

Performance Analysis of Deep Learning Techniques Detecting Black Fungus



Mallikarjuna Rao Gundavarapu, Bhargavi Lella, Shriya Manda, Lakshmi Prasanna Nekkanti, Prerana Panchareddy

Abstract: *Mucormycosis, which is also commonly known as "Black Fungus", is a deadly fungal infection that is caused by a group of molds called "mucormycetes". Black Fungus attacks when one gets in contact with fungus spores in the environment, especially when a person is having any kind of skin trauma like a cut, scrape, burn, etc. They noted that this infection affects a person who is having other health issues like diabetes, cancer, organ transplants, or who is using other drugs which compromise the immune system. The spread of coronavirus has hiked the cases of Black Fungus as coronavirus leaves its patient's immune systems in a weakened state, making them more susceptible to Mucormycosis. The symptoms of black fungus when the body gets attacked vary depending on where it is growing in your body, some of the recognized symptoms include fever, cough, chest pain, headache, swelling and redness of eyes, discoloration over the eyes, nose, and sides of one's face, blurred vision and in severe cases loss of vision, blood vomits, shortness of breath. It enters the body via the nose, mouth, or eyes and can affect the brain and other organs if not treated on time. The primary aim of this project is to detect black fungus using Deep Neural Networks based on the iris dataset. We have experimented with various deep learning techniques for detecting black fungus from eye image datasets, such as VGG16, ResNet-50, InceptionV3, and AlexNet.*

Keywords: *Mucormycosis, Discoloration, Redness, Swelling, Coronavirus, Vision.*

I. INTRODUCTION

Fungus is harmful to human health, it may cause various life-threatening diseases. There are millions of different fungal species present in the environment of which few cause fungal infections which are extremely hazardous to human health [1].

Manuscript received on 27 March 2022

Revised Manuscript received on 30 March 2022

Manuscript published on 30 April 2022

* Correspondence Author

Mallikarjuna Rao Gundavarapu, Professor, Department of Computer Science and Engineering, Gokaraju Rangaraju Institute of Engineering and Technology, Hyderabad (Telangana), India.

Bhargavi Lella*, Department of Computer Science and Engineering, Gokaraju Rangaraju Institute of Engineering and Technology, Hyderabad (Telangana), India.

Shriya Manda, Department of Computer Science and Engineering, Gokaraju Rangaraju Institute of Engineering and Technology, Hyderabad (Telangana), India.

Lakshmi Prasanna Nekkanti, Department of Computer Science and Engineering, Gokaraju Rangaraju Institute of Engineering and Technology, Hyderabad (Telangana), India.

Prerana Panchareddy, Department of Computer Science and Engineering, Gokaraju Rangaraju Institute of Engineering and Technology, Hyderabad (Telangana), India.

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an [open access](http://creativecommons.org/licenses/by-nc-nd/4.0/) article under the CC-BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>)

One such deadly fungal infection is "mucormycosis" which is commonly referred to as black fungus, it is a very rare but fatal disease caused by a group of molds called "mucormycetes", which is ubiquitously present in the environment, especially in soil, air, plants, manure, dead and decaying organic matter and also have traces in the nose and mucus of healthy humans [2]. Although the root causes of a person being infected with black fungus are still uncertain, in most cases, people get infected when they get in contact with the fungal spores present in the environment, especially when a person is having any kind of skin trauma like a cut, scrape, burn, etc [3]. They also noted that this infection affects people who have other health issues like diabetes, cancer, organ transplants, or who are using other drugs which may compromise their immune system [4]. It is a very rare fungal infection, but, in recent times, there has been a tremendous increase in the cases of black fungus compared to the last few years or decades. The doctors and medical experts suggest that this may be because of the spread of COVID-19 [5]. The primary reason for this is the use of various steroids and drugs which are used in the treatment of COVID-19, which helps reduce the inflammation in the lungs while the body's immune system is going into overdrive to combat coronavirus. However, then it lessens inside the affected person's immunity stage and additionally shoots up blood sugar levels in both diabetics and non-diabetic Covid-19 sufferers [6]. It's the concept that this drop in immunity triggers and puts the patients at extra risk for mucormycosis. The mortality rate has also increased significantly because of this. Some of the recognized symptoms of black fungus include fever, cough, chest pain, headache, swelling and redness of eyes discoloration over the eyes, nose, and sides of one's face, blurred vision, and in severe cases loss of vision, blood vomits, and shortness of breath [7]. It shows the severe impact on the brain and other internal organs if not detected and treated on time. The primary aim of this project is to analyze and predict the probability of a person being infected with mucormycosis based on the black fungus symptoms with help of a black fungus disease detection model using deep neural networks [8].

