The Scientific Temper (2023) Vol. 14 (3): 582-590
E-ISSN: 2231-6396, ISSN: 0976-8653
Doi: 10.58414/SCIENTIFICTEMPER.2023.14.3.04
https://scientifictemper.com/

RESEARCH ARTICLE

Deep learning hyperparameter's impact on potato disease detection

Temesgen A. Asfaw

Abstract

In this study, we reviewed various published works that used deep learning techniques to detect potato leaf disease. Deep learning techniques have shown remarkable detection performance for potato leaf disease. In particular, CNN has been shown to be efficient in extracting features from images and in identifying patterns that are challenging to identify using machine learning techniques. However, CNN architectures with different activation functions, batch sizes, and optimizers can cause different results. Therefore, in this work, a CNN model has been implemented to analyze the effect of different activation functions, batch sizes, and optimizers for the detection of potato leaf diseases. Based on the findings of three experiments, the leaky rectifier function performed best as the activation function for the convolutional neural network (CNN) model. AdaGrad's optimizer showed superior accuracy compared to stochastic gradient descent (SGD), Adam, Adamax, and RMSProp algorithms. We also discovered that the model's performance was even better, but only when the batch size used in the model was smaller than the size of the test dataset. The work is based on deep learning to identify potato leaf disease and provide researchers and practitioners with heuristic knowledge to help increase potato production when CNN is employed in the agricultural sector.

Keywords*:* Deep learning, CNN, Batch size, Optimizer, Activation function, Potato.

Introduction

The potato has been part of the human diet for thousands of years, first in the southern portion of North America and then throughout the rest of the world. Potatoes are consumed as a vegetable in many developed countries, with adult consumption ranging from 50 to 150 g per day. On the other hand, in some rural areas of Africa and the highlands of Latin American countries, potato is considered a staple crop. It is consumed in significant quantities, with adult consumption ranging from 300 to 800 g per day (Burgos *et al.,* 2020). The main reasons for the decline in worldwide potato production are early and late potato diseases (Tsedaley, 2014; Yellareddygari *et al.,* 2019). This phenomenon directly impacts the production of potatoes.

Department of Computer Science, Mizan Tepi University, Tepi, Ethiopia

***Corresponding Author:** Temesgen A. Asfaw, Department of Computer Science, Mizan Tepi University, Tepi, Ethiopia, E-Mail: temesgenabera@mtu.edu.et

How to cite this article: Asfaw, T. (2023). Deep learning hyperparameter's impact on potato disease detection. The Scientific Temper, **14**(3): 582-590.

Doi: 10.58414/SCIENTIFICTEMPER.2023.14.3.04

Source of support: Nil

Conflict of interest: None.

Received: 12/07/2023 **Accepted:** 13/08/2023 **Published** : 25/09/2023

The global food system will need to be significantly improved, if it can sustainably and nutritiously feed the growing world population in the following decades (Devaux *et al.,* 2020). Therefore, detecting potato leaf disease in the agricultural sector becomes essential for the agricultural industry to protect food security.

Recently, there has been a significant increase in the usage of deep learning techniques in agricultural sectors (Kamilaris & Prenafeta-Boldú, 2018; Sujatha *et al.,* 2021; Zhong & Zhao, 2020). With the help of these techniques and information from plant communities, the suggested model may demonstrate the impact of hyperparameters on deep learning techniques for detecting potato leaf disease.

In this work, we compiled several published studies that utilized deep learning techniques and algorithms to detect potato leaf disease in the agriculture sector.

Deep learning is now widely used to detect potato leaf disease, but choosing the training hyperparameters for the disease detection model is time-consuming in this case (Kietzmann *et al.,* 2018; Lee *et al.,* 2020; Tiwari *et al.,* 2020a), making it difficult for researchers and practitioners to make further advancements in the field. As a result, there isn't much research that has examined how hyperparameters affect how well a deep learning model detects potato leaf disease in the agriculture sector. As a result, there appears to be limited empirical support for the idea that hyperparameters influence how well the deep learning

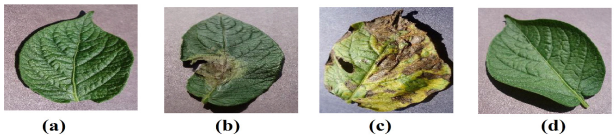

Figure 1: Sample leaf of (a) potato healthy, (b) potato late blight, (c) potato early blight, and (d) potato healthy (Hughes & Salathé, 2015).

model performs when it comes to detecting potato leaf disease. In order to pick hyperparameters, empirically generated heuristic knowledge still needs to be improved.

In this study, we'll look into the effect of hyperparameters on the deep learning models used to detect potato leaf disease in agriculture. In the current study, our contribution can be summarized as follows:

- Effect of different activation functions on CNN model performance
- The effect of different optimizers on CNN model performance
- The effect of different activation functions on CNN model performance

This paper is organized as follows: Section 2 briefly summarizes published studies that have successfully detected potato leaf disease in the agricultural sector using different deep-learning techniques. Section 3 summarizes the dataset used for the study, including pre-processing and evaluating hyperparameters on the chosen dataset. Section 4 presents the model results. Finally, this study ends with conclusions and suggestions for further research.

Literature Review

Deep learning techniques

Deep learning techniques have been used in a number of applications in the past, including image classification, pattern recognition, and natural language processing (Elsharif *et al.,* 2020; Khalifa *et al.,* 2021). Deep learning techniques for image classification and detection frequently employ convolutional neural networks (CNNs) (Albawi *et al.,* 2017; Kagaya *et al.,* 2014; Sharma *et al.,* 2018).

