

Глубокое обучение для эффективной визуализации и классификации маркировок перерабатываемых материалов

В.О. Кузеванов¹, Д.В. Тихомирова²

Национальный исследовательский ядерный университет "МИФИ", Москва, Россия

¹ ORCID: 0009-0005-7093-7709, slava.kuzevanov@mail.ru

² ORCID: 0000-0002-0812-2331, dvsulim@mail.ru

Аннотация

В данной работе представлен пример системы, позволяющей улучшить процесс разделения вторсырья посредством использования методов глубокого обучения для автоматического обнаружения, визуализации и классификации кодов переработки на упаковках продуктов. В работе авторами рассматриваются различные подходы к оптическому распознаванию символов и детекции объектов в видеопотоке или на изображении. Авторами разработан и предложен вариант комбинации нейронных сетей для детекции и классификации кодов переработки. Предлагаемая система нейронных сетей призвана способствовать эффективным процессам переработки за счет автоматизации идентификации символов переработки, тем самым облегчая сортировку и переработку вторсырья.

Ключевые слова: Глубокое обучение, нейронные сети, коды переработки, автоматизированная сортировка вторсырья, компьютерное зрение, распознавание изображений.

1. Введение

По мере роста глобального внимания к устойчивым практикам переработки растет потребность в передовых технологиях для поддержания процессов управления переработки вторсырья. Коды переработки, представленные такими символами, как треугольник переработки с числовым идентификатором (например, PETE 1, HDPE 2), играют решающую роль в определении возможности вторичной переработки упаковочных материалов. Такие условные символы применяются для обозначения материала, из которого изготовлен предмет. В каждой стране коды переработки могут отличаться друг от друга, однако страны Европейского Союза утвердили решение комиссии 97/129/ЕС [1] о создании системы идентификации упаковочных материалов. В дальнейшем, в странах Евразийского Экономического Союза был утвержден русскоязычный список материалов [2] с соответствующими кодами, основанный на документе, принятом в ЕС. Поэтому условные знаки, обозначающие коды перерабатываемых материалов на территории этих стран одинаковые. В этой статье представлено решение проблемы, в основе которого лежат методы глубокого обучения для автоматизации распознавания и классификации кодов переработки на упаковках продуктов.

В последние годы стремление решить проблемы с эффективностью вторичной переработки стимулировало обширные исследования и технологические разработки в области обработки вторсырья. В этом контексте подходы глубокого обучения стали мощным инструментом для автоматизации и оптимизации различных аспектов управления и переработки вторсырья. Однако критическим аспектом, который еще предстоит всесторонне рассмотреть, является точное распознавание и классификация кодов пе-

перерабатываемых материалов, указанных на упаковках продуктов. Чтобы решить эту сложную задачу, наш подход требует разработки системы, состоящей из двух различных типов нейронных сетей.

Сложность с классификацией перерабатываемого материала возникает из-за визуального сходства кодов переработки, наносимых на упаковки товаров. Для более точной классификации возникает необходимость в многоэтапном подходе. Предложенная авторами система состоит из поэтапного использования двух нейронных сетей. Первая нейронная сеть предназначена для обнаружения и определения ограничивающих рамок кодов перерабатываемых материалов в видеопотоке и на изображении. Понимая, что визуальное сходство кодов переработки представляет собой серьезную проблему для точной классификации, следующим этапом мы предлагаем использовать вторую нейронную сеть, специализирующуюся на оптическом распознавании символов (OCR). Эта сеть работает с обнаруженными кадрами, расшифровывая цифры и подписи кодов переработки.

Стратегически разделив задачу между двумя специализированными нейронными сетями, авторы статьи стремятся повысить точность и надежность всей системы.

1.1. Обзор работ в области применения машинного обучения для детекции объектов

В контексте управления процессами переработки вторсырья важность обнаружения объектов и оптического распознавания символов становится первостепенной. Хотя многие существующие исследования заложили основу для применения машинного обучения при сортировке мусора, недавние достижения в области обнаружения объектов и оптического распознавания символов открыли новые возможности для точности и эффективности этого метода.

Системы обнаружения объектов и системы оптического распознавания символов (OCR) были разработаны (и продолжают совершенствоваться) для решения широкого спектра задач в различных областях, демонстрируя свою универсальность и влияние на различные отрасли. В области обнаружения объектов эти системы превосходно справляются с такими задачами, как автономная навигация транспортных средств [3] [4], где они идентифицируют пешеходов, транспортные средства и дорожные знаки, способствуя повышению безопасности дорожного движения. В розничной торговле обнаружение объектов используется для управления запасами и для анализа клиентов, автоматизации мониторинга запасов и оптимизации покупательского опыта [5]. В то же время системы оптического распознавания символов играют ключевую роль в преобразовании рукописного или печатного текста в цифровые форматы [6], облегчая оцифровку документов и улучшая их доступность для людей с нарушениями зрения. Более того, OCR широко используется в сфере финансов для автоматизации извлечения данных из счетов и квитанций [7]. Адаптивность технологий обнаружения объектов и оптического распознавания символов распространяется и на сферу здравоохранения, помогая медицинским работникам анализировать различные изображения и обрабатывать документы [8].

Помимо вышеупомянутых приложений, системы обнаружения объектов и оптического распознавания символов играют важную роль в обеспечении безопасности и наблюдения. В сфере безопасности обнаружение объектов используется для мониторинга общественных мест в реальном времени [9], распознавания и отслеживания потенциально подозрительных действий или объектов. С другой стороны, OCR имеет решающее значение для идентификации лицензионных номеров транспортных средств, расширяя возможности систем автоматического распознавания номерных знаков (ALPR) [10], которые находят применение в работе правоохранительных органов и в управлении парковками.

Такая обширная сфера применения подчеркивает потенциал этих систем, демонстрируя их способность решать широкий спектр сложных задач с достаточной точностью и эффективностью.

Значительный объем исследований посвящен изучению интеграции методов машинного обучения в сферу переработки вторсырья, что отражает важность этой области для промышленности и общества. В ходе исследований были изучены способы применения алгоритмов машинного обучения для сортировки вторсырья, анализа его состава и идентификации материалов в потоках переработки.

В большинстве работ на эту тему прослеживается тенденция использования машинного обучения для классификации вторсырья по широким категориям, таким как пластик, бумага и стекло, в первую очередь, на основе визуальных особенностей, извлеченных из изображений на упаковочных материалах. Например, авторы статьи «Comparing deep learning and support vector machines for autonomous waste sorting» [11] сравнивали эффективность применения сверточных нейронных сетей (CNN) и метода, основанного на опорных векторных машинах (SVM), для автономной сортировки мусора по трем основным категориям: пластик, бумага и стекло.

Хотя эти усилия внесли значительный вклад в автоматизацию процессов сортировки вторсырья, в литературе существует заметный пробел в отношении детального анализа кодов переработки на упаковках продуктов. Большинство существующих работ концентрируются на более широкой классификации материалов, упуская из виду конкретную идентификацию и интерпретацию символов переработки — важнейшего аспекта в экосистеме переработки.

1.2. Компьютерное зрение и оптическое распознавание символов

С развитием технологий и появлением первых электронно-вычислительных машин возникла необходимость в оцифровке печатных документов и текстов в «понимаемую» компьютерами информацию. Для этого велись работы над созданием систем оптического распознавания символов.

Оптическое распознавание символов (OCR) — это часть программного обеспечения, которое преобразует печатный текст и изображения в оцифрованную форму. В таком виде с этими данными можно работать при помощи машины. В отличие от человеческого мозга, который способен довольно легко распознавать текст/символы, нанесенные на изображения, машины недостаточно «интеллектуальны», чтобы воспринимать всю доступную на изображениях информацию. В процессе оптического распознавания символов возникает ряд проблем, связанных с тем, что текст на изображении может быть написан на разных языках, а сам код переработки может быть по-разному оформлен (выбор гарнитуры и начертания текста зависит от фирменного стиля производителя). Следовательно, методы из разных дисциплин информатики (например, обработка изображений, классификация образов и обработка естественного языка) используются для решения различных аспектов данной проблемы.

Именно из-за своей комплексности задача оптического распознавания символов относится к области компьютерного зрения в машинном обучении. Машинное обучение позволяет компьютерам учиться, делать прогнозы и принимать решения, основываясь на наборе данных, а не выдавать жестко запрограммированный и однозначный результат.

