Proceedings of the 2023 International Conference on Software Engineering and Machine Learning
DOI: 10.54254/2755-2721/8/20230228

Improving performance of VGG-19 model using dual input block for skin disease classification

S. Malarkhodi[1], Tharun Raj T.R.[2,3], Shailesh Kanna R.[2], Sanjeevi G.G.[2]

[1]Professor, Department of Electronics and Communication Engineering, K. S. Rangasamy College of Technology, Namakkal.
[2]UG Student, Department of Electronics and Communication Engineering, K. S. Rangasamy College of Technology, Namakkal.

[3]tharunraj002@gmail.com

Abstract. Skin diseases are a growing threat to humans. They can cause severe damage to the skin. The damage to the skin sometimes can lead the patient to have a loss of confidence and depression. By using emerging technologies like deep learning, we can identify skin diseases and treat them effectively. These computer-powered techniques can detect skin diseases without any help from a professional. People can have a user-friendly experience if some effort is put into the interface. This can help people identify the disease in its early stages instead of just ignoring it as an allergy. This will help people avoid suffering from the severe consequences of the diseases. By using these technologies, we can accurately find the disease in the patient. By using deep learning algorithms like those at CNN, we can make things simpler and less time-consuming. By using advanced models like Inception Net and ResNet and optimizing the model's hyperparameters like learning rate, optimizers, etc., the model can give high performance and accuracy. The model can be trained to give accurate results even if the images used in the training dataset and the image given by the patient are of low quality. We propose a dual-input CNN model to identify skin diseases. We use a modified VGG-19 neural network with dual input blocks and batch normalization to classify the diseases. The model outperforms the original VGG19 by about 5 percent. The original model, trained from scratch, was also modified to have batch normalization. The models did not learn, and the accuracy did not increase without batch normalization. The models were trained for 150 epochs. Both were trained under the same parameters and methods. The dual input model yields about 94%, while the original VGG19 architecture trained from scratch gives about 89%. The models were trained on P100 GPU are available in Kaggle. The model is implemented on hardware that can give results without any delay to the patient.

Keywords: VGG19, dual input block model, deep learning, CNN, skin diseases.

1. Introduction

Skin diseases are diseases that affect the skin of a person. Many types of skin diseases affect humans. Many of them can cause more than just damage to the person's skin. Many of them can cause more than just damage to the person's skin. They can sometimes kill someone if the disease is ignored or untreated. In America, one in four people is affected by these diseases. Not only in America, but people all over

DOI: 10.54254/2755-2721/8/20230228

the world are affected. Many people ignore the early symptoms of these diseases and think they are just allergies or that they will go away on their own. If these people don't get treated, at least some of them will get to the severe stage of the disease due to ignorance and a lack of medical assistance. To prevent these kinds of situations and possibly avoid the death of a person, technologies like machine learning can be used to reduce the number of patients with severe cases significantly.The sub-branch of machine learning called "deep learning" uses images to perform the tasks of classification, segmentation, etc. A single photo of a diseased area of a person can be used to detect skin disease. It can even accurately identify which disease has affected the person. One of the algorithms used in deep learning is called a "convolutional neural network," which is the algorithm used in this paper.CNN has a lot of architectures, like ResNet, Inception Net, Xception, etc. They can perform better if we use custom loss functions, optimization, hyperparameter tuning, etc [3,11,13].

Usually, the standard architectures in deep learning like VGG nets, ResNet (residual networks), Inception nets, etc. use a single input for the classification. By using two input blocks for feature extraction and adding the output of these two input blocks, we can get better feature extraction than with the original single-input network. In this paper, we have demonstrated the architecture using a modified VGG-19 network to have dual input and batch normalization. The basic idea is that by giving the same image to the 2-input block of the model, we can achieve better accuracy than the original methods. The input blocks do not depend on each other and extract features independently, and by adding the outputs of these 2 blocks, we can have a better feature extraction than the single input models. This, in turn, can increase the accuracy of the model and its performance.

