УДК 631.5

**Определение вида заболеваний поражающих листья картофеля с использованием алгоритмов машинного обучения**

***Александр Григорьевич Возмилов1🖂, Андрей Анатольевич Лисов 2, Виль Губаевич Урманов 3,***

***Галина Николаевна Синева 4***

*1,2 Южно-Уральский государственный университет (НИУ), Челябинск, Россия*

*3* *Башкирский государственный аграрный университет, Уфа, Россия*

*4* *Волгоградский государственный аграрный университет, Волгоград, Россия*

1. *vozmiag44@rambler.ru🖂, https://orcid.org/0000-0002-1292-3975*
2. *lisov.andrey2013@yandex.ru, https://orcid.org/* *0000-0001-7282-8470*
3. *uvg55@mail.ru*
4. *galina.sineva1954@yandex.ru*

**Аннотация**

**Введение.** Картофель является одним из основных продуктов питания, занимающий 4-е место, среди продуктов потребляемых во всем мире. Он богат питательными веществами, в первую очередь витаминами С и В6, а также минералами, калием, магнием и железом Болезни картофеля являются основной причиной снижения его качества и урожайности.

**Материалы и методы.** Для успешной борьбы с болезнями картофеля важное значение имеет ранее определение вида заболевания по внешнему виду листьев. Предлагаемый данной статье метод определения больных листьев картофеля на основе их внешнего состояния с применением алгоритмов машинного обучения, в частности свёрточных нейронных сетей, позволяет автоматизировать данный процесс. Веса и смещения в модели данной нейронной сети оптимизируются путем обучения на данных из датасета «Potato Disease Leaf Dataset (PLD)» и с помощью алгоритма обратного распространения ошибки.

**Результаты и их обсуждения.** Проведённые эксперименты показали, что предлагаемый метод и архитектура нейросети позволяет достичь средней эффективности определения типа заболевания 98,23% и скоростью 0,105 секунды. Если убрать компонент визуализации результата предсказания, то это позволит добиться скорости распознавания – 0,00079±5% секунды для всех изображений из тестовой выборки.

**Заключение.** Предлагаемый вариант решения проблемы снижения качества урожая из-за возникновения болезни листьев картофеля позволяет предотвратить данный процесс на раннем этапе. Код разрабатывался на основе существующего фреймворка Google.Colab и облачного хранилища, что позволяет другим исследователем ознакомиться с данным алгоритмом и улучшить его работу при необходимости.

***Ключевые слова*** – болезни листьев картофеля, машинное обучение, сверточные нейронные сети.

**Determination of the type of potato leaves diseases with using machine learning**

***Aleksander G. Vozmilov1🖂, Andrey A. Lisov 2, Vil G. Urmanov 3, Galina N. Sineva 4***

*1,2 South Ural State University (NRU), Chelyabinsk, Russia*

*3* *Bashkir State Agrarian University, Ufa, Russia*

*4* *Volgograd State Agrarian University, Volgograd, Russia*

1. *vozmiag44@rambler.ru🖂, https://orcid.org/0000-0002-1292-3975*
2. *lisov.andrey2013@yandex.ru, https://orcid.org/* *0000-0001-7282-8470*
3. *uvg55@mail.ru*
4. *galina.sineva1954@yandex.ru*

***Abstract***

**Introduction.** The potato is one of the staple foods, ranking 4th among the foods consumed worldwide. It is rich in nutrients, primarily vitamins C and B6, as well as minerals, potassium, magnesium and iron Potato diseases are the main cause of reduced quality and yield.

**Materials and methods.** For the successful fight against potato diseases, early identification of the type of disease by the appearance of the leaves is important. The method proposed in this article for determining diseased potato leaves based on their external state using machine learning algorithms, in particular convolutional neural networks, allows you to automate this process. The weights and biases in the model of this neural network are optimized by training on data from the Potato Disease Leaf Dataset (PLD) and using the backpropagation algorithm

**Results and discussion.** The conducted experiments showed that the proposed method and architecture of the neural network makes it possible to achieve an average efficiency of determining the type of disease of 98.23% and a speed of 0.105 seconds. If the visualization component of the prediction result is removed, this will allow achieving a recognition speed of 0.00079 ± 5% of a second for all images from the test sample.

