

А.Ю. Юрченко, М.Ю. Поленов

РАСПРЕДЕЛЕННАЯ СИСТЕМА РАСПОЗНАВАНИЯ ШТРИХКОДОВ С ИСПОЛЬЗОВАНИЕМ НЕЙРОСЕТЕЙ

Представлена распределённая программно-аппаратная система для автоматизированного распознавания штрихкодов на движущихся объектах в условиях производственной среды. Целью исследования является разработка надёжного и адаптивного решения, обеспечивающего устойчивое считывание штрихкодов вне зависимости от положения, скорости или высоты объектов, перемещающихся по транспортной ленте. Основной акцент сделан не на максимальной скорости обработки, а на обеспечении широкого угла обзора и надёжности распознавания при движении объектов. В отличие от традиционных сканеров, требующих точного позиционирования и дорогостоящего оборудования, предложенное решение базируется на использовании одной сетевой камеры и сервера с нейросетевыми модулями обработки. Это делает систему более универсальной и доступной для широкого круга предприятий. Ключевым элементом архитектуры выступает нейросетевой модуль восстановления изображений, основанный на модели MPRNet, способной устранять размытие и оптические искажения в кадрах видеопотока. После этапа предобработки изображения поступают в модуль детекции объектов, построенный на базе архитектуры YOLO, адаптированной под задачи распознавания штрихкодов. Распознанные данные сохраняются в базе с использованием ORM-интерфейса, что обеспечивает гибкую интеграцию в существующие информационные системы. Для предотвращения потери кадров и обеспечения высокой пропускной способности используется система асинхронной обработки с применением потоков и буферизованных очередей. Актуальность исследования обусловлена широкой распространённостью штрихкодов как основного средства промышленной маркировки и необходимостью автоматизации процессов учёта и отслеживания продукции в условиях гибкого производства. Несмотря на наличие решений в области сканирования и компьютерного зрения, большинство из них не рассчитаны на работу с нестабильным или низкокачественным видеопотоком. Предложенная система демонстрирует устойчивость к ряду искажений и может быть реализована на бюджетном оборудовании, что открывает перспективы для её применения в промышленности, логистике и складском хозяйстве.

Штрихкод; распознавание образов; компьютерное зрение; нейросеть; распределенная система.

A.Yu. Yurchenko, M.Yu. Polenov

DISTRIBUTED SYSTEM FOR BARCODE RECOGNITION USING NEURAL NETWORKS

This work presents a distributed software-hardware system for automated barcode recognition on moving objects in industrial environments. The primary objective of the research is to develop a reliable and adaptive solution capable of consistently reading barcodes regardless of the orientation, speed, or height of objects moving along a conveyor belt. The main focus is not on achieving maximum processing speed, but rather on providing a wide field of view and ensuring reliable recognition of moving objects. Unlike traditional scanners that require precise positioning and expensive hardware, the proposed approach leverages a single network camera and a server equipped with neural processing modules, providing a cost-effective and versatile alternative suitable for a wide range of industrial applications. A key component of the system architecture is a neural image restoration module based on the MPRNet model, which effectively reduces motion blur and optical distortions in video frames. After preprocessing, frames are passed to an object detection module built upon the YOLO architecture, which has been adapted specifically for barcode recognition. Detected barcode data is stored in a database using an ORM interface, enabling seamless integration with existing enterprise systems. To prevent frame loss and maintain high throughput, the system incorporates asynchronous processing mechanisms using multithreading and buffered queues. The relevance of this research stems from the widespread use of barcodes as the primary method of product marking in industrial settings and the increasing demand for automation in product tracking and inventory control. Despite the availability of various vision-based and scanning solutions, most existing systems are not designed to handle unstable or low-quality video streams. The proposed system demonstrates robustness to visual distortions and motion-related artifacts, making it suitable for deployment in real production environments. Its affordability and adaptability also open up possibilities for implementation in logistics, warehousing, and supply chain management.

Barcode; image recognition; computer vision; neural network; distributed system.

Введение. Штрихкоды остаются одним из наиболее универсальных и надёжных средств маркировки продукции в современной промышленности [1]. Они широко применяются для идентификации, отслеживания и учёта товаров на производственных линиях, складах и в логистических цепочках. Доступность технологии и высокая скорость считывания делают её привлекательной для большинства предприятий. Однако классические методы считывания, основанные на лазерных сканерах, сталкиваются с рядом ограничений: они требуют точной ориентации объектов, неэффективны при перемещении по широким конвейерным лентам, а также чувствительны к оптическим искажениям и смазанности изображения, возникающим при движении объектов.

