Journal of Applied Data Sciences
Vol. 5, No. 2, May 2024, pp. 532-545

Teachable Machine: Optimization of Herbal Plant Image Classification Based on Epoch Value, Batch Size and Learning Rate

Edwin Ariesto Umbu Malahina[1,*], Mardhalia Saitakela[2], Semlinda Juszandri Bulan[3], Marinus Ignasius Jawawuan Lamabelawa[4], Yohanes Suban Belutowe[5]

[1,2,3,4,5] *STIKOM Uyelindo Kupang, Jl. Perintis Kemerdekaan I, Kupang City and 85111, Indonesia*

(Received: February 3, 2024; Revised: March 2, 2024; Accepted: April 13, 2024; Available online: May 31, 2024)

Abstract

Herbal plants are a source of natural materials used in alternative medicine and traditional therapies to maintain health. The purpose of this research is to develop an intelligent system application that is able to assist people in independently detecting herbal plants around them, provide education, and most importantly, find the optimal value based on certain parameters. This research uses several values for the parameters studied, namely the epoch value which varies between 10, 50, 100, 250, 750, and 1000; the batch size value which varies between 16, 32, 64, 128, 256, and 512; and the learning rate value which varies between 0.00001, 0.0001, 0.001, 0.01, 0.1, and 1. A total of 10,000 training data samples (1,000 samples in 10 classes) were used in Teachable Machine. The method used is to utilize the TensorFlow framework in the Teachable Machine service to train image data. This framework provides convolutional neural networks (CNN) algorithms that can perform image classification with a high degree of accuracy. The test results for more than three months showed that the highest optimal value was achieved at the 50th epoch value, with a learning rate of 0.00001, and a batch size of 32, which resulted in an accuracy rate between 98% and 100%. Based on these results, a mobile web-based intelligent system application service was developed using the TensorFlow framework in Teachable Machine. This application is expected to be widely implemented for the benefit of the community. However, the challenges and limitations in training this test data are the large number of data classes that will be very good so that machine learning can learn to recognize objects but will take hours to train, then the training image object data has a clean background from other objects so that when tested it is not detected and influenced as another object or can result in a decrease in the percentage value.

Keywords: Teachable Machine, Herbal Plants, Convolutional Neural Networks, Image Optimization.

1. Introduction

Digital image detection classification technology with CNN has become the gold standard in the field of pattern recognition and image processing [1] ,[2]. CNN is a type of artificial neural network specifically designed to recognize visual patterns in image data. [3], [4]. The CNN structure allows the network to automatically and adaptively learn hierarchical spatial features from image data [5]. This is done through the use of convolutional layers, where each neuron receives input from a subset of neurons in the previous layer, rather than all neurons as in a regular artificial neural network [6], [7]. With the ability to learn from large amounts of data and extract important features from images [8], [9], CNNs have opened up new opportunities in the field of digital image detection and classification, where these algorithms are able to learn important features from images and use these features to classify objects [10].

In other research, herbal plants have been studied by previous researchers with various case studies, such as the classification of herbal plants according to the type of disease identified through the texture, shape and color of herbal plant diseases [11]. Identification of herbal medicinal plant names, scientific names and remedies on leaf objects [12], [13], so that the identification of herbal plants is able to increase the accuracy of herbal plant recognition with CNN well which is able to achieve a recognition rate above 95% near perfect depending on the quality of the object and other approaches [14], [15], because some herbal plants have a surpa level of similarity such as turmeric and ginger, so a good classification technique approach is needed such as CNN and other algorithms [16].

***Corresponding author: Edwin Ariesto Umbu Malahina (edwinariesto@gmail.com)**
DOI: https://doi.org/10.47738/jads.v5i2.206

In the following study, researchers took the topic of classification detection of several herbal plant image objects in Indonesia. Indonesia is known for its rich biodiversity, including various types of herbal plants [17], [18], [19]. These plants have been used for centuries in traditional medicine and many of them are now being researched for potential human health benefits [20], [21], [22], [23]. Some examples of herbs popular in Indonesia include Turmeric, Ginger, and Temulawak, all of which are known to have anti-inflammatory and antioxidant properties [24], [25]. In addition, plants such as Sambiloto and Gotu Kola are also used in traditional medicine for various health conditions [26], [27], [28], [29].

Teachable Machine is a platform that allows researchers to train their own machine learning models without the need to write code [30], [31], [32], that can produce accuracy up to 90% to 100% [33], [34], [35]. In the context of herbal plant image detection, this can be very useful. For example, users can collect images of different types of herbal plants and use Teachable Machine to train a model that can recognize and classify these plants based on images of these objects. This process can be optimized by adjusting parameters such as the number of epochs (iterations through the entire dataset), batch size (the number of samples processed before the model is updated), and learning rate (how fast the model learns) [36], [37], [38], [39], [40]. Furthermore, this model can be integrated into a mobile web application, allowing users to identify herbs by simply taking an image of the object.

The results of this process can vary greatly depending on the quality of the data and the number of datasets used to train the model and the parameters chosen for optimization [41], [42]. However, with a large enough dataset and proper parameter adjustments, it is possible to achieve a high level of accuracy [43], [44], [45] in the classification of herbal plants. This can open up new opportunities in the field of herbal medicine, allowing practitioners and researchers to easily and quickly identify plants and utilize their health benefits. In addition, the integration of this model into mobile web applications can make this technology more accessible to the general public, facilitating the spread of knowledge about herbal plants and their potential health benefits.

2. Methods and Algorithms

The following discussion will outline the meaning of the methods and algorithms used in this research, starting from the explanation of transfer learning, Teachable Machine, CNN, Tensorflow.js and the three main measures in machine learning optimization, namely epoch, learning rate and batch size as follows.

2.1. Transfer Learning and Tensorflow.js

Transfer learning is a technique in machine learning where knowledge learned from a task is reused to improve performance on a related task. For example, in image classification, knowledge gained when learning to recognize cars can be applied when trying to recognize trucks [46]. Transfer learning is used in the Teachable Machine service where this technique utilizes knowledge from previously trained models, eliminating the need to start from scratch. With this technique, the user needs fewer examples of new items to classify, saving time and resources. The training process is usually faster as it only needs to train the last few layers of the model architecture instead of the entire network. This technique is particularly suitable for web browser environments, where resources may vary by device, but also have direct access to sensors for easy data acquisition [47]. The transfer learning formula is as follows:

$$Y = f(X, W) \quad (1)$$

Where, Y is the output of a pre-trained model, X the input of a pre-trained model, W the parameters of a pre-trained model, and f a function that connects the input and output of a pre-trained model.

In the transfer learning used by Teachable Machine, there is the Tensorflow.js library, where TensorFlow.js is an open-source machine learning library that can be used anywhere JavaScript can be used. It is based on the original TensorFlow library written in Python, and aims to provide developers with the same experience and a set of APIs that are compatible with the JavaScript ecosystem [48].