**Conclusion.** The proposed solution to the problem of reducing the quality of the crop due to the occurrence of potato leaf disease makes it possible to prevent this process at an early stage. The code was developed on the basis of the existing Google.Colab framework and cloud storage, which allows other researchers to get acquainted with this algorithm and improve its performance if necessary.

***Keywords*** – potato leaf diseases, machine learning, convolutional neural networks.

**Введение**

Продовольственная безопасность и улучшение питания населения являются одной из важных задач любого государства. Благодаря тому, что картофель является богатым питательными веществами, витаминами С и В6, а также минералами, калием, магнием и железом [1] он является одним из основных продуктов питания, занимающий 4-е место, среди продуктов потребляемых во всем мире.

Известно, что ряд заболеваний картофеля можно определить по внешнему виду листьев. В связи с этим имеется возможность определения заболевания картофеля на основе алгоритмов машинного обучения, в частности, свёрточных нейронных сетей, позволяющих автоматизировать процесс распознавания видов заболевания картофеля. Известно, что картофель поражается многими болезнями в период как до, так и после уборочной стадии [2]. Наличие болезней картофеля в период выращивания значительно снижает его качество и урожайность.

Одной из проблем потери урожая связано с поздним выявлением болезней картофеля, а также ошибками в их диагностике. В настоящее время определение заболеваний картофеля осуществляется фермерами вручную, что связано с большими трудозатратами и потерями времени.

В результате осуществляемая профилактика заболеваний, как правило, не эффективна и за частую связана с определением неправильного вида болезни картофеля, что приводит к снижению урожайности [3, 4]. Предлагаемый метод определения заболеваний картофеля по внешнему виду листьев, на основе алгоритмов машинного обучения, показал свою хорошую эффективность в ряде исследований [5-9].

Традиционно, проверки на основе машинного обучения, включают обычно три этапа: предварительная обработка изображений, извлечение признаков и классификация. В качестве методов классификации часто использовались машина опорных векторов (SVM) [10], метод k-ближайших соседей (k-NN) [11], линейная регрессия [12]. Общие проблемы этих методов включают их неспособность обрабатывать сложные фоновые изображения или обнаруживать несколько объектов на изображении [13]. Для решения этой задачи подходит алгоритм на основе сверточных нейронных сетей (CNN) [4,5].

В последние годы качество классификации изображений быстро улучшалось за счет увеличения скорости вычислений и продвижения различных алгоритмов, таких как ImageNet, Large-Scale, Visual Recognition Challenge (ILSVRC) [14], VGG [15], GoogLeNet [16] и ResNet [17]. Такие методы глубокого обучения, как YOLO [18–20], RCNN [21–23], SSD [24, 25] и RFCN, хорошо зарекомендовали себя в задачах обнаружения объектов, включая Pascal VOC. Архитектура CNN обычно состоит из двух основных этапов: выделение признаков и классификация. Представления извлеченных признаков передаются компоненту классификации архитектуры, после чего модель делает вывод о вероятности принадлежности дефекта к определенному классу. Веса и смещения в модели оптимизируются путем обучения нейронной сети алгоритмом обратного распространения ошибки.

**Материалы и методы**

В данной работе проводились исследования в области машинного обучения для решения задачи обнаружения больных листьев картофеля на изображении. Основой разработанной программы для решения поставленной задачи является свёрточная нейронная сеть, которая работает с открытым датасетом «Potato Disease Leaf Dataset (PLD)», который можно найти на «Kaggle». Датасет содержит 3 класса изображений: здоровые листья (Healthy), ранняя болезнь листьев (Early\_Blight) и поздняя (Late\_Blight) и соответственно 809, 152 и 1000 изображений для каждого класса. Изображения (рис. 1) листьев представлены в формате цветового пространства RGB, в jpeg формате размером 256х256 пикселей. Исходная выборка изображений была разделена на 2 класса «test» и «train», разделение выборки произведено в соотношении 1:9.