In computer vision applications, CNN is a widely used technique. It belongs to a type of deep neural network that is used to evaluate visual data. A convolutional phase is used to process an input image before assigning a label to it. First of all, an image is sent to the network; this is referred to as the input image. The input image is then processed in infinite steps; this is the convolutional part of the network. Finally, the neural network can detect the images of a potato leaf. An image is made up of an array of pixels that have height and width. A grayscale image has one channel, whereas a color image has three (one each for red, green, and blue). A channel is piled on top of another. Each pixel has a value ranging from 0 to 255 to represent the color's intensity. For example, a pixel with a value of 0 will be white, whereas a pixel with a value close to 255 will be darker.

The most crucial element of the CNN model is the convolutional layer. To locally extract an object's features from an image, convolution is utilized. It suggests that the network will be able to detect specific patterns throughout the image. A multiplication done element by element is called convolution. A feature map is the result of element-wise multiplication. A small array of pixels in the image will receive the filter's application during the convolutional phase. Generally, the filter will follow the input image in a 33 or 55 pattern. This implies that the network will perform the convolution while swiping these windows across every input image.

An activation function is applied to the output after the convolution procedure to accommodate non-linearity. The pooling layer attempts to reduce the dimensionality further and include more key characteristics into the image using the 3D feature map of the convolution layer as input. The pooling process is used to make the supplied image's dimensions smaller. Typically, the largest value of the feature map is used to pool the input image. After that, the feature map is flattened and given to the fully connected layer. The number of fully connected layers may vary depending on the problem and the network.

Deep learning techniques for potato leaf disease detection

A model for detecting potato leaf disease was designed by many researchers using deep learning techniques such as CNN and transfer learning. These strategies, according to recent studies, enhance prediction accuracy. For instance, Saeed *et al.,* (2021) suggested deep learning techniques for the early diagnosis and detection of potato diseases. The method trains deep convolutional neural networks like ResNet-152 and InceptionV3 with an accuracy of 98.34 and 95.24%, respectively, on the Kaggle potato dataset at a learning rate of 0.0005.

Asif *et al.,* (2020) suggested a model that uses image processing methods to identify and diagnose diseases in potato leaves effectively. The five methods employed in the study are AlexNet, VggNet, ResNet, LeNet, and the Sequential Model. This provided model had a precision of 97.5%. Rashid *et al.,* (2021) proposed a multi-level deep learning model for recognizing potato leaf disease. The proposed deep learning algorithm obtained 99.75% accuracy on the potato leaf disease dataset. The study by Chakraborty *et al.,* (2022) demonstrated the mask region-wbased convolutional neural network architecture and the residual network for detecting blight disease patches on potato leaves. In a field scenario with complex backgrounds, a manual study of the detection performance showed an overall precision of 98% on leaf images. Johnson *et al.,* (2021) proposed a novel network architecture named MobOca_Net to recognize potato diseases. The lightweight MobileNetV2 was chosen as the foundation network to improve the learning capability of classical MobileNetV2 by incorporating the attention mechanism behind the pre-trained network, which was followed by an octave convolution block for extracting high-dimensional features. The proposed procedure outperformed other methods in terms of performance gain, with an average identification accuracy of 97.73% on various potato disease types. A potato leaf disease detection model was created by Kukreja *et al.,* (2021) to identify and detect potato leaf diseases. They used a CNN-based deep learning multi-classification model for classifying 900 real-time images of potato crop plants with healthy and potato blight disease images based on their potato blight disease severity level, as well as a binary classification to classify the healthy and diseased crop leaves. Four disease severity levels were considered, resulting in a binary classification accuracy of 90.77% and the highest multi-classification accuracy of 94.77%.

Tiwari *et al.* (2020b) proposed a technique that fine-tunes pre-trained models like VGG19 to extract significant features from a dataset. The results were then analyzed using several classifiers, with logistic regression outperforming others by a significant margin of classification accuracy, achieving 97.8% over the test dataset. To detect disease in potato leaves, Baskar *et al.* (2013) developed a CNN architecture. According to the findings, the 70:30 data splitting produces more accuracy than the 80:20 data splitting. By using 20 batch sizes and 10 epochs, the accuracy was 97% for training data and 92% for validation data. The study by M. Islam *et al.,* (2017) proposed a new method for identifying diseases using leaf images that combines image processing and machine learning. The experimental result shows that support vector machines outperformed other models with 95% accuracy in disease classification over 300 images. The framework proposed by Hou *et al.,* (2021) compared and analyzed the effectiveness of the graph-cut algorithm with various machine-learning techniques for detecting potato leaf disease. The proposed method's performance was tested on 2840 images of healthy and diseased potato leaves. According to the segmentation data, the average intersection over union for the five classes was 93.70%. Compared to k-NN, ANN, and RF, the SVM classifier had the highest overall accuracy of 97.41% for disease classification. When it came to determining the degree of infection, the SVM classifier had the highest overall accuracy of 91.0%.

The research proposed by Patil *et al.* (2017) employed machine learning techniques to identify potato leaf diseases. According to the data, ANN has the highest accuracy of 92%, followed by SVM at 84.9% and RF at 79.6%. Singh & Kaur (2021) proposed a machine-learning-based approach for the detection and classification of diseases that affect potato plants. The K-means approach was explored for image segmentation; the gray level co-occurrence matrix concept was considered for feature extraction; and the multiclass support vector machine methodology was considered for classification. The proposed methodology has a 95.99% accuracy rate. Islam *et al.,* (2019) built a pre-trained model that detects potato leaf disease with high accuracy. Images of 152 healthy leaves, 1000 late blight leaves, and 1000 early blight leaves were used in the studies. With 20% test data and 80% training data, the model predicts with an accuracy of 99.4% in testing.