We have used four different models to train and compare the results. The models used are Inception v3, VGG16, ResNet50, and AlexNet.

A. Inception V3

The Inception v3 model is a popular image recognition model. On the ImageNet dataset, they have proved it to achieve higher than 78.1 percent accuracy. The model has 48 layers [9].

Performance Analysis of Deep Learning Techniques Detecting Black Fungus

Convolutions, average pooling, max pooling, concatenations, dropouts, and completely connected layers are some of the symmetric and uneven building blocks [10].

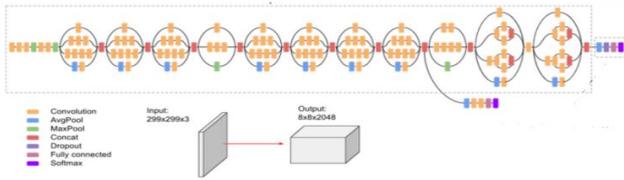


Fig. 1.1 Inception V3 Architecture

Batch normalization is applied to activation inputs and is used extensively throughout the model. Softmax is used to calculate the loss. The very final layer includes a thick layer with a sigmoid function as an activation feature [11].

B. VGG16

VGG16 is a convolution neural network. The network contains convolution layers of a 3x3 filter with a stride 1. VGG16 was one of the best performing architectures in the ILSVRC Imagenet competition held in 2014 [12].

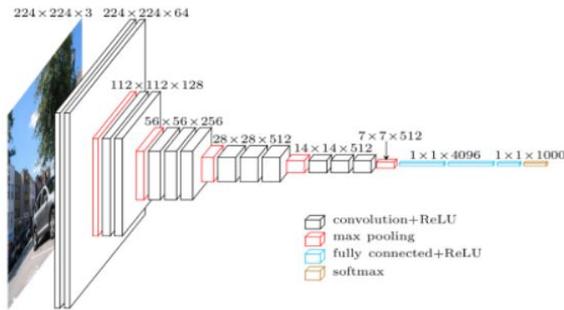


Fig. 1.2 VGG16 Architecture

The same padding and maxpool layer of a 2x2 filter of stride 2 are used throughout. There is a consistent arrangement of convolutional and max pool layers used throughout the architecture of the network. In the end, it has 2 fully connected layers followed by a softmax, which is removed to use the sigmoid function for output. The number 16 in VGG16 refers to the network having 16 layers that have weights. This network is pretty large, and it has about 138 million parameters.

C. ResNet-50

ResNet was one of the best-performing architectures in the ILSVRC Imagenet competition held in 2015. It has 152 layers. Residual is what the Res in the ResNet stands for. The ResNet50 is a deep residual network. The deep residual network comprises an identity connection between the layers. An identity connection is a connection from the input to the end of the residual block.

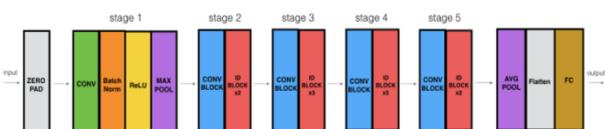


Fig. 1.3. ResNet-50 Architecture

The ResNet-50 model consists of 5 stages. Each stage consists of a convolution and identity block. The

convolution and identity blocks contain 3 convolution layers each. The ResNet-50 has 23 million trainable parameters.

D. AlexNet

While the AlexNet architecture was first developed in 2012 to answer the ImageNet classification issue, it yielded first-rate outcomes. Its top-1 and top-5 accuracy had been 57 percent and 80.3 percent, respectively. Dropout becomes additionally integrated to shield the version against overfitting. Introducing Dropout ended in a discount of some million parameters from AlexNet's 60 million. AlexNet is made up of eight layers. Those layers are composed of five convolutional layers and three completely connected layers. AlexNet makes use of filters with sizes of 11x11, 5x5, and 3x3. AlexNet has almost sixty-two million parameters.

