

УЛУЧШЕНИЕ КАЧЕСТВА РАСПОЗНАВАНИЯ ПАРАМЕТРОВ ФЛОТОМАШИНЫ С ИСПОЛЬЗОВАНИЕМ НЕЙРОННОЙ СЕТИ

А.В. Затонский¹, С.В. Кузнецов¹, К.А. Саломатова¹

¹ Пермский национальный исследовательский политехнический университет,
Березниковский филиал, Березники, Россия, e-mail: zxenon@narod.ru

Аннотация: Рассмотрены алгоритмы решения одной из задач контроля за процессом флотации с применением системы компьютерного зрения. Выявлено, что существующий алгоритм синхронизации с пеногоном флотационной машины, основанный на изменениях уровня освещенности, обрабатывает не все проходы пеногона. Предложено улучшить существующие алгоритмы за счет использования нейронной сети для распознавания лопастей пеногона. Представлены результаты проверки работоспособности и адекватности обученной модели распознавания лопастей пеногона. Произведена сравнительная оценка разработанного алгоритма с использованием модели нейронной сети и ранее предложенного алгоритма. Сделан вывод, что использование обученной модели нейронной сети позволит сократить количество пропущенных проходов по сравнению с предыдущим алгоритмом на 5,1%.

Ключевые слова: калийные удобрения, пенная флотация, автоматизация, компьютерное зрение, машинное обучение, распознавание объектов, распознавание в реальном времени, YOLO.

Для цитирования: Затонский А. В., Кузнецов С. В., Саломатова К. А. Улучшение качества распознавания параметров флотомашин с использованием нейронной сети // Горный информационно-аналитический бюллетень. – 2025. – № 6. – С. 153-167. DOI: 10.25018/0236_1493_2025_6_0_153.

Improvement of detection quality for parameters of flotation machines using neural network

A.V. Zatonskiy¹, S.V. Kuznetsov¹, K.A. Salomatova¹

¹ Perm National Research Polytechnic University, Berezniki Branch,
Berezniki, Russia, e-mail: zxenon@narod.ru

Abstract: The algorithms of solution of a problem connected with the control over flotation process using computer vision are reviewed. It is revealed that the algorithm of foam generator unit synchronization with flotation machine, with the measurement of an illuminance level, covers not all necessary runs. It is proposed to improve the existing algorithms by using neural network for the identification of blades of the foam generator unit. The check-up data on the efficiency and adequacy of the learned model of identifying blades of the foam generator unit are described. The developed algorithm using the neural network model is compared with the ear-

lier proposed algorithm. It is concluded that the learned neural network model makes it possible to reduce the number of the missed runs as compared with the previous algorithm by 5.1%.

Key words: potassium fertilizers, foam flotation, automation, computer vision, machine learning, object detection, real-time detection, YOLO.

For citation: Zatonskiy A. V., Kuznetsov S. V., Salomatova K. A. Improvement of detection quality for parameters of flotation machines using neural network. *MIAB. Mining Inf. Anal. Bull.* 2025;(6):153-167. [In Russ]. DOI: 10.25018/0236_1493_2025_6_0_153.

Введение

В настоящее время из-за роста численности населения в мире возрастает значимость проблемы по созданию устойчивой продовольственной системы, что, в свою очередь, ведет к повышению ценности минеральных удобрений, от которых зависит продуктивность сельскохозяйственных культур. Увеличение их урожайности является одним из основных способов обеспечения продовольственной безопасности. Исходя из вышесказанного, можно сделать вывод, что глобальная потребность в минеральных удобрениях будет расти и дальше [1].

Калий является одним из ключевых элементов, необходимых для жизни. Он является естественным компонентом почвы и представляет собой незаменимый элемент питания для растений. Для полноценного роста и развития растения должны получать калий на всех этапах вегетации. Калийные соли, которые требуются для производства калийных удобрений, представляют собой ценный и востребованный ресурс, добыча которого постоянно увеличивается из-за растущего спроса в мировой экономике [2, 3].

На данный момент технологии, которые применяются в процессе добычи полезных ископаемых, сталкиваются с такими ограничениями, которые не позволяют обеспечить получение чистых минеральных ресурсов. Поэтому добы-

тые минералы отгружаются для дальнейшей переработки в смешанном состоянии, наряду с такими компонентами, которые залегают в тех же слоях с желаемыми минералами и представляют малую ценность или не представляют ее вообще [4].

В силу этого сложного процесса переработка ресурсов становится многогранной задачей, которая требует применения различных методов. Изначально после добычи происходит процесс предварительной обработки, на котором осуществляется первичное выделение ценных компонентов и удаление наиболее очевидных примесей. На следующем этапе полезные ископаемые подвергаются обогащению. Этот процесс включает в себя такие операции, как дробление и измельчение, в результате которых крупные куски минералов разрушаются на более мелкие фрагменты, что помогает отделить ценные компоненты от менее значимых, а также флотация, которая является одним из ключевых этапов обогащения и применяется для увеличения концентрации полезных ископаемых.

Основным методом обогащения руд является пенная флотация, которая основывается на отделении ценной руды от неценной. Подобный процесс происходит за счет гидрофобных свойств частиц, которые имеют склонность к избеганию контакта с водой, в результате чего такие частицы, которые обладают

низкой смачиваемостью, поднимаются на поверхность, тем самым образуя флотационную пену. В зависимости от состояния пены будет изменяться качество конечного продукта. Например, форма пузырьков влияет на способность пены удерживать минералы, поэтому стабильная пена обладает более округлыми пузырьками, а нестабильная пена, напротив, имеет пузырьки неправильной формы. Это, в свою очередь, ведет к тому, что минералы будут неэффективно отделяться и в результате увеличится потеря полезных компонентов руды. Кроме формы, состояние пены зависит от размера пузырьков, который влияет на скорость их подъема и способность захватывать материалы, соответственно, маленькие пузырьки более эффективны для захвата более мелких частиц, в то время как большие пузырьки — для захвата более крупных. Также процесс обогащения характеризуется цветом пены. Если он изменяется, это может свидетельствовать о том, что нежелательные минералы начинают всплывать.