Компьютерное зрение [12] — это область машинного обучения, которая получает значимую информацию из графических файлов, т.е. с изображениями и видеопотоками, занимается ее обработкой и предоставлении результатов. Компьютерное зрение используется в таких задачах, как:

1. Идентификация
2. Распознавание объектов

3. Сегментация изображений
4. Оценка положения
5. Распознавание текста
6. Генерация объектов
7. Анализ видео

Рассмотрим основные этапы, которые необходимы для разработки системы оптического распознавания символов [13].

Таблица 1– Основные этапы разработки OCR системы

Этап	Описание	Подходы
Сбор информации	Процесс получения информации об изображении	Оцифровка, бинаризация, сжатие
Предобработка	Повышения качества изображения	Удаление шума, удаление перекоса, утончение, морфологические операции
Сегментация	Разделение изображения на его составные части	Техники сегментации
Классификация	Категоризация символа или объекта к его истинному классу	Методы машинного обучения (байесовский классификатор, деревья решений, нейронные сети и т. д. [6])
Постобработка	Повышение качества результатов OCR системы	Контекстуальные подходы, использование нескольких классификаторов

В книге «Optical character recognition: An Illustrated Guide to the Frontier» [14] авторы описывают основные проблемы, с которыми сталкиваются системы оптического распознавания символов. Даже если текст печатается легко считываемой гарнитурой и в стандартном формате, многие ошибки вызваны дефектами изображения (рис. 1). Такие дефекты могут возникнуть на разных этапах, например, в процессе печати изображения или во время его оцифровки. Как правило, дефекты состоят из тяжелых, смазанных, соприкасающихся знаков (рис. 1.а), светлых и ломаных знаков (рис. 1.б) или их сочетаний. Блуждающие метки (рис. 1.в) и изогнутые базовые линии (рис. 1.г) также приводят к искажению результатов системы. Кроме того, схожесть основных кодов переработки и дефекты от средств съемки также усложняют задачу классификации.

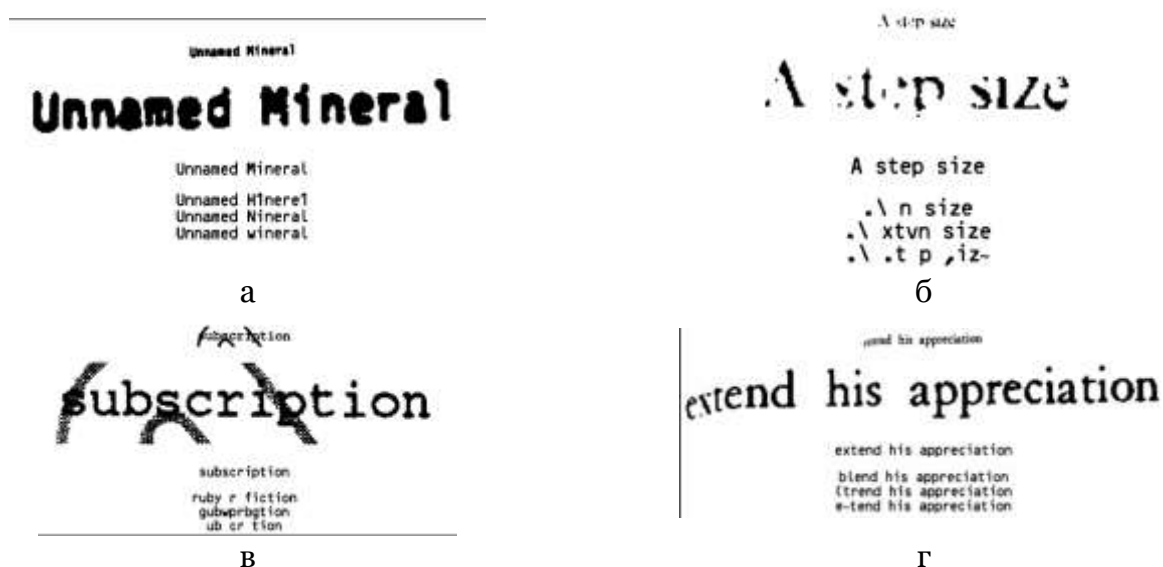


Рисунок 1. Примеры дефектов изображений: а) смазанные знаки б) светлые знаки в) блуждающие метки г) изогнутые базовые линии

Одной из задач данной работы является построение такой системы, которая бы с достаточной точностью распознавала символы переработки на упаковках даже при условии наличия дефектов, отмеченных авторами.

1.3. Основные способы классификации объекта на изображении

На данный момент существует три основных способа распознавания и классификации объектов на изображении и в видеопотоке:

1. Сравнение с шаблоном (template matching)
2. Сегментация изображений и анализ BLOB-объектов
3. Нейронные сети

Сопоставление с шаблоном является одним из самых простых для понимания способов распознавания объекта на изображении (видеопоток является последовательностью изображений). Этот метод основывается на нахождении места на изображении, наиболее похожего на шаблон. Сходство изображения с шаблоном задается определенной метрикой: шаблон «накладывается» на изображение, и считается расхождение между изображением и шаблоном. Положение шаблона, при котором это расхождение будет минимальным, и будет означать место искомого объекта.

В качестве метрики можно использовать разные варианты, например — сумму квадратов разниц между шаблоном и картинкой (sum of squared differences, SSD) (формула 1), или использовать кросс-корреляцию (cross-correlation, CCORR) (формула 2). Пусть f и g — изображение и шаблон размерами (k, l) и (m, n) соответственно (каналы цвета пока будем игнорировать); i, j — позиция на изображении, к которой применяется шаблон.

$$SSD_{i,j} = \sum_{a=0..m, b=0..n} (f_{i+a, j+b} - g_{a,b})^2 \quad (1)$$

$$CCORR_{i,j} = \sum_{a=0..m, b=0..n} (f_{i+a, j+b} * g_{a,b})^2 \quad (2)$$

Можно заметить, что эти метрики требуют попиксельного соответствия шаблона в искомом изображении. Любое отклонение гаммы, света или размера приведут к тому, что методы не будут работать. Это именно то, что происходит с кодами переработки, так как изображения кодов переработок на упаковках не стандартизированы (как мы уже упоминали, они могут быть разного размера, цвета и формы в зависимости от фирменного стиля производителя). Например, один из кодов переработки пластика PET 1 на разных упаковках может выглядеть по-разному (смотреть рисунки 2.а и 2.б).



а



б

Рисунок 2.а – Образец нанесения кода PET 1 на упаковке; б – Образец нанесения кода PET 1 на упаковке

Сегментация изображений обычно использует такие свойства объекта, как размер, цвет или форма. Поэтому представляется возможным правильно классифицировать данные, зная основные характеристики объекта, при условии, что на изображении они похожи.

BLOB-объект (Binary Large Object) — это последовательность двоичных данных, которая может представлять любую форму неструктурированной информации, такую как изображения, аудио, видео или другие файлы. В контексте обработки изображений, BLOB-объекты представляют собой связные области на бинаризованном изображении, которые соответствуют различным объектам или интересующим частям изображения. BLOB-объекты помогают выделить и идентифицировать отдельные связные области на изображении, которые представляют собой потенциальные объекты интереса. Каждый BLOB-объект характеризуется набором признаков, таких как площадь, форма, периметр, моменты инерции и другие геометрические свойства. Эти признаки можно использовать для описания и различения различных объектов.

Однако изображение кодов переработки может разительно отличаться от установленного шаблона, что может говорить о том, что у них не будет однозначно схожих признаков.

Поэтому для классификации и распознавания кодов переработки является целесообразным использование третьего подхода – нейронных сетей, тем более что использование нейронных сетей в области сортировки мусора по фотографиям уже было описано в книге «Waste Segregation System Using Artificial Neural Networks» [15]. Кроме того, проблему сортировки материалов с помощью искусственного интеллекта также исследовали R.S.Sandhya Devi, Vijaykumar VR, M.Muthumeena в своей научной работе [16]. Это еще раз подтверждает эффективность таких систем в задачах компьютерного зрения.