Развитие технологий компьютерного зрения и глубокого обучения открыло новые возможности для замены традиционных сканеров программными решениями, способными обрабатывать видеопоток и выявлять маркировку в реальном времени. Существующие решения предлагают методы распознавания штрихкодов на статичных изображениях с применением библиотек для декодирования и классических фильтров. Однако при переходе к видео, особенно в условиях перемещающихся объектов, большинство существующих решений оказываются малоприменимыми: они не способны обрабатывать смазанные кадры, не адаптируются к изменяющимся ракурсам и требуют стабильного видеопотока.

Настоящая работа направлена на разработку гибкой и устойчивой к внешним факторам системы распознавания штрихкодов в режиме реального времени. Предлагаемый подход отличается интеграцией современных нейросетевых моделей и архитектур асинхронной обработки, что позволяет добиваться надёжности и качества даже при использовании недорогого оборудования.

Научная новизна заключается в объединении нейросетевого восстановления изображений, высокоэффективной детекции объектов и модулей асинхронной передачи и хранения данных в распределённой системе. В отличие от решений, ориентированных исключительно на скорость, данная разработка делает упор на устойчивость и полноту обработки. В перспективе такое решение может быть реализовано как компактное устройство, способное заменить несколько сканеров одной камерой и сервером обработки.

Постановка задачи. Процесс автоматического считывания штрихкодов на производстве требует высокой надёжности при обработке большого потока продукции с разными параметрами движения. Стационарные сканеры с фиксированным фокусом и ограниченным углом обзора оказываются неэффективными в ситуациях, где объекты перемещаются с различной ориентацией, скоростью или высотой. Помимо этого, такие решения требуют дорогостоящего оборудования и точной настройки, что ограничивает их применимость в гибких производственных системах.

Разработка программной системы на основе методов компьютерного зрения и нейросетей, способной выполнять считывание маркировки с видеопотока, имеет высокую прикладную значимость. Она позволяет автоматизировать учёт и контроль продукции, снизить стоимость оборудования, а также повысить точность идентификации за счёт интеллектуальной обработки изображений. Системы такого рода востребованы не только в промышленности, но и в логистике, складском учёте, торговле. Адаптируемость решения и независимость от дорогой аппаратной базы делают его перспективным для масштабного внедрения.

В последние годы большое внимание уделяется применению сверточных нейросетей для анализа изображений в реальном времени [2]. Архитектура YOLO (You Only Look Once) [3, 4], показала высокую эффективность в задачах детекции объектов. В то же время проблема устранения смазанности изображений рассматривается в контексте моделей восстановления, таких как MPRNet [5, 6]. Однако интеграция этих моделей в промышленную систему реального времени до сих пор требует исследований, особенно при работе на бюджетном оборудовании и с нестабильным видеопотоком.

Таким образом ставится задача – разработать программно-аппаратную систему для автоматизированного распознавания и декодирования штрихкодов на видеопотоке с движущихся объектов в условиях производственной среды. Система должна обеспечи-

вать устойчивую работу в режиме реального времени, используя современные методы компьютерного зрения и нейросетевые подходы, при этом оставаться доступной по стоимости и легко масштабируемой.

Достижение этой цели позволит заменить дорогое специализированное оборудование универсальной цифровой системой, повысить надёжность и точность считывания в сложных условиях, создать основу для адаптации решения под различные производственные сценарии.

Задачи исследования. Основными задачами являются:

- 1) анализ существующих методов детекции и декодирования штрихкодов;
- 2) организация захвата видеопотока с камеры и разбиение его на отдельные кадры;
- 3) реализация асинхронной архитектуры обработки с применением потоков и очередей;
- 4) внедрение нейросетевой модели восстановления изображения (MPRNet) для устранения их смазанности;
- 5) интеграция алгоритма YOLO для точного выявления области штрихкода;
- 6) реализация процедуры вырезки и декодирования штрихкодов с использованием специализированных библиотек;
- 7) реализация инструментов мониторинга системы и фильтрации входных данных по заданным критериям.

Объект исследования – процесс считывания штрихкодов в производственной среде.

Предмет исследования – методы видеопотоковой обработки изображений, нейросетевые модели восстановления качества и алгоритмы детекции и декодирования штрихкодов.

Если применить асинхронную архитектуру обработки, дополненную нейросетевыми алгоритмами устранения смазанности и объектной детекции, то можно получить устойчивую и адаптивную систему считывания штрихкодов, способную эффективно работать в условиях нестабильного видеопотока, при низкой стоимости оборудования и высокой вариативности производственной среды.

