VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

CSE - 3013 Artificial Intelligence

Skin Cancer Detection

Submitted by -

Aayushi Tiwari 19BCE0666
Archisha Sharma 19BCE0502
Tanishq Shah 19BCE2067
Vansh Aggarwal 19BCE0987

Submitted to -

Rajakumar K

ABSTRACT

- *An increasing instant of skin cancers every year with regards to malignant melanoma, the dangerous type of skin cancer, has been observed. The detection of skin cancer is difficult from the skin lesion due to artifacts, low contrast, and similar visualization like mole, scar etc. Hence Automatic detection of skin lesion is performed using techniques for lesion detection for accuracy, efficiency and performance criteria.*
- *Early detection of skin cancer is very important as it is one of the dangerous forms of cancer spreading vigorously among humans. With the advancements in the machine learning domain and Deep Neural Network, we now have ways and ability to construct systems that real time detect cancer which is not only faster than humans can but also possibly more accurate than humans. In the proposed system we make use of a Convolutional Neural Network (CNN). Computer based skin cancer detection is more advantageous to patients, by which patients can identify the skin cancer without going to hospital or without the help of a doctor. Computer based detection uses imaging techniques and Artificial Intelligence.*

INTRODUCTION

This project aims to identify the type of skin cancer using image classification using deep convolutional neural networks. Automated classification of skin cancer using images may be difficult depending upon the visibility of skin lesions. Skin cancer is the uncontrolled growth of abnormal skin cells. There are three main types of skin cancer namely-

1. Melanoma (including nodular melanoma)
2. Basal cell carcinoma
3. Squamous cell carcinoma

This carcinoma can be further classified into 2 categories namely benign and malignant. In benign, the carcinoma is localized and doesn't spread to the surrounding structures. Whereas in case of malignant, it spreads to the surrounding areas and distant organs by the process of metastasis. This type of skin cancer is very harmful and can lead to severe systemic complications. The prognosis of benign is good, but for malignant, it is very poor. These skin cancers often start as changes in an individual's skin and then proceed to becoming lesions. These lesions can be detected and are thus used to detect the type of cancer caused by the tumor cells that start to multiply, leading to the genesis of cancer in a body. This nonmelanoma skin cancer may appear as a firm red nodule, a scaly growth that bleeds or develops a crust, or a sore that doesn't heal. It most often occurs on the nose, forehead, ears, lower lip, hands, and other sun-exposed areas of the body. Squamous cell carcinoma is curable if caught and treated early. Cumulative sun exposure causes mainly basal cell and squamous cell skin cancer, while episodes of severe sunburns, usually before age 18, can cause melanoma later in life. Other less common causes are repeated X-ray exposure, scars from burns or disease and occupational exposure to certain chemicals. It is proven that more than one-third of the skin cancer lesions itch. Melanoma lesions were the least likely to be painful or itchy. Other skin cancers, especially basal cell carcinoma and squamous cell carcinoma, were more likely to be itchy or painful. The CNN is trained using a clinical dataset to verify the correct type of skin cancer using just images of the lesions. One of the most dangerous types of skin cancer, Melanoma, develops when unrepaired DNA damage to skin cells triggers mutation leading to the skin cells multiplying rapidly and forming tumors which can be fatal if not treated at the right time. Thus, to avoid such dangerous consequences, detection of cancer is very important.

LITERATURE REVIEW

[1] Skin malignant growth is the uncontrolled development of weird skin cells. It happens when unrepaired DNA harms skin cells which further triggers changes, or hereditary deformities, that could eventually lead the skin cells to duplicate promptly and structure threatening tumours. Picture preparation is a normally utilized technique for skin malignant growth identification from the presence of an influenced zone on the skin. Artificial Neural Network (ANN) is one of the significant parts of Artificial Intelligence, which has been acknowledged as a fresh out of the plastic new innovation in software engineering for picture handling. Neural Networks are as of now the territory of enthusiasm for medication, especially in the fields of radiology, urology, cardiology, oncology, and so on. Neural Network assumes a crucial job in an exceedingly call arrange. In this paper, an electronic strategy has been created to utilize Neural Networks in the field of medicinal picture preparation. A definitive point of this paper is to actualize savvy crisis emotionally supportive networks, to process the medicinal pictures. It has been utilized to examine Melanoma parameters Like Asymmetry, Border, Colour, Diameter, (ABCD), and so on which are determined utilizing MATLAB from skin malignancy pictures meaning to creating symptomatic calculations that may improve triage rehearses in the crisis office. Diagnosing an obscure skin injury is the first venture to decide fitting treatment.