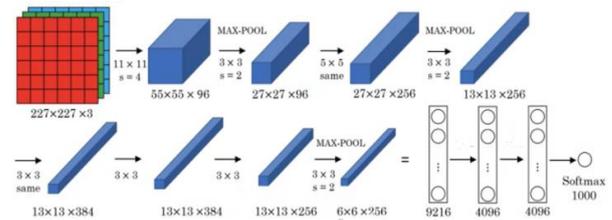


Fig. 1.4. AlexNet Architecture

II. LITERATURE SURVEY

A. Fungus Detection using Computer Vision and Machine Learning Techniques

Yeast and yeast-like fungi are an element of the human microbiota. These pathogens could be the main reason for systemic infections. One of the main reasons that cause fungal infections is impaired characteristics of the immune system and imbalanced microbiota composition within the human body. Different elements of fungal infections consist of steroid remedies, invasive clinical approaches, and lengthy-time period antibiotic remedies with a wide spectrum of antimicrobial agents. The same old manner in mycological diagnostics starts with collecting diverse forms of materials like swabs, scraps of pores and skin lesions, urine, blood, or cerebrospinal fluid. Then, the scientific substances are at once cultured on unique media, at the same time as the blood and cerebrospinal fluid samples require previous cultivation in automated closed structures for extra 2-3 days. Material incubates below particular temperature situations (typically for 2-4 days in case of yeast-like fungi). In this paper, they practice a machine learning approach primarily based on deep neural networks and bag-of-words strategies to categorize microscopic photographs of diverse fungus species. As a result, the ultimate degree of biochemical identification makes no sense, which shortens the identification by about 2-3 days and decreases the cost of analysis. It accelerates the choice of approximately the creation of the right antifungal drug, which prevents the development of the ailment and shortens the time of the affected person's healing.



B. Blood Diseases Detection using Classic Machine Learning Algorithms

Blood has many secrets that have an important effect on human existence. It is the postman that circulates through the body and visits all organs. The growth in age has to be pondered in blood. This variation can be detected with the aid of the values of parameters inside blood analysis tests. Relying on numerous attributes like age, gender, signs, and any fitness conditions. The doctor can select several blood examinations to diagnose the sickness. Many blood examinations are widespread and important for everybody to get. Blood examinations are massive due to that, Maximum doctors might also suggest blood examinations to expect the health stage of the affected person's body. Blood evaluation is a crucial indicator for lots of ailments; it consists of numerous parameters that are an indicator for particular blood ailments. For predicting the disorder in line with the blood evaluation, one needs to figure out the cause of the ailment exactly and have to be identified. Machine learning is the sector answerable for constructing models for predicting the output primarily based on previous records. The data set is collected and confirmed by using professional doctors from highly trusted sources. Numerous classical machine learning algorithms are examined and acquired promising outcomes.

III. PROPOSED SYSTEM ARCHITECTURE

Architectural diagrams present the diagrammatic representation of the summary of a model. It helps us look into the outline of the program and shows us the association between different elements of the model.

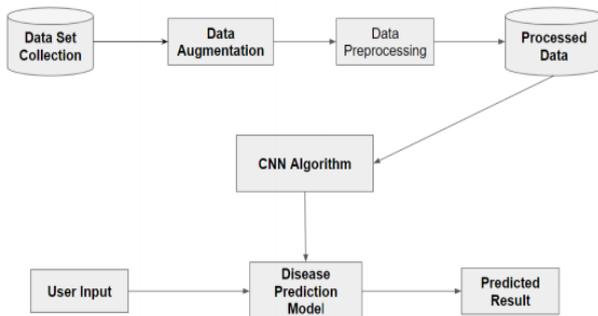


Fig. 3.1 System Architecture

We collected the eye dataset from Google images. The dataset contains Black fungi infected eyes and normal eyes. Since the Black fungi infected eyes are not available in huge numbers on the internet, the images collected are not enough to train a model. To increase the data in the dataset, we next performed Data Augmentation. After this, the images are passed onto Preprocessing. Here, the images are resized and are labeled according to their category. This creates the database required for training. We then create the models InceptionV3, VGG16, ResNet50, and AlexNet, which are CNNs used for classification. Now we pass the training data into the models created. The models are trained and the weights of the CNN are updated. Now, we take the input eye image for the prediction. The image is preprocessed and given as input to the trained models. The models are then used to predict the probability of the eye being infected. Then, based on the probability we predict if the eye is

infected or not. These results have one of the two values, “infected” or “normal”.