Обзор подходов. Обнаружение и распознавание штрихкодов в видеопотоке представляет собой задачу, лежащую на пересечении компьютерного зрения и обработки изображений. Существуют два основных подхода: классические алгоритмы обработки изображений (например, Canny, Hough, морфология) и современные методы, основанные на глубоком обучении (YOLO, SSD, Faster R-CNN, Detectron и др.) [7]. Классические алгоритмы, такие как детектор границ Canny в сочетании с преобразованием Хафа, демонстрируют приемлемые результаты при наличии чётких, контрастных изображений. Однако в условиях производственной среды, где присутствует шум, нестабильное освещение, движения объектов и фоны с текстурами, такие методы проявляют серьёзные ограничения: высокая чувствительность к шуму и артефактам съёмки, особенно на видеопотоке с низким *bitrate*, невозможность адаптации к различным типам и ориентациям штрихкодов – в частности, перспективные искажения, наклоны или частичное перекрытие делают традиционные методы малоэффективными, ограниченная масштабируемость – адаптация таких алгоритмов под разные условия требует ручной настройки параметров (порогов, размеров ядер и т.д.), что трудно реализуемо в универсальной системе.

Современные модели глубокого обучения были исследованы в рамках задачи локализации штрихкодов на кадрах. Рассматривались и тестировались следующие архитектуры:

SSD (Single Shot Detector) – модель работает быстро, но в реальных экспериментах показала худшую производительность на мелких объектах (штрихкоды часто занимают небольшую часть кадра), особенно при сжатии и низкой резкости;

Faster R-CNN – демонстрирует высокую точность, но её скорость существенно уступает другим моделям. Для задач в реальном времени, особенно с дешёвым оборудованием, использование этой модели делает систему инертной и перегруженной по ресурсам;

Detectron2 от Facebook AI – мощный фреймворк с поддержкой множества детекторов и сегментаторов. Однако для задачи распознавания штрихкодов он оказался избыточным, а его интеграция потребовала бы значительных затрат на настройку и обучение под специфичные датасеты.

В данном исследовании по итогам сравнения для локализации используется глубокая нейросетевая модель YOLOv8, она имеет оптимальный баланс между точностью и скоростью, особенно при работе на GPU, поддерживает работу с малыми объектами благодаря улучшенной архитектуре, обладает высокой устойчивостью к разным масштабам и ориентациям объектов, легко интегрируется и обучается на пользовательских датасетах, что позволило адаптировать модель под конкретные условия съёмки (например, наклон или размытие).

Для декодирования используются библиотеки на Python (pyzbar, zxing-cpp), но традиционные декодеры без предварительной нейросетевой обработки хорошо работают только на чётких изображениях и сильно теряют точность при наличии смазанности, вызванной движением или нестабильной фокусировкой. Для сложных условий съёмки требуется предобработка изображения, которую традиционные декодеры не выполняют автоматически.

В этой связи было принято решение об интеграции нейросетевой модели восстановления качества изображений MPRNet, которая используется только в случае неудачи при первичной попытке декодирования. Это позволяет существенно повысить устойчивость и полноту распознавания, особенно на смазанных или зашумлённых кадрах.

Также были протестированы и другие модели восстановления, в том числе SwinIR [8], однако они показали либо избыточное время инференса, либо более низкое качество восстановления при наличии сложных искажений. В отличие от них, MPRNet использует многоуровневую стратегию восстановления и демонстрирует стабильную работу на широком спектре реальных искажающих факторов.

Таким образом, выбранный гибридный подход – использование YOLOv8 для локализации, классических декодеров с фильтрацией, и нейросетевого восстановления качества при необходимости – обеспечивает высокую адаптивность, точность и устойчивость всей системы в реальных производственных условиях.

Выбор стратегии. Была выбрана экспериментальная стратегия исследования, направленная на разработку, реализацию и тестирование многоэтапной архитектуры для надежного распознавания штрихкодов в видеопотоке. Эксперименты проводились на заранее записанных видеороликах с различными условиями съёмки (освещение, движение, шум), а результаты сравнивались по критериям точности и скорости.

Разработка архитектуры. Разработка основывалась на следующих архитектурных принципах.

Модульность: каждый этап обработки (захват, детекция, декодирование, улучшение качества) выделен в отдельный поток.

Масштабируемость: архитектура использует очереди (Queue) и потоки (Thread), что позволяет масштабировать обработку видеопотока.

Отказоустойчивость: при неудачной декодировке изображение повторно обрабатывается с применением модели повышения качества.

Логирование и мониторинг: реализован сбор системной статистики (FPS, RAM, GPU).

Компоненты системы. На объекте размещается промышленная IP-камера, обеспечивающая постоянный видеопоток с высоким разрешением (например, 4K). Камера может передавать видеопоток в реальном времени по сети (RTSP или аналогичный протокол) или периодически сохранять видеофайлы на промежуточный накопитель.

На стороне сервера располагается основная логика системы. Сервер принимает видеопоток с камеры и обрабатывает его покадрово. Вся обработка реализована в виде набора параллельных потоков, работающих независимо, что обеспечивает высокую пропускную способность и устойчивость к нагрузке.

Распознанные и прошедшие проверку штрихкоды передаются в централизованную базу данных. Каждая запись сопровождается временной меткой (timestamp), отражающей время захвата соответствующего кадра. Это обеспечивает трассировку и хронологию операций. Вся система работает в отдельной виртуальной сети VLAN для безопасности данных [9]. Передача в БД реализована через отдельный интерфейс.

