

В качестве основных технологий для построения веб-решения будут использоваться фреймворки *angular* (как удобное по структуре решение для построения *UI*) и *firebase platform* (как идеальный инструмент связи веб-решения на *node.js* и облачной платформы *google cloud*).

В качестве основных технологий для построения *back-end* составляющей системы будут использоваться инструменты *google cloud* платформы:

- *Cloud-functions* – для реализации нано сервисов, соответствующих *serverless* модели;

- *Firebase-ml* – для реализации технологий нейронных сетей, вспомогательного инструмента для таргетной рекламы, рекомендаций и умного добавления медикаментов;

- *Pub-sub* – для реализации *message query* между наносервисами и компонентами системы;

- *Firebase/firestore – nosql* бд для хранения данных;

- *Security* – для защиты пользовательских данных;

- *Pay-api* – для обработки и осуществления транзакций используя платформу *google*;

- *Cloud-storage* – для хранения открытых данных пользователей;

- *Firebase-analytics* – для составления аналитики по всей системе.

С помощью *API*-маркировки изображений *ML Kit Firebase* мы можем распознавать сущности в изображении без необходимости предоставлять какие-либо дополнительные контекстные метаданные, используя *API* на устройстве или облачный *API*. Данные технологии используются для распознавания упаковки препарата без необходимости заполнять форму для добавления его в инвентарь.

Маркировка изображений дает представление о содержании изображений. При использовании *API* мы получаем список распознанных сущностей: люди, вещи, места, действия и т. д. (в частности, упаковки различных фармацевтических препаратов). Каждая найденная метка поставляется с оценкой, которая указывает на уверенность модели *ML* в своей актуальности. С помощью этой информации можно выполнять такие задачи, как автоматическое создание метаданных и модерация контента.

УДК 004.891:004.93

ОБНАРУЖЕНИЕ ПЛОДОВ ОГУРЦА ДЛЯ АВТОМАТИЗАЦИИ ПРОЦЕССА СБОРА МНОГОФУНКЦИОНАЛЬНЫМ АГРАРНЫМ РОБОТОМ НА ОСНОВЕ СВЕРТОЧНЫХ НЕЙРОННЫХ СЕТЕЙ

К. С. Курочка, В. С. Ермашкевич, Д. А. Дорошко, Н. И. Расшивалов

Учреждение образования «Гомельский государственный технический университет имени П. О. Сухого», Республика Беларусь

Описан процесс реализации алгоритма по обнаружению плодов огурца на основе сверточных нейронных сетей для их сбора с помощью многофункционального аграрного робота.

Ключевые слова: сверточная нейронная сеть, обнаружение, робот, огурцы.

DETECTION OF CUCUMBER FRUITS FOR AUTOMATION OF THE COLLECTING PROCESS BY A MULTIFUNCTIONAL AGRICULTURAL ROBOT BASED ON CONVOLUTIONAL NEURAL NETWORKS

K. S. Kurochka, V. S. Ermashkevich, D. A. Doroshko, N. I. Rasshivalov
Sukhoi State Technical University of Gomel, the Republic of Belarus

The process of implementing an algorithm for detecting cucumber fruits based on convolutional neural networks for their collection using a multifunctional agricultural robot is described.

Keywords: Convoluted neural network, detecting, robot, cucumbers.

Сбор плодовых культур является одной из наименее автоматизированных на сегодняшний день отраслей [1]. Использование роботов для сбора плодов может позволить увеличить и площади сельскохозяйственных угодий, и эффективность садоводческих хозяйств за счет повышения производительности труда при сборе урожая и повышения качества данного сбора. Исходя из этого, рассматривается проектирование робота для автоматизации сбора плодов. В качестве предметной области выбраны огурцы. Одной из задач данной автоматизации является задача обнаружения плодов.

Предлагается решение, позволяющее при помощи сверточной нейронной сети *YOLOv5* осуществлять локализацию плодов огурца, готовых к сбору (рис. 1).



Рис. 1. Пример локализации огурца и вывода соответствующей вероятности

Подготовлен и размечен набор изображений огурцов (600 изображений), включающий и другие овощи и фрукты.

Набор данных представлен двумя классами:

- *cucumber* – представляет огурцы;
- *trash* – представляет любой другой овощ либо фрукт, чтобы позволить сети узнавать объекты, отличные от огурца и, следовательно, уменьшить вероятность того, что неизвестный объект будет принят за огурец.

YOLOv5 имеет множество разновидностей предварительно обученных моделей. Разница между ними заключается в компромиссе между размером модели и временем распознавания. Облегченная версия модели *YOLOv5s* имеет размер всего 14 МБ, но не очень точна. С другой стороны, есть *YOLOv5x*, размер которого составляет 168 МБ, но данный вариант является наиболее точной версией из всех разно-

видностей. Для обучения выбраны разновидности *nano* и *small* как наиболее производительные и достаточные для решения поставленной задачи. Разновидности *YOLOv5* представлены на рис. 2.

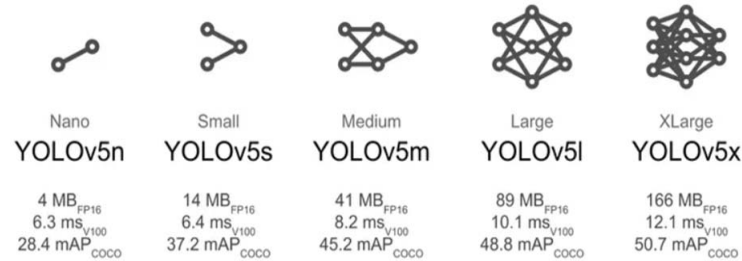


Рис. 2. Разновидности *YOLOv5*

Принцип работы *YOLOv5* заключается в следующем: исходное изображение сжимается таким образом, чтобы получить квадратную матрицу размером 13 на 13, в каждой клетке которой записана информация о наличии объекта и классе этого объекта на соответствующей части картинки. Таким образом, *YOLO* просматривает картинку один раз, что существенно увеличивает скорость обработки [2, с. 211].