IV. DEVELOPMENT FRAMEWORK

A. Software Installation

- 1) Step-1: Visit the following website:
<https://www.anaconda.com/products/individual>
- 2) Step-2: Download anaconda software according to your system requirements.
- 3) Step-3: Now Run the downloaded installer
 - a. Right-click on the installer
 - b. Select Run as Administrator and click on Yes
- 4) Step-4: Read the license agreement and click on I agree.
 - a. Click on next.
 - b. Note your installation location and click on next.
- 5) Step-5: Click on Finish.
- 6) Step-6: Open command prompt
 - a. On the taskbar, to the bottom left, there will be an option to search. Type cmd in search and open the command prompt.
 - b. Execute the following command to check if python is properly installed.
`conda --version python --version`
- 7) Step-7: Execute the following command to update pip.
`pip install --upgrade pip`
- 8) Step-8: Install pandas by executing the following command.
`pip install pandas`
- 9) Step-9: Install tensorflow by executing the following command.
`pip install tensorflow`
- 10) Step-10: Install numpy by executing the following command.
`pip install numpy`
- 11) Step-11: Install opencv by executing the following command.
`pip install opencv-python`
- 12) Step-12: Install scikit-learn by executing the following command.
`pip install opencv-python`

B. Execution

1. Image input: We took 30% of the dataset for testing using `train_test_split` from the `sklearn` library.
2. Preprocessing the images: We have used the `cv2` library to preprocess our images. We have used `imread` to load the image into a variable.
`img= cv2.imread(path,cv2.IMREAD_COLOR)`
Then we have resized the images using the `resize` function from the `cv2` library.
`img=cv2.resize(img,dsize=(150,150),interpolation=cv2.INTER_CUBIC)`
Now we have converted the images to numpy arrays.
`preprocessed_img=img_to_array(img)`

Performance Analysis of Deep Learning Techniques Detecting Black Fungus

3. Prediction using the model: We input the preprocessed image into the model using the predict function.

```
prediction=model.predict(preprocessed_img)
```

4. Since the model outputs the probability values in the range of 0 to 1, we have used a threshold of 0.8 to predict the eye is infected.

```
If(prediction>0.8): Pred_labels.append('infected')
Else: Pred_labels.append('normal')
```

5. We have converted the obtained results to a csv file.

```
df=DataFrame(res,columns=['Actual','Predicted'])
df.to_csv()
```

V. EXISTING APPROACHES

There are quite a few options. For the detection of fungal spores, three computer vision-based approaches were developed. One of them made advantage of HOG-based features and got good results. To achieve promising results, other strategies included a combination of Fourier transform and SIFT characteristics. Superpixel and handmade characteristics are used in the third method. All of these approaches' results point to the possibility of detecting fungal spores from clay particles early.

VI. EXPERIMENTAL RESULTS

Initially, we import all the necessary packages. Then, we create an object of the ImageDataGenerator class by providing all the required attribute values like rescale, rotation_range, etc.

We use the flow function and provide the path of the directory to store the augmented images and generate a required number of Images from the ImageDataGenerator object. Preprocessing is done on the image by using imread, resize functions from the cv2 library.

Now we take the image data and the labels as numpy arrays and split the data into training and test data using train_test_split.

Then we imported the required network from the respective library and created an object of the class by giving input_shape, weights values. Flatten, Dropout, and Dense layers are added to the existing model. We provided the activation functions as relu and sigmoid for Dropout and Dense layers, respectively. We have compiled the model and stored the final version in a variable by giving the following attribute values:

```
loss='binary_crossentropy'
optimizer='adam'
metrics=['accuracy']
```

Next, we trained the model by giving X_train, y_train as inputs and giving values of epochs as 10, batch_size as 8, and validation split as 0.25.