TensorFlow.js supports various execution environments, including CPU, GPU, and WebGL. This ensures compatibility and optimal performance. Currently, TensorFlow.js supports:

1) CPU: for efficient performance on mobile and desktop devices

2) GPU: for faster performance on GPU-enabled devices
3) WebGL: for optimal performance in web applications [48]

2.2. Teachable Machine

Teachable Machine is a web-based tool that makes it easy to build machine learning models, making it fast and accessible to everyone, including educators, artists, students, innovators, and developers of all kinds. Teachable Machine is designed in such a way that it requires no prior knowledge of machine learning. With Teachable Machine, users can train computers to recognize images, sounds, and poses without the need to write complex machine learning code. Once the model is trained, users can use it in various projects, websites, apps, and more [49]. The following is the TM display, which can be seen in figure 1, and the image recognition process stage can be seen in figure 2.

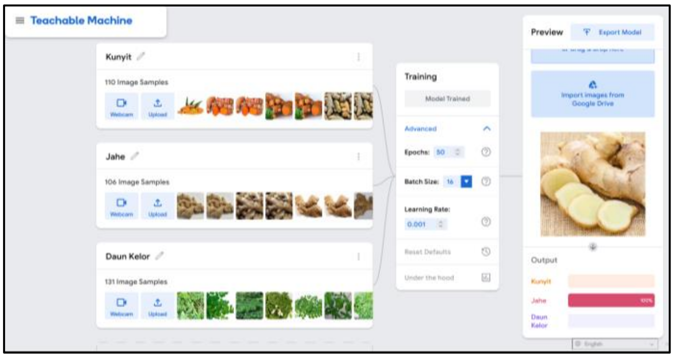

Figure 1. Teachable Machine object detection image website

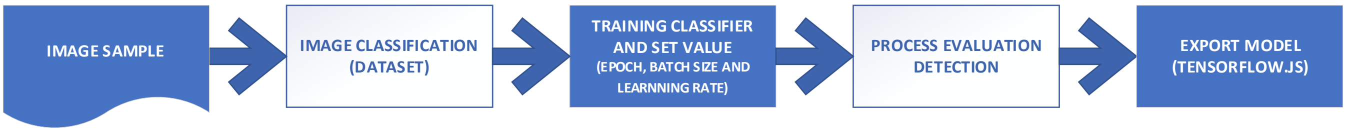

Figure 2. Image recognition stage (Teachable Machine)

To classify and test object class data on the Teachable Machine service, users can visit the site www.teachablemachine.withgoogle.com. When the user logs in and logs in using a Google account, it will display object detection options such as Audio, Image and Pose, please select Image to perform classification, users can also prepare image data either recorded directly in Teachable Machine using a webcam or upload image files that are owned and create class names in Teachable Machine as labels to be recognized. Then if the image data has been uploaded and made a class for each image object, please click the Training button or first determine the Epoch value, Batch Size or learning rate to get the best results. If the training has been completed, the user will test the data either in real-time or upload a test image to be trained and recognized in Teachable Machine, if you are satisfied with the percentage results and detection accuracy, the user can export the model to the types of programming languages whether web-based, desktop or mobile, because Teachable Machine provides source code for developers to develop applications based on the Tensorflow framework provided in Teachable Machine.

2.3. Convolutional Neural Networks

Google Teachable Machine, powered by convolutional neural networks, provides an easy-to-use yet extremely powerful tool for object classification tasks [50]. CNN consists of several specialized display layers namely convulution layers, pooling layers and fully connected, the architecture can be seen as follows in figure 3.

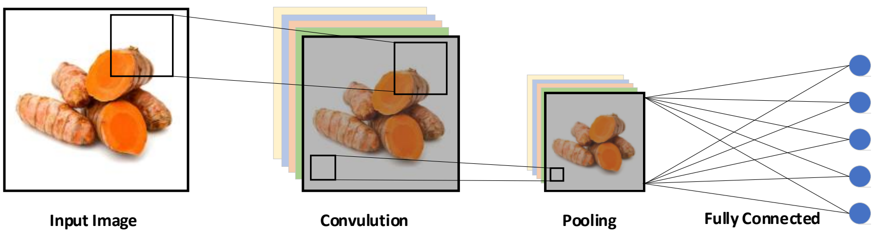

Figure 3. CNN Architecture

ISSN 2723-6471

Convolution layers are a Generalized Linear Model (GLM) for an image [51], The purpose of the convolution layer is to transform the raw input image into a more meaningful representation that is easily understood by the network, and there are also two important components in convolution layers: strides, which is a parameter that determines how far the filter moves on the image, and padding, an operation that increases the size of the input data. The convolution layer uses filters that perform convolution operations while scanning the input I with respect to its dimensions. Hyperparameters include filter size F and stride S. The resulting output O is called a feature map or activation map. [52]. The convolution layers can be seen in figure 4.

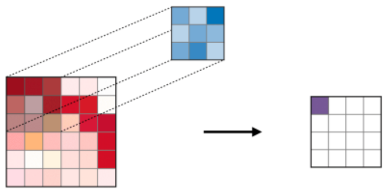

Figure 4. Convolution layers [52]

Pooling layers aims to reduce the resolution of image features while retaining important spatial information [53], as well as preventing overfitting which occurs when the model learns too much detail on the training data and its performance degrades on the test data [54]. As for the pooling layer, there are two main components: max pooling which aims to take the neuron that has the highest activation value in the local receptive field (grid cell), and average pooling which will produce output by taking the average value of all activations in each receptive field. The Max pooling and average pooling can be seen in figure 5.

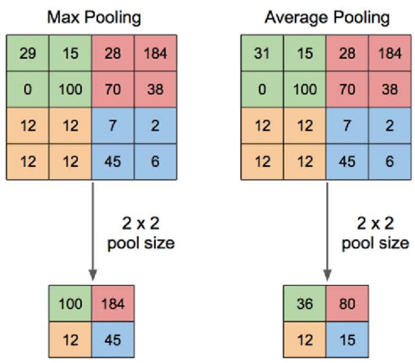

Figure 5. Max pooling and average pooling [55]

Fully connected is at the end of the CNN architecture which functions as the final classification of each network of previously interconnected neurons [56], so the purpose of the Fully connected layer is to transform the raw image input into a more meaningful representation that is easily understood by the network. The Fully connected layers can be seen in figure 6.

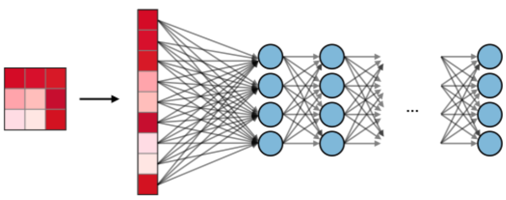

Figure 6. Fully connected [52]

The classic CNN is still used today by many researchers because the accuracy of image recognition can be very accurate depending on several influencing factors. However, the development of object classification models has emerged with various advantages and efficiencies that need to be tested to be able to distinguish various new algorithms such as VGGNet, ResNet, GoogleNet, and AlexNet.