2. Theory and method

2.1. Relatedwork

Many works in deep learning and skin disease classification are helpful to our work. The models trained with the help of the GPUs are extremely useful for disease classification. They can achieve higher accuracy than humans. This is extremely important because it can help make a better analysis of the patient. Recently, Belal Ahmad et al. [3] have used Resnet 152 and Inception Resnet-V2 with triplet loss functions. Using the triplet loss function has achieved better results than Resnet 152 and Inception Resnet-V2 without the triplet loss function. This shows that by adding to or improving the existing architectures and tuning the hyperparameters, we can achieve better results than the original or previous ones. Sung-Hyun Yoon et al. [5] have shown that we can extract more features and accuracy by using dual inputs instead of a single input. We are hoping to achieve better accuracy by adding batch normalization to the neural network. Seunghyeok back et al. [2] models to recognize the condition Herpes zostera, a prevalent cutaneous illness that affects one in five individuals, even in low-quality images. This is practical in real life; for example, in their case, they have trained it for a mobile app that identifies the disease. People in real life may not upload a good quality image due to some reasons. So, the model must perform well even in those conditions.Sergey Ioffe et al. [1] in their paper introduced the batch normalization layer. The batch normalization layer increases the performance, and the model learns features more accurately. The batch normalization is a layer that decreases the internal covariance shift, which is helpful in model training and classification. Using the batch normalization layer in the model is a good approach. The layer is used in ResNet, inception, and xception the layer solves the problem of accuracy and loss staying at the same percentage over and over after many epochs. It helps solve the problem as it helps the model grasp the hard-to-learn features and gives a good result.

2.2. Dataset

The Dataset for training the model is from Shubham Goel and Bill Hall in Kaggle. The dataset includes 23 types of diseases, including exanthems and drug eruptions. Other diseases in the dataset include actinic keratosis basal cell carcinoma and other malignant lesions, atopic dermatitis, bullous disease, cellulitis impetigo, and other bacterial infections. There are 4002 testing pictures and 15557 training images in the collection. The collection includes 19559 pictures overall for 23 classes. The dataset's

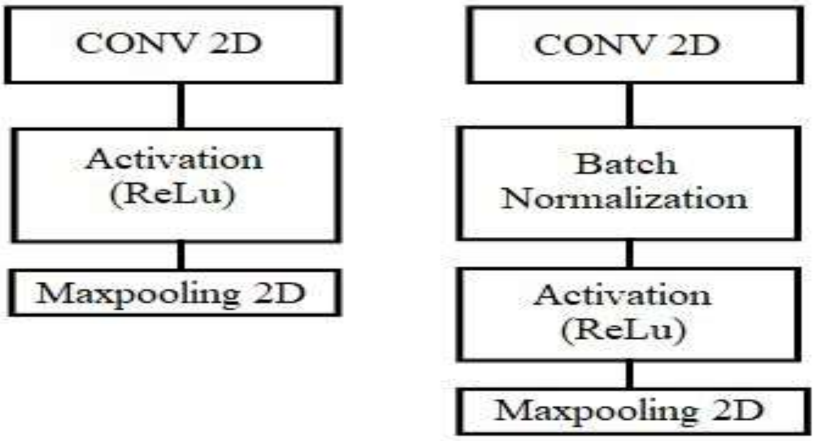
DOI: 10.54254/2755-2721/8/20230228

entire picture library was downloaded from the dermnet website. The three bands of the pictures are RGB, and they are in JPEG format. Although there are variations in the picture resolutions, they are not generally high-resolution photos. The dataset is imbalanced because in some classes there are over 1000 images and in some classes have about 400 and it had a forest image in the dataset. Before using the dataset, measures to balance the images must be taken. Overall, the dataset is good, but it lacks the balance of data that is essential for model training.[2] & [7]

Figure 1. Sample images from the Dataset.

2.3. *Batch normalization*

In their 2015 article, Sergey Ioffe et al. [1] proposed batch normalization. The group normalization reduces the output of the preceding layer to have a mean and variation of 0 and 1, respectively. Since it normalizes the numbers in the present batch, the layer is referred to as "Batch Normalization".The batch normalizing layer improves the applicability of the model and reduces the internal correlation shift. It decreases the internal correlation shift by aligning the distribution of the layer sources.



(a)Without Batch Normalization Layer

(b)With Batch Normalization layer

Figure 2. Placing the Batch Normalization layer in the model.

DOI: 10.54254/2755-2721/8/20230228

It is done by subtracting the one batch's mean and dividing it by the one batch's standard deviation. By using batch normalization, As the layer takes care of the vanishing gradient issue, we can set high learning rates. The disappearing gradient issue is fixed by the layer. Each layer's input is noisier after the Batch Normalization layer. As a result, the overfitting issue is also resolved. Between the convolution layer and the activation layer comes the batch normalization layer. The location of the Batch Normalization layer is depicted in Figure 2.