Таким образом, вся система физически состоит из трех компонентов: камеры, установленной в производственном помещении, сервера обработки данных, выполняющего весь цикл анализа и распознавания, и sql-сервера, где хранятся полученные результаты. Структура распределенной системы приведена на рис. 1.

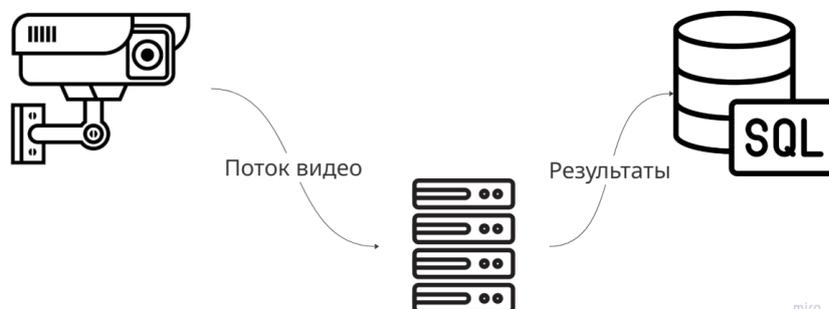


Рис. 1. Структура распределенной системы

Система на сервере состоит из следующих компонентов:

- 1) модуль захвата кадров: извлекает кадры из видео и помещает их в очередь;
- 2) YOLO-модуль: локализует предполагаемые участки со штрихкодами;
- 3) модуль обрезки: извлекает детектированные регионы;
- 4) декодер: пытается расшифровать штрихкод с помощью `ruzbar` и `zxing-cpp`;
- 5) модуль MPRNet: восстанавливает изображение при неудаче;
- 6) Uploader: отправка результата в базу данных PostgreSQL;
- 7) SystemMonitor: фиксирует показатели производительности и успешные распознавания.

Интерфейсы и протоколы. Взаимодействие между модулями реализовано через очереди Python Queue, что обеспечивает потокобезопасную передачу данных. Для хранения распознанных кодов используется PostgreSQL, подключение осуществляется через SQLAlchemy [10].

Методы и алгоритмы решения задачи. Рассмотрим основные методы и модели.

YOLOv8: нейросетевая модель используется для локализации штрихкодов. Выбрана за высокую точность и скорость. Входными данными для системы могут являться отдельные изображения, видео файлы, а также видео поток. Выходными данными являются информация о обнаруженных на изображениях объектах (их класс, расположение и другие данные, в зависимости от режима работы и типа модели). Данная информация может быть автоматически наложена на исходное изображение и сохранена/выведена в отдельный файл. YOLOv8 может использовать как CPU, так и GPU.

Ruzbar + ZXing-cpp: комбинация декодеров повышает устойчивость к различным типам штрихкодов. При объединении работы двух библиотек увеличивается количество распознанных объектов и при этом уменьшается общее время работы в сравнении с более сильной библиотекой

MPRNet: модель восстановления изображений, используется при неудаче начального декодирования. MPRNet была выбрана как одна из наиболее эффективных и универсальных архитектур глубокого обучения в задаче image restoration [12]. Она демонстрирует высокую точность восстановления при наличии как глобального размытия (например, вызванного движением камеры), так и локальных искажений. Модель построена на многоступенчатом подходе, при котором изображение обрабатывается на трёх уровнях: от грубого до детального восстановления (рис. 2). Это позволяет достигать более точных результатов по сравнению с одноуровневыми архитектурами. Дополнительным преимуществом является модульность MPRNet – её можно легко адаптировать под конкретные задачи, такие как

деблюринг, денойзинг, реставрация JPEG-артефактов, что делает её особенно удобной в составе гибкой системы обработки видеоданных. В нашей работе использована модель MPRNet для решения задачи deblurring, обученная на датасете GoPro.



Рис. 2. Сравнение изображений до и после восстановления нейросетью MPRNet

Фильтрация по шаблону: применён регулярный шаблон для фильтрации ложноположительных распознаваний. В ситуации, когда известен формат штрихкодов, которые должны декодироваться, это средство повышает точность работы и отсекает ненужные данные.

Методы предобработки изображений (OpenCV). Для повышения качества входных изображений перед попыткой декодирования штрихкодов применялись методы предобработки с использованием библиотеки OpenCV [13, 14]. Эти методы использовались как до подачи изображения в модель MPRNet, так и как самостоятельный этап в некоторых случаях. Основные использованные в системе приёмы [15]:

- ◆ преобразование в градации серого – уменьшает количество информации и повышает контрастность важных объектов;
- ◆ гауссово размытие (Gaussian Blur) – уменьшает шум, который может мешать декодированию;
- ◆ адаптивная бинаризация – помогает выделить штрихкод в условиях переменного освещения.