Рис. 1. Исходные изображения из датасета для распознавания болезней листьев картофеля.

Fig. 1. Initial images from the dataset for the recognizing of potato leaf diseases.

Источник: разработано авторами в ходе исследования

Код был разработан во фреймворке Google.Collab, на языке программирования «python», основной библиотекой для машинного обучения является TensorFlow» и в особенности её модуль «Keras». Код программы можно найти на GitHub одного из авторов по имени пользователя «AnLiMan» по названию «Potato-Disease-Leaf-Recognazing». Код является доступным к редактированию и коммерческому использованию. Модель свёрточной нейросети – последовательная (Sequential), содержащая в себе несколько слоёв свёртки для обработки изображений, визуализация архитектуры представлена на рис. 2.

В работе использовались следующие методы: эксперимент и математическое моделирование. Основными методами процесса обучения нейронной сети, использованными в данной работе, являются алгоритм обратного распространения ошибки и использование матриц свертки для обнаружения закономерностей на изображениях. Ошибка распознавания рассчитывалась с использованием метода расчета стандартного отклонения.

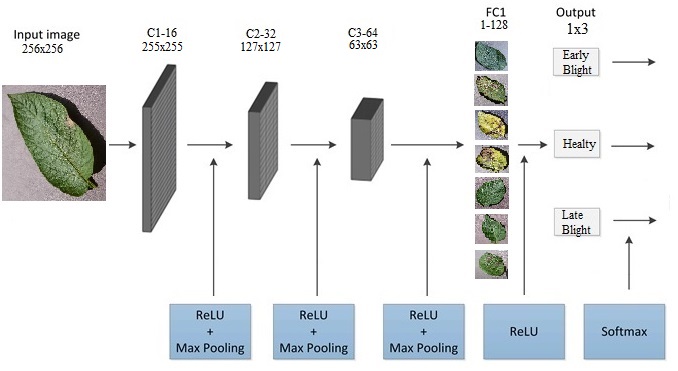


Рис. 2. Архитектура CNN для распознавания болезней листьев картофеля

Fig. 2. CNN architecture for recognizing potato leaf diseases

Источник: разработано авторами в ходе исследования

*Алгоритм работы и модель CNN*

Сверточная нейронная сеть (ConvNet / CNN) – это алгоритм глубокого обучения, который может принимать входное изображение, назначать важность различным аспектам или объектам на изображении и иметь возможность отличать один от другого. CNN может успешно фиксировать пространственные и временные зависимости в изображении с помощью применения соответствующих фильтров. Архитектура свёрточной нейронной сети аналогична схеме подключения нейронов в человеческом мозге и была вдохновлена ​​организацией зрительной коры. Отдельные нейроны реагируют на раздражители только в ограниченной области поля зрения, известной как рецептивное поле. Набор таких полей перекрывается, чтобы покрыть всю визуальную область.

Данные слои имеют следующее назначение:

*Convolutional layer* (свёрточный слой): «фильтр» проходит по изображению, сканируя несколько пикселей за раз и создавая карту объектов, которая предсказывает класс, к которому принадлежит каждый объект.

*Pooling layer* (слой пула, понижающая выборка): уменьшает количество информации в каждом признаке, полученном в сверточном слое, сохраняя при этом наиболее важную информацию (обычно есть несколько раундов свертки и объединения).

*Fully connected input layer – flatten* (полностью связанный входной слой – сглаживание): берет выходные данные предыдущих слоев, «сглаживает» их и превращает в единый вектор, который может быть входом для следующего этапа.

*The first fully connected layer* (первый полносвязный слой): принимает входные данные из анализа объектов и применяет веса для прогнозирования правильной метки.