Таким образом, пенная флотация основывается на визуальных характеристиках слоя пены, включая цвет, форму и размер пузырьков [5]. Эти параметры могут предоставлять важную информацию о состоянии производства, на основании чего работник может принимать решение.

Ручное распознавание параметров пены является нецелесообразным, потому что оно требует усилий и времени со стороны сотрудников. Кроме того, не стоит исключать человеческий фактор, так как работник может допустить ошибки. Использование автоматизированных систем технического зрения исключает участие оператора из процесса принятия решения о значениях измеряемых физических величин. Это экономит потенциальное время сотрудника,

который мог бы заняться другой полезной для производства деятельностью, а также приводит к объективному снижению риска ошибок, повышению точности и достоверности измерений [6]. Таким образом, задача автоматического распознавания параметров пены является актуальной.

Для распознавания параметров пены было разработано десктопное приложение на Python, которое обрабатывает изображения или видео и указывает количество пузырьков в кадре, среднее арифметическое и среднее медианное расстояние между центрами пузырьков, а также количество красной компоненты. В работе [7] для калийных флотомашин показана эффективность метода распознавания не границ пузырьков, а бликов, возникающих от точечного источника освещения, поэтому подсчет пузырьков в реализованном приложении производится за счет распознавания их бликов при бинаризации изображения. Особенностью пены при флотации хлорида калия, в отличие от флотации полиметаллов или угля, является блеклость, низкая контрастность изображения, отсутствие четко прорисованной границы пузырька. Поэтому из всех методов контроля состояния пены калийной флотомашины работоспособным оказался только бликовый. В работе [8] был предложен алгоритм обработки кадров в общем виде, используемый в разработанном приложении, включающий выделение кадров, в которых пеногон не мешает снимать пену, и бликовое распознавание параметров пены на них.

Однако проблема заключается в том, что пеногон во флотомашине вращается с непостоянной частотой, вследствие чего производить обработку кадра в приложении через постоянное количество секунд не представляется возможным. В таком случае со временем лопасти пеногона будут все больше попадать в вы-

деленную для обработки область кадра. Это приведет к искаженному распознаванию бликов, из-за чего могут быть приняты неверные решения. Решение данной проблемы обсуждалось в работах [9] и [10], но предложенные в данных работах алгоритмы не позволяют корректно идентифицировать наличие пеногона в области для обработки на всех имеющихся флотационных машинах на предприятиях ООО «Еврохим — Усольский калийный комбинат» и ПАО «Уралкалий». В диссертационной работе [9] был предложен алгоритм, который заключался в том, что наблюдался продолжительный период увеличения яркости, за которым следовал короткий последовательный отрезок уменьшения, после чего формировался сигнал синхронизации. Такой подход оказался работоспособным на флотомашинах ПАО «Уралкалий», где слабое внешнее освещение, однако условия съемки на ООО «Еврохим — Усольский калийный комбинат» отличаются сильным внешним освещением, где точечный источник света на выделяемую область не способен достичь вышеупомянутого эффекта, так как при слабой светимости точечного источника света блики остаются видны слабо, при сильной светимости обрабатываемая область засвечивается. Вследствие вышеупомянутых проблем использование данного алгоритма приводило к некорректным распознаваниям пеногона вплоть до 50%. Алгоритм, предложенный в диссертационных исследованиях К.А. Федосеевой [10], учитывая эту проблему, был нацелен на ее решение путем поиска участков со значительным увеличением яркости, выявления максимальной и минимальной освещенности за определенный период и формирования сигнала синхронизации в момент, когда яркость намного выше, чем в предыдущий момент времени. Такой подход оказался более

работоспособным, однако в таком случае алгоритму требуется некоторое количество отсчетов, чтобы «подхватить» нужное положение пеногона. Данная специфика работы алгоритма привела к тому, что при съемке пенной флотации в ООО «Еврохим — Усольский калийный комбинат» из-за высокой внешней освещенности кадра алгоритм пропускал некоторое количество «проходов», что в свою очередь искажает собираемые данные для поддержки принятия решений.

Решение вышеописанной проблемы возможно за счет использования такого алгоритма, который будет выполнять идентификацию пеногона в кадре на основе анализа не уровня освещенности, а изображения целиком. Метод построен на примере обработки видеорядов с калийных флотомашин, но, поскольку пеногоны других машин конструктивно схожи, должен быть работоспособен и на полиметаллических, и на угольных флотомашинах.

Идентификация положения пеногона на изображении возможна, наряду с другими методами, за счет использования нейронных сетей. Нейронные сети представляют из себя программы, которые нацелены на выполнение сложных когнитивных задач, например, распознавание образов. Нейронная сеть состоит из нескольких слоев. Входной слой принимает определенные параметры, затем полученные данные проходят через скрытые слои, где происходит обработка полученной информации, после чего выходной слой выдает результат. Например, в нашей задаче распознавания лопасти пеногона на вход будет подаваться изображение, а в качестве результата будет отображен объект со степенью уверенности, насколько объект похож на лопасть пеногона.

Помимо вышесказанного, нейронная сеть обладает множеством нейронов,

каждый из которых использует какую-либо математическую функцию, которая позволяет моделировать сложные зависимости, поэтому является необходимой для обработки данных и непосредственного принятия решений. Кроме того, нейроны обладают собственными весами, влияющими на соседние нейроны. Обучение нейронной сети заключается в поиске таких весов для нейронов, которые позволяют минимизировать ошибки между предсказанными значениями и реальными в обучающей выборке, представляющей из себя набор данных, на котором модель нейросети обучается. Помимо обучающей, существует тестовая выборка — набор данных, который не используется в процессе обучения и применяется для оценки, насколько хорошо модель умеет обобщать на новых данных [11, 12].

Таким образом, проблема может быть решена, если использовать предварительно обученную модель распознавания объектов. Такой подход позволит снизить вероятность ошибочных распознаваний и сократить количество пропущенных «проходов» по сравнению с предыдущими методами. Поэтому целью работы является улучшение существующих алгоритмов, реализованных в программном обеспечении за счет обучения и дальнейшего использования модели нейронной сети.

