

**DOI:10.26104/NNTIK.2022.52.39.007**

*Сабитов Б.Р., Сейтказиева Н.С., Кашикароева А.А.*

# **ЖҮГӨРҮ ООРУСУН БОЛЖОЛДОО ҮЧҮН КОНВОЛЮЦИЯЛЫК НЕЙРОН ТАРМАГЫН КУРУУ**

*Сабитов Б.Р., Сейтказиева Н.С., Кашикароева А.А.*

## **ПОСТРОЕНИЕ СВЕРТОЧНОЙ НЕЙРОННОЙ СЕТИ ДЛЯ ПРОГНОЗИРОВАНИЯ БОЛЕЗНЕЙ КУКУРУЗЫ**

*B. Sabitov, N. Seitkazieva, A. Kashkaroeva*

### **BUILDING A CONVOLUTIONAL NEURAL NETWORK FOR PREDICTING CORN DISEASES**

УДК: 004.93

Бул макалада азыркы учурдагы заманбап колдонмолор, конволюциялык нейрон тармактарынын сүрөттөрүн таануу технологиясы каралат. Жүгөрүнүн оорулуу жана дени сак сүрөттөрүнүн конкреттүү маалымат базасы үчүн өсүмдүктүн ар кандай ооруларын аныктоо модели курулган. Иш терең окутуу технологияларын жана Google Colabatory кызматтарын колдонуу менен аткарылды. Таануу тапшырмасы үчүн маалыматтарды көбөйтүү ыкмалары колдонулган, бул терең үйрөнүү моделинин сапатын бир топ жогорулатат. Бул макалада сунушталган модель жүгөрү өсүмдүктөрүнүн жалбырактарынын сүрөттөрүн колдонуу менен бактериялык тактарды аныктоо үчүн колдонулат, бирок ал ар кандай өсүмдүк ооруларын аныктоо үчүн колдонулушу мүмкүн. Бул макаладагы эксперименттер жүгөрү өсүмдүктөрүнүн жалбырактарын сүрөттөө үчүн жалпыга жеткиликтүү PlantVillage маалымат топтомун колдонот.

**Негизги сөздөр:** конволюциялык нейрон тармактары, модель, жүгөрү, моделдин тактыгы, моделдин катасы, моделди текшерүү, таануу.

В данной работе рассматривается применения современной в данное время, технологии распознавания изображений сверточных нейронных сетей. Для конкретной базы данных больных и здоровых изображений кукурузы построена модель обнаружения различных болезней данного растения. Работа выполнена с использованием технологий глубокого обучения и сервисов Google Colaboratory. Для задачи распознавания использованы методы увеличения данных, которая существенно увеличивает качество модели глубокого обучения. В этой работе предложенная модель применяется для обнаружения бактериальной пятнистости, присутствующей на растениях кукурузы, с использованием изображений их листьев, однако ее можно использовать для обнаружения любых болезней растений. Эксперименты, проведенные в этой статье, используют общедоступный набор данных PlantVillage, для получения изображений листьев растений кукурузы.

**Ключевые слова:** сверточные нейронные сети, модель, кукуруза, точность модели, ошибка модели, тестирование модели, распознавание.

In this paper, the application of the currently modern image recognition technology of convolutional neural networks is considered. For a specific database of sick and healthy images of corn, a model for detecting various diseases of this plant has been built. The work was performed using deep learning technologies and Google Colabatory services. For the recognition task, data augmentation methods were used, which significantly increases the quality of the deep learning model. In this work, the proposed model is used

to detect bacterial spotting present on corn plants using images of their leaves, but it can be used to detect any plant diseases. The experiments conducted in this article use the publicly available Plant Village dataset to obtain images of corn plant leaves.