As a result, we can guarantee that deep learning offers a reliable and practical answer to the agriculture sector's potato leaf disease problem. Additionally, this method yields noteworthy outcomes across numerous datasets. Despite the widespread use of CNN, we have observed that only a very limited amount of research has been done on the effects of choosing different hyperparameters as well as how to improve CNN performance. The same previously mentioned result has also been published and supported in some other pertinent papers that have employed the same deep-learning techniques to identify potato leaf disease in the agricultural field (Lee *et al.,* 2021; Radha & Swathika, 2021). Since this work has been extensively examined in this article, we can happily state that the methods used to configure the hyperparameters in deep learning techniques where CNN is used to model the detection of potato leaf disease seem to be still lacking on a larger scale in the agricultural sector.

Methods

Dataset

The dataset used for these studies was collected from the PlantVillage collection (Hughes & Salathé, 2015). The collection contains 2152 images of potato leaves that are divided into three groups: healthy, late blight, and early blight. There are 152 images of healthy potato leaves, 1000 images of early blight leaves, and 1000 images of late blight. Table 1 provides the dataset's comprehensive information. All of the images were taken into consideration for the study. Figure 1 depicts (a) healthy potatoes, (b) late-blight potatoes, (c) early-blight potatoes, and (d) healthy potatoes.

Image Pre-processing

The collected image sizes varied, making learning difficult for the model. As a result, before dividing the collected image dataset into the training, test, and validation sets, we performed image pre-processing, such as resizing, data normalization, and data splitting. The images have been reduced in size to 224 224 pixels. The image's pixel value is rescaled to the interval [0, 1] via data normalization. The

Table 1: Provides the dataset's comprehensive information.

Name of sample	Number of samples
Healthy leaf	152
Early blight	1000
Late blight	1000
Total	2152

Table 2: The output of the CNN architecture.

Layer(type)	Output shape	Turtles
conv2d (Conv2D)	(None, 224, 224, 32)	896
max_pooling2d (MaxPooling2D)	(None, 112, 112, 32)	0
conv2d_1 (Conv2D)	(None, 112, 112, 64)	18496
max_pooling2d_1 (MaxPooling2D)	(None, 56, 56, 64)	0
conv2d_2 (Conv2D)	(None, 56, 56, 128)	73856
max_pooling2d_2 (MaxPooling2D)	(None, 28, 28, 128)	0
flatten (Flatten)	(None, 100352)	0
dense (Dense)	(None, 1000)	100353000
dense_1 (Dense)	(None, 3)	3003

dataset was split into three sections: training, testing, and validation. The validation and test datasets were used to assess the performance of the proposed model, while the training dataset was used to train the CNN model. As a result, we divided the training, validation, and testing datasets by 80%, 10%, and 10%, respectively.

Performance evaluation.

We used accuracy, precision, recall, and F1-Score as performance evaluation metrics.

- Accuracy. Accuracy is the fraction of the number of true predictions to the entire number of input examples.

Accuracy = (True positive + False negative)/(Total Number of samples) (1)

- Precision. It is the number of true positive outcomes divided by the number of positive outcomes expected by the classifier.

Precision = True positive/(True positive + False positive) (2)

- Recall. It is the number of true positive outcomes divided by the total number of all patterns that should have been known as positive.

Recall = True positive/(True positive + False negative) (3)

- F1-Score. The F1 Score is the choral mean of recall and precision. Therefore, this score returns false positives and negatives into reason to assault a strength between recall and precision.

F1-Score = (2 * Precision * Recall)/(Precision + Recall) (4)

CNN Implementation

Stephen *et al.,* (2019) have developed procedures for training the CNN model. This study will serve as the foundation for the method we use to train this model.

The feature extractors and the classifier are the two main components of the CNN model's structure. The output of the immediately preceding layer is supplied as an input to the successive layers in the feature extraction layer, which each layer, in turn, uses as an input. As indicated in Table 2, the convolution, maximum pooling, and classification layers are merged to form the CNN architecture. The conv33, 32, conv33, 64, and conv33, 128, a max-pooling layer of size 22, and the activation function, which is a hyperparameter for CNN between them, make up the feature extractors. With an input image of size 224224, we acquired feature maps of the following sizes for the convolution and max-pooling procedures: 224224, 112112, 5656, and 2828 and 128 for the pooling operations, respectively. It is important to remember that each layer's plane was created by joining one or more planes from earlier layers. The classifier is at the end of the convolutional neural network (CNN) model. It is a dense layer, often called an artificial neural network (ANN). Like any other classifier, this one uses individual features (vectors) to carry out computations. As a result, a 1D feature vector is created from the output of the feature extractor (CNN component) for the classifiers. The result of the convolution operation is flattened in this phase, known as "flattening," to produce a single, lengthy feature vector that the dense layer will use for its final classification procedure. A flat layer with two thick layers of 1000 and 3, respectively, makes up the categorization layer. The CNN's performance hyperparameter and the detection functions' softmax activation function are the activation functions that operate between the two dense layers. The CNN's non-linearity is controlled by a hyperparameter known as the activation function of each layer's node. These operations restrict the output to a specific band or threshold.

We evaluated the effect of hyperparameters on a deep learning model for detection of potato leaf disease in the agricultural sector using the methodology of (Dalli (2022).

Activation Function

There are four main activation functions widely used in CNN; the description is presented below: Each neuron node in a neural network will take the output value from the layer before as its input value and pass it to the layer after. The nodes in the input layer must input the feature value to the following output layer. The relationship between the input values of the neuron nodes in one layer and the output values of those nodes in another layer is represented in a multilayer neural network by an activation function (Y. Zhang *et al.,* 2021). Similar to an activation function, a nonlinear function enables neural networks to achieve improved representation performance and get around the linear function's finite approximation limitation.