1) AlexNet, known for its two parallel CNN paths trained on two GPUs with cross-connections.

2) GoogleNet or Inception, is an Inception module that allows the use of deeper and wider networks without overfitting.
3) VGGNet, with its simple architecture, uses only a very small 3x3 convolution kernel but has many layers.
4) ResNet, is a very deep network with more than 100 layers and uses “Skip connections” and “batch normalization” which helps train deep layers without hampering performance. These four algorithms have made significant contributions in the field of Deep Learning.

2.4. Epoch, Learning Rate and Batch Size

In deep learning, there are three parameters that are important in measuring the training dataset: epoch, learning rate and batch size [57]. The epoch value in the context of Teachable Machine, one epoch means that the model has processed all the samples in the training dataset and updated the parameters based on the calculated loss. In the context of Teachable Machine, Learning Rate governs how often the neural network updates the learned knowledge. In the context of Teachable Machine, Batch Size controls the number of training samples that must be processed before the internal parameters of the model are updated [58]. It is important to find the detection optimization level in Teachable Machine to get an effective training dataset from the three parameters [59].

The epoch value in Teachabel Machine ranges from 1 to infinity, but it should be noted that the larger the number of epoch values, the more overfitting the model will be, i.e., the training model becomes too specific to the training data and its performance will degrade on new data. In general, a larger number of epochs will cause the machine learning model to learn more accurately, but take longer. A smaller number of epochs will cause the machine learning model to learn faster, but less accurately. Therefore, it is important to experiment with various number of epochs to find the right value for the machine learning model.

While the batch size value in Teachabel Machine ranges from 16; 32; 64; 128; 256 and 512, the selection of batch size will affect the training time, time per epoch, and model quality, so it is necessary to experiment with various batch sizes to find the best size. Batch size is also the number of data samples used for one iteration of machine learning model training. A smaller batch size will cause the machine learning model to learn more accurately, but it may also cause the machine learning model to take longer to train.

Then for the learning rate is not determined exactly but can start at size 1 (largest) to 0.00001 (smallest), the smaller the learning rate value, the slower it learns to process training data, but prevents overfitting. Therefore, a good learning rate value for CNN Teachable Machine is around 0.0001 to 0.00001, which is recommended.

To prove the use of the three parameter values above (epoch, batch size and learning rate) researchers have summarized from various sources for the minimum and maximum values used in object detection testing, but cannot be separated from the influence of image quality, the number of datasets and algorithms used. For epoch values ranging from 10; 50; 100; 250; 750 and 1000 epochs that have been done by previous researchers [60], [61], [62], then for batch sizes of 16; 32; 64; 128; 256 and 512 available on Teachable Machine, these values are the minimum and maximum values that are already available without the need for modification. Then, for the learning rate value based on the average of previous research ranges from 0.00001; 0.0001; 0.001; 0.01; 0.1 and 1 [63], [64], [65].

The three parameters above (epoch, batch size and learning rate) that have been explained, of course, the percentage and level of accuracy will affect the number of training datasets provided and trained, of course, then the image quality and the algorithm used to test the training data will also affect. So that through this research, exploratory testing will be carried out directly to find out the percentage results obtained through measurements and sample data provided and tested on the Teachable Machine site.

Epoch, learning rate, and batch size are important parameters in training machine learning models. Epoch is the number of times the entire dataset is skipped by the machine learning algorithm. Learning rate is a measure of how much change is made to the variables in each iteration. Batch size is the number of training samples used in one iteration. Determining the values for these three parameters is usually done through a trial-and-error process, with initial values either determined based on previous research or using Teachable Machine's default values of epoch 50, batch size 16 and learning rate of 0.001, and then adjusted based on the model training results. For example, if the model is overfitting, researchers may want to reduce the number of epochs. If the model training runs very slowly, researchers may want to increase the learning rate or change the batch size. However, keep in mind that any changes to these parameters can affect the performance of the model, so changes should be made with caution.

3. Methodology

This research uses an experimental method to test the effect of epoch value, learning rate, and batch size on the accuracy of herbal plant image classification. Image data of herbal plants is obtained from various sources, both from the internet and from the researcher's personal collection. The image data is then cleaned and grouped based on the type of herbal plant. Herbal plant data owned by the author is taken on internet sites, after the image is taken the researcher modifies the position change and rotates the image so that it has a different shape, this method is done because of limited data and variations in image models to be trained.

The herbal plant image classification process is carried out using Teachable Machine software. The classification model used is a CNNs model. The epoch value, learning rate, and batch size tested are as follows:

Table 1. Testing parameters with epoch value, learning rate and batch size

Epoch	10	50	100	250	750	1000
Learning rate	0.00001	0.0001	0.001	0.01	0.1	1
Batch size	16	32	64	128	256	512

From table 1 above, 216 tests were obtained to test the accuracy of the percentage value of the image recognition rate. Testing is also done by dividing the image data into two parts, namely training data and testing data. Training data is used to train the model, while testing data is used to evaluate the model.

The test data used is in the form of herbal plant image data taken from the internet as many as 10 types of herbal plants (turmeric, ginger, moringa leaves, aloe vera, basil leaves, mint leaves, chamomile, rosemary, coriander leaves and sambiloto leaves) as experimental samples in this study (although there are still many other herbal plants that can be explored again), where each type of herbal plant image there are 1,000 images per class as training data samples so that the total has 10,000 training data, and 15 images as validation/training data in Teachable Machine one by one. The data model set is shown in figure 7.

Figure 7. A collection of several types of herbal plants

4. Result and Discussion

Here is the table of testing results to find the most optimal and ideal values from the herbal plant datasets manually tested, identifying the best percentage from the sample data and the amount of training data in the Teachable Machine service. In the optimization value testing for the classification of herbal plant images using TM, a total of 216 data points were manually tested and tabulated. This testing included 6 categories of epochs (50, 100, 250, 500, 750, and 1,000) based on batch sizes (6, 32, 64, 128, 256, and 512) and learning rates (0.00001, 0.0001, 0.001, 0.01, 0.1, and 1). However, this study only considers the best accuracy values from each epoch to identify the best detection accuracy. The best accuracy results from each epoch category, consisting of 21 datasets (from 216 manually tested data variations), with accuracy ranked from highest to lowest based on epoch categories.

In selecting the best accuracy values, the researchers considered the values in the "Outside Test Class (%)" column and only used percentages below 100%. This approach was taken to maintain classification accuracy and avoid errors when the herbal plant detection program is tested with data not included in the training data. Thus, the program can detect that the data is not part of the original training data, and its accuracy is not perfect despite similarities. The best accuracy level selection testing results on teachable machine can be seen in table 2.