Методы

Разработанные ранее алгоритмы, как упоминалось выше, основывались на изменениях уровня освещенности обрабатываемой области. В нашем случае использование модели нейронной сети подразумевает распознавание объектов со всего кадра и, если их пространственное местоположение не попадает в выделенную область, тогда формировать сигнал синхронизации. Учитывая специфику нашей задачи, для обучения

будут использоваться сверточные нейронные сети. Их основная идея заключается в использовании локальных фильтров, которые проверяют малые регионы входного изображения. Это позволяет выявлять различные визуальные признаки (линии, текстуры и формы), что подходит для распознавания объектов с изображений [13 — 15].

Для обучения модели были рассмотрены алгоритмы обнаружения объектов R-CNN, Faster R-CNN и YOLO. R-CNN (Region-based Convolutional Network) представляет из себя алгоритм, в котором сначала происходит сканирование изображения для создания множества областей интереса ROI (Region of Interest), в которых потенциально могут находиться объекты. Эти области помогают определить все возможные места, где может находиться объект. Далее происходит обработка областей интереса для извлечения признаков, таких как края, текстуры и узоры, после чего выполняется классификация объектов в этих областях. Такой подход позволяет получить высокую точность в распознавании объектов, однако, несмотря на впечатляющие результаты, R-CNN обладает высокой вычислительной сложностью и, соответственно, медленной обработкой кадров.

В алгоритме Faster R-CNN используется более эффективная архитектура, в которой вводится RPN (Region Proposal Network), вместе с обучением нейронной сети автоматически генерирующей области интереса, что позволяет сократить время на обработку кадров [16, 17].

Однако Faster R-CNN по-прежнему является двухступенчатым алгоритмом, то есть сначала он использует RPN для генерации областей интереса, а уже затем области анализируются для классификации объектов и уточнения их границ.

Поскольку приложение обрабатывает видеопоток с камеры в режиме реального времени, для достижения высокой скорости наша модель должна использовать одноступенчатый алгоритм, то есть выполнять распознавание за 1 шаг. Кроме того, с учетом специфики расположения пеногона в кадре, задача состоит не в «классическом распознавании», при котором строятся рамки, в пределах которых находится предсказанный объект, а в сегментационном, что рассматривается как задача классификации изображения по пикселям [18].

Принцип работы YOLO заключается в следующем:

1. Подготовка данных. Входное изображение разбивается на сетку ячеек фиксированного размера;

2. Препроцессинг изображения. Сначала изображение масштабируется до заданного в параметрах обучения размера, затем пиксели изображения нормируются в диапазоне от 0 до 1;

3. Обнаружение объектов. Сверточные слои применяются для извлечения признаков изображения, которые затем подаются на полносвязный слой для классификации и определения границ объекта по пикселям с определенной вероятностью;

4. Пост-обработка. Фильтрация и устранение ложных срабатываний;

5. Отображение результатов. Получение координат объекта по пикселям.

Таким образом, YOLO основывается на том, что он разделяет изображение на сетку, каждая часть которой отвечает за обнаружение объектов в соответствующей области. Она делает несколько предсказаний для каждого участка и отфильтровывает менее уверенные результаты, оставляя только точные [19].

В отличие от предыдущих алгоритмов, упомянутых выше, YOLO рассматривает задачу обнаружения объектов как задачу регрессии, предсказывая координаты

и классы объектов одновременно; это упрощает модель и улучшает результаты. Кроме того, YOLO производит обработку за один проход, что значительно ускоряет процесс, соответственно, он способен выполнять обнаружение объектов в реальном времени, что требуется для нашего приложения. Также YOLO производит аугментацию данных, то есть генерацию новых изображений на основе имеющегося набора. Это не только экономит время, необходимое для ручного создания дополнительных примеров, но и значительно снижает риск переобучения модели; обеспечивая более разнообразное и репрезентативное представление данных, аугментация способствует улучшению обобщающей способности модели, позволяя ей успешно справляться с различными вариациями входных изображений в реальных сценариях (рис. 1).

Наиболее подходящими из рассмотренных алгоритмов для нашей задачи оказались алгоритмы Faster R-CNN и YOLO. В работе [20] были обучены модели, которые обрабатывают входящие кадры и распознают людей с использованием Faster R-CNN и YOLO. В результате точность с учетом некоторой погрешности оказалась равной у всех, однако модель, обученная на YOLO, оказалась быстрее. В качестве критерия быстроты модели было указано количество кадров в секунду, которое на первом наборе данных на модели с YOLO составило 56,5 кадров в секунду и на модели с Faster R-CNN — 6,56 кадров в секунду, а на втором — 100 и 6,1 кадров в секунду соответственно.

Учитывая все вышеупомянутые факторы, был выбран именно алгоритм YOLO. Стоит отметить, что конфигураций YOLO существует несколько — они различаются по размеру весов и скорости выполнения. График различных конфигураций, представленный разра-

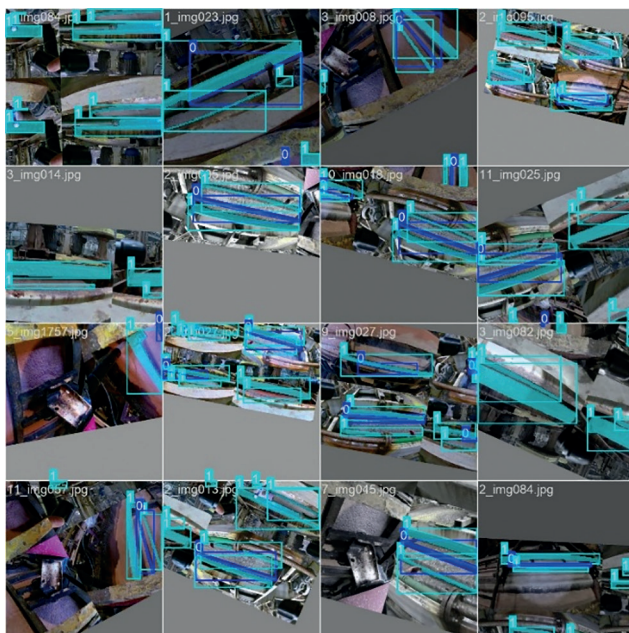


Рис. 1. Пример аугментации с помощью алгоритма YOLO

Fig. 1. Example of augmentation using the YOLO algorithm

ботчиками в репозитории GitHub, показан на рис. 2.