The rectified linear unit (ReLU) function was the most popular function used in CNN architecture. The threshold of the ReLU function is 0, and it has a real value. Figure 2 illustrates how this function replaces negative numbers with zero. In the scenario where the input values are negative, the ReLU activation function can be utilized. The model is set up so that it can alter the real positive scaled values.

f(x)=max(0,x) (6)

The disadvantage of ReLU is that it does not converge to the minima when the gradient becomes 0 for negative values, which will cause a dead neuron during backpropagation. Leaky ReLU, which permits a little negative value during the backpropagation if we have a dead ReLU problem, can solve this issue. The neuron will finally be brought down and activated as a result. Figure 3 illustrates a leaky ReLU.

f(x) = max(0.01*x,x) (7)

As depicted in Figure 3, the hyperbolic tangent (tanh) value is taken into account to be a real number that falls between -1 and 1. The tanh expression is based on negative values not being frequently scaled onto commonly used functions as zero.

$$f(x) = \frac{e^{x} - e^{-x}}{e^{x} - e^{-x}} \quad (8)$$

Optimizer

On a particular dataset, the model learns by contrasting the actual label of the input (available in the training set) with the predicted label and minimizing the cost function. Theoretically, the model has learned the dataset successfully if the cost function is zero. However, in order to minimize a cost function, an optimization procedure is required. In order to reduce the cost function, various optimization strategies are discussed in the following section.

Stochastic gradient descent (SGD) is also referred to as "online training" since it enables updating the network weights for each training image. We discovered that this strategy is significantly faster and is available online from numerous literature reviews. As a result, the SGD technique successfully carries out a parameter modification in response to each training example taken into account, as stated in (Halgamuge *et al.,* 2020).

The work of Zhang (2018) provides a description of the estimation of adaptive moments (Adam). This first-order optimization strategy uses gradients, stochastic objective functions, and adaptive lower-order moment estimations.

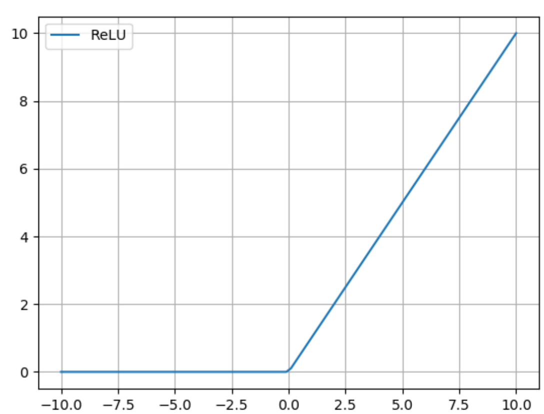

Figure 2: ReLU function.

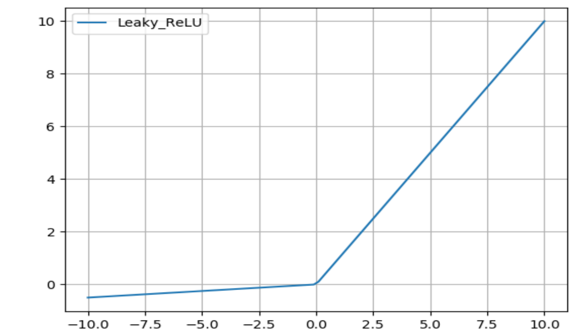

Figure 3: Leaky ReLU function.

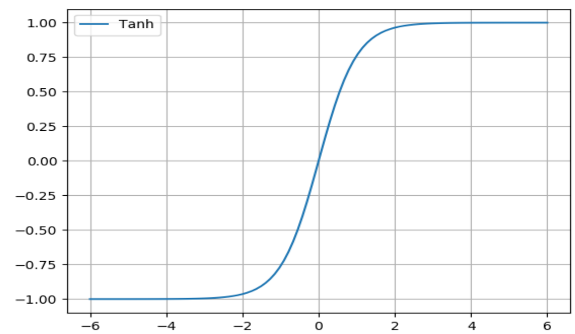

Figure 4: Tanh function.

The adaptive gradient algorithm (AdaGrad) (N. Zhang *et al.,* 2018) permits a decrease in the learning rate by increasing the numerator. Alternately, we can say that the gradient-based optimization technique has been used, which also adapts the learning rate that would then predict the parameters; it can undoubtedly give us smaller updates to the data (low learning rates) for all of the parameters that we would consider to be relevant to it. This is due to the fact that the AdaGrad technique uses highly frequent features in the model, and very large updates (high learning rates) consider a parameter that would be important to the rare features, thus, it would undoubtedly operate well with extra data in the network.

The "root mean square propagation" (RMSProp) technique is another term for an advanced AdaGrad modification that controls the learning rate at a rapid decline level. It is frequently compared to the Adadelta approach. Despite this, the Adadelta method unquestionably uses the RMSProp method of parameter changes carried out in the rule of numerator's updating. The AdaMax method, which is based on the infinite standard, is a variation of the Adam method (Z. Zhang, 2018).

Results

Effect of different batch sizes

In the first experiment, we tested the most commonly used batch size combinations for CNN. The batch sizes used in this experiment were BS = [16, 32, 64, 128, *256]; a*n SGD optimizer and a 0.001 learning rate were used. For consistency of results and due to the size of the dataset, the number of epochs was fixed at 100. Table 2 shows the effect of different batch sizes. The lowest accuracy was achieved for larger batches (BS_256). The highest performance was obtained by using a smaller batch size; the smaller the batch size, the higher the performance.