Существует два подхода к распознаванию объектов с использованием глубокого обучения. Первый из них – это обучение модели с нуля. Чтобы обучить глубокую сеть с нуля, необходимо собрать очень большой размеченный набор данных и разработать архитектуру сети, которая будет изучать характеристики объекта и строить модель. Результаты могут быть впечатляющими, но этот подход требует большого количества обучающих данных. Кроме того, необходимо настраивать уровни и веса в CNN. Вторым методом – это использование предварительно обученной модели глубокого обучения. Большинство приложений глубокого обучения используют подход трансферного обучения – процесс, который включает в себя точную настройку предварительно обученной модели. Для начала берется существующая сеть, такая как AlexNet [17] или GoogLeNet [18], и вводятся новые данные, содержащие ранее неизвестные классы. Этот метод требует меньше времени и может обеспечить более быстрый результат, поскольку модель уже обучена на тысячах или миллионах изображений. Глубокое обучение предлагает высокий уровень точности, но требует большого количества данных для точных прогнозов.

2. Детекция объектов и OCR

2.1. Нейронные сети – детекторы

Область применения компьютерного зрения очень обширна и позволяет решать целый ряд задач, связанных с обработкой изображений и видео. Было разработано много методов и архитектур нейронных сетей, позволяющих распознавать объекты в реальном времени. Авторы статьи Deep-learning based object detection in low altitude UAV dataset: A Survey [19] предлагают разделять архитектуры нейронных сетей на три вида (смотреть рис. 3): Двухстадийные (Two Stage Detectors), Одностадийные (One Stage Detectors) и Точечные / Продвинутые детекторы (Points-based Detectors / Advanced Detectors).

Предсказания фреймов в двухстадийных детекторах реализуются в два этапа. Во время первого этапа идет генерация «регионов интересов» (boxes proposals) и приведение их к одному размеру. На втором этапе осуществляется предсказание координат фреймов и принадлежности их к классу.

Одностадийные нейронные сети отличаются монолитной архитектурой нейронной сети без отдельного алгоритма генерации (Selective Search или Region-proposal network). Самыми популярными и часто используемыми одностадийными архитектурами нейронных сетей для детектирования на данный момент являются SSD (Single-Shot-Detector) [20] и YOLO (You-Only-Look-Once) [21].



Рисунок 3 – таксономия методов обнаружения объектов на основе нейронных сетей

Sahel, Alsahafi, Alghamadi и Alsubait [22] работали над моделью обнаружения логотипов и сравнивали модели RCNN, FRCNN, Retina Detection, чтобы определить, какая из них является наиболее точной. Набор данных, который используется в статье, взят из FlickrLogos32, который имеет 32 логотипа, а также имеет четко сформулированные аннотации, сделанные на веб-сайте makeense.ai. Авторы заявили, что достигли точности 99,8% для модели RCNN и 95,8% для Retina Net. Далее в исследовательской работе делается вывод о том, что модели CNN являются наиболее предпочтительными, когда речь идет о точности.

Brahim Jabir, Nouredine Falih, Khalid Rahmani [23] работали над сравнением различных моделей обнаружения объектов, а именно Detectron2, EfficientDet, YOLO и Faster R-CNN. Цель их исследования — создать модель, которая выявляет сорняки в посевах в режиме реального времени с помощью компьютерного зрения. Их набор данных был создан путем ручного захвата изображений сорняков с полей с помощью профессиональной камеры при различных условиях освещения. В заключительном разделе они заявили, что YOLO v5 является самой быстрой моделью для обнаружения сорняков в посевах в режиме реального времени по сравнению с Detectron2, EfficientDet и Faster R-CNN. Они также заявили, что Faster R-CNN лучше YOLOv5 с точки зрения точности.

Varad, Rohit, Rushal and Sohan [24] сравнили две модели YOLO и SSD для обнаружения объектов в режиме реального времени. Авторы собрали свой набор данных из coco.names, который имеет 80 классов объектов, на которых модели будут обучены выполнять детекцию. Исследователи обучили модели обнаруживать объекты в режиме реального времени с помощью беспилотника-наблюдателя. Авторы статьи завершили

свое исследование, заявив, что модель SSD быстрее, но менее точна, тогда как YOLO обладает как скоростью, так и точностью и считается эффективной моделью.

Основываясь на сравнительных анализах нейронных сетей для детекции объектов, приведенных в исследованиях выше, авторы статьи делают вывод, что на данный момент архитектура YOLO является наиболее подходящей для решения задачи детекции кодов перерабатываемых материалов на упаковках в реальном времени. В своей системе авторы статьи будут использовать нейронную сеть на основе архитектуры YOLO для определения ограничивающих рамок кодов переработки.

2.2. OCR алгоритмы

Основными подходами и механизмами для реализации OCR систем являются Tesseract Model [25] и EAST (An Efficient and Accurate Scene Text Detector) [26]. Нейронная сеть EAST, известная своей эффективностью и точностью, разработана специально для обнаружения текста в естественной сцене. Она использует уникальную архитектуру, которая делит текстовую область на четырехугольники, что позволяет более точно локализовать и распознавать текст в естественных сценах. Это делает EAST особенно надежным механизмом при обработке текста неправильной формы и различной ориентации. Благодаря этому нейронная сеть EAST хорошо подходит для таких задач, как распознавание номерных знаков и извлечение текста сцены (области, которая попадает в поле зрения камеры). С другой стороны, Tesseract OCR, механизм оптического распознавания символов с открытым исходным кодом, разработанный Google, известен своей универсальностью и широкой языковой поддержкой. Первоначально разработанный в Hewlett-Packard в 1980-х годах, Tesseract претерпел значительные усовершенствования и теперь основан на методах глубокого обучения. Tesseract превосходно распознает текст на изображениях и документах, обеспечивая точные результаты даже при наличии сложных макетов и шрифтов. Его гибкость и адаптируемость делают его популярным выбором для различных приложений OCR: от оцифровки печатных документов до извлечения текста из изображений для различных целей, таких как анализ контента и извлечение данных.

Tesseract OCR использует архитектуру глубокого обучения для оптического распознавания символов. Последние версии Tesseract, включая Tesseract 4, интегрировали нейронные сети в процесс распознавания. Сверточные нейронные сети (CNN) используются для анализа областей изображения и извлечения функций, важных для распознавания символов. Модель нейронной сети Tesseract работает с сегментами текстовой строки, извлеченными из входного изображения. Он включает в себя рекуррентные нейронные сети (RNN), в частности сети долгосрочной краткосрочной памяти (LSTM), для фиксации контекстных зависимостей и повышения точности распознавания, особенно в сценариях со сложными макетами или различными стилями шрифтов. Архитектура нейронной сети Tesseract OCR [27] спроектирована так, чтобы быть гибкой, и может быть точно настроена для конкретных задач распознавания, что способствует ее адаптации к широкому спектру приложений OCR.

Нейронная сеть EAST, предназначенная для обнаружения текста сцены, представляет архитектуру, оптимизированную для эффективности и точности. EAST использует U-образную архитектуру, часто называемую архитектурой U-Net, которая представляет собой тип сверточной нейронной сети (CNN). Эта архитектура позволяет EAST обрабатывать изображения с разными разрешениями, фиксируя как мелкие детали, так и грубую контекстную информацию. Уникальной особенностью EAST является выходной слой, который создает четырехугольные прогнозы для текстовых областей, включая координаты ограничивающего прямоугольника и параметр угла. Такой выбор архитектуры позволяет EAST эффективно обрабатывать текст с произвольной ориентацией, что делает его надежным в сценариях, где текст может появляться в разных ракурсах, например, на знаках или в природных сценах.

Таким образом, и Tesseract OCR, и EAST используют архитектуру нейронных сетей для выполнения своих конкретных задач. Tesseract фокусируется на распознавании символов с помощью CNN и LSTM, а EAST специализируется на обнаружении текста сцены с помощью своей архитектуры, вдохновленной U-Net, что обеспечивает точное и эффективное извлечение текста из сложных изображений.

3. Предлагаемый подход к улучшению процесса сортировки вторсырья

Авторы предлагают новый подход к улучшению процесса сортировки вторсырья, основанный на использовании комбинации двух специализированных моделей машинного обучения. Первая нейронная сеть – детектор на основе архитектуры YOLOv7 – необходима для определения ограничивающих рамок кодов перерабатываемых материалов и предварительной классификации символа (отнесения его к определенному типу перерабатываемых материалов). Вторая нейронная сеть EAST нужна для определения цифры и буквы внутри ограничивающей рамки, полученных в результате выхода первой сети. Использование второй сети предлагается только в том случае, если уверенность модели YOLOv7 в предсказанном классе ниже 0.5. Далее результат OCR модели по распознаванию текста сравнивается с шаблонами текстов, которые могут быть на условных знаках кодов переработки. Например, код переработки полиэтилена низкой плотности может быть как 4 LDPE, так и 4 PE-LD. Архитектура предлагаемой авторами системы представлена на рисунке 4.