[2] The author shows that a direct classifier, prepared on highlights extricated from a convolutional neural system pretrained on regular pictures, recognizes among up to ten skin injuries with a higher exactness than recently distributed cutting-edge results on the equivalent dataset. Further, as opposed to contending works, their methodology requires no lesion divisions nor complex preprocessing. An increase in predictable extra upgrades to exactness utilizing a for every picture standardization, a completely convolutional system to separate multi-scale highlights, and by pooling over an enlarged component space. Contrasted with condition-of-the heart, our proposed approach accomplishes an ideal exactness of 85.8% more than 5-classes (contrasted with 75.1%) with recognizable enhancements in precision for underrepresented classes (e.g., 60% contrasted with 15.6%). Over the whole 10-class dataset of 1300 pictures caught from a standard (non-dermoscopic) camera, the strategy accomplishes a precision of 81.8% outflanking the 67% exactness recently revealed.

[3] A large, deep convolutional neural network was trained in this paper to classify the 1.2 million high-resolution images in the ImageNet LSVRC-2010 contest into the 1000 different classes. On the test data, top-1 and top-5 error rates

of 37.5% and 17.0% were achieved which is considerably better than the previous state-of-the-art. The neural network, which has 60 million parameters and 650,000 neurons, consists of five convolutional layers, some of which are followed by max-pooling layers, and three fully-connected layers with a final 1000-way softmax. To make training faster, a non-saturating neuron and a very efficient GPU implementation of the convolution operation were used. To reduce overfitting in the fully-connected layers, a recently-developed regularization method called “dropout” was implemented that proved to be very effective. The authors achieved a test error rate of 15.3%.

[4] Melanoma, the most threatening sort of skin disease, is on the ascent. In this paper a usage of a deep learning framework on a PC server, furnished with a graphic processing unit (GPU), is proposed for recognition of melanoma lesions. Clinical (non-dermoscopic) pictures are utilized in the proposed framework, which could help a dermatologist in early finding of this kind of skin disease. In the proposed framework, input clinical pictures, which could contain illumination and noise impacts, are pre-processed so as to diminish such artefacts. A short time later, the improved pictures are nourished to a pre-prepared convolutional neural network (CNN) which is a member from deep learning models. The CNN classifier, which is prepared by an enormous number of training tests, recognizes melanoma and benign cases. Trial results show that the proposed technique is unrivalled as far as diagnostic precision in comparison with the cutting-edge methods.

[5] Convolutional neural networks (CNN) have recently shown outstanding image classification performance in the large-scale visual recognition challenge (ILSVRC2012). The success of CNNs is attributed to their ability to learn rich mid-level image representations as opposed to hand-designed low-level features used in other image classification methods. Learning CNNs, however, amounts to estimating millions of parameters and requires a very large number of annotated image samples. This property currently prevents application of CNNs to problems with limited training data. In this work, it is demonstrated that image representations learned with CNNs on large-scale annotated datasets can be efficiently transferred to other visual recognition tasks with limited amounts of training data. A method to reuse layers trained on the ImageNet dataset to compute mid-level image representation for images in the PASCAL VOC dataset has been designed. Despite the differences in image statistics and tasks in the two datasets, it is shown that the transferred representation leads to significantly improved results for object and action classification, outperforming the current state of the art on Pascal VOC 2007 and 2012 datasets.

[6] This paper proposes two ways to deal with the skin lesion picture segmentation issue. The first is an essentially locale-based segmentation strategy where an ideal threshold is resolved iteratively by an isodata algorithm. The subsequent technique proposed depends on neural network edge location and a rational Gaussian curve that is an approximate closed flexible curve between the perceived neural network edge designs. A quantitative comparison of the systems is empowered by the utilization of engineered lesions to which Gaussian noise is added. The proposed strategies are likewise contrasted with established skin segmentation techniques. It is shown that for lesions with a scope of various border anomaly properties the iterative thresholding strategy gives the best execution over a range of signal to noise ratios. Iterative thresholding method additionally exhibits comparable execution when tried on genuine skin lesions.

[7] An automatic skin cancer classification framework is created and the relationship of skin cancer pictures over various sorts of neural networks are examined with various kinds of pre-processing. The gathered pictures are fed into the framework, and through various image processing techniques to improve the image properties. At that point the ordinary skin is removed from the skin affected region and the cancer cell is left in the picture. Helpful data can be taken from these pictures and passed to the classification framework for training and testing. Recognition accuracy of the 3-layers back-propagation neural network classifier is 89.9% and auto-associative neural system is 80.8% in the image database that incorporates dermoscopy photographs and advanced photographs.

[8] In this paper, detection of skin cancer lesions as malignant (melanoma) or benign is performed using the CNN ReSNeT34. The performance of this system is studied using the accuracy and error rate with respect to the variations in number of epochs and learning rate. The accuracy increases with a decrease in learning rate. The Maximum accuracy of 90.12% is achieved when lR is decreased to 1x10-6 after 10 epochs. In this work, only the detection of skin cancer is considered. The data sets are processed with pre-trained CNN ResNet 34 networks to classify the type of skin cancer that is either benign or malignant. The final testing stage is done by choosing a random skin lesion image. It is then tested for accuracy and error rate. Skin cancer detection is done with python for Transfer learning.