С его использованием мы выбирали конкретную версию алгоритма для использования.

По оси абсцисс отмечено время в миллисекундах, затраченное на обработку одного кадра. По оси ординат отмечена метрика mAP50-95, которая усредняет

средние значения точности mAP, рассчитанные при нескольких пороговых значениях IoU (Intersection over Union), от 0,5 до 0,95 с шагом 0,05. Такая метрика рассчитывается как отношение области пересечения к области объединения истинной и предсказанной областей объекта, она дает более полное представление о том, насколько точно мо-

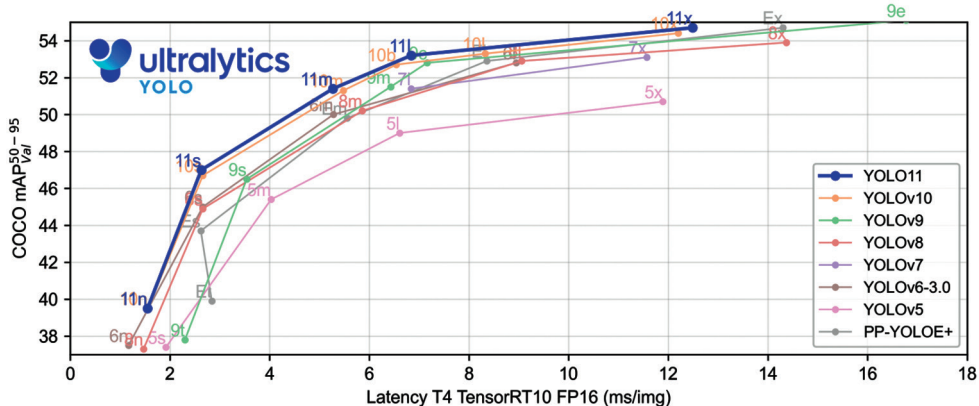


Рис. 2. Конфигурации YOLO

Fig. 2. YOLO configurations

дель может находить объекты на разных уровнях строгости.

Как видно из графика, конфигурация YOLO-11n является самой маловесной, при этом обладает самой низкой точностью. Конфигурации YOLO-11l и YOLO-11x, наоборот, тратят больше всех времени на обработку кадра — 6,84 и 12,49 соответственно. Так как для решения нашей задачи важна обработка кадров в режиме реального времени, была выбрана конфигурация YOLO-11s, которая точнее, чем YOLO-11n, на 41%, при этом быстрее, чем YOLO-11m, в 2 раза. Такая конфигурация имеет 355 слоев.

Обучение модели

Из-за того, что YOLO относится к методу обучения с учителем, требуется передавать модели информацию о местоположении полигонов. Поэтому необходимо выполнить аннотацию изображений.

В качестве обучающей выборки было использовано 410 изображений с флотомашинной, а также 88 изображений тестовой выборки, что составляет 20% от обучающей выборки и позволяет обеспечить достаточное количество изображений для обучения. Изображения являются кадрами видео с различными флотомашинами, разными положениями пеногона в кадре и разнообразной светимостью как внешней, так и от точечного источника освещения. Это делает наш датасет более разнообразным и повышает вероятность успешного обучения модели.

Существует множество инструментов для аннотации изображений. Было рассмотрено приложение LabelIMG, которое является бесплатным, а за счет готовых собранных файлов для Windows его установка производится быстро. Однако приложение не обладает достаточным для нашей задачи функционалом, так как оно поддерживает только огра-

ничивающие прямоугольники, тогда как для задачи сегментации требуется выделение объектов по полигонам.

Также было рассмотрено приложение VGG Image Annotator, которое имеет веб-версию, благодаря чему отдельная установка не требуется. Также оно поддерживает выделение объектов по полигонам, однако сохраняет получившиеся файлы в формате JSON. Так как YOLO обладает собственным форматом аннотаций, формат приложения VGG Image Annotator не подходит и нуждается в дополнительной конвертации форматов.

Поэтому, учитывая вышесказанное, необходимо выбрать такой инструмент для аннотации изображений, который поддерживал бы выделение объектов по полигонам, позволял сохранять получившиеся аннотации в формате YOLO и был бесплатным. Исходя из требований, был выбран инструмент CVAT (Computer Vision Annotation Tool), который имеет весь необходимый функционал. Кроме того, CVAT обладает встроенными возможностями искусственного интеллекта, которые позволяют автоматически выделять требуемые объекты, что значительно сокращает время на аннотацию изображений по сравнению с ручной аннотацией.

Использование инструмента возможно как через веб-версию, так и через локальное развертывание версии CVAT, доступной на GitHub. Веб-версия не требует установки, однако обладает некоторыми ограничениями для бесплатного использования, например, ограниченное использование автоматического выделения объектов с помощью искусственного интеллекта. Поэтому CVAT был развернут локально. Использование Docker позволило развернуть приложение в контейнере со всеми возможностями CVAT, которые недоступны в бесплатной веб-версии. Пример аннотации с

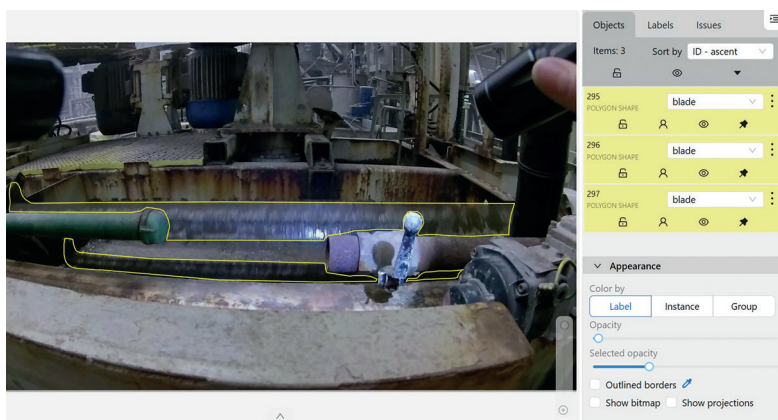


Рис. 3. Аннотация изображения с использованием инструмента CVAT
 Fig. 3. Image annotation using CVAT tool

помощью инструмента CVAT представлен на рис. 3.