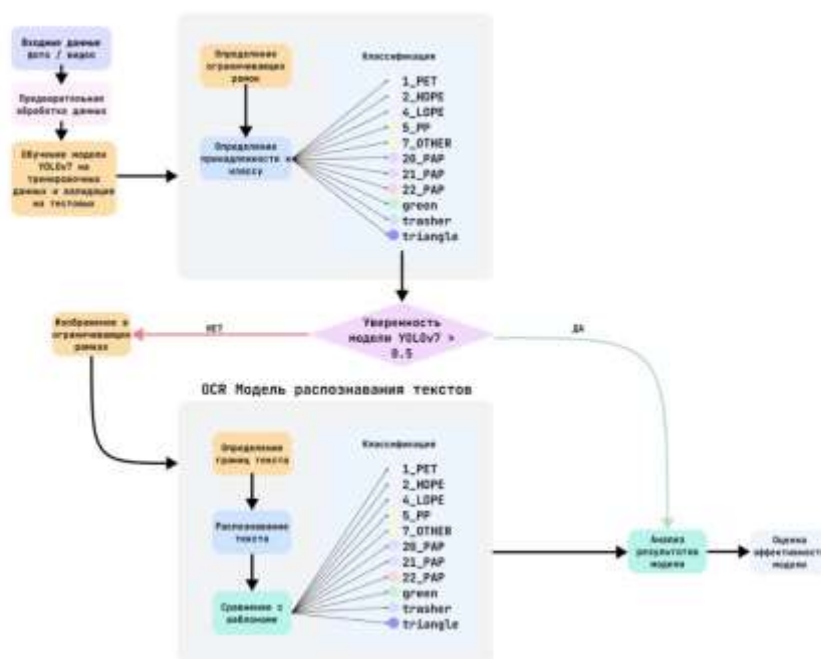


Рисунок 4 – предлагаемая архитектура системы по распознаванию и классификации кодов перерабатываемых материалов

3.1. Процесс сбора и предварительной обработки данных

Набор изображений для обучения нейронных сетей включает в себя фотографии упаковок продуктов питания и бытовых товаров, находящиеся в свободном доступе в сети интернет, а также снимков, сделанных авторами самостоятельно на камеру смартфона. Система обучается на изображениях с 11 условными обозначениями, которые чаще всего встречаются в повседневной жизни.

В итоге было собрано 531 изображение с маркировками кодов переработки на продуктах.

Исходя из собранного фотоматериала, авторы данной статьи определили, что очень часто коды переработки либо «выдавлены» на поверхности продукта и не имеют четкого цветового отличия от фона, либо находятся на прозрачных поверхностях. Поэтому было принято решение произвести предварительную обработку изображений таким образом, чтобы явно выделить границы маркировки переработки.

Были рассмотрены несколько ядер сверточного слоя для обработки изображений, такие как:

1. Ядро контура (outline) $\begin{pmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{pmatrix}$
2. Ядро контура второй версии (custom) $\begin{pmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{pmatrix}$
3. Ядро тиснения (emboss) $\begin{pmatrix} -2 & -1 & 0 \\ -1 & 1 & 1 \\ 0 & 1 & 2 \end{pmatrix}$
4. Совместное ядро контура и тиснения (outline+emboss) $\begin{pmatrix} -3 & -2 & -1 \\ -2 & 9 & 0 \\ -1 & 0 & 1 \end{pmatrix}$

Однако после получения результатов тестирования этих ядер сверточного слоя авторы статьи пришли к выводу, что они не позволяют достичь необходимого результата, и символы переработки, нанесенные на прозрачные поверхности или сделанные с помощью тиснения, выделяются на отсканированных изображениях нечетко. По этой причине было принято решение использовать оператор или фильтр Собеля. Пример изображения со сложными вариантами нанесения кодов перерабатываемых материалов представлен на рисунке 5, а результаты применения фильтра Собеля к этому изображению – на рисунке 6.



Рисунок 5 – Исходное изображение с примерами сложно читаемых кодов перерабатываемых материалов



Рисунок 6 – Результат применения оператора Собеля к изображению

Оператор Собеля используется для выделения границ и контуров на изображениях. Он применяет два свёрточных ядра (маски) к изображению для нахождения горизонтальных и вертикальных градиентов интенсивности. Эти свёрточные ядра, часто называемые фильтрами, вычисляют первую производную яркости изображения.

Для вычисления градиентов интенсивности оператор Собеля использует следующие ядра:

Горизонтальное ядро (выделяет горизонтальные границы):

$$G_x = \begin{pmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{pmatrix}$$

Вертикальное ядро (выделяет вертикальные границы):

$$G_y = \begin{pmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{pmatrix}$$

Изображение обрабатывается каждым из свёрточных ядер (горизонтальным и вертикальным) отдельно. Это создает два градиентных изображения G_x и G_y . Для каждого пикселя вычисляется результирующая градиентная величина, которая комбинирует горизонтальный и вертикальный градиенты. Это делается по следующей формуле:

$$G = \sqrt{G_x^2 + G_y^2}$$

На итоговом изображении, обработанном оператором Собеля, становятся видны контуры и границы объектов. Эти границы отображают резкие изменения яркости, что позволяет выделить и визуализировать ключевые структурные элементы изображения.

Как можно убедиться, оператор Собеля значительно улучшает выделение кода переработки на прозрачной поверхности, делая его более заметным и четким. На остальных поверхностях оператор Собеля также способствует выделению штампованного кода переработки, улучшая контраст и делая код более различимым на фоне.

Далее авторами статьи был написан класс аугментации изображений. Он предполагает, что изображение подгоняется под общий формат (640x640 пикселей), происходит случайный поворот, горизонтальное отражение и изменение цвета.

После создания аугментированных данных необходимо данные разметить. Сервис roboflow [28] предоставляет удобные инструменты для этих целей. С его помощью со-

здается необходимый для обучения сети YOLOv7 файл с описанием классов и координат ограничивающих рамок объектов на изображении. Процесс разметки данных представлен ниже на рисунке 7.