Table 2. The results of the best accuracy level selection testing on Teachable Machine

The results of the best accuracy level selection testing on Teachable Machine based on Epoch value categories.									
Dataset	**Epoch**	**Batch Size**	**Learning Rate**	**Loss per epoch**		**Accuracy per epoch**		**Original Test Class (%)**	**Outside Test Class (%)**
				Loss	***Test loss***	***Acc***	***Test acc***		
1,000	10	64	0.00010	0.04	0.03	0.99	0.99	97 - 100	41 - 96
1,000	10	128	0.00010	0.06	0.05	0.98	0.99	95 - 99	52 - 93
1,000	10	32	0.00010	0.00	0.01	0.99	0.99	92 - 100	46 - 97
1,000	10	32	0.00001	0.05	0.05	0.99	0.98	87 - 97	40 - 76
1,000	10	16	0.00001	0.33	0.33	0.99	0.99	82 - 92	47 - 80
1,000	50	32	0.00001	0.03	0.04	0.99	0.99	98 - 100	35 - 96
1,000	50	128	0.00010	0.00	0.03	1.00	0.98	97 - 100	46 - 99
1,000	50	16	0.00001	0.01	0.03	0.99	1.00	96 - 99	40 - 97
1,000	50	256	0.00010	0.01	0.03	0.99	0.98	93 - 100	41 - 94
1,000	50	512	0.00010	0.03	0.03	0.99	0.99	90 - 100	48 - 90
1,000	100	32	0.00001	0.00	0.01	0.99	0.99	95 - 100	50 - 99
1,000	100	64	0.00001	0.00	0.04	0.99	0.99	94 - 99	42 - 95
1,000	100	128	0.00001	0.05	0.06	0.99	0.98	75 - 99	40 - 92
1,000	100	256	0.00001	0.12	0.11	0.98	0.98	67 - 93	62 - 90
1,000	100	512	0.00001	0.24	0.26	0.97	0.97	44 - 92	36 - 70
1,000	250	128	0.00001	0.01	0.03	0.99	0.98	93 - 100	39 - 98
1,000	250	256	0.00001	0.03	0.04	0.99	0.98	91 - 100	34 - 96
1,000	250	512	0.00001	0.10	0.11	0.99	0.97	78 - 99	38 - 83
1,000	750	256	0.00001	0.01	0.02	1.00	0.99	97 - 100	37 - 95
1,000	750	512	0.00001	0.01	0.02	0.99	0.99	84 - 100	42 - 98
1,000	1,000	512	0.00001	0.01	0.02	0.99	0.99	90 - 100	46 - 94

With reference to the tabulated results, the search for the most optimal or ideal values of the three test parameters (epoch, batch size, and learning rate) has resulted in some analysis of the conclusions of the tests using Teachable Machine with 1,000 image samples per class (10 classes). The test results show an optimal and good object recognition rate ranking, which can be presented in the following table based on the percentage of the smallest "Outside Test Class" value, which is caused by a decrease in the recognition value of foreign image objects that are not included in the test category (such as: turmeric, ginger, moringa leaves, aloe vera, basil leaves, mint leaves, chamomile, rosemary, coriander leaves, and sambiloto leaves). However, this analysis is also affected by the accuracy of the original test data level of the herbal plant objects tested, known as the "Original Test Class". The higher the percentage value in the "Original Test Class" category, the better the results. The following 20 best test results are taken and have been analyzed based on their accuracy in the following table 3.

Table 3. Best test level based on Outside Test Class and Original Test Class on Teachable Machine

Best test level based on Outside Test Class and Original Test Class on Teachable Machine										
No	**Dataset**	**Epoch**	**Batch Size**	**Learning Rate**	**Loss per epoch**		**Accuracy per epoch**		**Original Test Class (%)**	**Outside Test Class (%)**
					Loss	***Test loss***	***Acc***	***Test acc***		
1	1,000	50	32	0.00001	0.03	0.04	0.99	0.99	98 - 100	35 - 96
2	1,000	750	256	0.00001	0.01	0.02	1.00	0.99	97 - 100	37 - 95
3	1,000	10	64	0.00010	0.04	0.03	0.99	0.99	97 - 100	41 - 96
4	1,000	50	128	0.00010	0.00	0.03	1.00	0.98	97 - 100	46 - 99
5	1,000	50	16	0.00001	0.01	0.03	0.99	1.00	96 - 99	40 - 97
6	1,000	10	128	0.00010	0.06	0.05	0.98	0.99	95 - 99	52 - 93
7	1,000	100	32	0.00001	0.00	0.01	0.99	0.99	95 - 100	50 - 99
8	1,000	100	64	0.00001	0.00	0.04	0.99	0.99	94 - 99	42 - 95
9	1,000	250	128	0.00001	0.01	0.03	0.99	0.98	93 - 100	39 - 99
10	1,000	50	256	0.00010	0.01	0.03	0.99	0.98	93 - 100	41 - 94
11	1,000	10	32	0.00010	0.00	0.01	0.99	0.99	92 - 100	46 - 97
12	1,000	250	256	0.00001	0.03	0.04	0.99	0.98	91 - 100	34 - 96
13	1,000	1,000	512	0.00001	0.01	0.02	0.99	0.99	90 - 100	46 - 94
14	1,000	50	512	0.00010	0.03	0.03	0.99	0.99	90 - 100	48 - 90
15	1,000	50	64	0.00001	0.07	0.08	0.99	0.98	88 - 98	44 - 94
16	1,000	10	32	0.00001	0.05	0.05	0.99	0.98	87 - 97	40 - 76
17	1,000	750	512	0.00001	0.01	0.02	0.99	0.99	84 - 100	42 - 98
18	1,000	10	16	0.00001	0.33	0.33	0.99	0.99	82 - 92	47 - 80
19	1,000	250	512	0.00001	0.10	0.11	0.99	0.97	78 - 99	38 - 83
20	1,000	100	128	0.00001	0.05	0.06	0.99	0.98	75 - 99	40 - 92

So, after analyzing the ability and reliability of the Teachable Machine service in this study, a conclusion can be drawn from the results of research conducted in three months to find the most optimal value to be used and developed into an intelligent system application for detecting herbal plants based on mobile web as follows:

1) Based on the results of research tabulation in Teachable Machine manually, the optimal and ideal epoch value for herbal plant image classification is at the 50th epoch value with a learning rate of 0.00001 and batch size value is 32, (See table 3).
2) The higher the epoch value, the better the accuracy value, but too many epoch values can cause overfitting, where the model learns the training data very well but fails to generalize well to new data in the Teachable Machine. Evidently at the 1,000th epoch, new test data that is outside the object in training can be detected up to 100% (for example: orange leaves are detected as marungga leaves).
3) The worst values tested for the learning rate were 0.01, 0.1, and 1. This is due to the learning process being too fast, which results in the model's lack of ability to recognize the trained images, and often results in 100% accuracy (overfitting), due to the image recognition being too specific and unstable, so the model is unable to recognize images that are not included in the herbal plant image object dataset in Teachable Machine. On the other hand, the best learning rate values are in the range of 0.0001 and 0.00001. These values provide enough time for the model to learn and recognize the patterns in the training data, so the results are more stable and able to recognize images better.
4) Improvement in classification accuracy also occurs as the learning rate value decreases. This is because a smaller learning rate value will cause the classification model to adjust more subtly and optimally. Conversely, the larger the learning rate, the worse the accuracy results and the unstable the training process.
5) The increase in classification accuracy also occurs as the batch size value increases. This is because a larger batch size value will cause the classification model to learn patterns from image data more efficiently but it takes a long time to achieve optimal accuracy. The most optimal batch size in this study is on average at batch size 32, (See table 3).