The highest overall accuracy achieved during the experiments was due to the batch size of 32. Our results agree with the ones obtained by Masters and Luschi (2018), where the authors stated that smaller batch sizes should be used. While the use of large batch size values is not recommended in our study, the results of Radiuk (2017) match our findings on the batch size. Finally, Bengio (2012) suggested that 32 is a good default value for the batch size. While this is corroborated by our experiments (in which a batch size of 32 provided good results), the best performance was achieved with a batch size of 32. The comparative results of the loss and accuracy obtained from the batch size effects are presented in Figures 5 and 6, respectively.

Effects of different optimizers

We continue the second experiment using a smaller batch size (BS_32), as we saw in the first experiment, that smaller batch sizes have higher performance. The second experiment focuses on the second research topic, that is, the effect of different optimizers on the CNN model's performance. When we talk about some optimizers, like SGD, AdaGrad, and derivatives, Other algorithms that we can take into account include the Adam algorithm and the root mean square propagation (RMSProp) algorithm. Table 3 shows the effect of different optimizers. For consistency of results and due to the size of the dataset, the number of epochs was fixed at 100. For the RMSProp optimizer, the lowest accuracy was achieved. The highest performance was obtained using the AdaGrad optimizer, demonstrating that AdaGrad optimizer had the highest performance.

The highest overall accuracy achieved during the experiments was by the AdaGrad optimizer. Our results agree with those obtained by Iqbal *et al.,* (2021), where the authors stated that the AdaGrad optimizer should be used. Finally, Vidushi *et al.,* (2021) suggested that AdaGrad is a good default optimizer. While this is corroborated by our experiments (in which an AdaGrad optimizer provided good results), the best performance was achieved with an optimizer. The comparative results of the loss and accuracy obtained from the optimizer effects are presented in Figures 7 and 8, respectively.

Effect of different activation functions

For the final test, we used the smallest batch size and AdaGrad optimizer, which have high performance, as seen in the first and second experiments. The final question of the study is to determine the effect of different activation functions on the performance of the CNN model. The activation functions used in this experiment are ReLU, Leaky ReLU, Tanh, and PReLU. For consistency of results and due to the size of the database, the number of epochs was limited to 100. Table 4 shows the effect of different activation functions. Tanh activation function was achieved with the lowest accuracy. The leaky ReLU activation function provides the best performance.

The highest overall accuracy achieved during the experiments was when using the leaky ReLU activation function. Our results agree with the one obtained by Nayef *et al.,* (2022), where the authors stated that the leaky ReLU activation function should be used. Finally, Mujhid *et al.,* (2022) suggested that leaky ReLU is a good default activation function. While this is corroborated by our experiments, in which a leaky ReLU activation function provided good results, the best performance was achieved with a leaky ReLU activation function. The comparative results of the loss

Table 2: shows the effect of different batch sizes.

Batch Sizes	Accuracy	Precision	Precision	F1-Score
16	78.0%	78.3%	78.0%	78.3%
32	89.3%	91.3%	89.3%	89.3%
64	89.1%	91.0%	89.3%	89.0%
128	80.5%	85.6%	80.6%	80.6%
256	77.7%	81.0%	77.6%	75.6%

Table 3: shows the effect of different optimizers.

Activation function	Accuracy	Precision	Recall	F1-Score
ReLU	95.9%	95.3%	96.0%	96.0%
Leaky ReLU	98.0%	97.3%	97.6%	97.6%
Tanh	93.9%	94.3%	94.0%	94.0%
PReLU	97.7%	97.6%	97.6%	97.6%

Table 4: shows the effect of different activation functions.

Optimizer	Accuracy	Precision	Recall	F1-Score
SGD	96.9%	97.0%	97.0%	97.0%
Adam	84.8%	88.0%	84.6%	84.6%
AdaGrad	97.9%	98.0%	98.0%	98.0%
Adamax	90.0%	91.3%	90.0%	90.3%
RMSProp	77.0%	78.3%	77.3%	76.0%