**Key words:** convolutional neural networks, model, corn, model accuracy, model error, model testing, recognition.

**Введение.** Прогнозирование болезней сельскохозяйственных культур для регионов страны актуально и имеет ценное практическое значение для экономики страны. В период роста, растения восприимчивы к различным заболеваниям. Раннее выявление болезней растений является одной из самых сложных задач в сельском хозяйстве. Если заболевания не выявить на ранних стадиях, то они могут отрицательно сказаться на общем урожае, что приведет к снижению прибыли фермеров и в целом экономики страны. Чтобы преодолеть эту проблему, многие исследователи представили различные современные системы, основанные на подходах глубокого обучения и машинного обучения. Однако большинство этих систем либо используют очень большое число обучающих параметров, либо имеют низкую точность классификации. В этой работе предложенная модель применяется для обнаружения бактериальной пятнистости, присутствующей на растениях кукурузы, с использованием изображений их листьев, однако ее можно использовать для обнаружения любых болезней растений. Эксперименты, проведенные в этой статье, используют общедоступный набор данных PlantVillage, для получения изображений листьев растений кукурузы. Предлагаемая система обеспечивает точность обучения 99,35% и точность при тестировании 98,38%, используя всего 9914 параметров обучения.

**Обзор научных публикаций и исследований распознавания болезней растений.** Алгоритмы машинного обучения в последнее время применительно к задачам распознавания болезней растений (Es-saady et al., 2016 [1]; Islam et al., 2017 [2]; Krithika and Selvarani, 2017 [3]; Padol and Yadav, 2016) [4] приобретает все большее значения в научных исследованиях.

Вот небольшой обзор современных систем, представленные в литературе, которые используются

для автоматического обнаружения болезней растений. Санга и др. (Sanga et al., 2020) [5] разработали инструмент для обнаружения болезней растений с пятью различными архитектурами CNN. Этими архитектурами были VGG-16, ResNet-152, ResNet-50, ResNet-18 и InceptionV3. Они обнаружили, что ResNet-152 превосходит другие с точностью 99,2%. Они также разработали мобильное приложение, чтобы фермеры могли легко выявлять болезни различных растений, загружая изображения листьев своих растений на свои смартфоны. Это мобильное приложение использовало модель InceptionV3 для прогнозирования заболеваний с достоверностью 99%. Как указано в статье ResNet, количество тренировочных параметров, используемых их самой эффективной моделью, т.е. ResNet-152, составляло 60 миллионов (He et al., 2016) [6]. Другая аналогичная работа была проделана в статье, автором которой является Chohan et al. (Чохан и др., 2020 г.) [7]. Они использовали архитектуры CNN VGG-19 и InceptionV3 для автоматического обнаружения болезней растений с использованием набора данных PlantVillage. В своей исследовательской работе они также использовали аугментацию данных для искусственного расширения набора данных. Модель VGG-19 превзошла модель InceptionV3 с точностью обучения 98% и точностью тестирования 95%, как они утверждают в своей статье. Количество тренировочных параметров, используемых их самой эффективной моделью, т. е.

VGG-19, составило 143 миллиона, как утверждают (Симонян и Зиссерман, 2015) в своей исследовательской работе. Моханти и др. (Mohanty et al., 2016) проанализировали производительность архитектур AlexNet и GoogLeNet CNN для обнаружения болезней растений с использованием набора данных PlantVillage.

**Методология и методы построения нейронных сетей.** Сверточные нейронные сети (CNN) – это метод глубокого обучения, используемые во многих приложениях компьютерного зрения. В CNN вместо простого матричного умножения используется операция свертки, и эта операция свертки используется для извлечения различных пространственных и временных характеристик из данных изображения. CNN используются для классификации входных изображений по соответствующим классам. По сравнению с другими методами глубокого обучения CNN наиболее эффективно работает с изображениями. Он извлекает из входных изображений различные пространственные и временные характеристики, которые играют важную роль в классификации изображений и других задачах компьютерного зрения. На рисунке 1 изображена архитектура типичной CNN, которая содержит один входной слой, один выходных слоя, набор сверточных слоев, объединяющих слоев и полносвязных слоев (каждый с функцией активации).