[9] This paper proposes a classification framework using the innovations made in deep learning. This research aims at investigating the efficiency of pre-trained convolutional neural networks and the transfer learning technique for classifying images of skin lesions. A dataset of images, with seven different types of skin lesions and a total of 10,015 images, was separated into two classes. Skin lesions of the seven forms were classified either malignant or benign. By augmenting the number of images, the dataset was balanced. Numerous pre-trained neural networks were trained on this dataset; ResNet50, VGG16, VGG19, MobileNet, DenseNet, Inception V3. A classification framework was proposed based on their results. The framework suggested combining the three best pre-trained networks to enhance the classification accuracy. The combined model was made up of ResNet50, VGG16, and DenseNet. The addition of convolution layers to each model, that were trainable, further improved the classification accuracy. The overall accuracy achieved was 84.01% for this final proposed framework.

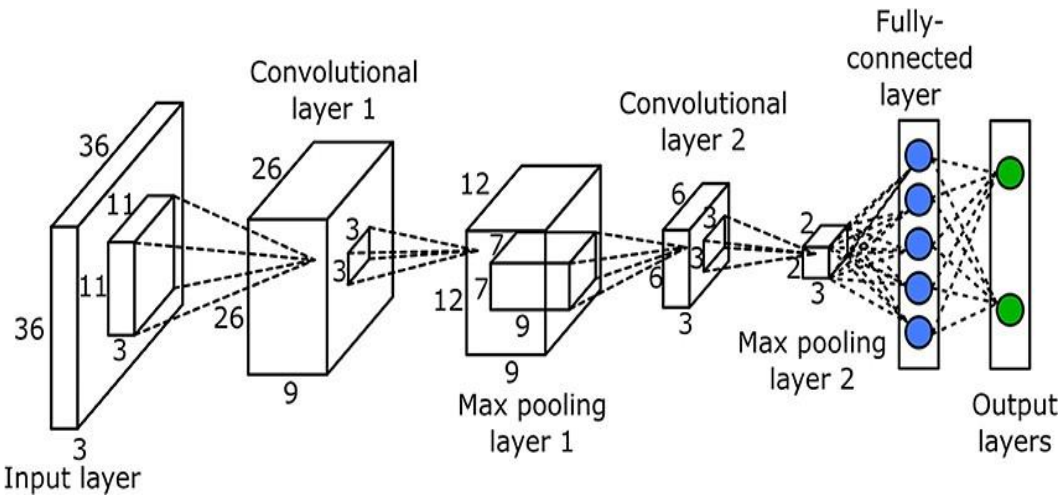
[10] In this paper, a new intelligent method of classifying benign and malignant melanoma lesions is implemented. The system consists of four stages; image pre-processing, image segmentation, feature extraction, and image classification. As the first step of the image analysis, pre-processing techniques are implemented to remove noise and undesired structures from the images using techniques such as median filtering and contrast enhancement. In the second step, a simple thresholding method is used to segment and localise the lesion, a boundary tracing algorithm is also implemented to validate the segmentation. Then, a wavelet approach is used to extract the features, more specifically wavelet packet transform (WPT). Finally, the dimensionality of the selected features is reduced with principal component analysis (PCA) and later supplied to an artificial neural network and support vector machine classifiers for classification. The ability to correctly discriminate between benign and malignant lesions was about 95% for the Artificial Neural Network and 85% for the Support Vector Machine classifier.

PROPOSED WORK WITH ARCHITECTURE

Methodology

We use a custom made Deep convolutional neural network consisting of 3 blocks each consisting of a convolutional layer, a max pooling layer and a batch normalization layer. We also use two dense layers at the end for higher accuracy. In addition to this dropout, l1 and l2 regularization have also been used to make sure the model does not overfit. We have used the relu activation function since it has been recorded to have performed better than all the other activation functions. The optimization algorithm used is Adam since experiments have shown that it has shown improvement over standard algorithms like SGD and Adagrad. We use data augmentation since the total number of images available is only 5000 which is minimal. In case of shift microscopical images we primarily need shift and rotational invariance as well as robustness to deformations and gray value variations.

Architecture



IMPLEMENTATION

Convolutional Neural Networks

Convolutional Neural networks are specialized deep neural networks which can process the data that has input shape like a 2D matrix. Images are easily represented as a 2D matrix and CNN is very useful in working with images. CNN is basically used for image classifications and identifying if an image is a bird, a plane or Superman, etc. It scans images from left to right and top to bottom to pull out important features from the image and combines the features to classify images. It can handle the images that have been translated, rotated, scaled and changes in perspective.