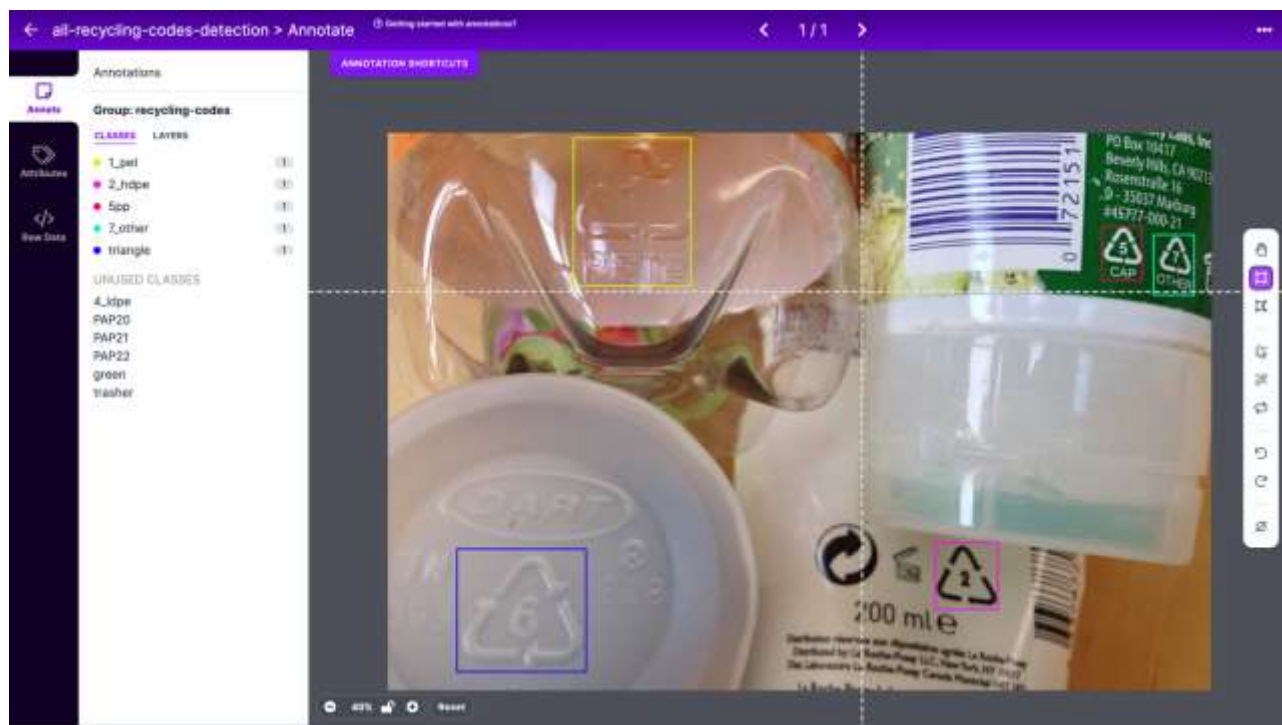


Рисунок 7 – Процесс разметки изображений в тренировочном наборе данных

3.2. Реализация нейронной сети YOLOv7 и её обучение

Модель, используемая авторами для реализации, представляет собой предварительно обученную модель YOLOv7 на данных MS COCO [29]. Данные MS COCO, в свою очередь, представляют собой изображения для 80 классов объектов – от машин до зубной щетки. Использование предварительно обученной модели позволяет сократить время на обучение модели на наших данных, так как сверточные слои, которые создают карту признаков, уже обучены на достаточно хорошем уровне. Модель, обученная на данных MS COCO, находится в открытом доступе [30]. Для обучения, предварительной обработки и получения предсказаний используется язык программирования Python и его стандартные библиотеки для анализа данных и машинного обучения: pandas, scikit-learn, Tensorflow и OpenC.

YOLOv7 состоит из 77 слоев, которые представляют собой последовательную комбинацию сверточных слоев, ищущих карты признаков, а также слоев конкатенации и MaxPool. Модель обучалась на 400 эпохах. Стоит отметить, что процесс обучения модели YOLOv7 занимает довольно продолжительное время в несколько часов.

Для оценки качества модели использовались 2 метрики: Mean Average Precision (mAP) [31] и Intersection over Union (IoU) [32].

Значение ошибки определяется тем, насколько далеко прогнозируемая ограничивающая рамка находится от истинной ограничивающей рамки. Если прогнозируемая ограничивающая рамка находится далеко, оценка потерь высока. График обучения модели представлен на рисунке 8.

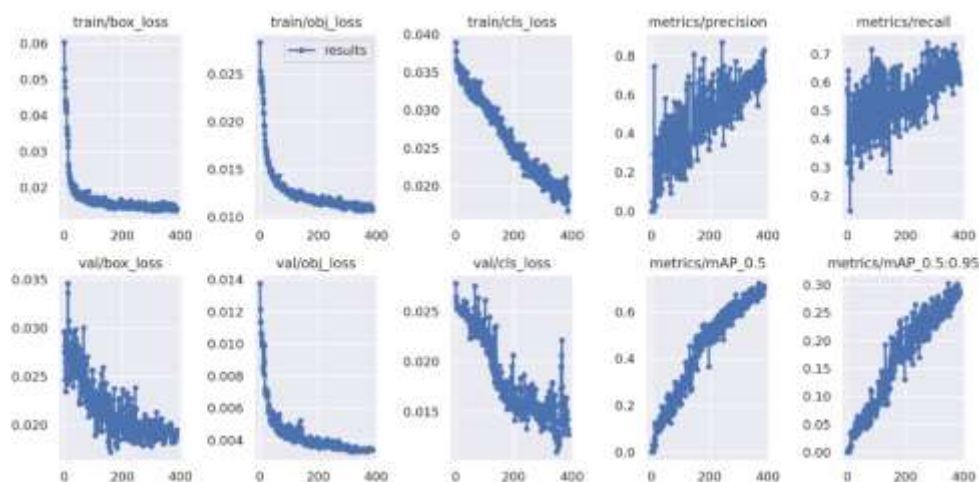


Рисунок 8 – Графики обучения YOLOv7 на тренировочных данных

После обучения первой нейронной сети из системы удалось достичь метрики качества mAP в 79% на тренировочных данных и в 70% на тестовых данных. Диаграммы со значениями средней точности по классам (average precision) представлены на рисунках 9 и 10.

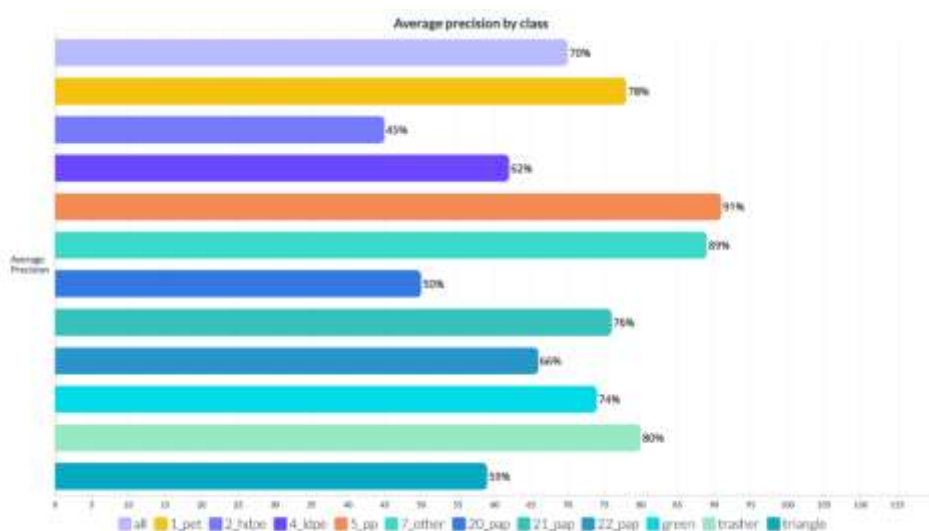


Рисунок 9 – Average Precision по классам на тренировочных данных

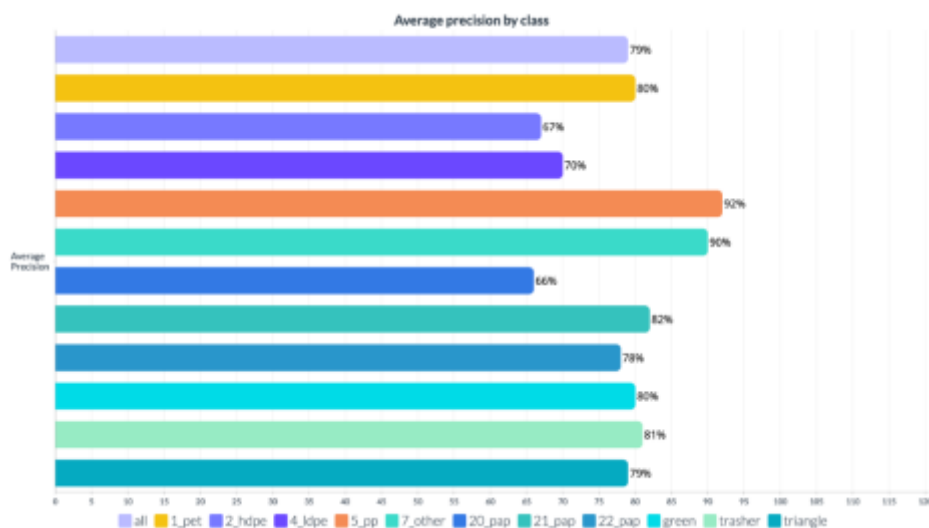


Рисунок 10 – Average Precision по классам на тестовых данных

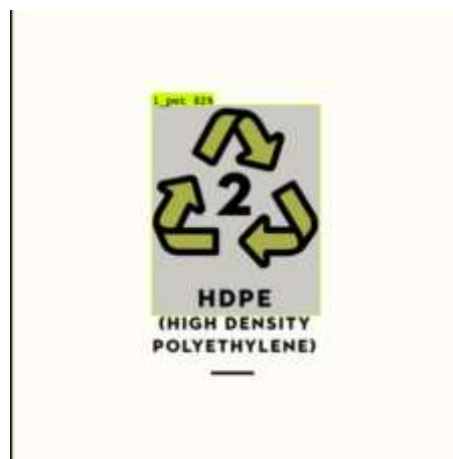
По итогу демонстрации работоспособности модели можно сказать, что она хорошо научилась обобщать знания обо всех классах, довольно точно и с большой уверенностью определяет области интереса (необходимые коды переработки на упаковках товаров). Однако можно заметить, что она ошибается в классах своего предсказания. Это может быть связано с тем, что логотипы классов переработки имеют множество вариантов представления. К тому же они все очень похожи друг на друга и отличаются в лучшем случае цифрой внутри треугольника или надписью. Тестирование предсказаний модели на изображениях представлено на рисунке 11.а – 11.в.