*Fully connected output layer* (полностью связанный выходной слой): дает окончательные вероятности для каждой метки.

*Dense* (плотностный слой): это слой с глубокой связью, что означает, что каждый нейрон в плотном слое получает входные данные от всех нейронов предыдущего слоя.

*Dropout* (выпадание). Техника выпадения работает путем случайного уменьшения количества взаимосвязанных нейронов в нейронной сети. На каждом шаге обучения у каждого нейрона есть шанс быть исключенным или, скорее, выпавшим из сопоставленного вклада связанных нейронов. Этот метод сводит к минимуму вероятность переобучения, поскольку каждый нейрон становится самодостаточным, в том смысле, что нейроны внутри слоев узнают значения веса, которые не основаны на сотрудничестве соседних нейронов.

*Max-pooling Layer* (cлой с максимальным пулом): это операция объединения, которая вычисляет максимальное значение для исправлений карты объектов и использует его для создания карты объектов с пониженной дискретизацией (объединенной). Обычно он используется после свёрточного слоя.

Также слои свёртки имеют активационные функции нейронов, которые определяют значимость полученных параметров для коррекции весов нейронной сети при помощи алгоритма обратного распространения ошибки (backpropagation), в данной работе были использованы функции активации «relu» и «softmax».

Гиперпараметры – это переменные, которые определяют структуру сети, такие как количество скрытых слоев и переменные, которые определяют, как сеть обучается (например, скорость обучения). Гиперпараметры устанавливаются перед обучением сети при её конфигурации, как в коде разработанной модели ниже. Следует отметить, что автоматическая оптимизация гиперпараметров позволяет уменьшить человеческие усилия, которые необходимы для определения оптимального количества гиперпараметров конечной модели сети. Однако на практике часто реализуют комбинированный метод – увеличение гиперпараметров каждого следующего слоя на «2», что позволяет сэкономить большое количество времени на разработку и проверку различных вариантов модели, т.к. в этом случае у нас есть только 1 переменная, а остальные зависимые. В данной работе была применена схема 16-32-64-128 нейронов на каждом новом слое свёртки (рис. 2).

Алгоритм обучения свёрточной нейросети для определения больных листьев картофеля представлен на рис. 3. Процесс обучения начинается с импорта библиотек для машинного обучения, работы массивами, архивами, изображениями и облачными хранилищами. После импорта изображений они разделяются на несколько классов и помещаются во временное хранилище для работы. Далее идет определение архитектуры сети, которая будет подробно описана позднее. Основной процесс обучения занимает порядка 30 минут. В случае, если эффективность распознавания превысит 93%, либо если закончится количество эпох обучения, то процесс обучения завершится. После модель и её веса сохраняются на облачном хранилище, чтобы не повторять процесс обучения в будущем.

model = tf.keras.models.Sequential([

    tf.keras.layers.Conv2D(16, (2,2), activation='relu', input\_shape=(256, 256, 3),

    tf.keras.layers.MaxPooling2D(2, 2),

    tf.keras.layers.Conv2D(32, (2,2), activation='relu'),

    tf.keras.layers.MaxPooling2D(2,2),

    tf.keras.layers.Conv2D(64, (2,2), activation='relu'),

    tf.keras.layers.MaxPooling2D(2,2),

    tf.keras.layers.Flatten(),

    tf.keras.layers.Dense(128, activation='relu'),

    tf.keras.layers.Dropout(0.2),

    tf.keras.layers.Dense(3, activation='softmax')

])

Начало (Start)

Подключение библиотек

(Import libraries)

Работа с изображениями

(Working with dataset)

Working with dataset

Определение архитектуры

(Defining architecture of CNN)

Обучение нейросети

(Training Neural network)

A > 93%; Epochs > 15

Сохранение модели

(Saving Model)

Конец (End)

Да (Yes)

Нет (No)

Рис. 3. Алгоритм обучения сверточной нейронной сети

Fig.3. Algorithm for training a convolutional neural network

Источник: разработано авторами в ходе исследования

**Результаты**

Резюме разработанной сверточной нейронной сети для определения больных листьев картофеля представлено в табл. 1, из данной таблицы можно увидеть, что конечная архитектура содержит примерно 7,8 миллиона параметров, что является, довольно, небольшим значением и поэтому скорость распознавания одного изображения составила 0,105 секунды.