Из-за того, что машинное обучение часто требует установки множества библиотек и пакетов, которые могут иметь конфликты версий или зависимости друг от друга, необходимо использовать виртуальные окружения, которые создают изолированную среду для каждого проекта, где устанавливаются только необходимые библиотеки и пакеты. Это предотвращает конфликты между различными проектами и гарантирует стабильное выполнение кода. В качестве виртуального окружения было рассмотрено окружение `venv`, которое является встроенным инструментом Python, поэтому его не требуется устанавливать отдельно, а также он прост в использовании. Однако `venv` только создает контейнер для установки пакетов, для управления пакетами используется встроенный в Python пакетный менеджер `pip`, который не всегда может автоматически разрешить конфликты версий между пакетами и зависимостями, так как `pip` изолирует только установленные пакеты, что может приводить к конфликтам с системными библиотеками.

Поэтому при обучении модели использовалось виртуальное окружение

`conda`, которое создает полностью изолированные окружения, включая как установленные пакеты, так и встроенные системные библиотеки Python. Помимо этого, `conda` использует собственный алгоритм для разрешения проблемы зависимостей между пакетами. Этот алгоритм находит такие версии библиотек, которые будут совместимы между собой, что позволяет избегать конфликтов; это является важным в задаче машинного обучения, так как перед обучением требуется установить пакет `Ultralytics`, содержащий множество «тяжелых» библиотек и зависимостей [21].

Для обучения были выставлены некоторые параметры аугментации, то есть генерации новых изображений, основой для которых являются изображения из датасета. К таким параметрам относятся:

- вертикальный и горизонтальный повороты с вероятностью 30%;
- максимальный угол поворота изображения на 30°;
- случайное уменьшение исходного изображения до 50% от своего первоначального размера;
- максимальное смещение по горизонтали и вертикали на 15%;
- возможное обрезание изображения до 50%.

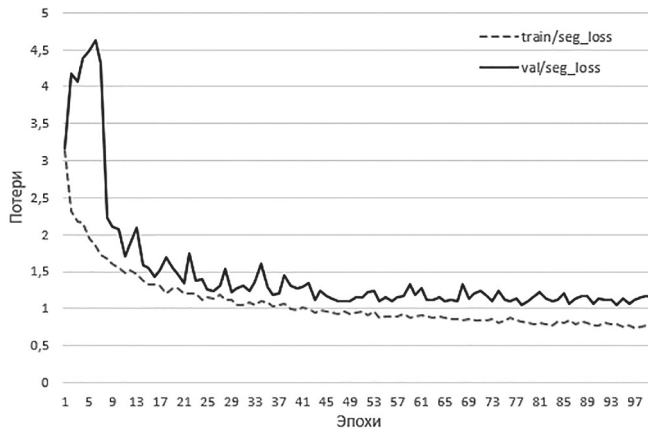


Рис. 4. График потерь при обучении
 Fig. 4. Graph of learning loses

Помимо параметров аугментации, были выставлены параметры для непосредственного обучения:

- 100 эпох;
- размер батча, то есть количество изображений, которые будут обрабатываться за один раз во время обучения, равный 16;
- добавление штрафа к функции потерь для предотвращения переобучения в размере 0,0005;
- вероятность выброса нейронов во время обучения для улучшения обобщения модели, равная 10%;

- количество эпох, через которое обучение прекратится, если не будет наблюдаться улучшения, равное 15.

О качестве обученной модели можно судить по таким метрикам, как потери на обучающей и тестовой выборках, а также mAP50, который рассчитывается на основе оценки, когда допустимый порог IoU (Intersection over Union) равен 0,5. Это означает, что, если предсказанное ограничивающее окно объекта совпадает с истинным ограничивающим окном хотя бы на 50%, это считается успешным обнаружением.

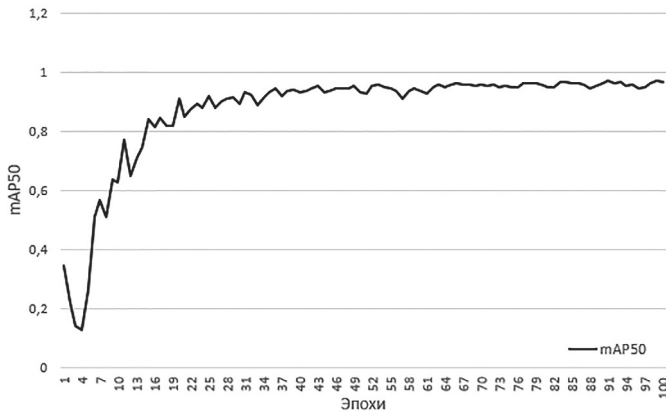


Рис. 5. График mAP50
 Fig. 5. Graph mAP50

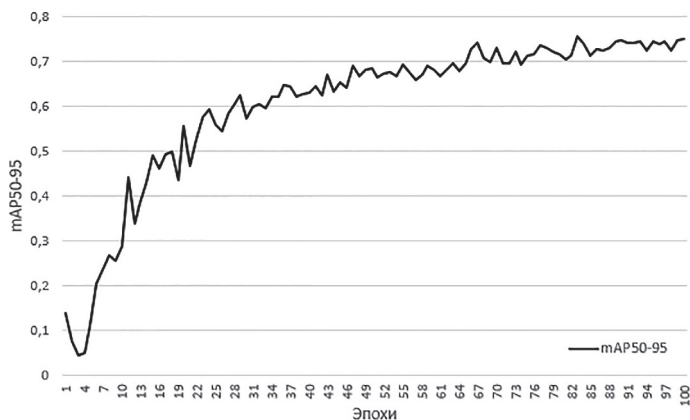


Рис. 6. График mAP50-95
Fig. 6. Graph mAP50-95

В результате обучения модели нейронной сети получились зависимости эпохи от метрики обучения, представленные на рис. 4 и 5.