а



б



в

Рисунок 11. Пример работы модели на тестовых изображениях: а) тест 1 б) тест 2 в) тест 3

На рисунках 11.а и 11.б представлена визуализация ограничивающих рамок для кодов перерабатываемых материалов на этикетках товаров. Цифры над ограничивающими рамками представляют собой визуализацию вероятности принадлежности выделенного объекта к предсказанному моделью классу. На рисунке 11.в отчетливо виден пример, когда модель качественно определила ограничивающую рамку кода перерабатываемого материала, однако ошибается в предсказанном классе.

Для уменьшения доли таких ложно положительных результатов классификации авторы статьи предлагают использовать дополнительно к модели YOLOv7 модели OCR, такую, как EAST.

3.3. Применение EAST для улучшения предсказаний системы

В качестве имплементируемой модели распознавания текста на изображении была выбрана модель Efficient and Accurate Scene Text Detector. Для этого используется библиотека OpenCV языка программирования Python.

Чтобы реализовать модель EAST для обнаружения текста в Python, необходимо выполнить следующие ключевые шаги:

Шаг 1: Установка необходимых зависимостей:

Перед началом реализации необходимо убедиться, что установлены необходимые зависимости. Основные библиотеки включают OpenCV, NumPy и argparse. OpenCV используется для обработки изображений, предоставляя различные инструменты и алгоритмы, такие как оператор Собеля. NumPy необходима для работы с массивами данных и выполнения численных операций. Argparse используется для обработки аргументов командной строки, что позволяет удобно передавать параметры и настройки в скрипт.

Шаг 2: Загрузка предварительно обученной модели EAST:

Модель EAST требует использования предварительно обученных весов и файла конфигурации. Их можно получить из официального репозитория EAST.

Шаг 3: Предварительная обработка изображения:

Прежде чем передать изображение через модель EAST, необходимо выполнить некоторые шаги предварительной обработки. Этот этап необходим для улучшения читабельности текстовых областей, так как различные погрешности значительно затрудняют работу моделей OCR. Обычно используется преобразование изображения в оттенки серого и применение методов размытия и шумоподавления. Кроме этого, необходимо привести ширину и высоту изображения к числам, делящимся на 32. Это обязательное условие модели EAST.

После получения изображений из выхода первой модели (детектора) остается лишь привести ширину и высоту изображения полученной области с кодом переработки к числам, кратным 32, так как к изображению уже предварительно применен оператор Собеля.

Шаг 4: Выполнение обнаружения текста:

Чтобы выполнить обнаружение текста с использованием модели EAST, необходимо передать предварительно обработанное изображение через сеть и извлечь координаты ограничивающей рамки обнаруженных текстовых областей.

Рисунок 12 иллюстрируют результаты работы модели EAST для распознавания текста на изображении, полученном в рамках предыдущего предсказания модели YOLOv7.



Рисунок 12 – Распознавание текста моделью EAST на основе результирующих ограничивающих рамок из модели YOLOv7

После применения OCR нейронной сети EAST дополнительно к результатам модели YOLOv7 удалось значительно улучшить результаты всей системы.

Предсказания системы стали точнее, а по метрике качества mAP удалось достичь уровня в 93%. Для проверки работы системы осуществляются предсказания на новых изображениях, а также построена матрица ошибок (рисунок 13).

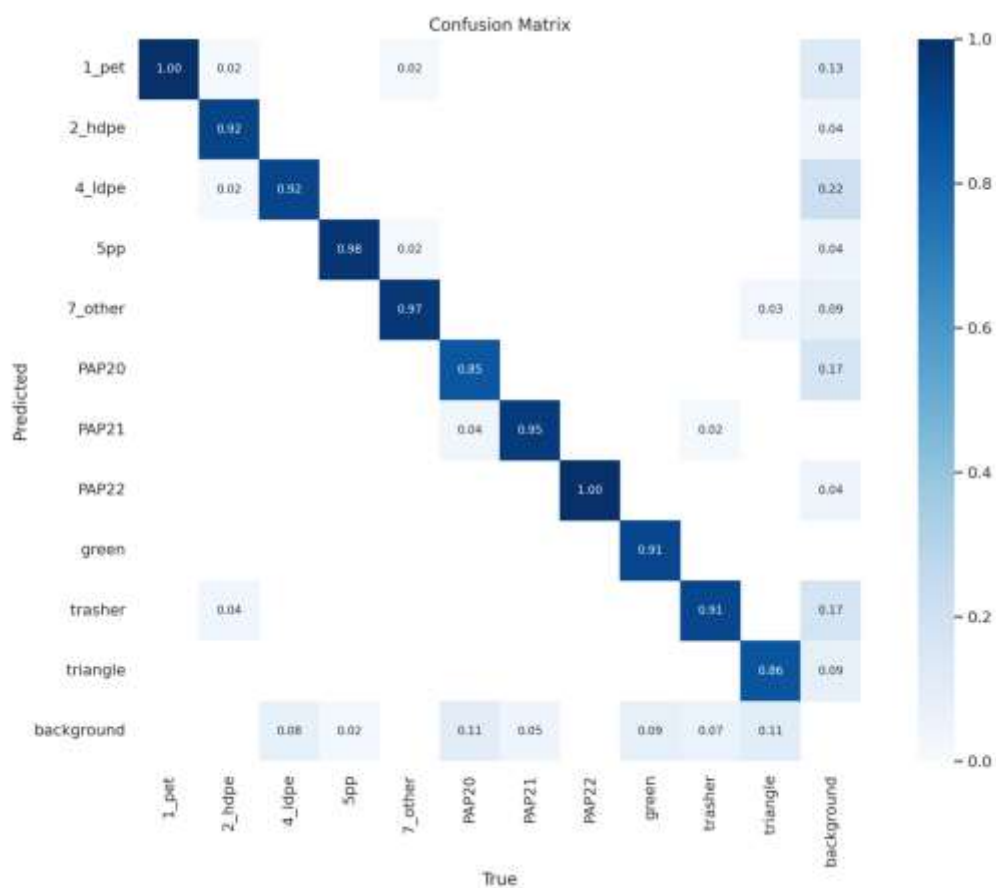


Рисунок 13 – Матрица ошибок

4. Заключение

В ходе данной работы были рассмотрены основные подходы к детекции объектов в видеопотоке и на изображениях в режиме реального времени, а также основные технологии в области оптического распознавания символов. Были проанализированы существующие исследования по применимости таких систем детекции объектов и оптического распознавания символов к различным областям. Авторами был предложен вариант системы по детекции и распознаванию кодов перерабатываемых материалов на упаковках для более точной сортировки вторсырья.

Предложенная система, основанная на передовых технологиях в области детекции объектов в реальном времени и оптического распознавания символов, в частности модели YOLOv7 и EAST, продемонстрировала высокие результаты в точном обнаружении и классификации кодов переработки на упаковках продуктов. Показатели производительности модели указывают на ее потенциал для реального применения в системах автоматизированной переработки вторсырья.

Практическое применение системы по распознаванию кодов перерабатываемых материалов представляет собой важный шаг в сфере устойчивого управления отходами. В контексте визуальной обработки изображений, система демонстрирует способность эффективно выделять и интерпретировать символы на упаковке, что имеет ключевое значение для создания мобильных приложений и автоматизированных пунктов приёма. Мобильное приложение, основанное на данной технологии, может стать надежным инструментом в повседневной практике сбора и сортировки мусора. Пользователи смогут легко сканировать специальные коды на упаковках, получая мгновенную информацию о том, как правильно классифицировать и утилизировать отходы. Это создает интерактивное и образовательное взаимодействие, способствуя повышению осведомлен-

ности и вовлеченности общества в процесс устойчивого потребления и управления ресурсами.

Более того, автоматизированные пункты приема с системой сортировки вторсырья усиливают эффективность процесса переработки отходов. Граждане могут легко сдавать отдельно собранные материалы, зная, что их усилия направлены на поддержание зеленого образа жизни и экологической устойчивости. Такие инновационные решения не только упрощают жизнь людей, но и активно способствуют формированию экологически ответственного общества, где утилизация ресурсов становится неотъемлемой частью повседневной рутины.