Таблица 1. **Параметры модели нейросети**

Table 1. **Parameters of the neural network model**

|  |  |  |
| --- | --- | --- |
| Тип слоя | Размер на выходе | Число параметров |
| conv2d (Conv2D) | (None, 255, 255, 16) | 208 |
| max\_pooling2d (MaxPooling2D) | (None, 127, 127, 16) | 0 |
| conv2d\_1 (Conv2D) | (None, 126, 126, 32) | 2080 |
| max\_pooling2d\_1 (MaxPooling 2D) | (None, 63, 63, 32) | 0 |
| conv2d\_2 (Conv2D) | (None, 62, 62, 64) | 8256 |
| max\_pooling2d\_2 (MaxPooling 2D) | (None, 31, 31, 64) | 0 |
| flatten (Flatten) | (None, 61504) | 0 |
| dense (Dense) | (None, 128) | 7 872 640 |
| dropout (Dropout) | (None, 128) | 0 |
| dense\_1 (Dense) | (None, 3) | 387 |
| Всего параметров | | 7 883 571 |

Источник: разработано авторами в ходе исследования

Как уже было сказано ранее примерное время обучения составило 30 минут график данного процесса показан на риc. 4. Для последней эпохи обучения и всей нейросети были получены следующие результаты: loss: 0.1498 - accuracy: 0.9657 - val\_loss: 0.0332 - val\_accuracy: 0.9823. Здесь loss – «потери», разница между полученным значением предсказания и реальным (в нашем случае это 0,1,2), accuracy – эффективность распознавания для текущей эпохи. Главные показатели: «val\_loss» и «val\_accuracy», отображающие итоговые результаты обучения сети.

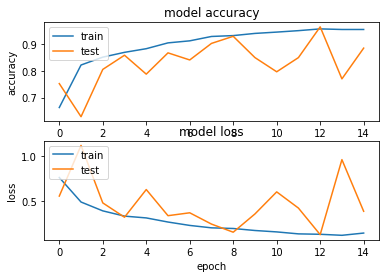


Рис. 4. Графики обучения нейросети

Fig. 4. Neural network training graphs

Источник: разработано авторами в ходе исследования

Ниже на рис. 5 приведён результат распознавания 16-ти случайных изображений из тестовой выборки. Зелёный цвет означает, что изображение было распознано правильно, за скобками указано предсказанное значение, а в скобках реальное. Как видно из рис. 5, эффективности в 98,23% позволяет достичь достаточно высоких результатов распознавания. Также был проведён тест скорости распознавания и вывода одного изображения (рис. 6), время составило примерно 0,105 секунды.



Рис. 5. Результат распознавания 16-ти случайных изображений из тестовой выборки

Fig. 5. The result of recognition of 16 random images from the test sample

Источник: разработано авторами в ходе исследования



Рис. 6. Проверка отдельного изображения из тестовой выборки

Fig 6. Checking a single image from the test set

Источник: разработано авторами в ходе исследования

**Обсуждение**

Для увеличения скорости работы нейронной сети мы предлагаем убрать компонент визуализации результата предсказания, что позволит добиться скорости распознавания – 0,00079±5% секунд для всех изображений из тестовой выборки.

Следует отметить важное ограничение данной работы – достаточно сложно добиться высокой точности распознавания без использования дополнительных методов фильтрации и обработки изображений. Поэтому для повышения производительности сети необходимо применять алгоритмы предварительной обработки необработанных изображений, например, фильтром Гаусса или аналогичным. Также следует провести постобработку результатов прогнозирования, путём усреднения или перепроверки. Для улучшения качества предсказания также можно повысить порог срабатывания (threshold) нейрона, но в этом случае потребуется увеличить размер датасета.