After the training is done, we have predicted the test results by inputting the X_test to the function predicted to the model and the results produced are in y_pred. We have calculated the confusion matrix and accuracy by using metrics from the sklearn library. Now, the model is ready to predict if the eye is infected or not for a given image.

The images shown below are the Infected and normal eye datasets that are used.



Fig. 6.1. Infected Eye



Fig. 6.2. Normal Eye

Batch size=8, learning rate=0.001				
Epoch	VGG16	ResNet50	AlexNet	Inception V3
5	0.9395	0.5531914894	0.5744680851	0.6612903226
8	0.939516	0.609929078	0.8581560284	0.7741935484
10	0.9516129032	0.7588652482	0.6453900709	0.75
12	0.9395	0.8085106383	0.609929078	0.693548
15	0.887	0.7517730496	0.7730496454	0.709
20	0.959677	0.6737588652	0.609929078	0.709
Batch size=8, learning rate=0.002				
Epoch	VGG16	ResNet50	AlexNet	Inception V3
5	0.95161	0.5460992908	0.8794326241	0.6330645161
8	0.94758	0.5460992908	0.7872340426	0.7137096774
10	0.92338	0.5460992908	0.8368794326	0.709
12	0.9153	0.5460992908	0.829787234	0.7459677419
15	0.9516129032	0.5460992908	0.6312056738	0.713
20	0.94758	0.5460992908	0.5035460993	0.681
Batch size=8, learning rate=0.003				
Epoch	VGG16	ResNet50	AlexNet	Inception V3
5	0.9435	0.5460992908	0.5531914894	0.7177419355
8	0.883064	0.5460992908	0.4113475177	0.721774193
10	0.8951	0.5460992908	0.9290780142	0.705
12	0.9032	0.8156028369	0.8085106383	0.66129
15	0.939516129	0.8156028369	0.4822695035	0.725
20	0.88709	0.8368794326	0.695035461	0.691

Fig. 6.3. Performance Table 1: Accuracy



Batch size=8, learning rate=0.001				
Epoch	time taken - vgg16	time taken - ResNet50	time taken - AlexNet	time taken - Inception V3
5	10	2	1	2
8	11	3	3	3
10	20	4	3	4
12	13	4	4	5
15	29	5	5	6
20	31	7	6	8

Batch size=8, learning rate=0.002				
Epoch	time taken - vgg16	time taken - ResNet50	time taken - AlexNet	time taken - Inception V3
5	10	2	2	2
8	11	2	3	3
10	19	4	3	4
12	17	4	4	5
15	29	5	5	6
20	30	6	7	9

Batch size=8, learning rate=0.003				
Epoch	time taken - vgg16	time taken - ResNet50	time taken - AlexNet	time taken - Inception V3
5	10	2	2	2
8	11	3	3	3
10	20	4	3	4
12	22	4	4	5
15	29	6	5	6
20	31	7	6	8

Fig. 6.4. Performance Table 2: Time-Taken

The maximum accuracy is attained by the VGG16 architecture. We can also see that VGG16 took the longest time to execute among all four models.

The graphs for the accuracy vs epochs and time-taken vs epochs for the learning rates 0.001, 0.002, 0.003 respectively are plotted as shown below.

