.1109/CVPR52688.2022.00531.
9. Liang B., Wang X., Zhao W., Wang X. 2024. Vol. 24. No. 1. p. 12. DOI: 10.3390/s24010012.
10. Yang J., Wu S., Gou L., Yu H., Lin C., Wang J., Wang P., Li M., Li X. 2022. Vol. 22. No. 10. p. 3617. DOI: 10.3390/s22103617.
11. Polyantseva K.A. 2022. No. 5. pp. 32–37. DOI: 10.25728/avtprom.2022.05.09.
12. Babu K.S., Rani D., Giri S.V., Teja N.S., Rao K.R., Vamsi K., Komali M. 2023. Proceedings of the KILBY 100 7th International Conference on Computing Sciences (ICCS 2023). DOI: 10.2139/ssrn.4483835.
13. Solon'ko M.K., Krasnov K.A. 2020. Vol. 7. No. 1. pp. 65–69.

Авторы согласны на обработку и хранение персональных данных.

Дата поступления: 14.01.2026

Дата публикации: 28.02.2026