Также о качестве обучения можно судить по метрике mAP50-95, которая уже описывалась ранее при выборе конфигурации модели YOLO. График зависимости эпохи от метрики mAP50-95 представлен на рис. 6.

Исходя из результатов, одинаковые тренды падения потерь говорят о том, что модель не переобучилась и хорошо обобщает данные. Высокие показатели mAP50 и mAP50-95 означают, что модель в большинстве случаев правильно распознает лопасти пеногона.

Таким образом, полученные результаты свидетельствуют о том, что модель успешно обучена и ее можно интегрировать в разработанное ранее ПО.

Использование модели в ПО

Теперь добавим исключение пеногона непосредственно в реализованное ранее ПО. Как упоминалось ранее, будем считать, что пеногон имеется в обрабатываемой области тогда, когда его маска совпадает с маской этой области. Для нашей задачи в первую очередь необходимо идентифицировать наличие

пеногона не в кадре с высокой уверенностью, а хотя бы с уверенностью 0,5, с чем нейросеть справляется хорошо. Поэтому была разработана функция, которая ищет объект в кадре. Если объект найден и пересекается с выделенной областью, то делается вывод о наличии пеногона в обрабатываемой области, в противном случае — о его отсутствии. Далее, в зависимости от наличия пеногона, возвращается истина или ложь и, при наличии пеногона в кадре, на изображении рисуются его контуры.

Разработанная функция будет вызываться в том случае, если по сравнению с предыдущим обработанным кадром прошло больше времени, чем заданный минимальный интервал между кадрами в секундах. Кроме того, если при обработке кадра был обнаружен пеногон, вызов функции может быть прекращен на заданное количество кадров, по истечении которых снова произойдет вызов функции на проверку наличия пеногона в кадре.

Поскольку распознавание объектов — ресурсоемкая для компьютера задача, такой подход позволяет уменьшить количество ненужных распознаваний, что важно для обработки видеопотока в режиме реального времени.

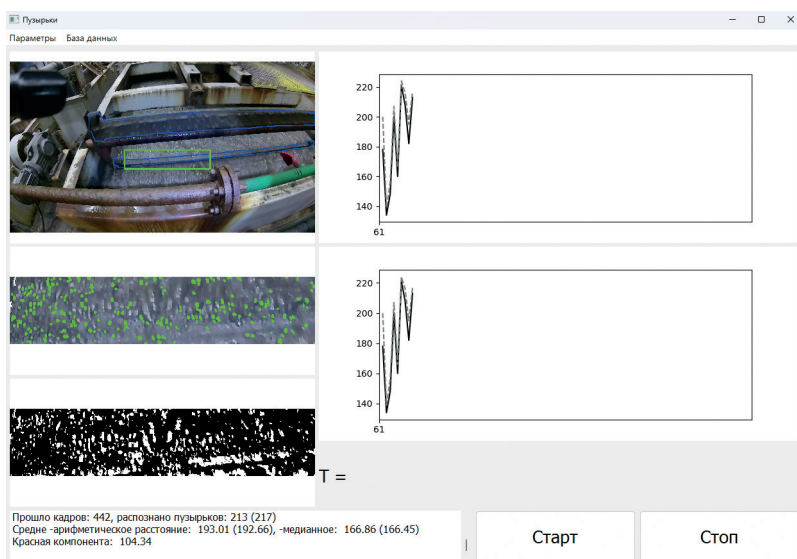


Рис. 7. Пример работы программы с распознаванием параметров пены с новым алгоритмом синхронизации с пеногоном

Fig. 7. Example of program execution with recognition of foam parameters using a new synchronization algorithm with foam blades

Пример работы программы представлен на рис. 7.

Результаты

Для проверки работоспособности и адекватности разработанного алгоритма было проведено сравнение с алгоритмом, разработанным ранее и описанным в работе [10]. Всего было проведено 8 испытаний, критерием сравнения

выступали количество обработанных кадров с использованием алгоритма из работы [10] и нового алгоритма, который использует модель нейронной сети, а также разница между ними и общее количество проходов на видео.

Результаты сравнительного анализа представлены в таблице.

Как видно из таблицы, использование модели распознавания объектов по-

№ съемки	Положение пеногона	Обработано кадров старым алгоритмом	Обработано кадров новым алгоритмом	Всего проходов	Разница между старым и новым алгоритмами, %
1	горизонтальное	46	46	46	0
2	горизонтальное	38	38	38	0
3	вертикальное	208	221	221	5,9
4	вертикальное	92	98	98	6,1
5	горизонтальное	33	35	35	5,7
6	горизонтальное	56	61	61	8,2
7	горизонтальное	62	67	67	7,5
8	горизонтальное	41	41	41	0
Всего		576	607	607	5,1

зволило сократить пропущенные проходы пеногона на 5,1%

Заключение

Таким образом, была обучена модель распознавания лопастей пеногона, у которой показатели mAP50 и mAP50-95 составили 0,96 и 0,75 соответственно, что, учитывая специфику задачи, может свидетельствовать об успешном распознавании лопастей пеногона. Помимо вышесказанного, последовательное уменьшение потерь в результате обучения,

по сравнению с первой эпохой, на 24% на обучающей выборке и на 36,5% на тестовой выборке демонстрирует отсутствие переобучения. То есть модель способна обобщать и хорошо справляется с новыми наборами данных.

Эксперимент, проведенный в ходе модификации программного обеспечения, показал, что использование нейросети позволяет уменьшить количество пропущенных проходов на 5,1% по сравнению с предыдущим алгоритмом. Гипотеза исследования подтверждена.