2.4. Architecture

Initially, we used the original VGG-19 model without batch normalization for training. But despite running several hours and epochs, the accuracy was stuck at the same percentage. So, we added the batch normalization layer to the model. Both the single- and dual-input VGG19 models have a batch normalization layer. After adding the layer, the model started to learn the features well and increased its accuracy. We tested the VGG19 model without the Batch Normalization layer and saw the same results again and again after testing a few times.To increase the features that are learned by the neural network, we have used dual input block in VGG19 along with batch normalization.

We give the same image to the two input blocks, and by adding the result before making a prediction, the number of features learned can be increased, which can lead to better performance and accuracy of the model.Both dual input blocks of the model have the same layers as the VGG19 model. The layers after input layer and before flatten layer are used in the Input blocks. It is like the Ensemble learning where different models are trained with the same dataset and all their predictions are combined to give a higher accurate ensemble model. The model is like random forest which is an ensemble model of decision trees. But the difference in our model is that we are using the model to more features than the original model not to combine the prediction to give a combined ensemble model with higher accuracy.We use an "add" layer instead of a "concatenate" layer. So, the image features obtained are not stacked side by instead added into a better feature map of the image. Also, by including batch normalization layers in the model, the accuracy can be further increased. The Batch Normalization layers are used in models like ResNet, etc., enhancing the model's functionality and producing a model with greater precision. The architecture is shown in Figure 3.

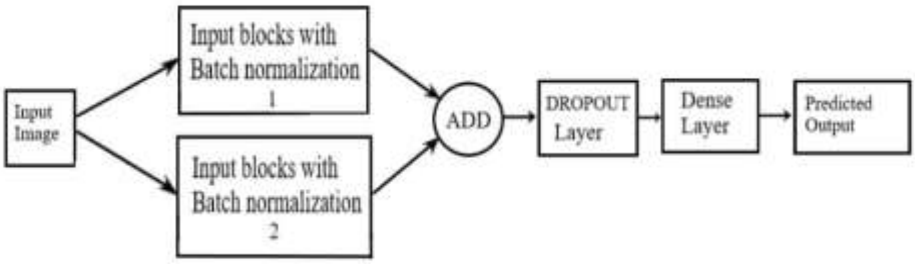


Figure 3. Model Architecture.

The model has been trained with the same image on both input blocks. So instead of using two input layers to get the input image, a single input layer is used. Since the image given to both inputs is the same, it creates extra complexity to have two input layers to get the input image, so a single input layer is used. Since the image given to both inputs is the same, having two input layers creates extra complexity. The output of the single input layer is given to both input blocks. If we use two input layers, then we must make changes to the pre-processing functions, like the image data generator. The model was first used with two input layers. We created a function to use the image data generator to give two inputs to the model.Despite the length of time needed to train the model, it was successful. A step of an epoch was completed by the model in around 28 seconds. The dataset had about 20,000 images, so it was not an efficient way to train, and with many images, the training time increases. By using the image data generator, good and accurate results can be obtained, but it increases the training time. So, using

DOI: 10.54254/2755-2721/8/20230228

our modified image data generator function was not possible and would not be a good solution. A simple solution we found was to implement the model with a single input layer. This allowed us to use the image data generator function without any prior modifications. The new model resulted in less training time than the previous model.There is one input layer, and its output is fed forward into the dual input blocks based on the VGG-19 architecture. They both have the same layers and activation or they are mirror images of each other. Then their output is added together and given to the flatten, dropout, and dense layers for prediction.

2.5. Training and results

Each model completed 150 epochs of training. The models were developed using P100 Processors found in Kaggle. The training dataset was also obtained from Kaggle. In terms of loss and model precision, the dual input block model outperforms the VGG19 model. The figures below show the loss and accuracy. In comparison to the initial VGG19 model, the model is learning more characteristics. The original model has 89% accuracy, while the dual-input model has 94%. which is 5% higher than the original VGG19 architecture. Both models were trained from scratch. The comparison of the accuracy between the original is shown in figure 5 below.