**Заключение**

Использование разработанной технологии определения вида заболевания картофеля больных листьев на основе их внешнего вида с применением алгоритмов машинного обучения, в частности свёрточных нейронных сетей позволяет автоматизировать процесс распознавания вида заболевания, снизить трудозатраты на определение эффективного способа профилактики заболеваний картофеля, повысить его качество и урожайность.

Эффективность определения вида заболеваний картофеля составила 98,23% со скоростью определения данного заболевания 0,00079 секунды при использовании интерпретируемого языка Python, это время также можно уменьшить, если экспортировать модель на компилируемый язык программирования, например, C#.

Предлагаемый вариант решения проблемы снижения качества урожая из-за возникновения болезни листьев картофеля позволяет предотвратить данный процесс на раннем этапе. Код разрабатывался на основе существующего фреймворка Google.Colab и облачного хранилища, что позволяет другим исследователем ознакомиться с данным алгоритмом и улучшить его работу при необходимости.

**СПИСОК ИСТОЧНИКОВ**

1. *Beals K. A.* Potatoes, Nutrition and Health // American Journal of Potato Research. 2019 №96. P. 102-110.

2. *Arora R. K., Sharma S.* Pre and Post Harvest Diseases of Potato and Their Management // in Future Challenges in Crop Protection Against Fungal Pathogens, New York: Springer. 2014. P. 149-183.

3. P. TM, P. Alla, K. S. Ashirta, N. B. Chittaragi and S. G. Koolagudi, "Tomato Leaf Disease Detection using Convolutional Neural Network," International Conference on Contemporary Computing (IC3), 2018, pp. 125-131.

4. *Rizqi A. S., Indra A. S*. Potato Leaf Disease Classification Using Deep Learning Approach // 2020 International Electronics Symposium (IES). 2020. P. 392-397.

5. *Afonso M., Blok P. M.* Blackleg Detection in Potato Plants using Convolutional Neural Networks // IFAC (International Federation of Automatic Control). 2019. P. 6-11.

6. *Qinghua S., Kondo N.* Potato Quality Grading Based on Depth Imaging and Convolutional Neural Network // Hindawi Journal of Food Quality. 2020. P. 322-331

7. *Moallem P., Razmjooy N., Ashourian M.* Computer vision-based potato defect detection using neural networks and support vector machine, International Journal of Robotics and Automation, Vol. 28, No. 2. 2013. P. 137-145.

8. *Vozmilov A., Andreev L., Lisov A.* Development of an Algorithm for the Program to Recognize Defects on the Surface of Hot-Rolled Metal //2022 International Conference on Industrial Engineering, Applications and Manufacturing (ICIEAM). – IEEE. 2022. P. 1004-1008.

9. *Abeer A. E., Ibtesam M. D.* Potato Classification Using Deep Learning // International Journal of Academic Pedagogical Research (IJAPR). Vol. 3 No. 12. 2019. P. 1-8

10. *You D., Gao X., Katayama S.* WPD-PCA-based laser welding process monitoring and defects diagnosis by using FNN and SVM // IEEE Trans. Ind. Electron. Vol. 62 No.1. 2014. P. 628–636

11. *Guo G., Wang H., Bell D., Bi Y., Greer K.* KNN Model-Based Approach in Classification // OTM Confederated International Conferences. 2003. P. 986–996.

12. *Su X., Yan X., Tsai C.* Linear regression // WIREs Comput Stat 2012. 2012. Vol 4. P. 275-295

13. *Russakovsky O., Deng J., Su H.* Imagenet large scale visual recognition challenge // Int. J. Comput. Vision. 2015. Vol. 115 No. 3. P. 211–252.