6) Detection results also affect the amount and quality of training data, the more and better-quality datasets the better the model learns patterns and features, and increases the model's ability to make accurate predictions on new data. In this study, the number of samples per class used is 1,000 samples per class where there are 10 image classes.
7) The similarity of the new test image object also affects the resulting percentage value, so that similar images are sometimes detected as the name of one of the images contained in the Teachable Machine database.
8) A very large amount of data and with a large increase in the number of epochs, will affect the training time to take hours. In this study, a total of 10 image categories (turmeric, ginger, moringa leaves, aloe vera, basil leaves, mint leaves, chamomile, rosemary, coriander leaves and sambiloto leaves) are training samples with a total of 1,000 samples per class, and if the total reaches 10,000 image data, which takes around 8 hours to 9 hours for each change in learning rate per-epoch and batch size.

From the summary and testing on the Teachable Machine site, related studies can adjust to the case study along with the amount and quality of image data owned, so that the test results to find the optimal value in this study become a reference, but back again to other studies that will certainly have different values depending on the amount and quality of image data owned.

After the test results are obtained, the researcher automatically determines the epoch value of 50, batch size 32 and learning rate 0.00001 then the researcher exports directly on the “Export Model” menu available on the Teachable Machine site into a file or javascript programming language (Tensorflow.js) to be developed into a mobile web-based application and designs the visual form of the application for easy use. And the test results are as expected ranging from 98% to 100% of the direct detection test results. Figure 8 below illustrates implementation in mobile web.

Figure 8. Implementation in mobile web

In figure 8, users can directly use the application then users can direct the application to the plants found or plants that want to be identified either in real-time or upload images manually from storage media. Then the system will automatically display the name of the plant type, accuracy and also the benefits of information from these herbal plants when detected and identified. Thus, the developed service is able to provide recommendations, good and accurate education and also become a reference service in the health sector, because all sources of information are obtained on credible health sites such as Halodoc and the website of the Ministry of Health (KEMENKES) of the Republic of Indonesia.

5. Conclusion

In this research that has been carried out for three months to find the most optimal value in image classification using the Teachable Machine service by testing epoch values ranging from 10; 50; 100; 250; 750 and 1000, then batch size

ISSN 2723-6471

values of 16; 32; 64; 128; 256 and 512 available on Teachable Machine, and learning rate values ranging from 0.00001; 0.0001; 0.001; 0.01; 0.1 and 1, using a total of 10,000 training data samples (1,000 samples per 10 classes). So that by testing all these variable values, 20 variation results are obtained which can be seen in table 3, with the most optimal value being the 50th epoch value with a learning rate of 0.00001 and a batch size value of 32 with an accuracy rate of 98% to 100%. The results of the hyperparameter values (epoch, batch size, and learning rate) are used by the researcher as a reference. Subsequently, the TensorFlow.js code file, available on the Teachable Machine website, is exported after determining the epoch, batch size, and learning rate parameters. This code is then developed into a mobile web-based application to educate the broader community through a publicly available application.

The potential implications of the research findings studied for the application of herbal medicine in the future with accurate classification can provide potential results that can be tested that the plants used in herbal medicine are the right plants, can increase public knowledge about herbal medicine, Teachable Machine is also an open-source service that can be developed and can also train new data and technology that is updated every time by Google.

6. Declarations

6.1. Author Contributions

Conceptualization: E.A.U.M., M.S., S.J.B., M.I.J.L., and Y.S.B.; Methodology: M.S.; Software: E.A.U.M.; Validation: E.A.U.M., M.S., S.J.B., M.I.J.L., and Y.S.B.; Formal Analysis: E.A.U.M., M.S., S.J.B., M.I.J.L., and Y.S.B.; Investigation: E.A.U.M.; Resources: M.S.; Data Curation: M.S.; Writing Original Draft Preparation: E.A.U.M. and Y.S.B.; Writing Review and Editing: M.S. and E.A.U.M.; Visualization: E.A.U.M.; All authors have read and agreed to the published version of the manuscript.

6.2. Data Availability Statement

The data presented in this study are available on request from the corresponding author.

6.3. Funding

The authors received no financial support for the research, authorship, and/or publication of this article.

6.4. Institutional Review Board Statement

Not applicable.

6.5. Informed Consent Statement

Not applicable.

6.6. Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] Darmatasia and M. I. Fanany, "Handwriting recognition on form document using convolutional neural network and support vector machines (CNN-SVM)," in *2017 5th International Conference on Information and Communication Technology (ICoIC7)*, Malaysia: IEEE, May 2017, vol. 5, no. May, pp. 1–6, 2017. doi: 10.1109/ICoICT.2017.8074699.

[2] M. Zhang, H. Lin, G. Wang, H. Sun, and J. Fu, "Mapping Paddy Rice Using a Convolutional Neural Network (CNN) with Landsat 8 Datasets in the Dongting Lake Area, China," *Remote Sens (Basel)*, vol. 10, no. 11, pp. 1840–1862, Nov. 2018, doi: 10.3390/rs10111840.

[3] A. H. Fakhrulddin, X. Fei, and H. Li, "Convolutional neural networks (CNN) based human fall detection on Body Sensor Networks (BSN) sensor data," in *2017 4th International Conference on Systems and Informatics (ICSAI)*, Tiongkok: IEEE, Nov. 2017, vol. 4, no. November, pp. 1461–1465, 2017. doi: 10.1109/ICSAI.2017.8248516.

[4] C. Liu *et al.*, "TX-CNN: Detecting tuberculosis in chest X-ray images using convolutional neural network," in *2017 IEEE International Conference on Image Processing (ICIP)*, Tiongkok: IEEE, Sep. 2017, vol. 1, no. September, pp. 2314–2318, 2017. doi: 10.1109/ICIP.2017.8296695.

[5] A. Botalb, M. Moinuddin, U. M. Al-Saggaf, and S. S. A. Ali, "Contrasting Convolutional Neural Network (CNN) with Multi-Layer Perceptron (MLP) for Big Data Analysis," in *2018 International Conference on Intelligent and Advanced System (ICIAS)*, Malaysia: IEEE, Aug. 2018, vol. 1, no. August, pp. 1–5, 2018. doi: 10.1109/ICIAS.2018.8540626.