Морфологические операции (расширение/эрозия) использовались в отдельных случаях для устранения мелких шумов или соединения разорванных линий.

Фильтрация шумов (Non-local Means, Bilateral Filter) – проверялась экспериментально, но оказалась менее эффективной, чем вышеуказанные методы [16].

В результате предобработки изображения становились более пригодными для декодирования, особенно в случаях сильного шума или низкого контраста. Конкретный алгоритм восстановления изображений в системе выбирается в зависимости от характера повреждений или искажений в изображении, доступного времени и доступных ресурсов для вычислений [17].

Алгоритм обработки и программная реализация. Алгоритм состоит из следующих основных этапов:

1. Захват кадра → YOLO → обрезка → попытка декодирования.
2. Если неудача, то:
 - ◆ улучшение изображения через MPRNet;
 - ◆ повторная попытка декодирования.
3. Если успех – сохранить штрихкод, метку времени, изображение.

Все эти действия происходят асинхронно и последовательно [18, 19]. На рис. 3 представлена схема обработки кадра и передачи между очередями и потоками.

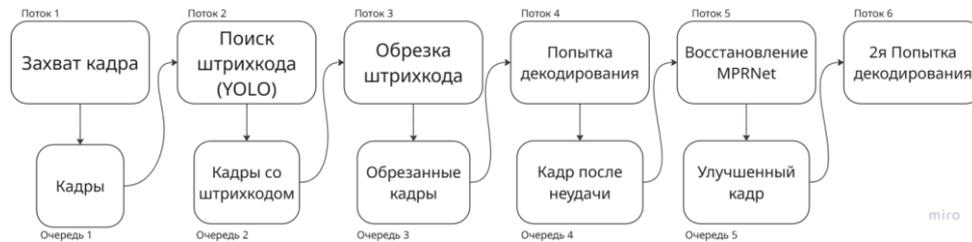


Рис. 3. Взаимодействие потоков и очередей

Программное приложение содержит 6 независимых потоков и 5 очередей. Сервер получает видео, в первом потоке разбивает на кадры и каждый кадр помещает в очередь, это необходимо для того, чтобы при простое на более долгих операциях кадры не терялись, а просто хранились в этой очереди, обрабатываясь в возможном темпе. Во втором потоке работает нейросеть детектирования YOLO, которая обучена на нахождение штрихкодов. Она возвращает координаты объекта. В очередь передается кадр, координаты объекта, временная метка (здесь она будет самой точной). В третьем потоке происходит обрезка по координатам с паддингом 10. Локализованная область передается в третью очередь вместе с той же временной меткой. В четвертом потоке происходит попытка декодирования с помощью двух библиотек ZXing (в обертке ZXing-cpp) и Zbar (в обертке PyZbar), если первая, более быстрая, не справляется, происходит попытка распознавания более долгой, но точной PyZbar. Если распознавание прошло удачно – сохраняются результаты, если нет, то кадр передается в четвертую очередь. В пятом потоке кадр отправляется для обработки нейросети MPRNet, обученной восстанавливать смазанные снимки, так как кадры камера получает в динамике (высока вероятность, что кадр просто смазан). После восстановления кадр передается в пятую очередь и в шестом потоке происходит аналогичная попытка распознавания. При успешном распознавании кадр сохраняется с разметкой, а данные отправляются на сервер.

Анализ данных. Данные логировались с помощью SystemMonitor [20]: количество кадров, число распознанных кодов, загрузка системы. Также сохранялись изображения с распознанными кодами и временной меткой для последующего анализа.

В результате работы каждый успешный кадр сохраняется, все уникальные штрихкоды записываются. Пример сохраненного кадра представлен на рис. 4.

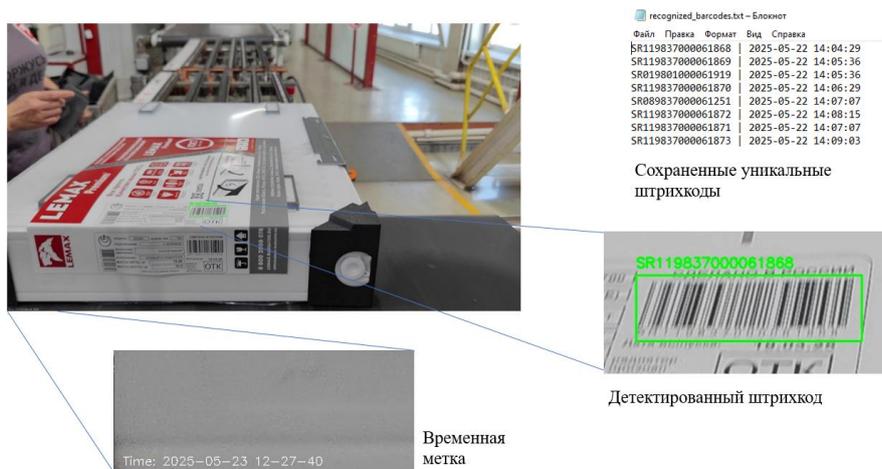


Рис. 4. Экспериментальные результаты

Трудности и ограничения, выявленные в процессе реализации. В ходе практической реализации системы автоматизированного распознавания штрихкодов в видеопотоке были выявлен ряд технических сложностей, связанных как с производительностью оборудования, так и с особенностями работы алгоритмов.