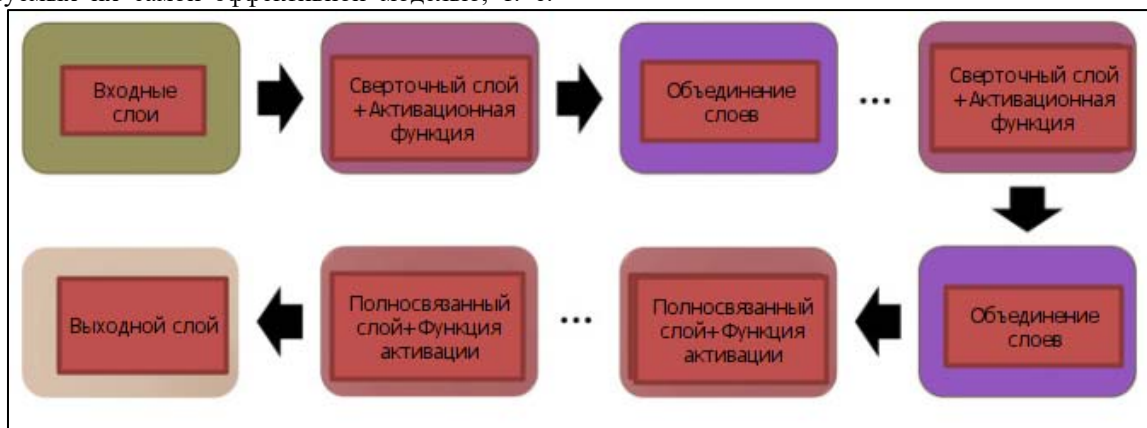


Рис. 1. Типичная архитектура сверточной нейронной сети.

Программная реализация CNN. Согласно требованиям, мы сначала с использованием, учетной записи Google монтируем рабочую среду (Листинг 1) диск, где будем хранить данные и загружать программное обеспечение.

Листинг 1

Монтирование программной среды в Colaboratory.

```
from google.colab import drive
drive.mount('/content/drive')

Mounted at /content/drive
```

Все данные изображения больных и здоровых растений в виде классов, мы предварительно загружаем их в Google диск. Далее загружаем базу данных в рабочую среду. База данных загружаем в каталог /content/drive/MyDrive/Dataset\_Full/Dataset\_Corn\_Train\_Valid\_Test и посмотрим какие классы имеются в

нашем хранилище. Для построения модели распознавания мы должны иметь три важных набора данных. Это набор данных для обучения модели, для проверки-валидации модели и набор данных тестирования модели (Листинг 2).

Листинг 2

#### Определение классов обучения, валидации и тестирования модели.

```
data
['/content/drive/My Drive/Dataset_Full/Dataset_Corn_Train_Valid_Test/train',
'/content/drive/My Drive/Dataset_Full/Dataset_Corn_Train_Valid_Test/test',
'/content/drive/My Drive/Dataset_Full/Dataset_Corn_Train_Valid_Test/valid',
```

В нашем наборе данных есть три папки, которые содержат базу данных больных и здоровых изображений листьев кукурузы. Эти папки train, valid и test. Давайте рассмотрим, Листинг 3 какие классы, типы болезней имеются в этих папках.

Листинг 3

#### Элементы классов обучения модели.

```
[ ] train_classes
['/content/drive/My Drive/Dataset_Full/Dataset_Corn_Train_Valid_Test/train/Corn_(maize)___healthy',
'/content/drive/My Drive/Dataset_Full/Dataset_Corn_Train_Valid_Test/train/Corn_(maize)___Northern_Leaf_Blight',
'/content/drive/My Drive/Dataset_Full/Dataset_Corn_Train_Valid_Test/train/Corn_(maize)___Cercospora_leaf_spot_Gray_leaf_spot',
'/content/drive/My Drive/Dataset_Full/Dataset_Corn_Train_Valid_Test/train/Corn_(maize)___Common_rust_']
```

#### Обработка базы данных больных растений.

**Методы увеличения данных.** При обработке базы данных в настоящее время широко используется метод увеличения данных. Увеличение данных, это метод искусственного создания новых обучающих данных из существующих обучающих данных. В данном случае мы путем применения специальных параметров, специфических предметной области и к примерам из обучающих данных создаем новые и разные обучающие наборы данных. Причиной использования метода увеличения является, факт, что, мы не знаем какой тип изображений мы получаем от пользователя. В процессе работы с базой данных изображений возможно различные случаи, мы можем получить изображение просто с названием, он присылается пользователем увеличенным или растянутым. Он может быть перевернутым на любой угол. С этой целью для обучения модели у нас должно быть классы изображений любой конфигурации. Для эффективной классификации любого типа изображений мы во всех

обучающих классах должны выполнить увеличение данных.