Данная система позволяет сохранить текущие процессы производства упаковок со стороны производителей, для унификации условных обозначений кодов перерабатываемых материалов. Предложенная авторами система обеспечивает высокую эффективность классификации материалов вторсырья, даже при наличии различий в изображении условных обозначений перерабатываемых материалов. Визуальная точность и надёжность системы подчёркивают её значимость для практического применения в области переработки и управления отходами.

Благодарности

Работа выполнена в рамках государственного задания по проекту «Аналитические и численные методы исследования сложных систем и нелинейных задач математической физики» (мнемокод FSWU-2023-0031).

Список литературы:

1. 97/129/EC: Commission Decision of 28 January 1997 establishing the identification system for packaging materials pursuant to European Parliament and Council Directive 94/62/EC on packaging and packaging waste. EUR-Lex. 28.01.1997. URL: <https://eur-lex.europa.eu/legal-content/EN/ALL/?uri=CELEX%3A31997D0129> (accessed: 02.10.2023).
2. Цифровой код и буквенное обозначение (аббревиатура) материала, из которого изготавливается упаковка (укупорочные средства). Технический регламент Таможенного союза О безопасности упаковки ТР ТС 005/2011. 18.10.2016. ЕЭК. URL: <https://eec.eaeunion.org/comission/departement/deptexreg/tr/bezopypakovki.php> (дата обращения: 01.10.2023).
3. Feng D., Harakeh A., Waslander S., Dietmayer K. A Review and Comperative Study on Probabilistic Object Detection in Autonomous Driving // IEEE Transactions on Intelligent Transportation Systems (<https://arxiv.org/abs/2011.10671>)
4. Li Y., Hu H., Liu Z., Zhao D. Influence of Camera-LiDAR Configuration on 3D Object Detection for Autonomous Driving (<https://arxiv.org/abs/2310.05245>)
5. Fuchs K., Grundmann T., Fleisch E. Towards Identification of Packaged Products via Computer Vision: Convolutional Neural Networks for Object Detection and Image Classification in Retail Environments // 9th International Conference on the Internet of Things. 2019. Vol. 9. P. 1-3. (https://www.researchgate.net/publication/337068624_Towards_Identification_of_Packaged_Products_via_Computer_Vision_Convolutional_Neural_Networks_for_Object_Detection_and_Image_Classification_in_Retail_Environments)
6. Christy M., Gupta A., Grumbach E., Mandell L., Furuta R., and Gutierrez-Osuna R. Mass Digitization of Early Modern Texts with Optical Character Recognition // ACMJ. Comput. Cult. Herit. 11, 1, Article 6 (December 2017). P. 1-17. (<https://psi.engr.tamu.edu/wp-content/uploads/2018/01/christy2017jcch.pdf>)
7. Yimyam W., Ketcham M., Jensuttiwetchakult T., Hiranchan S. Enhancing and Evaluating an Impact of OCR and Ontology on Financial Document Checking Process // 2020 15th International Joint Symposium on Artificial Intelligence and Natural Language Processing (iSAI-NLP). 2020

8. Li X., Hu G., Teng X., Xie G. Building structured personal health records from photographs of printed medical records // American Medical Informatics Association Annual Symposium (AMIA). 2016. P. 1-5.
(https://www.researchgate.net/publication/308724882_Building_structured_personal_health_records_from_photographs_of_printed_medical_records)
9. Ko T. A survey on behavior analysis in video surveillance for homeland security applications // Video Surveillance. 2011. P. 279-294.
10. Saluja R., Maheshwari A., Ramakrishnan G., Chaudhuri P., Carman M. OCR On-the-Go: Robust End-to-end Systems for Reading License Plates & Street Signs // International Conference on Document Analysis and Recognition (ICDAR). 2019. P. 1-6.
(<https://www.cse.iitb.ac.in/~ganesh/papers/icdar2019b.pdf>)
11. Sakr G., Mokbel M., Darwich A., Mia Nasr Khneisser, Hadi A. Comparing deep learning and support vector machines for autonomous waste sorting // IEEE International Multi-disciplinary Conference on Engineering Technology (IMCET). 2016.
12. Klette, R. Concise Computer Vision: An introduction into Theory and Algorithms // Springer. USA. 2014. P. 339-379.
13. N. Islam, Z. Islam, N. Noor. A Survey on Optical Character Recognition System // Journal of Information & Communication Technology-JICT V. 10. 2016. P. 1-3.
14. Nagy G., Nartker N. A., Rice S. V. Optical character recognition: an illustrated guide to the frontier // Society of Photo-Optical Instrumentation Engineers. 1999. P.7-57.
15. Singh S., Mamatha K., Ragothaman S., Raj K.D., Anusha N., Susmi Z. Waste Segregation System Using Artificial Neural Networks // HELIX. 2017. № 7. P. 2053-2058.
16. Devi R.S., Vijaykumar V., Muthumeena M. Waste Segregation using Deep Learning Algorithm // International Journal of Innovative Technology and Exploring Engineering (IJITEE). V. 8. 2018. P. 1-3.
17. AlexNet. PyTorch documentation // URL:
<https://pytorch.org/vision/main/models/generated/torchvision.models.alexnet.html> (дата обращения: 11.10.2023)
18. GoogLeNet. PyTorch documentation // URL:
https://docs.openvino.ai/latest/omz_models_model_googlenet_v3.html (дата обращения 11.10.2023)
19. Mittal P., Singh R. Deep learning-based object detection in low-altitude UAV datasets: A survey // Image and Vision Computing. 2020. P. 4-8.
(https://www.researchgate.net/publication/346201758_Deep_learning-based_object_detection_in_low-altitude_UAV_datasets_A_survey)
20. J. Hui. SSD THESIS. Dec. 2020 // URL: <https://jonathan-hui.medium.com/ssd-object-detection-single-shot-multibox-detector-for-real-time-processing-9bd8deac0e06>
21. Li C., Li L., Jiang H., Weng K., others. YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications. 2022. P. 3-10.
22. S. Sahel, M. Alsahafi, M. Alghamdi, and T. Alsubait. Logo detection using deep learning with pretrained CNN models. Engineering, Technology & Applied Science Research, vol. 11, no. 1. 2021. P. 6724-6729.
23. B. Jabir, N. Falih, and K. Rahmani. REF2 THESIS. International Journal of Online & Biomedical Engineering, vol. 17, no. 5, 2021.
24. V. Choudhari, R. Pedram, and S. Vartak. Comparison between YOLO and SSD MobileNet for Object Detection in a Surveillance Drone. Tech. Rep. Oct. 2021.
25. Tesseract documentation // URL:
<https://tesseract-ocr.github.io/tessdoc/AddOns.html#tesseract-wrappers> (дата обращения: 9.04.2023)
26. Xinyu Zhou, Cong Yao, He Wen, Yuzhi Wang, Shuchang Zhou, Weiran He and Jiajun Liang. EAST: An Efficient and Accurate Scene Text Detector. Computer Vision Foundation. Megvii Technology Inc., Beijing, China // URL:

https://openaccess.thecvf.com/content_cvpr_2017/papers/Zhou_EAST_An_Efficient_CVP_R_2017_paper.pdf

27. Github. Overview of the new neural network system in Tesseract 4.00 // URL: <https://github.com/tesseract-ocr/tessdoc/blob/main/tess4/NeuralNetsInTesseract4.00.md> (дата обращения: 10.11.2023)

28. Roboflow. E-service // URL: <https://roboflow.com/>

29. MS COCO. Dataset description. // URL: <https://cocodataset.org/#home>

30. Github. Pretrained YOLOv7 model on MS COCO dataset // URL: <https://github.com/WongKinYiu/yolov7> (дата обращения: 9.11.2023)

31. R. J. Tan. Map thesis. Mar. 2022 // URL: <https://towardsdatascience.com/breaking-down-mean-average-precision-map-ae462f623a52>

32. Zhang H., Xu C., Zhang S. Inner IoU: More Effective Intersection over Union Loss with Auxiliary Bounding Box. 2023. P. 1-3. (<https://arxiv.org/abs/2311.02877>)

Deep Learning for Effective Visualization and Classification of Recyclable Material Labels

V.O. Kuzevanov¹, D.V. Tikhomirova²

National Research Nuclear University “MEPhI”, Moscow, Russia

¹ ORCID: 0009-0005-7093-7709, slava.kuzevanov@mail.ru

² ORCID: 0000-0002-0812-2331, dvsulim@mail.ru

Abstract

This paper presents an example of a system to improve the process of sorting recyclables by using deep learning techniques to automatically detect, classify and visualize recycling codes on product packages. In this paper, the authors discuss various approaches to optical character recognition and object detection in a video stream or image. The authors have developed and proposed a combination of neural networks for detection and classification of recycling codes. The proposed neural network system is designed to facilitate efficient recycling processes by automating the identification of recycling symbols, thereby facilitating the sorting and processing of recyclables.