[6] M. K. Chegeni, A. Rashno, and S. Fadaei, "Convolution-layer parameters optimization in Convolutional Neural Networks," *Knowl Based Syst*, vol. 261, no. 1, p. 110210, Feb. 2023, doi: 10.1016/j.knosys.2022.110210.

[7] K. Uchida, M. Tanaka, and M. Okutomi, "Coupled convolution layer for convolutional neural network," *Neural Networks*, vol. 105, no. 1, pp. 197–205, Sep. 2018, doi: 10.1016/j.neunet.2018.05.002.

[8] H.-C. Shin *et al.*, "Deep Convolutional Neural Networks for Computer-Aided Detection: CNN Architectures, Dataset Characteristics and Transfer Learning," *IEEE Trans Med Imaging*, vol. 35, no. 5, pp. 1285–1298, May 2016, doi: 10.1109/TMI.2016.2528162.

[9] Y. Shen *et al.*, "CS-CNN: Enabling Robust and Efficient Convolutional Neural Networks Inference for Internet-of-Things Applications," *IEEE Access*, vol. 6, no. 1, pp. 13439–13448, 2018, doi: 10.1109/ACCESS.2018.2810264.

[10] R. Chauhan, K. K. Ghanshala, and R. C. Joshi, "Convolutional Neural Network (CNN) for Image Detection and Recognition," in *2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC)*, India: IEEE, Dec. 2018, vol. 1, no. December, pp. 278–282, 2018. doi: 10.1109/ICSCCC.2018.8703316.

[11] A. Khandelwal, A. Shukla, and M. Sain, "A Survey on Automated Disease Diagnosis and Classification of Herbal Plants Using Digital Image Processing," in *2023 6th International Conference on Information Systems and Computer Networks (ISCON)*, IEEE, Mar. 2023, vol. 6, no. March, pp. 1–5, 2023. doi: 10.1109/ISCON57294.2023.10112042.

[12] R. M. Laire Ang and N. B. Linsangan, "Herbal Medicinal Plant Identification Using Leaf Vein Through Image Processing and Convolutional Neural Network," in *TENCON 2023 - 2023 IEEE Region 10 Conference (TENCON)*, IEEE, Oct. 2023, vol. 1, no. 1, pp. 1046–1051, 2023. doi: 10.1109/TENCON58879.2023.10322529.

[13] A. J. M. Lozada, N. L. Monsanto, and G. B. Pepito, "Comparative Study on Image Filtering For Herbal Plant Identification Using Xception Based Convolutional Neural Network," in *2022 International Seminar on Application for Technology of Information and Communication (iSemantic)*, IEEE, Sep. 2022, pp. 268–272. doi: 10.1109/iSemantic55962.2022.9920424.

[14] I. A. Md Zin, Z. Ibrahim, D. Isa, S. Aliman, N. Sabri, and N. N. A. Mangshor, "Herbal plant recognition using deep convolutional neural network," *Bulletin of Electrical Engineering and Informatics*, vol. 9, no. 5, pp. 2198–2205, Oct. 2020, doi: 10.11591/eei.v9i5.2250.

[15] I. V. P. Delos Reyes, "Fused Random Pooling in Convolutional Neural Network for Herbal Plants Image Classification," *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 8, no. 6, pp. 3208–3214, Dec. 2019, doi: 10.30534/ijatcse/2019/87862019.

[16] U. Andayani, I. B. Sumantri, A. Pahala, and M. A. Muchtar, "The Implementation of Deep Learning Using Convolutional Neural Network to Classify Based on Stomata Microscopic Image of Curcuma Herbal Plants," *IOP Conf Ser Mater Sci Eng*, vol. 851, no. 1, pp. 12035–12044, May 2020, doi: 10.1088/1757-899X/851/1/012035.

[17] F. Mariani, R. Tammachote, I. W. Kusuma, W. Chavasiri, H. Punnapayak, and S. Prasongsuk, "Phenolic Content and Biological Activities of Ethanol Extracts from Medicinal Plants in East Kalimantan, Indonesia," *Sains Malays*, vol. 50, no. 8, pp. 2193–2205, Aug. 2021, doi: 10.17576/jsm-2021-5008-05.

[18] N. A. Siregar, "Studi Literatur: Herbal Indonesia Berkhasiat Sebagai Anti Hipertensi," *Herbal Medicine Journal*, vol. 6, no. 2, pp. 1–21, Aug. 2023, doi: https://doi.org/10.58996/hmj.v6i2.80.

[19] P. Herawati, "Berbagai Daun Tumbuhan Indonesia yang Dapat Menurunkan Tekanan Darah," *Herbal Medicine Journal*, vol. 6, no. 2, pp. 19–28, Aug. 2023, doi: 10.58996/hmj.v6i1.81.

[20] S. Muslichah, R. Azrianingsih, S. Indriyani, and E. L. Arumingtyas, "Papeja, herbal potions during breastfeeding in Madura (Indonesia): The use, perceived effectiveness, and documentation of the plants used," *Biodiversitas*, vol. 23, no. 10, pp. 5139–5148, Oct. 2022, doi: 10.13057/biodiv/d231021.

[21] N. Asnah Sitohang, E. De lux Putra, H. Kamil, and M. Musman, "Phytochemical Screening of Putat Air [Barringtonia racemosa (L.)] Spreng Herbal Plants Found in Bireun, Aceh, Indonesia," *Res J Pharm Technol*, vol. 15, no. 6, pp. 2727–2732, Jun. 2022, doi: 10.52711/0974-360X.2022.00456.

ISSN 2723-6471

[22] M. Rahayu, D. Susan, A. P. Keim, S. Susiarti, and W. Sujarwo, “The diversity of plant species used in traditional herbal massage oil in Indonesia,” *Pacific Conservation Biology*, vol. 28, no. 6, pp. 505–516, Nov. 2021, doi: 10.1071/PC21022.

[23] D. Iryani, I. A. I. Pramestigiri, and P. J. Pihahey, “Edukasi Pembuatan Herbal Kunyit Asam Untuk Mengatasi Nyeri Haid dan Meningkatkan Imunitas Tubuh bagi Remaja Putri Dimasa Pandemi Covid-19 di Smp N 02 Manokwari,” *JURNAL KREATIVITAS PENGABDIAN KEPADA MASYARAKAT (PKM)*, vol. 5, no. 8, pp. 2507–2523, Aug. 2022, doi: 10.33024/jkpm.v5i8.6715.

[24] N. Verawati, N. Aida, and A. Yani, “Pengaruh Perbandingan Jenis Jahe dan Konsentrasi Jahe pada Karakteristik Kimia, Mikrobiologi Minuman Herbal Tradisional Minaserua,” *G-Tech: Jurnal Teknologi Terapan*, vol. 7, no. 4, pp. 1732–1739, Oct. 2023, doi: 10.33379/gtech.v7i4.3387.