В Keras для увеличения данных имеется генератор, пакетный модуль ImageDataGenerator для увеличения т.е. для выполнения всех выше связанных операций с изображениями:

```
from tensorflow.keras.preprocessing.image import
ImageDataGenerator train_agument = ImageDataGenerator(
rescale=1./255, rotation_range=40, width_shift_range=0.2,
height_shift_range=0.2, shear_range=0.2, zoom_range=0.2,
horizontal_flip=True, fill_mode='nearest')
```

Здесь применяются различные методы увеличения данных. Вращение, диапазон сдвига по ширине, диапазон сдвига по высоте, диапазон сдвига, диапазон масштабирования, горизонтальное отражение, режим заполнения. Этот генератор, который будет преобразовывать изображения, найденные в train\_data\_path, и неограниченно будет их генерировать, создавая пакет расширенных данных изображения.

Листинг 4

#### Определение классов и данных для обучения модели.

```
train_agumented_set = train_agument.flow_from_directory('/content/drive/My Drive/Dataset_Full/Dataset_Corn_Train_Valid_Test/train/',
target_size=(224, 224), # размер всех изображений будет изменен на 224x224
batch_size=32,
class_mode='categorical') # поскольку мы используем потери categorical, нам нужны классы

Found 7316 images belonging to 4 classes.
```

У нас есть 7316 изображений для обучения модели. Нам нужен размер входного изображения (224x224), поэтому мы даем размер в виде `target_size = (224, 224)`. Еще набор данных содержит цветные изображения, причем цветные изображения представлены в RGB. Оператор `train_agumented_set [0][0][0]` возвращает 3 dim, который содержит [R, G, B]. Ниже представлено графическое представление увеличенных данных.

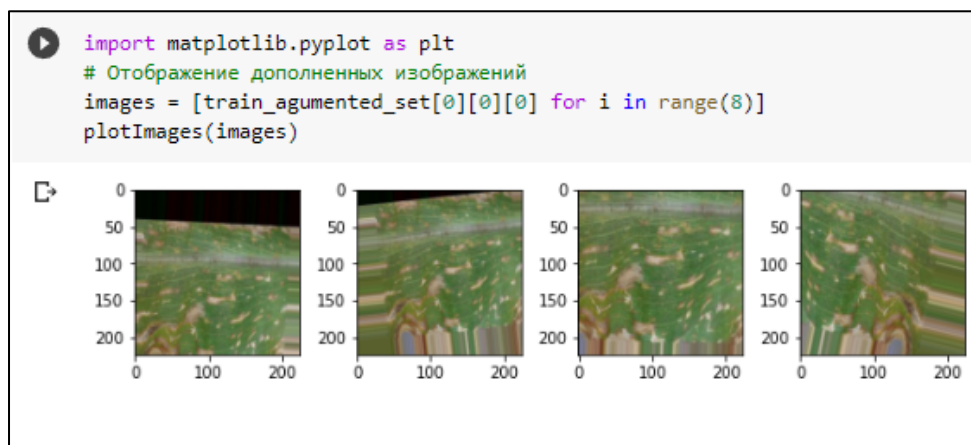


Рис. 2. Графическое представление увеличенных данных.

Рассмотрим теперь другую базу данных для проверки, уже обученной модели. Здесь необходимо отметить, что они не участвуют при обучении модели. Мы не применяем какое-либо увеличение данных к изображениям класса `valid`, потому что они предназначены для проверки обученной модели. Вообще если мы его увеличили, тогда наш модель переобучается. Мы просто изменяем масштаб изображения, например `rescale=1./255`. Ниже представлен код преобразования данных валидации-Листинг 5.