Высокая нагрузка на графические ресурсы при использовании MPRNet. Модель восстановления изображений MPRNet, несмотря на свою эффективность, обладает высокой вычислительной сложностью. При запуске на слабом или интегрированном GPU время обработки одного кадра значительно возрастает (до 1-2 секунд), что неприемлемо для систем, ориентированных на реальное время. Особенно ощутима эта задержка при параллельной обработке нескольких потоков или при масштабировании на слабых серверах. Для повышения скорости на этапе восстановления тестировалась облегчённая модификация MPRNet с урезанными слоями, что позволило снизить время инференса примерно на 40% без критичной потери качества. Также применялось ограничение количества кадров, отправляемых на восстановление, с помощью предфильтра на входе.

Проблемы при работе в условиях недостаточного освещения. В ночное время или при съёмке в неосвещённых участках производственной линии качество видеопотока резко падает: появляются шумы, контрастность снижается, возникают артефакты сжатия. В таких условиях детекторы работают нестабильно, а декодирование часто не удаётся даже после применения MPRNet. Рекомендовано использование внешней инфракрасной или светодиодной подсветки, а также предварительная фильтрация кадров с низкой освещённостью на стадии захвата. Также тестировались методы адаптивного выравнивания контраста и шумоподавления средствами OpenCV до подачи изображения в нейросеть.

Заключение. В ходе работы была разработана система для автоматизированного обнаружения и распознавания штрихкодов из сохранённых видео или из видеопотока в реальном времени. В качестве основы для детектирования использована модель YOLOv8, продемонстрировавшая высокую скорость и точность локализации штрихкодов на отдельных кадрах. Распознавание содержимого осуществлялось с использованием выбранных библиотек, в том числе с предварительной обработкой изображений, включающей масштабирование, повышение резкости и восстановление качества с помощью нейросетевой модели MPRNet.

Предложена многоступенчатая архитектура обработки видеопотока с применением MPRNet для повышения точности распознавания. Разработан механизм повторного декодирования после восстановления изображения. Сочетание YOLOv8, MPRNet и двойного декодера (ruzbar и zxing-cpp) впервые применено совместно для данной задачи.

Практическая значимость предложенного подхода заключается в том, что при последующей адаптации и доработке метод может применяться в логистике (автоматическое сканирование упаковок), производстве (слежение за продукцией), видеонаблюдении (инвентаризация), ритейле (анализ видеозаписей кассовой зоны). Полученная система устойчива к помехам, размытию и низкому освещению.

Таким образом, предложенные подход и алгоритм объединяют высокую точность, эффективность с точки зрения вычислительных ресурсов и хорошую масштабируемость. Рассмотренное решение обладает высокой практической значимостью и может быть внедрено в промышленное производство, торговые предприятия, а также использовано в системах мониторинга и контроля.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Арсланова Л.С. Использование штрихового кодирования // Экономика и социум. – 2016. – № 2 (21). – С. 42-44.
2. Tan M., Pang R., Le Q.V. EfficientDet: Scalable and Efficient Object Detection // arXiv. – 2020. – arXiv:1911.09070v7 [cs.CV].
3. Бурень М.Н. Обзор возможностей YOLO v8 в рамках решения задачи распознавания изображений // Компьютерные системы и сети: Сб. статей 60-й научной конференции аспирантов, магистрантов и студентов, БГУИР. – Минск: БГУИР, 2024. – С. 28-33.
4. Hansen D.K., Nasrollahi K., Rasmussen C.B., Moeslund T.B. Real-Time Barcode Detection and Classification Using Deep Learning // Proceedings of the 9th International Joint Conference on Computational Intelligence (IJCCI 2017). Vol. 1. – SciTePress, 2017. – P. 321-327. – DOI: 10.5220/0006508203210327.