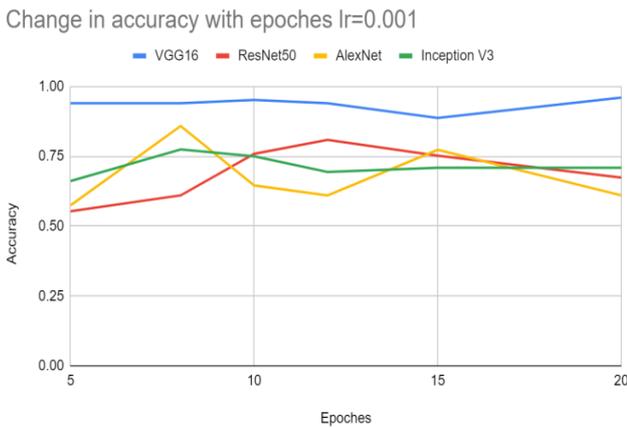


Fig. 6.5. Accuracy vs Epochs lr=0.001

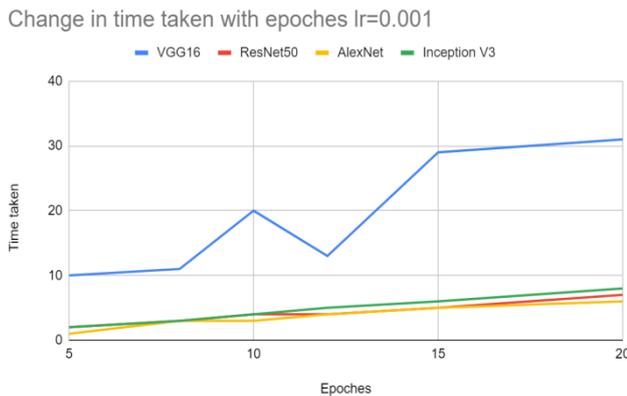


Fig. 6.6. Time taken vs Epochs lr=0.001

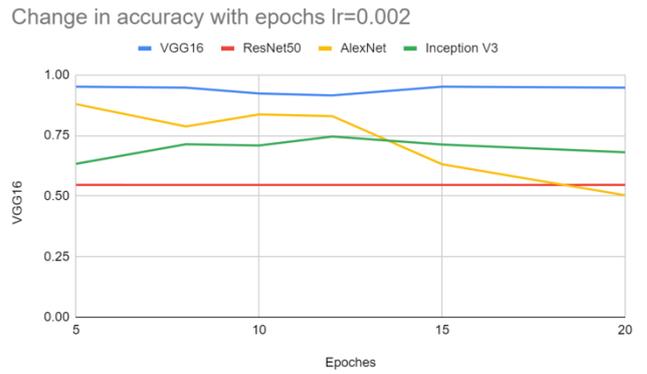


Fig. 6.7. Accuracy vs Epochs lr=0.002

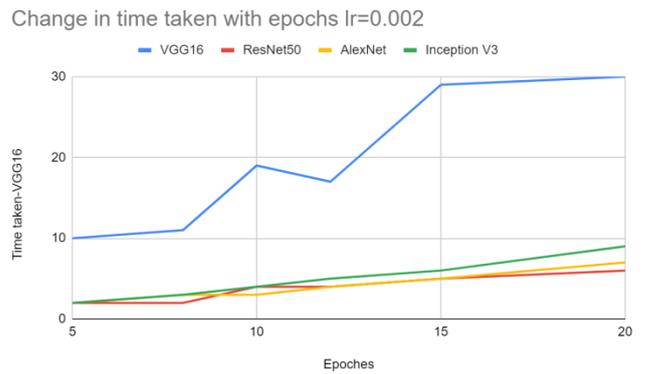


Fig. 6.8. Time taken vs Epochs lr=0.002

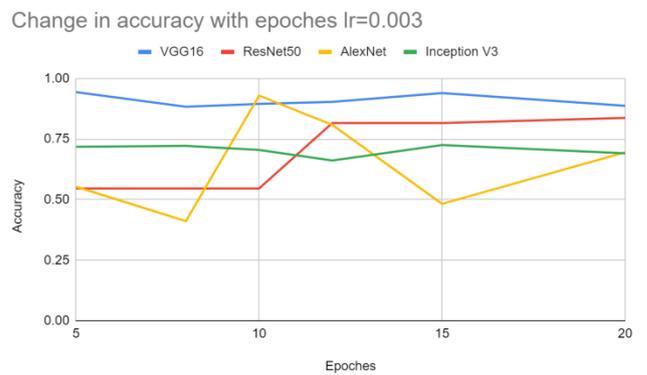


Fig. 6.9. Accuracy vs Epochs lr=0.003

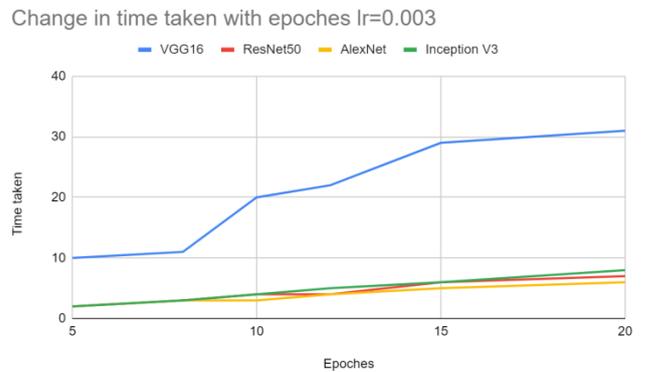


Fig. 6.10. Time is taken vs Epochs lr=0.003

Performance Analysis of Deep Learning Techniques Detecting Black Fungus

As the number of Epochs increased the time taken by the models also increased. Also VGG16 takes the longest time while the other models take almost similar amounts of time. This is because in the VGG16 the volume of the tensors decreases slowly compared to the other models. The VGG16 beats all the other models at accuracy. The major reason behind this is that the VGG16 doesn't have large kernel-sized filters. It contains multiple 3x3 filters throughout the network.

VII. PSEUDO CODE

```
//Load the dataset
//Load the image
img = cv2.imread(os.path.join(cpath,img),
cv2.IMREAD_COLOR)
//resize the images
res = cv2.resize(img, dsize=(150,150),
interpolation=cv2.INTER_CUBIC)
//passing into image recognition model, for example
InceptionV3
pre_trained_model = InceptionV3(input_shape = (150, 150,
3), # Shape of our images include_top = False, # Leave out
the
last fully connected weights = 'imagenet')
// Flatten the output layer to 1 dimension
x = layers.Flatten()(pre_trained_model.output)
// Add a fully connected layer with 1,024 hidden units and
ReLU activation function.
x = layers.Dense(1024, activation='relu')(x)
// Add a dropout rate of 0.2
x = layers.Dropout(0.2)(x)
// Add a sigmoid layer as the final layer for classification
x = layers.Dense(1, activation='sigmoid')(x)
// Compiling the model and storing the final version in a
variable
//Training the model
model.fit(X_train, y_train, batch_size = 8, epochs = 10,
validation_split=0.25)