Keywords: Deep learning, neural networks, recycling codes, automated sorting of recyclables, computer vision, image recognition.

References

1. 97/129/EC: Commission Decision of 28 January 1997 establishing the identification system for packaging materials pursuant to European Parliament and Council Directive 94/62/EC on packaging and packaging waste. EUR-Lex. 28.01.1997. URL: <https://eur-lex.europa.eu/legal-content/EN/ALL/?uri=CELEX%3A31997D0129> (accessed: 02.10.2023).
2. Цифровой код и буквенное обозначение (аббревиатура) материала, из которого изготавливается упаковка (укупорочные средства) [Numeric, Alphabetic (Abbreviation) Designation of the Material from Which the Packaging (Closures) is Made]. Технический регламент Таможенного союза О безопасности упаковки ТР ТС 005/2011 [Technical regulations of the Customs Union "On Safety of Packaging"]. 18.10.2016. ЕЭК [Eurasian Economic Commission]. URL: <https://eec.eaeunion.org/comission/departement/deptexreg/tr/bezopypakovki.php> (accessed 01.10.2023).
3. Feng D., Harakeh A., Waslander S., Dietmayer K. A Review and Comperative Study on Probabilistic Object Detection in Autonomous Driving // IEEE Transactions on Intelligent Transportation Systems (<https://arxiv.org/abs/2011.10671>)
4. Li Y., Hu H., Liu Z., Zhao D. Influence of Camera-LiDAR Configuration on 3D Object Detection for Autonomous Driving (<https://arxiv.org/abs/2310.05245>)
5. Fuchs K., Grundmann T., Fleisch E. Towards Identification of Packaged Products via Computer Vision: Convolutional Neural Networks for Object Detection and Image Classification in Retail Environments // 9th International Conference on the Internet of Things. 2019. Vol. 9. P. 1-3. (https://www.researchgate.net/publication/337068624_Towards_Identification_of_Packaged_Products_via_Computer_Vision_Convolutional_Neural_Networks_for_Object_Detection_and_Image_Classification_in_Retail_Environments)
6. Christy M., Gupta A., Grumbach E., Mandell L., Furuta R., and Gutierrez-Osuna R. Mass Digitization of Early Modern Texts with Optical Character Recognition // ACMJ. Com-

put. Cult. Herit. 11, 1, Article 6 (December 2017). P. 1-17. (<https://psi.engr.tamu.edu/wp-content/uploads/2018/01/christy2017jcch.pdf>)

7. Yimyam W., Ketcham M., Jensuttiwetchakult T., Hiranchan S. Enhancing and Evaluating an Impact of OCR and Ontology on Financial Document Checking Process // 2020 15th International Joint Symposium on Artificial Intelligence and Natural Language Processing (iSAI-NLP). 2020

8. Li X., Hu G., Teng X., Xie G. Building structured personal health records from photographs of printed medical records // American Medical Informatics Association Annual Symposium (AMIA). 2016. P. 1-5.

(https://www.researchgate.net/publication/308724882_Building_structured_personal_health_records_from_photographs_of_printed_medical_records)

9. Ko T. A survey on behavior analysis in video surveillance for homeland security applications // Video Surveillance. 2011. P. 279-294.

10. Saluja R., Maheshwari A., Ramakrishnan G., Chaudhuri P., Carman M. OCR On-the-Go: Robust End-to-end Systems for Reading License Plates & Street Signs // International Conference on Document Analysis and Recognition (ICDAR). 2019. P. 1-6. (<https://www.cse.iitb.ac.in/~ganesh/papers/icdar2019b.pdf>)

11. Sakr G., Mokbel M., Darwich A., Mia Nasr Khneisser, Hadi A. Comparing deep learning and support vector machines for autonomous waste sorting // IEEE International Multi-disciplinary Conference on Engineering Technology (IMCET). 2016.

12. Klette, R. Concise Computer Vision: An introduction into Theory and Algorithms // Springer. USA. 2014. P. 339-379.

13. N. Islam, Z. Islam, N. Noor. A Survey on Optical Character Recognition System // Journal of Information & Communication Technology-JICT V. 10. 2016. P. 1-3.

14. Nagy G., Nartker N. A., Rice S. V. Optical character recognition: an illustrated guide to the frontier // Society of Photo-Optical Instrumentation Engineers. 1999. P.7-57.

15. Singh S., Mamatha K., Ragothaman S., Raj K.D., Anusha N., Susmi Z. Waste Segregation System Using Artificial Neural Networks // HELIX. 2017. № 7. P. 2053-2058.

16. Devi R.S., Vijaykumar V., Muthumeena M. Waste Segregation using Deep Learning Algorithm // International Journal of Innovative Technology and Exploring Engineering (IJITEE). V. 8. 2018. P. 1-3.

17. AlexNet. PyTorch documentation // URL: <https://pytorch.org/vision/main/models/generated/torchvision.models.alexnet.html> (accessed: 11.10.2023)

18. GoogLeNet. PyTorch documentation // URL: https://docs.openvino.ai/latest/omz_models_model_googlenet_v3.html (accessed: 11.10.2023)

19. Mittal P., Singh R. Deep learning-based object detection in low-altitude UAV datasets: A survey // Image and Vision Computing. 2020. P. 4-8. (https://www.researchgate.net/publication/346201758_Deep_learning-based_object_detection_in_low-altitude_UAV_datasets_A_survey)

20. J. Hui. SSD THESIS. Dec. 2020 // URL: <https://jonathan-hui.medium.com/ssd-object-detection-single-shot-multibox-detector-for-real-time-processing-9bd8deacoe06>

21. Li C., Li L., Jiang H., Weng K., others. YOLOv6: A Single-Stage Object Detection Framework for Industrial Applications. 2022. P. 3-10.

22. S. Sahel, M. Alsahafi, M. Alghamdi, and T. Alsubait. Logo detection using deep learning with pretrained CNN models. Engineering, Technology & Applied Science Research, vol. 11, no. 1. 2021. P. 6724-6729.

23. B. Jabir, N. Falih, and K. Rahmani. REF2 THESIS. International Journal of Online & Biomedical Engineering, vol. 17, no. 5, 2021.

24. V. Choudhari, R. Pedram, and S. Vartak. Comparison between YOLO and SSD MobileNet for Object Detection in a Surveillance Drone. Tech. Rep. Oct. 2021.

25. Tesseract documentation // URL: <https://tesseract-ocr.github.io/tessdoc/AddOns.html#tesseract-wrappers> (accessed: 9.04.2023)
26. Xinyu Zhou, Cong Yao, He Wen, Yuzhi Wang, Shuchang Zhou, Weiran He and Jiajun Liang. EAST: An Efficient and Accurate Scene Text Detector. Computer Vision Foundation. Megvii Technology Inc., Beijing, China // URL: https://openaccess.thecvf.com/content_cvpr_2017/papers/Zhou_EAST_An_Efficient_CVP_R_2017_paper.pdf
27. Github. Overview of the new neural network system in Tesseract 4.00 // URL: <https://github.com/tesseract-ocr/tessdoc/blob/main/tess4/NeuralNetsInTesseract4.00.md> (accessed: 10.11.2023)
28. Roboflow. E-service // URL: <https://roboflow.com/>
29. MS COCO. Dataset description. // URL: <https://cocodataset.org/#home>
30. Github. Pretrained YOLOv7 model on MS COCO dataset // URL: <https://github.com/WongKinYiu/yolov7> (accessed: 9.11.2023)
31. R. J. Tan. Map thesis. Mar. 2022 // URL: <https://towardsdatascience.com/breaking-down-mean-average-precision-map-ae462f623a52>
32. Zhang H., Xu C., Zhang S. Inner IoU: More Effective Intersection over Union Loss with Auxiliary Bounding Box. 2023. P. 1-3. (<https://arxiv.org/abs/2311.02877>)