СПИСОК ЛИТЕРАТУРЫ

1. Дюжева Н. В., Тинькова А. А. Анализ конъюнктуры мирового рынка минеральных удобрений // Вестник Астраханского государственного технического университета. Серия: Экономика. — 2020. — № 1. — С. 91 — 100. DOI: 10.24143/2073-5537-2020-1-91-100.
2. Wakeel A., Ishaq M. Potash use and dynamics in agriculture. Springer, 2022, 144 p.
3. Белкин П. А., Катаев В. Н. Закономерности техногенной трансформации химического состава подземных вод в районах разработки калийных месторождений // Известия Уральского государственного горного университета. — 2018. — № 2 (50). — С. 55 — 64. DOI: 10.21440/2307-2091-2018-2-55-64.
4. Земсков А. Н., Лискова М. Ю., Заалишвили В. Б., Шамрин М. Ю. Современные технологические и технические решения при ведении горных работ на калийных рудниках // Известия Тульского государственного университета. Науки о Земле. — 2022. — № 2. — С. 284 — 296. DOI: 10.46689/2218-5194-2022-2-1-284-296.
5. Юркевич Н. В., Грошева Т. В., Еделев А. В., Гуреев В. Н., Мазов Н. А. Современные походы к обогащению баритовых руд // Записки Горного института. — 2024. — Т. 270. — С. 977 — 993.
6. Силаков А. В., Варламова С. А., Котков П. В. Программное распознавание дефектов изображений регулярных текстур в текстильной промышленности // Известия высших учебных заведений. Технология текстильной промышленности. — 2022. — № 2 (398). — С. 266 — 272. DOI: 10.47367/0021-3497_2022_2_266.
7. Затонский А. В., Варламова С. А. Использование бликовых отражений для автоматического распознавания параметров пены при флотации калийных руд // Обогащение руд. — 2016. — № 2. — С. 49 — 56. DOI: 10.17580/or.2016.02.09.
8. Варламова С. А., Затонский А. В., Федосеева К. А. Исследование чувствительности к освещению метода бликового распознавания пен калийных флотационных машин // Обогащение руд. — 2021. — № 6. — С. 29 — 33. DOI: 10.17580/or.2021.06.05.
9. Мальшева А. В. Модели и алгоритмы поддержки принятия решений при управлении процессом флотации калийной руды: Автореф. дисс. ... канд. техн. наук. — Пермь: ПНИПУ, 2020. — 19 с.
10. Федосеева К. А. Совершенствование методов идентификации состояния пенного слоя калийных флотомашин: Автореф. дисс. ... канд. техн. наук. — Пермь: ПНИПУ, 2023. — 19 с.
11. Chen L., Li S., Bai Q., Yang J., Jiang S., Miao Y. Review of image classification algorithms based on convolutional neural networks // Remote Sensing. 2021, vol. 13, no. 22, article 4712. DOI: 10.3390/rs13224712.
12. Zhao X., Wang L., Zhang Y., Han X., Deveci M., Parmar M. A review of convolutional neural networks in computer vision // Artificial Intelligence Review. 2024, vol. 57, article 99. DOI: 10.1007/s10462-024-10721-6.
13. Liu L., Ouyang W., Wang X., Fieguth P., Chen J., Liu X., Pietikainen M. Deep learning for generic object detection: A survey // International Journal of Computer Vision. 2020, vol. 128, pp. 261 — 318. DOI: 10.1007/s11263-019-01247-4.

14. Kaur R., Singh S. A comprehensive review of object detection with deep learning // *Digital Signal Processing*. 2023, vol. 132, article 103812. DOI: 10.1016/j.dsp.2022.103812.
15. Андриянов Н. А., Дементьев В. Е., Ташлинский А. Г. Обнаружение объектов на изображениях: от критериев Байеса и Неймана-Пирсона к детекторам на базе нейронных сетей // *Компьютерная оптика*. — 2022. — Т. 46. — № 1. — С. 139–159. DOI: 10.18287/2412-6179-CO-922.
16. Ren J., Wang Y. Overview of object detection algorithms using convolutional neural networks // *Journal of Computer and Communications*. 2022, vol. 10, no. 1, pp. 115–132. DOI: 10.4236/jcc.2022.101006.
17. Двойникова А. А., Кагиров И. А., Карпов А. А. Аналитический обзор методов автоматического распознавания вовлеченности пользователя в виртуальную коммуникацию // *Информационно-управляющие системы*. — 2022. — № 5. — С. 12–22. DOI: 10.31799/1684-8853-2022-5-12-22.
18. Diwan T., Anirudh G., Tembhrune J. V. Object detection using YOLO: challenges, architectural successors, datasets and application // *Multimedia Tools and Applications*. 2023, vol. 82, pp. 9243–9275. DOI: 10.1007/s11042-022-13644-y.
19. Cong X., Li S., Chen F., Liu C., Meng Y. A review of YOLO object detection algorithms based on deep learning // *Frontiers in Computing and Intelligent Systems*. 2023, vol. 4, no. 2, pp. 17–20. DOI: 10.54097/fcis.v4i2.9730.
20. Тимошкин М. С., Миронов А. Н., Леонтьев А. С. Сравнение YOLO v5 и Faster R-CNN для обнаружения людей на изображении в потоковом режиме // *Международный научно-исследовательский журнал*. — 2022. — № 6 (120). — С. 137–146. DOI: 10.23670/IRJ.2022.120.6.020.
21. Vaughan L. Python tools for scientists: An introduction to using Anaconda, JupyterLab, and Python's scientific libraries. No Starch Press, 2023, 744 p. **PLAB**