Dataset description:

Skin Cancer MNIST: HAM10000

Training of neural networks for automated diagnosis of pigmented skin lesions is hampered by the small size and lack of diversity of available dataset of dermatoscopic images. We tackle this problem by releasing the HAM10000 ("Human Against Machine with 10000 training images") dataset. It consists of dermatoscopic images from different populations, acquired and stored by different modalities. The final dataset consists of 10015 dermatoscopic images which can serve as a training set for academic machine learning purposes. Cases include a representative collection of all important diagnostic categories in the realm of pigmented lesions:

1. Actinic keratoses and intraepithelial carcinoma / Bowen's disease (akiec)
2. Basal cell carcinoma (bcc)
3. Benign keratosis-like lesions(solar lentigines/seborrheic keratoses and lichen-planus like keratosis, bkl)
4. Dermatofibroma (df)
5. Melanoma (mel)
6. Melanocytic nevi (nv)
7. Vascular lesions (angiomas, angiokeratomas, pyogenic granulomas and hemorrhage, vasc)

Columns in Dataset:

1. lesion_id
2. image_id
3. dx
4. dx_type
5. age
6. sex
7. Localization

Optimization algorithm: Adams Algorithms

Adam is an adaptive learning rate optimization algorithm that's been designed specifically for training deep neural networks. The algorithm leverages the power of adaptive learning rates methods to find individual learning rates for each parameter.

Steps:

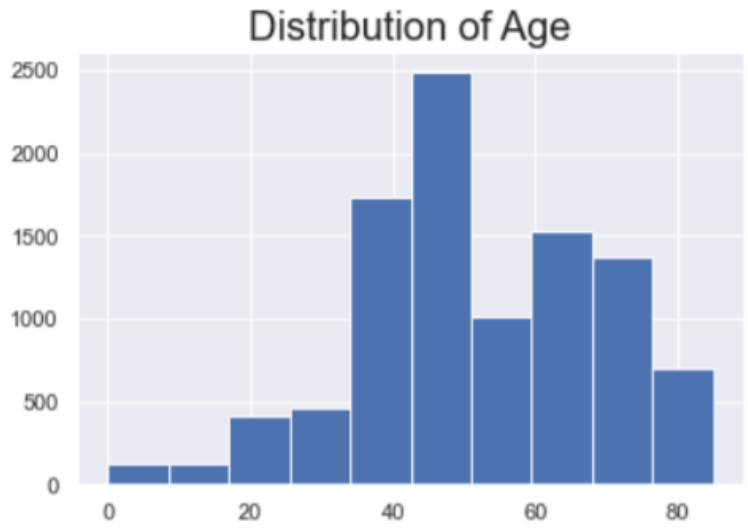
1. An input is given to the neural network.
2. Then the image undergoes “n” number of convolution layers and max pooling layers according to the developer.
3. In convolution, the features are then enhanced in an image to make it easier for the network to understand the patterns.
4. Max pooling down-samples an input representation reducing its dimensionality and allowing for assumptions to be made about features contained in the sub-regions binned.
5. In the end, the neural network outputs a number based on classification it thinks that the image belongs to.

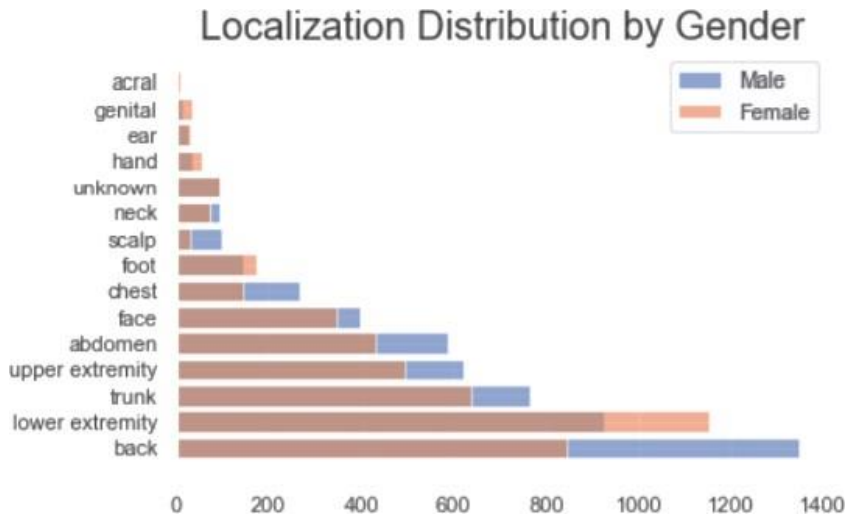
6. After images are passed through the neural network, all the outputs are recorded and compared to their original classification according to the dataset. The binary cross entropy is then calculated based on these results.
7. Then the weights of the nodes of the neural network are later adjusted based on the loss that has occured to reduce them as well as increase the accuracy of the system to predict the classification with higher accuracy than before.