Листинг 5

Определение классов и количество изображений данных валидации.

```

# это конфигурация дополнения, которую мы будем использовать для проверки:
# только масштабирование
valid_agument = ImageDataGenerator(rescale=1./255)

# это аналогичный генератор для данных проверки
valid_agumented_set = valid_agument.flow_from_directory('/content/drive/My Drive/'
'Dataset_Full/Dataset_Corn_Train_Valid_Test/valid/',
                target_size=(224,224),
                batch_size=32,
                class_mode='categorical')

Found 1829 images belonging to 4 classes.
    
```

Таким образом, у нас имеются 1829 изображений в наборе данных валидации - проверки модели. Обычно для обучения каждый класс индексирован и имеют метки, например:

Листинг 6

Код, который определяет индексы классов болезней кукурузы.

```

train_agumented_set.class_indices

{'Corn_(maize)___Cercospora_leaf_spot Gray_leaf_spot': 0,
 'Corn_(maize)___Common_rust_': 1,
 'Corn_(maize)___Northern_Leaf_Blight': 2,
 'Corn_(maize)___healthy': 3}
    
```

Классы в данном случае, представляют 3 типа болезней растения - первые три строки выхода в листинге 6, и один тип класса здоровых растения – четвертая строка Листинга 6.

Как мы говорили выше мы будем строить сверточные нейронные сети (CNN) для распознавания больных и здоровых изображений, т.е. будем решать задачу классификации с помощью нейронной сети. Будем создавать ее с нуля. В конце мы изучим трансферное обучение. Вот некоторые детали и структура сверточной нейронной сети. Базовые CNN имеют следующие разделы: слой свертки Convolution layer-Conv (CNN-(conv2d)), MaxPooling, полностью подключенный слой - Fully connected, исключение - Dropout, пакетная нормализация - Batch normalization, сглаживания слоя - Flatten layer, выходной слой - Output (softmax/sigmoid) layer. Dropout + Batch Normalization - выступает как регуляризация модели. Размер входного изображения - Our input image size = (224,224). Наша задача относится к мультиклассификации, т.е. у нас имеются 4 класса, и мы должны их в конце активизировать. В качестве функции активации выходного слоя берем - Softmax. Используемые библиотеки Keras с TensorFlow. Теперь рассмотрим модельную архитектуру CNN-сверточной нейронной сети. Шаги создания сверточной нейронной сети в Keras следующие. В следующем шаге определяется, сначала

архитектура нейронной сети, а затем компилируется. Сверточный слой, присутствующий в CNN-(conv 2d), выполняет операцию свертки. Начальные сверточные слои CNN извлекают простые низкоуровневые характеристики изображения, а сверточные слои, присутствующие в конце сети, извлекают сложные высокоуровневые характеристики изображения. Операция свертки определяется как бинарная операция (представленная символом  $*$ ) между двумя вещественными функциями (скажем,  $f(x)$  и  $g(x)$ ). В непрерывной области это может быть определено математически, как уравнение (1). Точно так же в дискретной области математическая формула для операции свертки может быть записана в виде уравнения (2) (Гудфеллоу и др., 2016 г.) [8].

$$f(x)*g(x)=\int f(x)g(x-y)dy, \quad (1)$$

$$f(x)*g(x)=\sum_{k=-\infty}^{\infty} f(x)g(x-k), \quad (2)$$

В работе для обучения нейронной сети, с 7312 изображениями, использованы оптимизатор Adam с  $learning\_rate=0.01$ , для оценки ошибки допущенной сети использовано  $loss='categorical\_crossentropy'$  и в качестве метрики использовано  $metrics=['accuracy']$ . После обучения построенной нейронной сети (50 эпох) мы имеем следующие точности и ошибки построенной модели для данных обучения и тестирования модели.

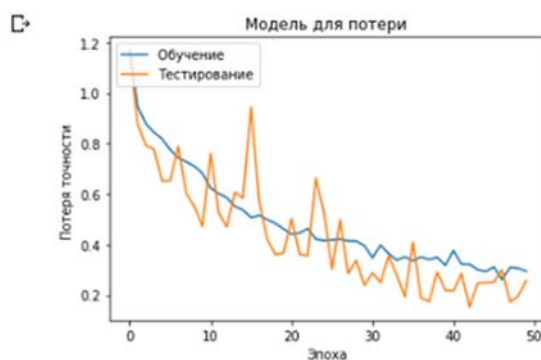


Рис. 3. Графическое представление ошибки модели на данных обучения и тестирования.