REFERENCES

1. Dioujeva N. V., Tinkova A. A. Global mineral fertilizer market analysis. *Vestnik of Astrakhan State Technical University. Series: Economics*. 2020, no. 1, pp. 91–100. [In Russ]. DOI: 10.24143/2073-5537-2020-1-91-100.
2. Wakeel A., Ishfaq M. *Potash use and dynamics in agriculture*. Springer, 2022, 144 p.
3. Belkin P. A., Kataev V. N. Regularities of the chemical composition of technogenic transformation which groundwater undergoes in the areas of the potash deposits development. *News of the Ural State Mining University*. 2018, no. 2 (50), pp. 55–64. [In Russ]. DOI: 10.21440/2307-2091-2018-2-55-64.
4. Zemskov A. N., Liskova M. Yu., Zaalishvili V. B., Shamrin M. Yu. Modern technological and technical solutions for mining operations potash mines. *News of the Tula state university. Sciences of Earth*. 2022, no. 2, pp. 284–296. [In Russ]. DOI: 10.46689/2218-5194-2022-2-1-284-296.
5. Yurkevich N. V., Grosheva T. V., Edelev A. V., Gureev V. N., Mazov N. A. Modern approaches to the enrichment of barite ores. *Journal of Mining Institute*. 2024, vol. 270, pp. 977–993. [In Russ].
6. Silakov A. V., Varlamova S. A., Kotkov P. V. Software recognition of image defects of regular textures in the textile industry. *Proceedings of higher education institutions. Textile industry technology*. 2022, no. 2 (398), pp. 266–272. [In Russ]. DOI: 10.47367/0021-3497_2022_2_266.
7. Zatonkiy A. V., Varlamova S. A. Use of reflection flare spots for automatic recognition of froth parameters in potassium ores flotation. *Obogashchenie Rud*. 2016, no. 2, pp. 49–56. [In Russ]. DOI: 10.17580/or.2016.02.09.
8. Varlamova S. A., Zatonkiy A. V., Fedoseeva K. A. Illumination sensitivity analysis for the speck-based froth detection method using potash flotation machines. *Obogashchenie Rud*. 2021, no. 6, pp. 29–33. [In Russ]. DOI: 10.17580/or.2021.06.05.
9. Malysheva A. V. *Modeli i algoritmy podderzhki prinyatiya resheniy pri upravlenii protsessom flotatsii kaliynoy rudy* [Models and algorithms for decision support in managing the flotation process of potash ore], Candidate's thesis, Perm, 2020, 19 p.
10. Fedoseeva K. A. *Sovershenstvovanie metodov identifikatsii sostoyaniya pennogo sloya kaliynykh flotomashin* [Improvement of methods for identifying the state of the foam layer in potash flotation machines], Candidate's thesis, Perm, 2023, 19 p.
11. Chen L., Li S., Bai Q., Yang J., Jiang S., Miao Y. Review of image classification algorithms based on convolutional neural networks. *Remote Sensing*. 2021, vol. 13, no. 22, article 4712. DOI: 10.3390/rs13224712.

12. Zhao X., Wang L., Zhang Y., Han X., Deveci M., Parmar M. A review of convolutional neural networks in computer vision. *Artificial Intelligence Review*. 2024, vol. 57, article 99. DOI: 10.1007/s10462-024-10721-6.

13. Liu L., Ouyang W., Wang X., Fieguth P., Chen J., Liu X., Pietikainen M. Deep learning for generic object detection: A survey. *International Journal of Computer Vision*. 2020, vol. 128, pp. 261 – 318. DOI: 10.1007/s11263-019-01247-4.

14. Kaur R., Singh S. A comprehensive review of object detection with deep learning. *Digital Signal Processing*. 2023, vol. 132, article 103812. DOI: 10.1016/j.dsp.2022.103812.

15. Andriyanov N. A., Dementiev V. E., Tashlinskii A. G. Detection of objects in the images from likelihood relationships towards scalable and efficient neural networks. *Computer Optics*. 2022, vol. 46, no. 1, pp. 139 – 159. [In Russ]. DOI: 10.18287/2412-6179-CO-922.

16. Ren J., Wang Y. Overview of object detection algorithms using convolutional neural networks. *Journal of Computer and Communications*. 2022, vol. 10, no. 1, pp. 115 – 132. DOI: 10.4236/jcc.2022.101006.

17. Dvoynikova A. A., Kagirov I. A., Karpov A. A. Analytical review of methods for automatic detection of user engagement in virtual communication. *Information and Control Systems*. 2022, no. 5, pp. 12 – 22. [In Russ]. DOI: 10.31799/1684-8853-2022-5-12-22.

18. Diwan T., Anirudh G., Tembhurne J. V. Object detection using YOLO: challenges, architectural successors, datasets and application. *Multimedia Tools and Applications*. 2023, vol. 82, pp. 9243 – 9275. DOI: 10.1007/s11042-022-13644-y.

19. Cong X., Li S., Chen F., Liu C., Meng Y. A review of YOLO object detection algorithms based on deep learning. *Frontiers in Computing and Intelligent Systems*. 2023, vol. 4, no. 2, pp. 17 – 20. DOI: 10.54097/fcis.v4i2.9730.

20. Timoshkin M. S., Mironov A. N., Leont'yev A. S. Comparison of YOLO v5 and Faster R-CNN for detecting people in the image in streaming mode. *International Research Journal*. 2022, no. 6 (120), pp. 137 – 146. [In Russ]. DOI: 10.23670/IRJ.2022.120.6.020.

21. Vaughan L. *Python tools for scientists: An introduction to using Anaconda, JupyterLab, and Python's scientific libraries*. No Starch Press, 2023, 744 p.

ИНФОРМАЦИЯ ОБ АВТОРАХ

Затонский Андрей Владимирович¹ — д-р техн. наук, профессор, зав. кафедрой, e-mail: zxenon@narod.ru,

Саломатова Кристина Александровна¹ — старший преподаватель, e-mail: kristya_0103@mail.ru,

Кузнецов Станислав Викторович¹ — студент, e-mail: kuznetsovstas04@yandex.ru,

¹ Пермский национальный исследовательский политехнический университет, Березниковский филиал.

Для контактов: Затонский А.В., e-mail: zxenon@narod.ru.

INFORMATION ABOUT THE AUTHORS

A.V. Zatonliy¹, Dr. Sci. (Eng.), Professor, Head of Chair, e-mail: zxenon@narod.ru,

K.A. Salomatova¹, Senior Lecturer, e-mail: kristya_0103@mail.ru,

S.V. Kuznetsov¹, Student, e-mail: kuznetsovstas04@yandex.ru,

¹ Perm National Research Polytechnic University, Berezniki Branch, Berezniki, Russia.

Corresponding author: A.V. Zatonliy, e-mail: zxenon@narod.ru.

Получена редакцией 25.02.2025; получена после рецензии 06.03.2025; принята к печати 10.05.2025.

Received by the editors 25.02.2025; received after the review 06.03.2025; accepted for printing 10.05.2025.