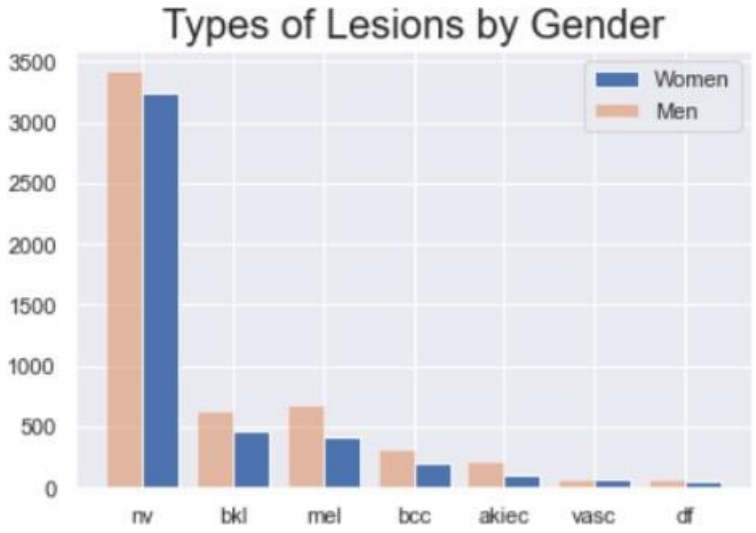
The use of this technology enables us to create a system which is faster and better. It helps us get more and improved accuracy in the results and testing while keeping the technical footprint required to run the model as minimal as possible. In turn, all these sum to increase its accessibility for use by the end-user.

Step-wise implementation of the model -

1. Data import and exploration







In [20]:
```
data_info.head()
```

Out[20]:

	lesion_id	image_id	dx	dx_type	age	sex	localization	pixel_avg
0	HAM_0000118	ISIC_0027419	bkl	histo	80.0	male	scalp	184.320017
1	HAM_0000118	ISIC_0025030	bkl	histo	80.0	male	scalp	177.003825
2	HAM_0002730	ISIC_0026769	bkl	histo	80.0	male	scalp	181.487038
3	HAM_0002730	ISIC_0025661	bkl	histo	80.0	male	scalp	165.210795
4	HAM_0001466	ISIC_0031633	bkl	histo	75.0	male	ear	188.305992

2. Model

In [1]:
```
#imports
from keras.layers import Dense, Conv2D, MaxPooling2D, Flatten, Dropout
from keras.models import Sequential
from keras import optimizers
from keras.preprocessing.image import ImageDataGenerator

import matplotlib.pyplot as plt

import os
from os import listdir
os.environ['KMP_DUPLICATE_LIB_OK']='True' #needed this so that my kernel didn't die
import warnings
warnings.filterwarnings("ignore")
```

```
INFO:tensorflow:Enabling eager execution
INFO:tensorflow:Enabling v2 tensorshape
INFO:tensorflow:Enabling resource variables
INFO:tensorflow:Enabling tensor equality
INFO:tensorflow:Enabling control flow v2
```

In [12]:
```
#list of directories
base_directory = "C:/Users/Vansh Aggarwal/Desktop/ai skin pro run"

test_directory = os.path.join(base_directory, 'image_data_test')
train_directory = os.path.join(base_directory, 'image_data_train')
validation_directory = os.path.join(base_directory, 'image_data_validation')
aug_train_directory = os.path.join(base_directory, 'image_data_train_augmented')
bal_validation_directory = os.path.join(base_directory, 'image_data_test_balanced')
```

In [4]:
```
#importing pre-trained model
# from keras.applications import MobileNetV2
from keras.applications.mobilenet_v2 import MobileNetV2
```

In [5]:
```
#assigning model to variable to add to new model
#top fully connected layers not included
prior = MobileNetV2(include_top =  False, weights = "imagenet", input_shape = (150,
```

In [6]:
```
model = Sequential()

model.add(prior)

model.add(Flatten())
# model.add(Dropout(rate = 0.5))

model.add(Dense(512, activation='relu'))
model.add(Dropout(rate = 0.5))

model.add(Dense(256, activation='relu'))
# model.add(Dropout(rate = 0.5))

model.add(Dense(128, activation='relu'))
model.add(Dropout(rate = 0.5))
```

```
model.add(Dense(64, activation='relu'))

model.add(Dense(7, activation='softmax'))

#freeze MobileNet layers so that imagenet weights can be used
for layer in model.layers[0].layers:
    layer.trainable = False
```

In [7]:
```
model.summary()
```

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
mobilenetv2_1.00_224 (Functi (None, 5, 5, 1280)        2257984
_________________________________________________________________
flatten (Flatten)            (None, 32000)             0
_________________________________________________________________
dense (Dense)                (None, 512)               16384512
_________________________________________________________________
dropout (Dropout)            (None, 512)               0
_________________________________________________________________
dense_1 (Dense)              (None, 256)               131328
_________________________________________________________________
dense_2 (Dense)              (None, 128)               32896
_________________________________________________________________
dropout_1 (Dropout)          (None, 128)               0
_________________________________________________________________
dense_3 (Dense)              (None, 64)                8256
_________________________________________________________________
dense_4 (Dense)              (None, 7)                 455
=================================================================
Total params: 18,815,431
Trainable params: 16,557,447
Non-trainable params: 2,257,984
_________________________________________________________________
```

In [8]:
```
adam = optimizers.Adam(lr = 0.001)
model.compile(loss = 'categorical_crossentropy',
              optimizer = adam,
              metrics = ['acc'])
```

In [9]:
```
train_datagen = ImageDataGenerator(rescale = 1./255)
val_datagen = ImageDataGenerator(rescale=1./255)
```

In [10]:
```
#class weights used to give more emphasis to the melanomic classes
class_weights = {
    0: 1,
    1: 1,
    2: 2,
    3: 1,
    4: 1,
    5: 1,
    6: 2
}
```

In [13]:
```
#Largest target size used for images
#increasing batch size made a difference
train_generator = train_datagen.flow_from_directory(
```