//predicting the test set
//Calculating the accuracy and confusion matrix
```

VIII. RESULT AND DISCUSSION

A critical issue of black fungus is addressed in this study. It is an important component of the ecosystem, but it is also well-known for posing a serious threat to human life. Existing technologies for detecting black fungus are time-consuming, inefficient, and inconvenient. Furthermore, traditional approaches necessitate the purchase of expensive equipment as well as technical skills.

A revolutionary and entirely automatic black fungus detection system is proposed that can detect people infected with black fungus using their iris samples, allowing us to take prompt preventative actions to save human lives. In this study, image data from the web was mined to see if a person was infected with the black fungus during the epidemic.

Deep Neural Networks and computer vision techniques were utilized to classify photos as infected or normal, allowing the model to determine whether or not the individual is afflicted with black fungus.

Data Augmentation strategies are used to improve the

dataset. VGG16, InceptionV3, ResNet, and AlexNet were used to test several deep learning approaches for detecting black fungus from eye image datasets. VGG16 was discovered to have a high accuracy of 95% when compared to other approaches. Other models take less time to complete. VGG16, on the other hand, takes longer to reach convergence. It has a high level of precision. To determine the performance consequences, several epoch numbers and learning rates were used. The findings of this study, on the other hand, should aid researchers in better understanding how to analyze images contaminated with black fungus during the global COVID-19 outbreak. Furthermore, this research can help the government and legislators make key decisions and take action in the fight against the black fungus and COVID-19. Hybridization with different algorithms can be explored for having faster and more accurate responses.