Архитектура *YOLOv5* представлена на рис. 3.

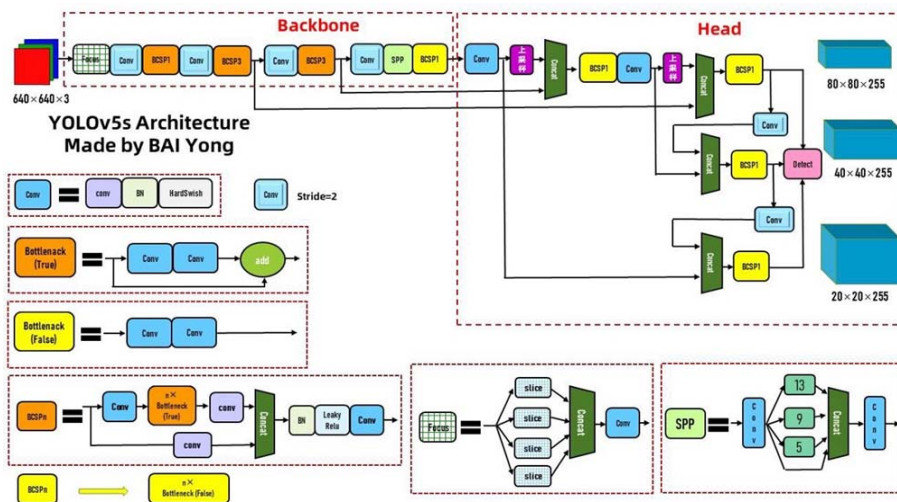


Рис. 3. Архитектура *YOLOv5*

Был проведен ряд обучений, выявлены оптимальные значения таких гиперпараметров обучения, как скорость обучения, *shear*, *scale*, *momentum* и др. Обучение состояло из 200 эпох с 11 шагами в рамках эпохи. Набор данных был разделен в пропорциях 80 к 20 %. В результате обучения модели *YOLOv5* получена точность в 95 % на валидационной выборке, а значение ошибки поиска – 5 % при использовании разновидности сети *small*. При обучении разновидности сети *nano* получена точность в пределах 88 %, что значительно хуже в сравнении с разновидностью сети *small*. В качестве функции потерь использовалась бинарная кросс-энтропия. Графики точности и потерь в зависимости от эпохи для *YOLOv5s* показаны на рис. 4.

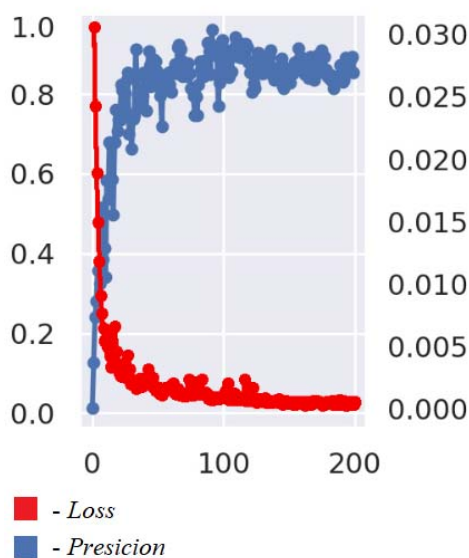


Рис. 4. Графики точности и потерь в зависимости от эпохи

Исходя из полученных графиков, можно сделать вывод, что для обучения модели на данном наборе данных достаточным количеством эпох является 100.

Для тестирования *YOLOv5* выбран одноплатный компьютер *Orange Pi*. Результат обработки одного кадра занял 0,8 с, что является достаточно хорошим результатом для данного одноплатного компьютера; в дальнейшем предполагается проведение вычислений на удаленном сервере для достижения необходимой производительности.

Литература

1. Сбор огурцов. – Режим доступа: <https://robroy.ru/robot-dlya-sbora-ogkrczov.html>. – Дата доступа: 18.09.2022.
2. Курочка, К. С. Нейросетевая обработка данных : учеб.-метод. пособие / К. С. Курочка, К. А. Панарин. – Гомель : ГГТУ им. П. О. Сухого, 2021. – 260 с.

УДК 62-83-52

ИССЛЕДОВАНИЕ ЭЛЕКТРОМАГНИТНОГО МОМЕНТА АСИНХРОННО-ВЕНТИЛЬНОГО КАСКАДА ПРИ РАЗЛИЧНЫХ ТИПАХ ПРЕОБРАЗОВАТЕЛЕЙ В РОТОРНОЙ ЦЕПИ

И. В. Дорощенко, М. Н. Погуляев

Учреждение образования «Гомельский государственный технический университет имени П. О. Сухого», Республика Беларусь

С использованием функциональных схем и имитационных моделей асинхронно-вентильного каскада произведен расчет и анализ действующего значения электромагнитного момента электроприводов на основе асинхронно-вентильного каскада с транзисторным и тиристорным преобразователями в роторной цепи.

Ключевые слова: асинхронно-вентильный каскад, асинхронный двигатель с фазным ротором, имитационная модель, компьютерное моделирование, моделирование электропривода.