14. *Simonyan K., Zisserman A.* Very deep convolutional networks for large-scale image recognition // arXiv preprint. 2014. Vol. 15. P. 1011–1045.

15. *Szegedy C., Liu W., Jia Y.* Going deeper with convolutions // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015. P. 1–9.

16. *He K., Zhang X., Ren S.* Deep residual learning for image recognition // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016. P. 770–778.

17. *Redmon J., Divvala S., Girshick R.* You only look once: Unified, real-time object detection // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016. P. 779–788.

18. *Redmon J., Farhadi A.* YOLO9000: better, faster, stronger // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017. P. 7263–7271.

19. *Redmon J., Farhadi A.* An incremental improvement // arXiv preprint. 2018. P. 180–201.

20. *Girshick R., Donahue J*., *Darrell T.* Region-based convolutional networks for accurate object detection and segmentation // IEEE Trans. Pattern Anal. Mach. Intell. Vol. 38 No. 1. 2015. P. 142–158.

21. *Ren S., He K., Girshick R.* Faster R-CNN: towards real-time object detection with region proposal networks // Advances in Neural Information Processing Systems. 2015. P. 91–99.

22. *Liu W., Anguelov D., Erhan D.* SSD: single shot multibox detector // European Conference on Computer Vision: Springer, Cham. 2016. P. 21–37.

23. Fu C.Y., Liu W., Ranga A. Dssd: deconvolutional single shot detector // arXiv preprint. 2017. P. 1011–1023.

24. *Behnke S.* Hierarchical Neural Networks for Image Interpretation // Lecture Notes in Computer Science: Springer. Vol. 27. 2003. P. 67–77.

25. *Dai J., Li Y., He K.* R-FCN: object detection via region-based fully convolutional networks // Neural Inform. Process. Syst. 2016. P. 379–387.

*Информация об авторах:*

**А. Г. Возмилов** – д.т.н., профессор, Spin-код: 2893-8730;

**А. А. Лисов** – магистр, аспирант, Spin-код: 1956-3662;

**В. Г. Урманов** – доцент, к.т.н, Spin-код: 4735-0050;

**Г.Н. Синева** – доцент, к.т.н.

*Заявленный вклад авторов:*

**Возмилов А. Г.** – общее руководство проектом, анализ и дополнение текста статьи.

**Лисов** **А. А.** – проведение критического анализа материалов, доработка текста статьи.

**Урманов В. Г.** – сбор и обработка материалов, работа над текстом статьи.

**Синева Г.Н.** – сбор и обработка материалов, участие в обсуждении материалов статьи.

*Авторы заявляют об отсутствии конфликта интересов.*

**REFERENCES**

1. *Beals K. A.* Potatoes, Nutrition and Health // American Journal of Potato Research. 2019 №96. P. 102-110.

2. *Arora R. K., Sharma S.* Pre and Post Harvest Diseases of Potato and Their Management // in Future Challenges in Crop Protection Against Fungal Pathogens, New York: Springer. 2014. P. 149-183.

3. P. TM, P. Alla, K. S. Ashirta, N. B. Chittaragi and S. G. Koolagudi, "Tomato Leaf Disease Detection using Convolutional Neural Network," International Conference on Contemporary Computing (IC3), 2018, pp. 125-131.

4. *Rizqi A. S., Indra A. S*. Potato Leaf Disease Classification Using Deep Learning Approach // 2020 International Electronics Symposium (IES). 2020. P. 392-397.

5. *Afonso M., Blok P. M.* Blackleg Detection in Potato Plants using Convolutional Neural Networks // IFAC (International Federation of Automatic Control). 2019. P. 6-11.

6. *Qinghua S., Kondo N.* Potato Quality Grading Based on Depth Imaging and Convolutional Neural Network // Hindawi Journal of Food Quality. 2020. P. 322-331

7. *Moallem P., Razmjooy N., Ashourian M.* Computer vision-based potato defect detection using neural networks and support vector machine, International Journal of Robotics and Automation, Vol. 28, No. 2. 2013. P. 137-145.