```
        aug_train_directory,
        target_size=(150, 150),
        batch_size= 256,
        class_mode='categorical'
)

validation_generator = val_datagen.flow_from_directory(
        bal_validation_directory,
        target_size=(150, 150),
        batch_size= 256,
        class_mode='categorical')
```

```
Found 29051 images belonging to 7 classes.
Found 196 images belonging to 7 classes.
```

In [14]:

```
for data_batch, labels_batch in train_generator:
    print('data batch shape:', data_batch.shape)
    print('labels batch shape:', labels_batch.shape)
    break
```

```
data batch shape: (256, 150, 150, 3)
labels batch shape: (256, 7)
```

In [15]:

```
from keras.callbacks import EarlyStopping, ReduceLROnPlateau

callbacks = [EarlyStopping(monitor= "val_loss", patience = 5, restore_best_weights=
             ReduceLROnPlateau(monitor="loss", patience = 2)]
```

In [16]:

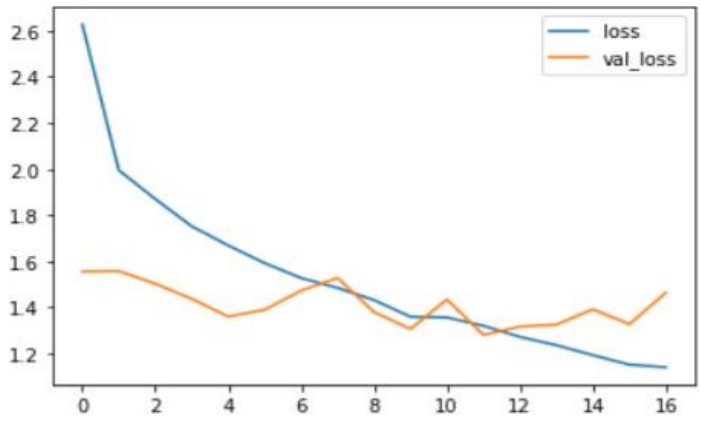
```
import numpy as np

#important to make sure steps are related to the batch size and image count
history = model.fit_generator(
      train_generator,
      steps_per_epoch= len(train_generator),
      epochs=20,
      validation_data=validation_generator,
      validation_steps= len(validation_generator),
      callbacks= callbacks,
      class_weight= class_weights
)
```

In [17]:

```
model.save("mymodel5.h5")
```

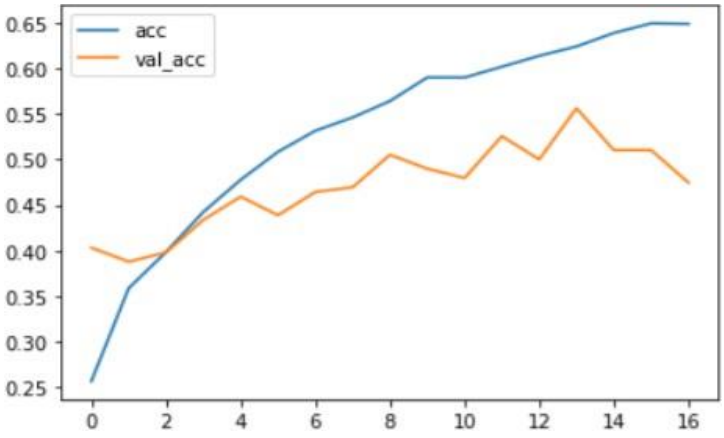
In [18]:

```
#plot of loss for train and val
plt.plot(history.history["loss"], label = "loss")
plt.plot(history.history["val_loss"], label = "val_loss")
plt.legend();
```



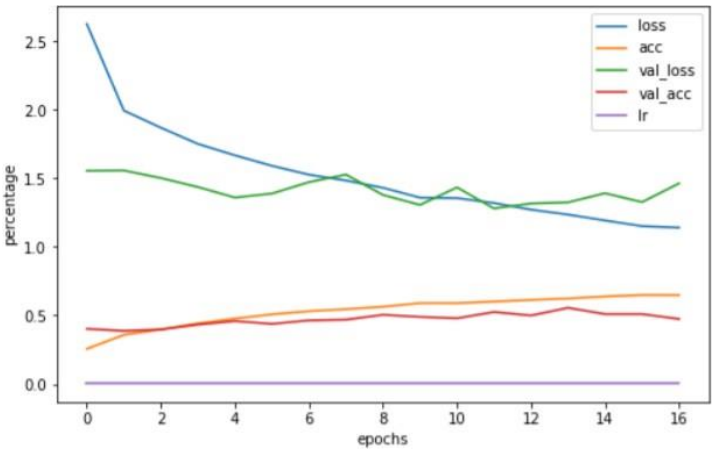
In [19]:

```
#plot of accuracy for train and val
plt.plot(history.history["acc"], label = "acc")
plt.plot(history.history["val_acc"], label = "val_acc")
plt.legend();
```



In [87]:

```
pd.DataFrame(history.history).plot(figsize=(8,5))
plt.xlabel("epochs")
plt.ylabel("percentage")
plt.show()
```



In [20]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

from keras.models import Model, load_model
import keras
from keras.preprocessing.image import ImageDataGenerator

from sklearn.metrics import confusion_matrix, accuracy_score, classification_report

import os
```

In [45]:
```
#directory imports
base_dir = "C:/Users/Vansh Aggarwal/Desktop/ai skin pro run"

test_directory = os.path.join(base_dir, "image_data_test_balanced")
```

In [46]:
```
#instantiating ImageGenerator to pull test images for predictions
test_data_gen = ImageDataGenerator(rescale= 1./255)
```

In [47]:
```
batch_size = 256

test_generator7 = test_data_gen.flow_from_directory(test_directory, target_size = (1

model = load_model("C:/Users/Vansh Aggarwal/Desktop/skin_lesion_cnn-master/skin_lesi
```

```
Found 196 images belonging to 7 classes.
```

In [48]:
```
len(test_generator7)
```

Out[48]:
```
1
```

In [49]:
```
preds = model.predict(test_generator7, steps = len(test_generator7))
preds7 = np.argmax(preds, axis =1)
```

In [50]:
```
y_true = test_generator7.classes
accuracy_score(y_true, preds7)
```

Out[50]:
```
0.5255102040816326
```

In [51]:
```
print(classification_report(y_true, preds7))
```

```
              precision    recall  f1-score   support
```

1	0.60	0.54	0.57	28
2	0.29	0.68	0.40	28
3	0.89	0.29	0.43	28
4	0.57	0.89	0.69	28
5	0.95	0.75	0.84	28
6	0.48	0.43	0.45	28
accuracy			0.53	196
macro avg	0.63	0.53	0.51	196
weighted avg	0.63	0.53	0.51	196

In [52]:
```
pd.DataFrame(confusion_matrix(y_true, preds7),
             columns = ["pred_0", "pred_1", "pred_2", "pred_3", "pred_4", "pred_5",
             index = ["true_0", "true_1", "true_2", "true_3", "true_4", "true_5", "t
```

Out[52]:

	pred_0	pred_1	pred_2	pred_3	pred_4	pred_5	pred_6
true_0	3	4	16	0	0	0	5
true_1	2	15	9	0	1	0	1
true_2	0	3	19	0	4	0	2
true_3	0	2	7	8	9	0	2
true_4	0	0	3	0	25	0	0
true_5	0	0	2	0	2	21	3
true_6	0	1	10	1	3	1	12

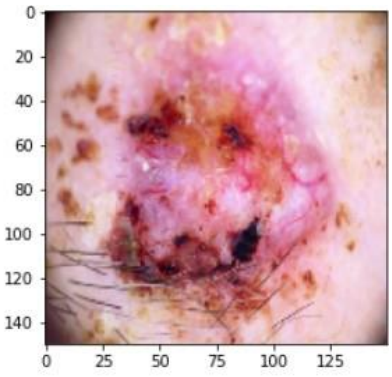
In [53]:
```
print(classification_report(y_true, preds7))
```

	precision	recall	f1-score	support
0	0.60	0.11	0.18	28
1	0.60	0.54	0.57	28
2	0.29	0.68	0.40	28
3	0.89	0.29	0.43	28
4	0.57	0.89	0.69	28
5	0.95	0.75	0.84	28
6	0.48	0.43	0.45	28
accuracy			0.53	196
macro avg	0.63	0.53	0.51	196
weighted avg	0.63	0.53	0.51	196

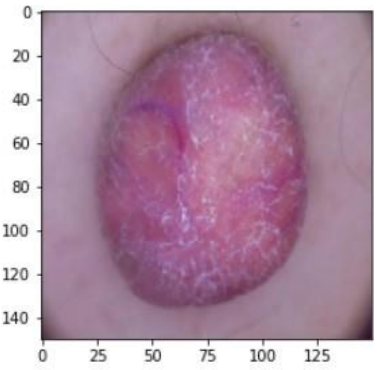
In [69]:
```
from tensorflow.keras.preprocessing import image
from PIL import Image
import PIL
```

In [88]:
```
dir_path='C:/Users/Vansh Aggarwal/Desktop/skin_lesion_cnn-master/skin_lesion_cnn-mas
for i in os.listdir(dir_path):

    img=image.load_img(dir_path+'//'+i,target_size=(150,150))
    img1=img
    plt.imshow(img)
    plt.show()
    x=image.img_to_array(img)
    x=np.expand_dims(x,axis=0)
```