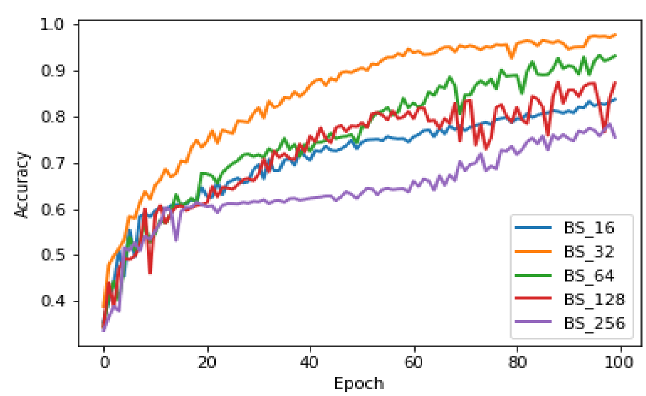

Figure 5: Comparative results of batch sizes accuracy

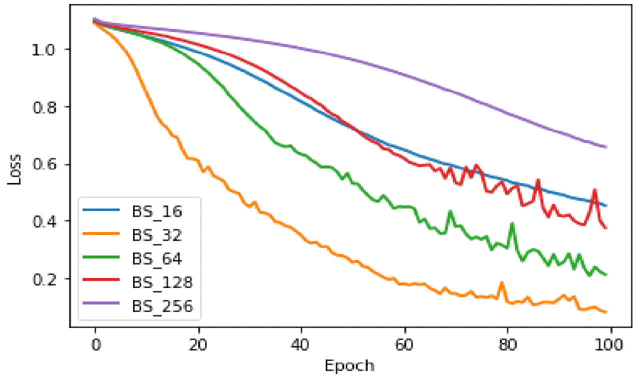

Figure 6: Comparative results of batch sizes loss

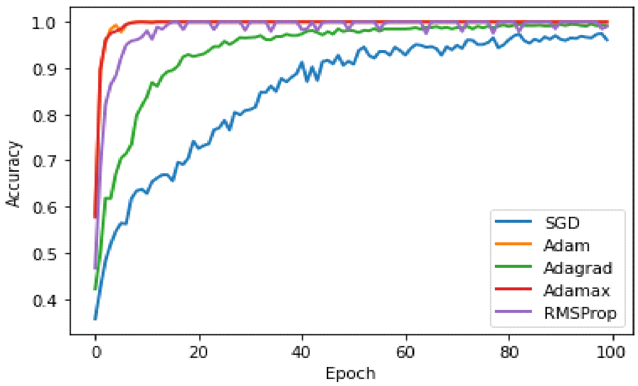

Figure 7: Comparative results of the accuracy

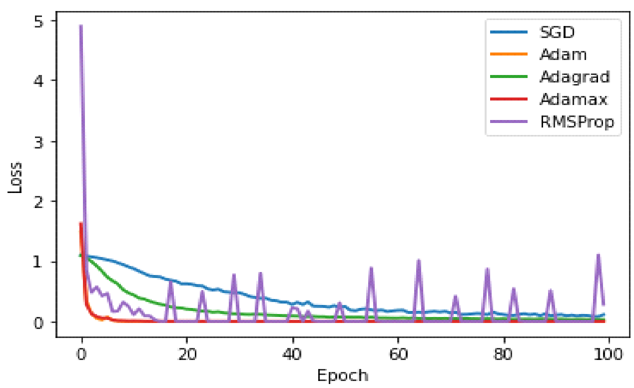

Figure 8: Comparative results of the loss

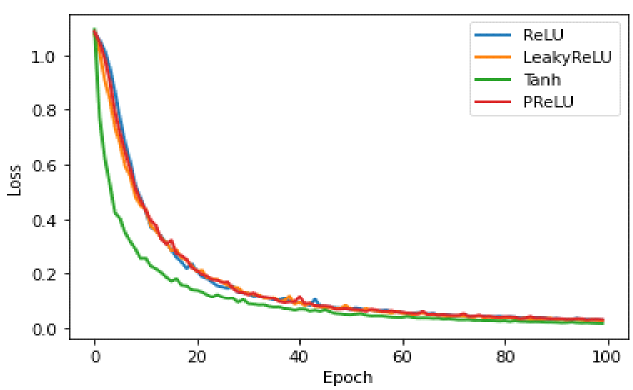

Figure 9: Comparative results of the loss

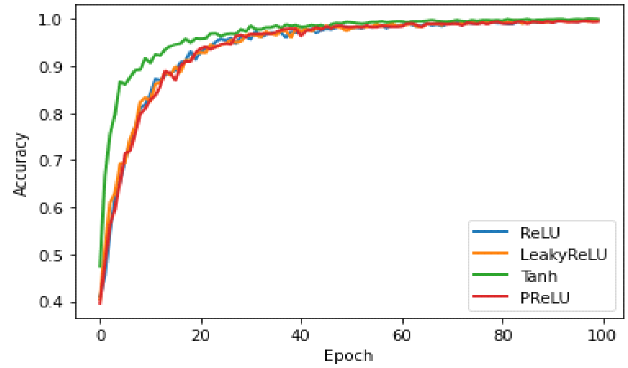

Figure 10: Comparative results of the accuracy

and accuracy obtained from the activation function effects are presented in Figures 9 and 10, respectively.

Analysis and Discussion

The data from the first experiment indicate that the best results are obtained by using a CNN with a smaller batch size of (BS_32). A CNN model can effectively detect leaf images in this situation. The second experiment shows that when AdaGrad is used as the training algorithm, we get better results, but when RMSProp is used, we get worse results. Finally, using the Leaky ReLU function as an activation function, we can determine that the model can converge to the minimum if the efficiency is 0 for negative values, which was the original goal of the study. Furthermore, since they allowed a small negative value during the backpropagation, it was shown that the neuron eventually descends and activates as a result. When the Leaky ReLU activation function is used in the CNN architecture, CNN provides the best throughput.

A crucial aspect of the new study is its theoretical and applied contribution. First, a few studies in the agricultural field examine how the hyperparameter setting affects the effectiveness of deep neural networks. It would not be a stretch to say that our research would undoubtedly contribute to theoretical data, particularly by laying the groundwork for a better understanding of the influence work of hyperparameter configurations. In contrast to earlier studies, its purpose is to make it easier for us to understand the effects of several commonly used activation functions in the model when it is used to detect potato leaf disease using CNN. We looked into how different activation functions, types of optimizers, and batch sizes affect CNN performance when it comes to detecting potato leaf disease.

On a more practical level, this research establishes the foundation for producing useful heuristic information that can assist researchers in employing deep learning techniques in the agricultural sector to detect potato leaf disease. This research contributes to the effectiveness of hyperparameter tweaking when training CNN to detect potato leaf disease.

Despite the encouraging findings, the study had some limitations. To begin with, the dataset used for the study was too small to train the CNN model. This limitation, however, provides an opportunity for additional research to assess the repeatability of the findings, and we plan to eventually summarize findings from larger databases collected from a variety of agricultural supplies across agricultural sectors. All three goals of the study outlined in Section 1 were achieved.

Conclusion

The effects of various hyperparameter configurations on the performance of a CNN were investigated in this study. Three experiments were designed to confirm this. First, we look at how different batch size combinations affect the CNN architecture. Secondly, we investigate the performance of CNN by using various optimizers. Finally, we investigate the CNN model's performance using various activation functions.