5. Abuolaim A. et al. NTIRE 2021 Challenge for Defocus Deblurring Using Dual-pixel Images: Methods and Results // CVPRW. – 2021. – P. 578-587. – DOI: 10.1109/CVPRW53098.2021.00070.
6. Nah S. et al. NTIRE 2021 Challenge on Image Deblurring // CVPRW. – 2021. – P. 149-165. – DOI: 10.1109/CVPRW53098.2021.00025.
7. Ren S., He K., Girshick R., Sun J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks // arXiv. – 2016. – arXiv:1506.01497v3 [cs.CV].
8. Liang J., Cao J., Sun G., Zhang K., Van Gool L., Timofte R. SwinIR: Image Restoration Using Swin Transformer // arXiv. – 2021. – arXiv:2108.10257v1 [eess.IV].
9. Стрельцов Д.С., Бутенко Е.А. Vlan id enumeration. Методы защиты // Мирская наука. – 2024. – № 12 (93). – С. 120-133.
10. Неустров А.В. Сохранить JSON данные в базу данных sqlalchemy // Проблемы науки. – 2016. – № 12 (13).
11. Zhao T., Ma X., Li X., Zhang C. MPR-Net: Multi-Scale Pattern Reproduction Guided Universality Time Series Interpretable Forecasting // Proceedings of the AAAI Conference on Artificial Intelligence. – 2023. – arXiv:2307.06736.
12. Chen X., Li Z., Pu Y., Liu Y., Zhou J., Qiao Y., Dong Ch. A Comparative Study of Image Restoration Networks for General Backbone Network Design // Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). – 2024. – P. 74-91.
13. Буэно Г.Г., Суарес О.Д., Эспиноса А. и др. Обработка изображений с помощью OpenCV: пер. с англ. Слинкин А.А. – М.: ДМК Пресс, 2016. – 210 с.
14. Boracchi G., Foi A. Modeling the Performance of Image Restoration from Motion Blur // IEEE Transactions on Image Processing. – 2012. – Vol. 21, Iss. 8. – P. 3502-3517. – DOI: 10.1109/TIP.2012.2192126.
15. Вербицкая В.И., Коршикова Д.В., Купчина Е.В. Реализация алгоритма восстановления «Фильтр Винера» на Python // Информационные технологии и управление: Матер. 60-ой научной конференции аспирантов, магистрантов и студентов, 2024. – Минск: БГУИР, 2024. – С. 69.
16. Иващенко В.Ю. Восстановление изображений и различные техники реставрации используемые в сфере обработки визуальных цифровых ресурсов // Теория и практика современной науки. – 2019. – № 12 (54). – С. 143-147.
17. Коршикова Д.В., Кукин Д.П., Купчина Е.В. Алгоритмы восстановления изображений // Информационные технологии и системы 2023 (ИТС 2023): Матер. Международной научной конференции. – Минск: БГУИР, 2023. – С. 149-150.
18. Абакаров Г.М. Анализ и оценка многопоточности для эффективности приложений // Вестник науки. – 2023. – № 12 (69). – С. 1094-1098.
19. Кадомский А.А., Захаров В.А. Эффективность многопоточных приложений // Научный журнал. – 2016. – № 7 (8).
20. Злобина Н.В., Волжанкин Н.В., Пособилов Н.Е. Обеспечение централизованного мониторинга для систем сложной архитектуры с большим объёмом данных // Тр. НГТУ им. Р.Е. Алексеева. – 2017. – № 4 (119). – С. 18-23.

REFERENCES

1. Arslanova L.S. Ispol'zovanie shtrikhovogo kodirovaniya [Use of barcode coding], *Ekonomika i sotsium* [Economics and Society], 2016, No. 2 (21), pp. 42-44.
2. Tan M., Pang R., Le Q.V. EfficientDet: Scalable and Efficient Object Detection, *arXiv*, 2020. arXiv:1911.09070v7 [cs.CV].
3. Buren' M.N. Obzor vozmozhnostey YOLO v8 v ramkakh resheniya zadachi raspoznavaniya izobrazheniy [Review of YOLO v8 capabilities for image recognition tasks], *Komp'yuternye sistemy i seti: Sb. statey 60-y nauchnoy konferentsii aspirantov, magistrantov i studentov, BGUIR* [60th Anniversary Scientific Conference of Postgraduates, Master's Students and Students of BSUIR]. Minsk: BGUIR, 2024, pp. 28-33.
4. Hansen D.K., Nasrollahi K., Rasmussen C.B., Moeslund T.B. Real-Time Barcode Detection and Classification Using Deep Learning, *Proceedings of the 9th International Joint Conference on Computational Intelligence (IJCCI 2017)*. Vol. 1. SciTePress, 2017, pp. 321-327. DOI: 10.5220/0006508203210327.
5. Abuolaim A. et al. NTIRE 2021 Challenge for Defocus Deblurring Using Dual-pixel Images: Methods and Results, *CVPRW*, 2021, pp. 578-587. DOI: 10.1109/CVPRW53098.2021.00070.
6. Nah S. et al. NTIRE 2021 Challenge on Image Deblurring, *CVPRW*, 2021, pp. 149-165. DOI: 10.1109/CVPRW53098.2021.00025.
7. Ren S., He K., Girshick R., Sun J. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks, *arXiv*, 2016. arXiv:1506.01497v3 [cs.CV].