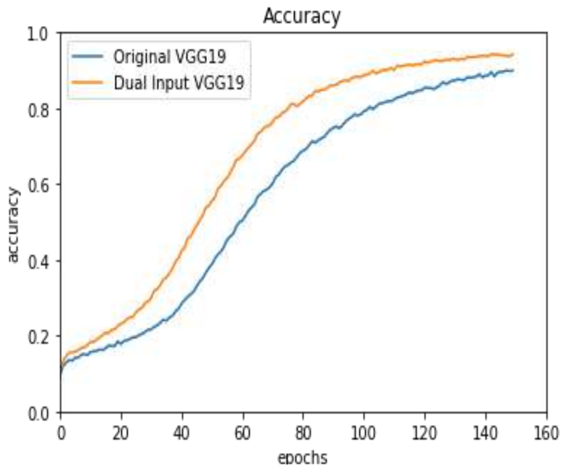


Figure 4. Accuracy comparison.

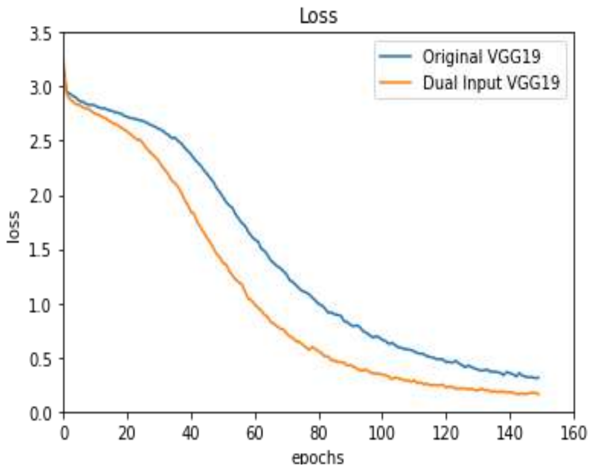


Figure 5. Loss comparison.

In figures 4&5 the X axis is the no of epochs We trained, and the accuracy is shown on the Y axis in Figure 4. The loss of the model is seen on the y-axis of Figure 4. The original VGG19 model's accuracy and loss are shown by the blue line, while the dual input block model's accuracy and loss are represented by the orange line. A planner for the learning rate was used to teach the models. The learning rate would be reduced by multiplying the current rate by the 0.98 decay rate. The learning rate would decrease every 10 epochs by multiplying by the decay rate. Figure 6 shows the learning rate decreasing over epochs. Both models were set and trained with the same decay rate. As a result, both models experience the same decrease in learning rate. The learning rate starts at 2E-04 and eventually drops to 0.00014.

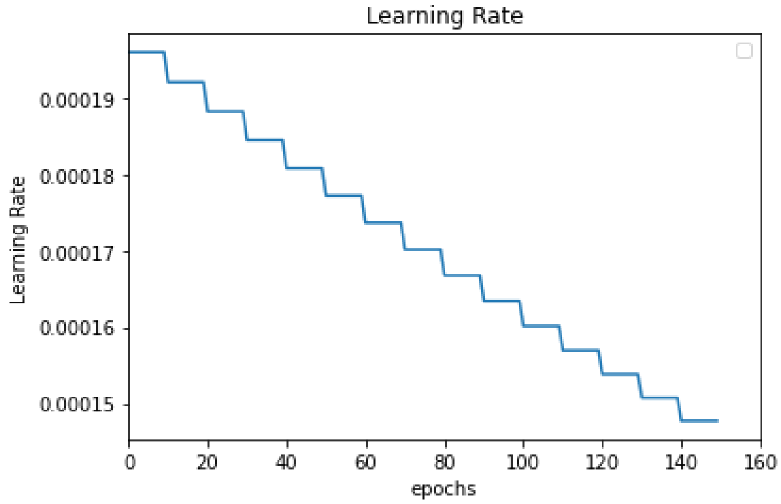


Figure 6. Learning rate decay.

3. Conclusion

This paper proposes a modified VGG-19 model with batch normalization for high feature extraction and improved accuracy. Any other models, like ResNet, etc., can be modified to have the same dual input structure for feature extraction. Further, this model can also be trained to use two images instead of one by using two input layers. If the model gets two inputs instead of one, this approach can be used. This can be done by using a concatenate layer instead of adding a layer. So, the features do not confuse the model by being added to one. The concatenate layer will stack both image features side by side. The model can then provide an accurate prediction of the diseases. The dual input block model performs well in predicting the 23 classes of the dataset.It initially struggles to distinguish the characteristics in the illness pictures. However, the algorithm was able to pick up on the characteristics of the pictures because of batch normalization. Additionally, by combining the two outcomes of the dual-input model's input components, better features are obtained than in the original single-input VGG19 model. The original VGG19 model achieved 89% accuracy, while the dual input block model got 94% accuracy. The loss of the dual input block model is also significantly lower than that of the original VGG19 model. A learning rate scheduler is also used in this model to allow the model to learn slowly as the training goes on. The model's learning rate is reduced every epoch for better learning of features. By adding more convolutional layers, the model can learn even more features than the one proposed in this paper. More layers can be added, and by tuning the hyperparameters of the model, the accuracy can be further increased. And we can even use a triple-input block model, which will perform even better than the dual-input block model.