[25] T. Hasan, N. Ida, and S. F. Qifni, “Skrining Fitokimia dan Uji Aktivitas Antioksidan Ekstrak Etanol Rimpang Kunyit Hitam (Curcuma caesia Roxb.) Asal Luwu Utara Dengan Metode DPPH,” *Jurnal Riset Kefarmasian Indonesia*, vol. 5, no. 3, pp. 439–457, Sep. 2023, doi: 10.33759/jrki.v5i3.397.

[26] R. G. Tribuana and A. Supriyatna, “Pemanfaatan Daun Herbal Sambiloto (Andrographis Paniculata) untuk Meningkatkan Daya Tahan Tubuh Pasca COVID-19,” *Matriks Jurnal Sosial dan Sains*, vol. 5, no. 1, pp. 121–126, May 2023, doi: 10.59784/matriks.v5i1.550.

[27] R. G. Absa and R. Riga, “Aktivitas Antioksidan Jamur Endofitik RS-1 Dari Ranting Sambiloto Menggunakan Media Beras Hitam,” *Jurnal Penelitian Farmasi Indonesia*, vol. 12, no. 1, pp. 31–36, Jun. 2023, doi: 10.51887/jpfi.v12i1.1756.

[28] M. L. Yudhi Purwoko, S. Syamsudin, and P. Simanjuntak, “Kombinasi Ekstrak Herba Pegagan dan Daun Kelor terhadap Kerusakan Otak dengan Metode Radial Eight-Arm Test,” *PHARMACY: Jurnal Farmasi Indonesia (Pharmaceutical Journal of Indonesia)*, vol. 19, no. 1, pp. 142–150, Sep. 2022, doi: 10.30595/pharmacy.v19i1.13490.

[29] N. S. Vinolina and R. Sigalingging, “Analyses of Bioactive Compounds of Pegagan (Centella Asiatica (L.) Urb) from Samosir – Indonesia Accession,” *Indonesian Journal of Agricultural Research*, vol. 5, no. 01, pp. 42–49, Aug. 2022, doi: 10.32734/injar.v5i01.6797.

[30] E. A. U. Malahina, “Teachable Machine: Deteksi Dialek Sumba Timur (Kambera) Menggunakan Layanan Open Source,” *Jurnal Nasional Teknik Elektro dan Teknologi Informasi*, vol. 12, no. 4, pp. 280–286, Nov. 2023, doi: 10.22146/jnteti.v12i4.8174.

[31] T. Machine, “Teachable Machine: FAQ,” Teachable Machine. Accessed: Jan. 24, 2024. [Online]. Available: https://teachablemachine.withgoogle.com/faq

[32] M. Carney *et al.*, “Teachable Machine: Approachable Web-Based Tool for Exploring Machine Learning Classification,” in *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems*, New York, NY, USA: ACM, Apr. 2020, pp. 1–8. doi: 10.1145/3334480.3382839.

[33] D. Agustian, P. P. G. P. Pertama, P. N. Crisnapati, and P. D. Novayanti, “Implementation of Machine Learning Using Google’s Teachable Machine Based on Android,” in *2021 3rd International Conference on Cybernetics and Intelligent System (ICORIS)*, Makasar: IEEE, Oct. 2021, pp. 1–7. doi: 10.1109/ICORIS52787.2021.9649528.

[34] J. J. N. Wong and N. Fadzly, “Development of species recognition models using Google teachable machine on shorebirds and waterbirds,” *Journal of Taibah University for Science*, vol. 16, no. 1, pp. 1096–1111, Dec. 2022, doi: 10.1080/16583655.2022.2143627.

[35] E. A. U. Malahina, R. P. Hadjon, and F. Y. Bisilisin, “Teachable Machine: Real-Time Attendance of Students Based on Open Source System,” *The IJICS (International Journal of Informatics and Computer Science)*, vol. 6, no. 3, pp. 140–146, Nov. 2022, doi: 10.30865/ijics.v6i3.4928.

[36] I. Kandel and M. Castelli, “The effect of batch size on the generalizability of the convolutional neural networks on a histopathology dataset,” *ICT Express*, vol. 6, no. 4, pp. 312–315, Dec. 2020, doi: 10.1016/j.icte.2020.04.010.

[37] S. Salim, M. M. A. Jamil, R. Ambar, W. S. W. Zaki, and S. Mohammad, “Learning Rate Optimization for Enhanced Hand Gesture Recognition using Google Teachable Machine,” in *2023 IEEE 13th International Conference on Control System, Computing and Engineering (ICCSCE)*, Penang: IEEE, Aug. 2023, pp. 332–337. doi: 10.1109/ICCSCE58721.2023.10237148.

[38] N. B. Aldin and S. S. A. B. Aldin, “Accuracy Comparison of Different Batch Size for a Supervised Machine Learning Task with Image Classification,” in *2022 9th International Conference on Electrical and Electronics Engineering (ICEEE)*, Alanya: IEEE, Mar. 2022, pp. 316–319. doi: 10.1109/ICEEE55327.2022.9772551.

[39] F. N. Fajri, K. Malik, and G. Q. O. Pratamasunu, “Metode Pengumpulan Data Pada Deteksi Pakaian Hijab Syar’I Berdasarkan Citra Digital Menggunakan Teachable Machine Learning,” *JUSTEK : Jurnal Sains dan Teknologi*, vol. 5, no. 2, pp. 194–203, Nov. 2022, doi: https://doi.org/10.31764/justek.vXiY.ZZZ.

[40] N. Rochmawati, H. B. Hidayati, Y. Yamasari, H. P. A. Tjahyaningtijas, W. Yustanti, and A. Prihanto, “Analisa Learning Rate dan Batch Size pada Klasifikasi Covid Menggunakan Deep Learning dengan Optimizer Adam,” *Journal of Information Engineering and Educational Technology*, vol. 5, no. 2, pp. 44–48, Dec. 2021, doi: 10.26740/jieet.v5n2.p44-48.

[41] X. Liu *et al.*, “Automatic taxonomic identification based on the Fossil Image Dataset (>415,000 images) and deep convolutional neural networks,” *Paleobiology*, vol. 49, no. 1, pp. 1–22, Feb. 2023, doi: 10.1017/pab.2022.14.

[42] H. L. Dawson, O. Dubrule, and C. M. John, “Impact of dataset size and convolutional neural network architecture on transfer learning for carbonate rock classification,” *Comput Geosci*, vol. 171, no. 2, pp. 105284–105295, Feb. 2023, doi: 10.1016/j.cageo.2022.105284.

[43] M. K. Balwant, “A Review on Convolutional Neural Networks for Brain Tumor Segmentation: Methods, Datasets, Libraries, and Future Directions,” *IRBM*, vol. 43, no. 6, pp. 521–537, Dec. 2022, doi: 10.1016/j.irbm.2022.05.002.