8. Liang J., Cao J., Sun G., Zhang K., Van Gool L., Timofte R. SwinIR: Image Restoration Using Swin Transformer, *arXiv*, 2021. arXiv:2108.10257v1 [eess.IV].
9. Strel'tsov D.S., Butenko E.A. Vlan id enumeration. Metody zashchity [VLAN ID enumeration: security methods], *Mirovaya nauka* [World Science], 2024, No. 12 (93), pp. 120-133.
10. Neustroev A.V. Sokhranit' JSON dannye v bazu dannykh sqlalchemy [Saving JSON data to a database with SQLAlchemy], *Problemy nauki* [Problems of Science], 2016, No. 12 (13).
11. Zhao T., Ma X., Li X., Zhang C. MPR-Net: Multi-Scale Pattern Reproduction Guided Universality Time Series Interpretable Forecasting, *Proceedings of the AAAI Conference on Artificial Intelligence*, 2023. arXiv:2307.06736.
12. Chen X., Li Z., Pu Y., Liu Y., Zhou J., Qiao Y., Dong Ch. A Comparative Study of Image Restoration Networks for General Backbone Network Design, *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024, pp. 74-91.
13. Bueno G.G., Suares O.D., Espinosa A. i dr. Obrabotka izobrazheniy s pomoshch'yu OpenCV [Image Processing with OpenCV]: transl. from engl. Slinkin A.A. Moscow: DMK Press, 2016, 210 p.
14. Boracchi G., Foi A. Modeling the Performance of Image Restoration from Motion Blur, *IEEE Transactions on Image Processing*, 2012, Vol. 21, Iss. 8. – P.3502-3517. DOI: 10.1109/TIP.2012.2192126.
15. Verbitskaya V.I., Korshikova D.V., Kupchina E.V. Realizatsiya algoritma vosstanovleniya «Fil'tr Vinera» na Python [Implementation of the Wiener Filter image restoration algorithm in Python], *Informatsionnye tekhnologii i upravlenie: Mater. 60-oy nauchnoy konferentsii aspirantov, magistrantov i studentov, 2024* [Information Technologies and Control: Proceedings of the 60th Scientific Conference of Postgraduates, Master's Students and Students, 2024]. Minsk: BGUIR, 2024, pp. 69.
16. Ivashchenko V.Yu. Vosstanovlenie izobrazheniy i razlichnye tekhniki restavratsii ispol'zuemye v sfere obrabotki vizual'nykh tsifrovyykh resursov [Image restoration and restoration techniques used in visual digital resource processing], *Teoriya i praktika sovremennoy nauki* [Theory and Practice of Modern Science], 2019, No. 12 (54), pp. 143-147.
17. Korshikova D.V., Kukin D.P., Kupchina E.V. Algoritmy vosstanovleniya izobrazheniy [Image restoration algorithms]. *Informatsionnye tekhnologii i sistemy 2023 (ITS 2023): Mater. Mezhdunarodnoy nauchnoy konferentsii* [Information Technologies and Systems 2023 (ITS 2023): Proceedings of the International Scientific Conference]. Minsk: BGUIR, 2023, pp. 149-150.
18. Abakarov G.M. Analiz i otsenka mnogopotochnosti dlya effektivnosti prilozheniy [Analysis and Evaluation of Multithreading for Application Efficiency], *Vestnik nauki* [Science Bulletin], 2023, No. 12 (69), pp. 1094-1098.
19. Kadomskiy A.A., Zakharov V.A. Effektivnost' mnogopotochnykh prilozheniy [Efficiency of Multithreaded Applications], *Nauchnyy zhurnal* [Scientific Journal], 2016, No. 7 (8).
20. Zlobina N.V., Volzhankin N.V., Posobilov N.E. Obespechenie tsentralizovannogo monitoringa dlya sistem slozhnoy arkhitektury s bol'shim ob'emom dannykh [Ensuring centralized monitoring for complex architecture systems with large data volumes], *Tr. NGTU im. R.E. Alekseeva* [Proceedings of NSTU nam. R.E. Alekseev], 2017, No. 4 (119), pp. 18-23.

Юрченко Анастасия Юрьевна – Южный федеральный университет, e-mail: iurchen@sfedu.ru; г. Таганрог, Россия; тел.: 89185609741, кафедра вычислительной техники; студент.

Поленов Максим Юрьевич – Южный федеральный университет; e-mail: mypolenov@sfedu.ru; г. Таганрог, Россия; тел.: 88634371656; кафедра вычислительной техники; к.т.н.; доцент.

Yurchenko Anastasiya Yuryevna – Southern Federal University; e-mail: iurchen@sfedu.ru; Taganrog, Russia; phone: +79185609741; the Department of Computer Engineering; student.

Polenov Maxim Yuryevich – Southern Federal University; e-mail: mypolenov@sfedu.ru; Taganrog, Russia; phone: +78634371656; the Department of Computer Engineering; cand. of eng. sc.; associate professor.