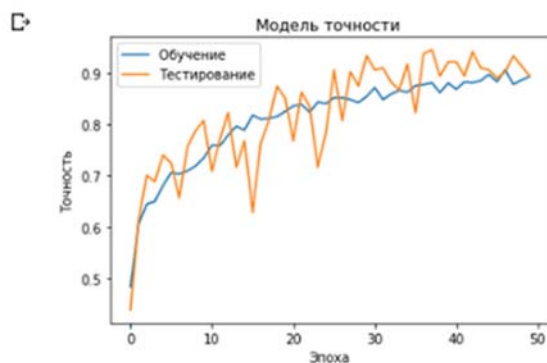


Рис. 4. Графическое представление точности модели на данных обучения и тестирования.

**Заключение.** Выявление болезней растений на ранних стадиях, является одним из сложных задач компьютерного зрения. Многие исследователи использовали различные методы машинного обучения и глубокого обучения для автоматического обнаружения болезней растений. Однако большинство из этих методов либо используют миллионы обучающих параметров, либо имеют низкую точность классификации. В этой статье была предложена модель, для обнаружения болезней растений, основанная на глубоком обучении нейронной сети, называемое сверточной нейронной сетью (CNN). Из-за увеличения размерности данных обучения значительно сократилось по сравнению с существующими современными системами. Для проверки модели ее применяли для выявления серой пятнистости листьев растений кукурузы. Модель достигла точности обучения 99% и точности тестирования 98%.

#### Литература:

1. Эс-Саади и др., Ю. Эс-Саади, И. Эль Масси, М. Эль Ясса, Д. Маммасс, А. Беназун. Автоматическое распознавание болезней листьев растений на основе серийной комбинации двух классификаторов SVM. / Международная конференция по электрическим и информационным технологиям (ICEIT), 2016г., IEEE, Танжер, Марокко (2016 г.). - С. 561-566, 10.1109/ EITech. 2016.75 1966.
2. Ислам М., Ань Дин, Вахид К., Бховмик П. Обнаружение болезней картофеля с помощью сегментации изображений и мультиклассовой машины опорных векторов. 2017. / 30-я Канадская конференция IEEE по электротехнике и вычислительной технике (CCECE), 2017г., IEEE, Виндзор, Онтарио, Канада (2017 г.). - С. 1-4, 10.1109/CCECE.2017.7946594
3. Критика Н., Селварани А.Г. Индивидуальная идентификация болезней листьев винограда с использованием скелетов листьев и классификации KNN. 2017. / Международная конференция по инновациям в информационных, встроенных и коммуникационных системах (ICSPICS), 2017 г., IEEE, Коимбатур, Индия (2017г.). - С. 1-5, 10.1109/ICSPICS. 2017.8275951.
4. Падол П.Б., Ядав А.А. Обнаружение болезней виноградных листьев на основе классификатора SVM, в: Конференция по достижениям в обработке сигналов (CASP) 2016 г. IEEE, Пуна, Индия (2016г.). - С. 175-179. 10.1109/CASP.2016.7746160.
5. Санга С.Л., Мачуве Д., Йоманга К. Мобильные модели глубокого обучения для выявления болезней бананов. / Технол. заявл. науч. Рез., 10 (3) (2020), стр. 5674-5677.
6. Хэ К., Чжан С., Рен С., Сун Дж. Глубокое остаточное обучение для распознавания изображений. / Конференция IEEE по компьютерному зрению и распознаванию образов (CVPR), 2016 г., IEEE, Лас-Вегас, Невада, США (2016 г.), стр. 770-778, 10.1109/ CVPR. 2016.90.
7. Чохан М., Хан А., Катпер С., Махар М. Обнаружение болезней растений с помощью глубокого обучения. / Межд. Дж. Недавние технологии. англ., 9 (1) (2020). - PP. 909 - 914, 10.35940 / ijrt A2139.059120
8. Гудфеллоу И., Бенжио Ю., Курвиль А. Глубокое обучение (1-е изд.), The MIT Press, Кембридж, Соединенное Королевство (2016 г.).
9. Сейтказиева Н.С., Токтогулова Г.А., Ибраева А.Т. Внедрение в образовательный процесс изучения информатики и информационно- коммуникационные технологии сервиса learningapps.org и google sites. Наука, новые технологии и инновации Кыргызстана. 2020. №. 12. С. 239-243