[44] S. N. Mohd Safuan, M. R. Md Tomari, and W. N. Wan Zakaria, “Cross Validation Analysis of Convolutional Neural Network Variants with Various White Blood Cells Datasets for the Classification Task,” *International Journal of Online and Biomedical Engineering (iJOE)*, vol. 18, no. 02, pp. 123–140, Feb. 2022, doi: 10.3991/ijoe.v18i02.27321.

[45] S. Chen, “CNN combined with data augmentation for face recognition on small dataset,” *J Phys Conf Ser*, vol. 2634, no. 1, p. 012040, Nov. 2023, doi: 10.1088/1742-6596/2634/1/012040.

[46] A. Farahani, B. Pourshojae, K. Rasheed, and H. R. Arabnia, “A Concise Review of Transfer Learning,” in *2020 International Conference on Computational Science and Computational Intelligence (CSCI)*, Las Vegas: IEEE, Dec. 2020, pp. 344–351. doi: 10.1109/CSCI51800.2020.00065.

[47] C. Google, “TensorFlow.js: Make your own ‘Teachable Machine’ using transfer learning with TensorFlow.js,” Google. Accessed: Jan. 24, 2024. [Online]. Available: https://codelabs.developers.google.com/tensorflowjs-transfer-learning-teachable-machine#0

[48] C. Google, “What is TensorFlow.js?,” Google Codelabs. Accessed: Jan. 25, 2024. [Online]. Available: https://codelabs.developers.google.com/tensorflowjs-transfer-learning-teachable-machine#1

[49] T. Machine, “What is Teachable Machine?,” Teachable Machine. Accessed: Sep. 26, 2022. [Online]. Available: https://teachablemachine.withgoogle.com/faq

[50] T. Toivonen, I. Jormanainen, J. Kahila, M. Tedre, T. Valtonen, and H. Vartiainen, “Co-Designing Machine Learning Apps in K–12 With Primary School Children,” in *2020 IEEE 20th International Conference on Advanced Learning Technologies (ICALT)*, Tartu: IEEE, Jul. 2020, pp. 308–310. doi: 10.1109/ICALT49669.2020.00099.

[51] J. Gu *et al.*, “Recent advances in convolutional neural networks,” *Pattern Recognit*, vol. 77, no. 5, pp. 354–377, May 2018, doi: 10.1016/j.patcog.2017.10.013.

[52] A. Amidi and S. Amidi, “Convolutional Neural Networks cheatsheet,” University Stanford. Accessed: May 18, 2024. [Online]. Available: https://stanford.edu/~shervine/teaching/cs-230/cheatsheet-convolutional-neural-networks

[53] C. Ozdemir, Y. Dogan, and Y. Kaya, “A new local pooling approach for convolutional neural network: local binary pattern,” *Multimed Tools Appl*, vol. 83, no. 12, pp. 34137–34151, Nov. 2023, doi: 10.1007/s11042-023-17540-x.

[54] A. Zafar *et al.*, “A Comparison of Pooling Methods for Convolutional Neural Networks,” *Applied Sciences*, vol. 12, no. 17, pp. 8643–8664, Aug. 2022, doi: 10.3390/app12178643.

[55] M. Yani, S. Si. , M. T. Budhi Irawan, and S. T. , M. T. Casi Setiningsih, “Application of Transfer Learning Using Convolutional Neural Network Method for Early Detection of Terry’s Nail,” *J Phys Conf Ser*, vol. 1201, no. 1, p. 012052, May 2019, doi: 10.1088/1742-6596/1201/1/012052.

[56] L. Alzubaidi *et al.*, “Review of deep learning: concepts, CNN architectures, challenges, applications, future directions,” *J Big Data*, vol. 8, no. 1, p. 53, Mar. 2021, doi: 10.1186/s40537-021-00444-8.

[57] S. Shafi and A. Assad, “Exploring the Relationship Between Learning Rate, Batch Size, and Epochs in Deep Learning: An Experimental Study,” in *Soft Computing for Problem Solving*, Singapore: Springer, 2023, pp. 201–209. doi: 10.1007/978-981-19-6525-8_16.

[58] R. Lin, “Analysis on the Selection of the Appropriate Batch Size in CNN Neural Network,” in *2022 International Conference on Machine Learning and Knowledge Engineering (MLKE)*, Guilin: IEEE, Feb. 2022, pp. 106–109. doi: 10.1109/MLKE55170.2022.00026.

[59] S. Salim, M. M. A. Jamil, R. Ambar, W. S. W. Zaki, and S. Mohammad, “Learning Rate Optimization for Enhanced Hand Gesture Recognition using Google Teachable Machine,” in *2023 IEEE 13th International Conference on Control System, Computing and Engineering (ICCSCE)*, Penang: IEEE, Aug. 2023, pp. 332–337. doi: 10.1109/ICCSCE58721.2023.10237148.

[60] R. D. Nurfita and G. Ariyanto, “Implementasi Deep Learning berbasis Tensorflow untuk Pengenalan Sidik Jari,” *Emitor: Jurnal Teknik Elektro*, vol. 18, no. 1, pp. 22–27, Jun. 2018, doi: 10.23917/emitor.v18i01.6236.

[61] Y. Sari, Y. F. Arifin, Novitasari, and M. R. Faisal, “The Effect of Batch Size and Epoch on Performance of ShuffleNet-CNN Architecture for Vegetation Density Classification,” in *7th International Conference on Sustainable Information Engineering and Technology 2022*, New York, NY, USA: ACM, Nov. 2022, pp. 39–46. doi: 10.1145/3568231.3568239.

[62] N. Das and S. Das, “Epoch and accuracy based empirical study for cardiac MRI segmentation using deep learning technique,” *PeerJ*, vol. 11, no. 3, p. e14939, Mar. 2023, doi: 10.7717/peerj.14939.

[63] N. K. R. Mirayanti, S. Sariyasa, and I. G. A. Gunadi, “Batch size and learning rate effect in covid-19 classification using CNN,” *Jurnal Mantik*, vol. 7, no. 3, pp. 1752–1765, Nov. 2023, doi: https://doi.org/10.35335/mantik.v7i3.4177.

[64] J. Pan, “The impact of learning rate and data size on CNN for skin cancer detection,” in *Second International Conference on Medical Imaging and Additive Manufacturing (ICMIAM 2022)*, Y. Yusof, Ed., Xiamen: SPIE, Jun. 2022, p. 28. doi: 10.1117/12.2636723.

[65] A. Johny and K. N. Madhusoodanan, “Dynamic Learning Rate in Deep CNN Model for Metastasis Detection and Classification of Histopathology Images,” *Comput Math Methods Med*, vol. 2021, no. 2, pp. 1–13, Oct. 2021, doi: 10.1155/2021/5557168.