In this paper, the data we have mentioned where the first experiment shows that the model developed using small batch size (BS_32) has high performance and the widely used softmax is applied to the output layer of the network, is a sign that the CNN potato leaf detection model is definitely working very well in the agricultural sector. Similarly, the second experimental data shows that the AdaGrad optimizer performed well if it was selected as the training method in the study. Finally, the last experiment showed us that using the leaky RELU function with CNN in the model definitely works well in the agricultural sector.

In the future, we expect to use our approach to detect more varieties of potato leaf and other types of plant diseases, evaluate further optimization algorithms, and use more data augmentation strategies. In addition, we plan to expand optimization for a few more hyperparameters. Meanwhile, the trained model can be flexibly combined with mobile devices to allow agricultural producers to make fast and fair decisions about potato disease knowledge.

References

Albawi, S., Mohammed, T. A., & Al-Zawi, S. (2017). Understanding of a convolutional neural network. *2017 International Conference on Engineering and Technology (ICET)*, pp. 1–6.

Asif, M. K. R., Rahman, M. A., & Hena, M. H. (2020). CNN based Disease Detection Approach on Potato Leaves. *2020 3rd International Conference on Intelligent Sustainable Systems (ICISS)*, 428–432. https://doi.org/10.1109/ICISS49785.2020.9316021

Baskar, P., Joseph, M. A., Narayanan, N., & Loya, R. B. (2013). Experimental investigation of oxygen enrichment on performance of twin cylinder diesel engine with variation of injection pressure. *2013 International Conference on Energy Efficient Technologies for Sustainability*, 682–687.

Bengio, Y. (2012). Practical recommendations for gradient-based training of deep architectures. In *Neural networks: Tricks of the trade* (pp. 437–478). Springer.

Burgos, G., Zum Felde, T., Andre, C., & Kubow, S. (2020). The potato and its contribution to the human diet and health. In *The potato crop* (pp. 37–74). Springer, Cham.

Chakraborty, K. K., Mukherjee, R., Chakroborty, C., & Bora, K. (2022). Automated recognition of optical image based potato leaf blight diseases using deep learning. *Physiological and Molecular Plant Pathology*, *117*, 101781. https://doi.org/https://doi.org/10.1016/j.pmpp.2021.101781

Dalli, A. (2022). Impact of Hyperparameters on Deep Learning Model for Customer Churn Prediction in Telecommunication Sector. *Mathematical Problems in Engineering*, *2022*.

Devaux, A., Goffart, J.-P., Petsakos, A., Kromann, P., Gatto, M., Okello, J., Suarez, V., & Hareau, G. (2020). Global food security, contributions from sustainable potato agri-food systems. In *The potato crop* (pp. 3–35). Springer, Cham.

Elsharif, A. A., Dheir, I. M., Mettleq, A. S. A., & Abu-Naser, S. S. (2020). Potato Classification Using Deep Learning. *International Journal of Academic Pedagogical Research (IJAPR)*, *3*(12).

Halgamuge, M. N., Daminda, E., & Nirmalathas, A. (2020). Best optimizer selection for predicting bushfire occurrences using deep learning. *Natural Hazards*, *103*(1), 845–860.

Hou, C., Zhuang, J., Tang, Y., He, Y., Miao, A., Huang, H., & Luo, S. (2021). Recognition of early blight and late blight diseases on potato leaves based on graph cut segmentation. *Journal of Agriculture and Food Research*, *5*, 100154. https://doi.org/https://doi.org/10.1016/j.jafr.2021.100154

Hughes, D., & Salathé, M. (2015). An open-access repository of images on plant health to enable the development of mobile disease diagnostics. *ArXiv Preprint ArXiv:1511.08060*.

Iqbal, I., Odesanmi, G. A., Wang, J., & Liu, L. (2021). Comparative Investigation of Learning Algorithms for Image Classification with Small Dataset. *Applied Artificial Intelligence*, *35*(10), 697–716. https://doi.org/10.1080/08839514.2021.1922841

Islam, F., Hoq, M. N., & Rahman, C. M. (2019). Application of Transfer Learning to Detect Potato Disease from Leaf Image. *2019 IEEE International Conference on Robotics, Automation, Artificial-Intelligence and Internet-of-Things (RAAICON)*, 127–130. https://doi.org/10.1109/RAAICON48939.2019.53

Islam, M., Dinh, A., Wahid, K., & Bhowmik, P. (2017). Detection of potato diseases using image segmentation and multiclass support vector machine. *2017 IEEE 30th Canadian Conference on Electrical and Computer Engineering (CCECE)*, 1–4.

Johnson, J., Sharma, G., & Srinivasan, S. (2021). *Enhanced field-based detection of potato blight in complex backgrounds using deep learning. Plant Phenomics 2021: 1–13*.

Kagaya, H., Aizawa, K., & Ogawa, M. (2014). Food detection and recognition using convolutional neural network. *Proceedings of the 22nd ACM International Conference on Multimedia*, 1085–1088.

Kamilaris, A., & Prenafeta-Boldú, F. X. (2018). Deep learning in agriculture: A survey. *Computers and Electronics in Agriculture*, *147*, 70–90.

Khalifa, N. E. M., Taha, M. H. N., El-Maged, A., Lobna, M., & Hassanien, A. E. (2021). Artificial intelligence in potato leaf disease classification: a deep learning approach. In *Machine Learning and Big Data Analytics Paradigms: Analysis, Applications and Challenges* (pp. 63–79). Springer.

Kietzmann, T. C., McClure, P., & Kriegeskorte, N. (2018). Deep neural networks in computational neuroscience. *BioRxiv*, 133504.