REFERENCES

1. Zieliński B, Sroka-Oleksiak A, Rymarczyk D, Piekarczyk A, Brzychczy-Wloch M (2020) Deep learning approach to describe and classify fungi microscopic images PloS one, Public Library of Science one, PLoS 1 2020
2. Suman Dee, Baisakhi Chakraborty Disease Detection System (DDS) Using Machine Learning Technique Machine Learning with Health Care Perspective 2020.
3. Fahad Kamal Alsharif Blood Diseases Detection using Classical Machine (IJACSA) International Journal of Advanced Computer Science and Applications 2019.
4. Muhammad Waseem Tahir Fungal Detection using Computer Vision and Machine Learning Open Access Theses and Dissertations 2019
5. Monika A. Myszczyńska Fungus Detection Using Computer Vision and Machine Learning Techniques 2020
6. John I Pitt, Ailsa Diane Hocking, et al. Fungi, and food spoilage. Vol. 519. Springer, 2009.
7. Chandra B Louise and Tom G Obring. "Specific interaction of Escherichia coli 0157: H7-derived Shiga-like toxin II with human renal endothelial cells". In: Journal of Infectious Diseases 172.5(1995)
8. Hena-Da Cheng et al. "Color image segmentation: advances and prospects". In: Pattern recognition 34.12(2001),pp.2259-2281.
9. Li Bin and Mehdi Samiei Yeganeh. "Comparison for image edge detectors for image segmentation". In: IOSR Journal of Computer Engineering 2.6 (2012), pp.1-4.
10. Pinaki Pratim Acharjya, Ritaban Das, and Dibyendu Ghoshal." Study and comparison of different edge detectors for image segmentation". In: Global Journal of Computer Science and Technology(2012).
11. Gabriel Leiva et al. "The automatic sorting using image processing improves post-harvest blueberries storage quality". In: Proceedings of the international congress on engineering and food. 2011.
12. Krishna Kant Singh and Akansha Singh. "A Study of Image Segmentation Algorithms For Different types of Images". In: International Journal of Computer Science Issues 7.5(2010).

AUTHORS PROFILE



Mallikarjuna Rao Gundavarapu, working as a professor in the Computer Science and Engineering Department of Gokaraju Rangaraju Institute of Engineering and Technology. He has a rich academic experience of 32 years. He has been teaching many technologies including Design and Analysis of Algorithms, Micro Controllers & Internet of things, and Digital Logic Design. He has mentored students to help them in their projects, research, and advancing their careers. He is interested in the domains of Pattern Recognition, IoT, and Soft computing. He is the instrument in setting up the "Parallel Computing and Operating Systems Lab" under the MODROB AICTE funded project.



Bhargavi Lella is a final year Computer Science and Engineering student at Gokaraju Rangaraju Institute of Engineering and Technology. Her skills include Data Structures, Algorithms, SQL, Python, and Machine Learning. She is actively learning new technologies every day. Her interests are in the field of Data Science. She has worked on various basic algorithms of Machine learning and Deep Learning. She has previously worked on the project Toxic Comment Classifier as a part of a team. They have built a model which classifies a comment into different classes of toxicity. She has also worked on solo mini projects like Titanic Survival prediction. She is currently learning technologies like Cloud and UNIX.



Shriya Manda is a final year Computer Science and Engineering student at Gokaraju Rangaraju Institute of Engineering and Technology. Her skills include Data Structures, Database Management Systems, Python, Machine Learning and AWS Cloud Foundations, and AWS Cloud Architecting. She is interested in Cloud Computing, Data Science, Customer Relationship Management, and Salesforce Services. Recently, She completed the AWS Cloud

Virtual Internship for ten weeks supported by AWS academy. She successfully completed a couple of projects including, Smart Security Camera used to monitor activity in and around your house working with several Image Processing Techniques and Toxic Comment Classifier which classifies a comment under different types of toxicity.



Lakshmi Prasanna Nekkanti is a final year Computer Science and Engineering student at Gokaraju Rangaraju Institute of Engineering and Technology, Hyderabad. She is interested in the Internet of Things (IoT), cloud computing, and machine learning. She is passionate and strives hard to learn new technologies every day. She has previously worked on the project “ Smart Parking

System”, where she has worked with various sensors, made an effort for the concept of “smart cities” by building an web application that shows the real-time information of parking slot status. Currently she is working on a cloud migration project and learning various cloud computing technologies.



Prerana Panchareddy is a student of Gokaraju Rangaraju Institute of Engineering and Technology. She is a final year Computer Science and Engineering student. She is enormously interested in Data Structures. As we know that Data Structures provide the right way to organize information in the digital space, it is the key component of Computer science. She has exerted herself in numerous basic

algorithms of Machine learning and Deep learning. She has dealt with a project called Toxic Comment Classifier. She is acquiring new skills daily. These are largely used in the areas of Artificial Intelligence, Machine Learning operating systems, graphics etc.