8. *Vozmilov A., Andreev L., Lisov A.* Development of an Algorithm for the Program to Recognize Defects on the Surface of Hot-Rolled Metal //2022 International Conference on Industrial Engineering, Applications and Manufacturing (ICIEAM). – IEEE. 2022. P. 1004-1008.

9. *Abeer A. E., Ibtesam M. D.* Potato Classification Using Deep Learning // International Journal of Academic Pedagogical Research (IJAPR). Vol. 3 No. 12. 2019. P. 1-8

10. *You D., Gao X., Katayama S.* WPD-PCA-based laser welding process monitoring and defects diagnosis by using FNN and SVM // IEEE Trans. Ind. Electron. Vol. 62 No.1. 2014. P. 628–636

11. *Guo G., Wang H., Bell D., Bi Y., Greer K.* KNN Model-Based Approach in Classification // OTM Confederated International Conferences. 2003. P. 986–996.

12. *Su X., Yan X., Tsai C.* Linear regression // WIREs Comput Stat 2012. 2012. Vol 4. P. 275-295

13. *Russakovsky O., Deng J., Su H.* Imagenet large scale visual recognition challenge // Int. J. Comput. Vision. 2015. Vol. 115 No. 3. P. 211–252.

14. *Simonyan K., Zisserman A.* Very deep convolutional networks for large-scale image recognition // arXiv preprint. 2014. Vol. 15. P. 1011–1045.

15. *Szegedy C., Liu W., Jia Y.* Going deeper with convolutions // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015. P. 1–9.

16. *He K., Zhang X., Ren S.* Deep residual learning for image recognition // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016. P. 770–778.

17. *Redmon J., Divvala S., Girshick R.* You only look once: Unified, real-time object detection // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016. P. 779–788.

18. *Redmon J., Farhadi A.* YOLO9000: better, faster, stronger // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017. P. 7263–7271.

19. *Redmon J., Farhadi A.* An incremental improvement // arXiv preprint. 2018. P. 180–201.

20. *Girshick R., Donahue J*., *Darrell T.* Region-based convolutional networks for accurate object detection and segmentation // IEEE Trans. Pattern Anal. Mach. Intell. Vol. 38 No. 1. 2015. P. 142–158.

21. *Ren S., He K., Girshick R.* Faster R-CNN: towards real-time object detection with region proposal networks // Advances in Neural Information Processing Systems. 2015. P. 91–99.

22. *Liu W., Anguelov D., Erhan D.* SSD: single shot multibox detector // European Conference on Computer Vision: Springer, Cham. 2016. P. 21–37.

23. Fu C.Y., Liu W., Ranga A. Dssd: deconvolutional single shot detector // arXiv preprint. 2017. P. 1011–1023.

24. *Behnke S.* Hierarchical Neural Networks for Image Interpretation // Lecture Notes in Computer Science: Springer. Vol. 27. 2003. P. 67–77.

25. *Dai J., Li Y., He K.* R-FCN: object detection via region-based fully convolutional networks // Neural Inform. Process. Syst. 2016. P. 379–387.

Information about authors:

**A. G. Vozmilov** Dr. Sci. (Engineering), associate professor, Spin-code: 2893-8730;

**A.A. Lisov**, master, post-graduate student, Spin-code: 1956-3662;

**V.G. Urmanov**, Ph. D. (Engineering), assistant professor, Spin-code: 4735-0050;

**G. N. Sineva**, Ph. D. (Engineering), assistant professor.

*Contribution of the authors:***Vozmilov A. G.** – general project management, justification of parameters, formulation of the conclusion.

**Lisov A.A.** – conducting a critical analysis of materials, finalizing the text of the article.

**Urmanov V.G.**  – collection and processing of materials, work on the text of the article.

**Sineva G. N.**  – collection and processing of materials, participation in the discussion of the materials of the article.

*The authors declare no conflicts of interests*