Kukreja, V., Baliyan, A., Salonki, V., & Kaushal, R. K. (2021). Potato Blight: Deep Learning Model for Binary and Multi-Classification. *2021 8th International Conference on Signal Processing and Integrated Networks (SPIN)*, 672–967. https://doi.org/10.1109/SPIN52536.2021.9566079

Lee, T.-Y., Lin, I.-A., Yu, J.-Y., Yang, J., & Chang, Y.-C. (2021).

High Efficiency Disease Detection for Potato Leaf with Convolutional Neural Network. *SN Computer Science*, *2*(4), 297. https://doi.org/10.1007/s42979-021-00691-9

Lee, T.-Y., Yu, J.-Y., Chang, Y.-C., & Yang, J.-M. (2020). Health detection for potato leaf with convolutional neural network. *2020 Indo–Taiwan 2nd International Conference on Computing, Analytics and Networks (Indo-Taiwan ICAN)*, 289–293.

Masters, D., & Luschi, C. (2018). Revisiting small batch training for deep neural networks. *ArXiv Preprint ArXiv:1804.07612*.

Mujhid, A., Surono, S., Irsalinda, N., & Thobirin, A. (2022). Comparison and Combination of Leaky ReLU and ReLU Activation Function and Three Optimizers on Deep CNN for COVID-19 Detection. In *Fuzzy Systems and Data Mining VIII* (pp. 50–57). IOS Press.

Nayef, B. H., Abdullah, S. N. H. S., Sulaiman, R., & Alyasseri, Z. A. A. (2022). Optimized leaky ReLU for handwritten Arabic character recognition using convolution neural networks. *Multimedia Tools and Applications*, *81*(2), 2065–2094. https://doi.org/10.1007/s11042-021-11593-6

Patil, P., Yaligar, N., & Meena, S. M. (2017). Comparision of Performance of Classifiers - SVM, RF, and ANN in Potato Blight Disease Detection Using Leaf Images. *2017 IEEE International Conference on Computational Intelligence and Computing Research (ICCIC)*, 1–5. https://doi.org/10.1109/ICCIC.2017.8524301

Radha, N., & Swathika, R. (2021). A Polyhouse: Plant Monitoring and Diseases Detection using CNN. *2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS)*, pp. 966–971. https://doi.org/10.1109/ICAIS50930.2021.9395847

Radiuk, P. M. (2017). *Impact of training set batch size on the performance of convolutional neural networks for diverse datasets*.

Rashid, J., Khan, I., Ali, G., Almotiri, S. H., AlGhamdi, M. A., & Masood, K. (2021). Multi-Level Deep Learning Model for Potato Leaf Disease Recognition. In *Electronics* (Vol. 10, Issue 17). https://doi.org/10.3390/electronics10172064

Saeed, Z., Khan, M. U., Raza, A., Sajjad, N., Naz, S., & Salal, A. (2021). Identification of Leaf Diseases in Potato Crop Using Deep Convolutional Neural Networks (DCNNs). *2021 16th International Conference on Emerging Technologies (ICET)*, 1–6. https://doi.org/10.1109/ICET54505.2021.9689807

Sharma, N., Jain, V., & Mishra, A. (2018). An analysis of convolutional neural networks for image classification. *Procedia Computer Science*, *132*, 377–384.

Singh, A., & Kaur, H. (2021). Potato Plant Leaves Disease Detection and Classification using Machine Learning Methodologies. *IOP Conference Series: Materials Science and Engineering*, *1022*(1), 12121. https://doi.org/10.1088/1757-899x/1022/1/012121

Stephen, O., Sain, M., Maduh, U. J., & Jeong, D. (2019). *An Efficient Deep Learning Approach to Pneumonia Classification in Healthcare*. *2019*.

Sujatha, R., Chatterjee, J. M., Jhanjhi, N. Z., & Brohi, S. N. (2021). Performance of deep learning vs machine learning in plant leaf disease detection. *Microprocessors and Microsystems*, *80*, 103615.

Tiwari, D., Ashish, M., Gangwar, N., Sharma, A., Patel, S., & Bhardwaj, S. (2020a). Potato leaf diseases detection using deep learning. *2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS)*, 461–466.

Tsedaley, B. (2014). Late blight of potato (Phytophthora infestans) biology, economic importance and its management approaches. *Journal of Biology, Agriculture and Healthcare*, *4*(25), 215–225.

Vidushi, Agarwal, M., Rajak, A., & Shrivastava, A. K. (2021). Assessment of Optimizers impact on Image Recognition with Convolutional Neural Network to Adversarial Datasets. *Journal of Physics: Conference Series*, *1998*(1), 12008. https://doi.org/10.1088/1742-6596/1998/1/012008

Yellareddygari, S. K. R., Taylor, R. J., Pasche, J. S., & Gudmestad, N. C. (2019). Quantifying control efficacy of fungicides commonly applied for potato early blight management. *Plant Disease*, *103*(11), 2821–2824.

Zhang, N., Lei, D., & Zhao, J. F. (2018). An improved Adagrad gradient descent optimization algorithm. *2018 Chinese Automation Congress (CAC)*, 2359–2362.

Zhang, Y., Zhu, R., Chen, Z., Gao, J., & Xia, D. (2021). Evaluating and selecting features via information theoretic lower bounds of feature inner correlations for high-dimensional data. *European Journal of Operational Research*, *290*(1), 235–247.

Zhang, Z. (2018). Improved adam optimizer for deep neural networks. *2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS)*, pp. 1–2.

Zhong, Y., & Zhao, M. (2020). Research on deep learning in apple leaf disease recognition. *Computers and Electronics in Agriculture*, *168*, 105146. https://doi.org/https://doi.org/10.1016/j.compag.2019.105146