References

[1] Ioffe S, Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *In International conference on machine learning 2015*, 448-456 (2015, Jun 1).

[2] S. Back et al: "Robust Skin Disease Classification by Distilling Deep Neural Network Ensemble for the Mobile Diagnosis of Herpes Zoster". *IEEE Access, vol 9*, 20156-20169. doi:10.1109/ACCESS.2021.3054403 (2021).

[3] B. Ahmad, M. Usama.-M. Huang, K. Hwang, M. S. Hossian and G. Muhammad: "Discriminative Feature Learning for Skin Disease Classification Using Deep Convolutional Neural Network". *IEEE Access, Vol 8*, 39025-39033. doi:10.1109/ACCESS.2020.2975198 (2020).

[4] T. Goswami, V. K. Dabhi and H. B. Prajapati: "Skin Disease Classification from Image - A Survey". *6th International Conference on Advanced Computing and Communication Systems (ICACCS)*, 599-605. doi:10.1109/ICACCS48705.2020.9074232 (2020).

[5] S. -H. Yoon H. -J. Yu: "Multiple Points Input For Convolutional Neural Networks in Replay Attack Detection". *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 6444-6448. doi:10.1109/ICASSP40776.2020.9053303 (2020).

[6] Z. Wu et al: "Studies on Different CNN Algorithms for Face Skin Disease Classification Based on Clinical Images". *IEEE Access, Vol 7*, 66505-66511. doi:10.1109/ACCESS.2019.2918221 (2019).

[7] A. Mohanty, A. Sutherland, M.Bezbradica and H. Javidnia: "Skin Disease Analysis with Limited Data in Particular Rosacea : A Review and Recommended Framework". *IEEE Access*, 39045-39068. doi:10.1109/ACCESS.2022.3165574 (2022).

[8] L. -F. Li, X. Wang, W.-J.Hu, N. N. Xiong, Y. -X. Du and B. -S. Li: "Deep Learning in Skin Disease Image Recognition: A Review". *IEEE Access, Vol 8*, 208264-208280. doi:10.1109/ACCESS.2020.3037258 (2020).

[9] S. J. K. Jagadeesh Kumar, P. Parthasarathi , Mofreh A. Hogo, Mehedi Masud, Jehad F. Al-Amri and Mohamed Abouhawwash, “Breast Cancer Detection Using Breastnet-18 Augmentation with Fine Tuned Vgg-16”, Intelligent Automation & Soft Computing, Vol. 36, No.2, pp. 2363–2378, 2023.

[10] H. Q. Yu and S. Reiff-Marganiec: "Targeted Ensemble Machine Classification Approach for Supporting IoT Enabled Skin Disease Detection". *IEEE Access, Vol 9*, 50244-50252. doi:10.1109/ACCESS.2021.3069024 (2021).

[11] T. -C. Pham, A. Doucet,C.-M.Loung,C.-T. Tran and V. -D. Hoang: "Improving Skin-Disease Classification Based on Customized Loss Function Combined With Balanced Mini-Batch Logic and Real-Time Image Augmentation". *IEEE Access, Vol 8*, 150725-150737. doi:10.1109/ ACCESS.2020.3016653 (2020).

[12] P. B. Manoorkar, D. K. Kamat and P. M. Patil: "Analysis and classification of human skin diseases". *International Conference on Automatic Control and Dynamic Optimization Techniques (ICACDOT)*, 1067-1071. doi:10.1109/ICACDOT.2016.7877750 (2016).

[13] E. Goceri: “Analysis of Deep Networks with Residual Blocks and Different Activation Functions: Classification of Skin Diseases”. *Ninth International Conference on Image Processing Theory, Tools and Applications (IPTA)*, 1-6. doi:10.1109/.

[14] Prabu Kanna, G., Jagadeesh Kumar, S.J.K., Parthasarathi, P., Yogesh Kumar (February 2023), “A Review on Prediction and Prognosis of the Prostate Cancer and Gleason Grading of Prostatic Carcinoma Using Deep Transfer Learning Based Approaches”, Archives of Computational Methods in Engineering, 2023.