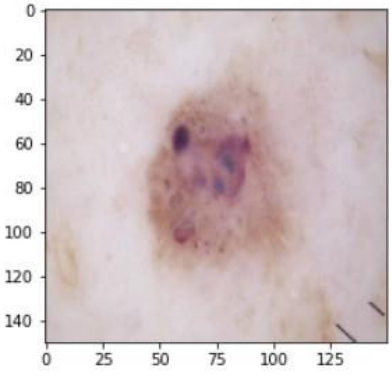
```
    images=np.vstack([x])
#     print(images)
    val=model.predict(images)
    hu=np.argmax(val, axis =1)
    if hu==0:
        print("Melanocytic nevi")
    elif hu==1:
        print("dermatofibroma")
    elif hu==2:
        print("Benign keratosis-like lesions")
    elif hu==3:
        print("Basal cell carcinoma")
    elif hu==4:
        print("Actinic keratoses")
    elif hu==5:
        print("Vascular lesions")
    elif hu==6:
        print("melanoma")
```



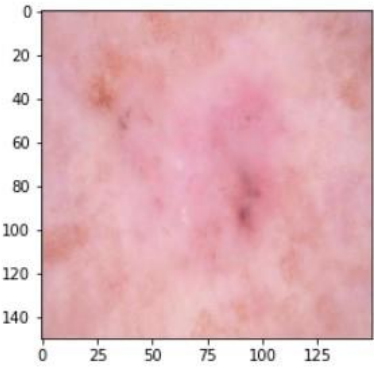
Vascular lesions



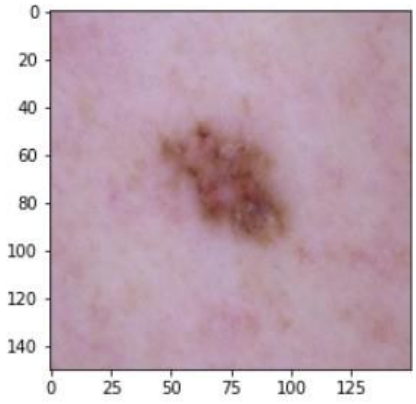
Basal cell carcinoma



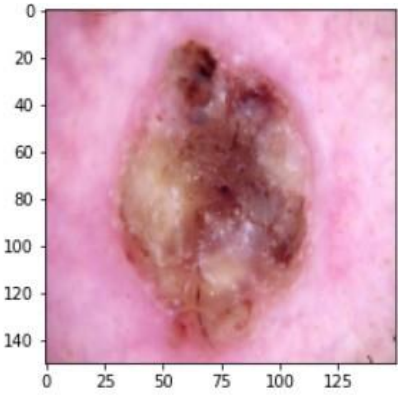
Vascular lesions



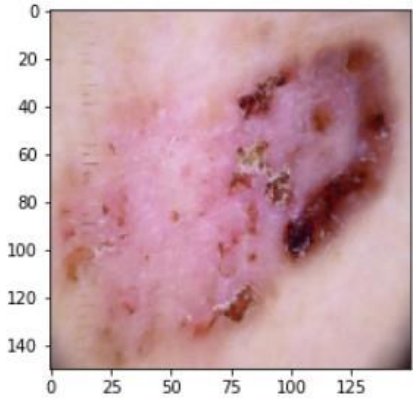
Actinic keratoses



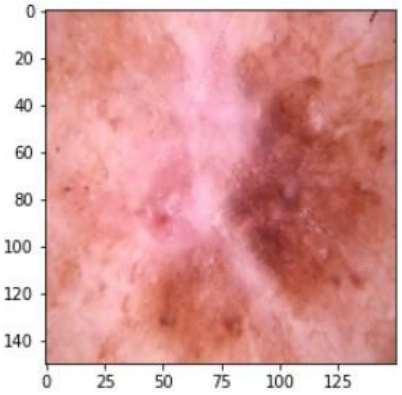
Vascular lesions



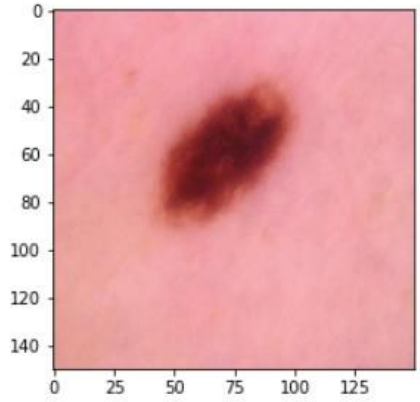
Actinic keratoses



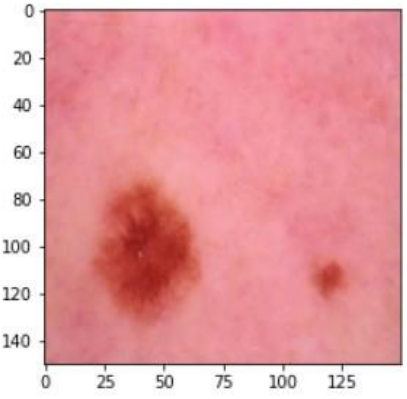
Basal cell carcinoma



Benign keratosis-like lesions

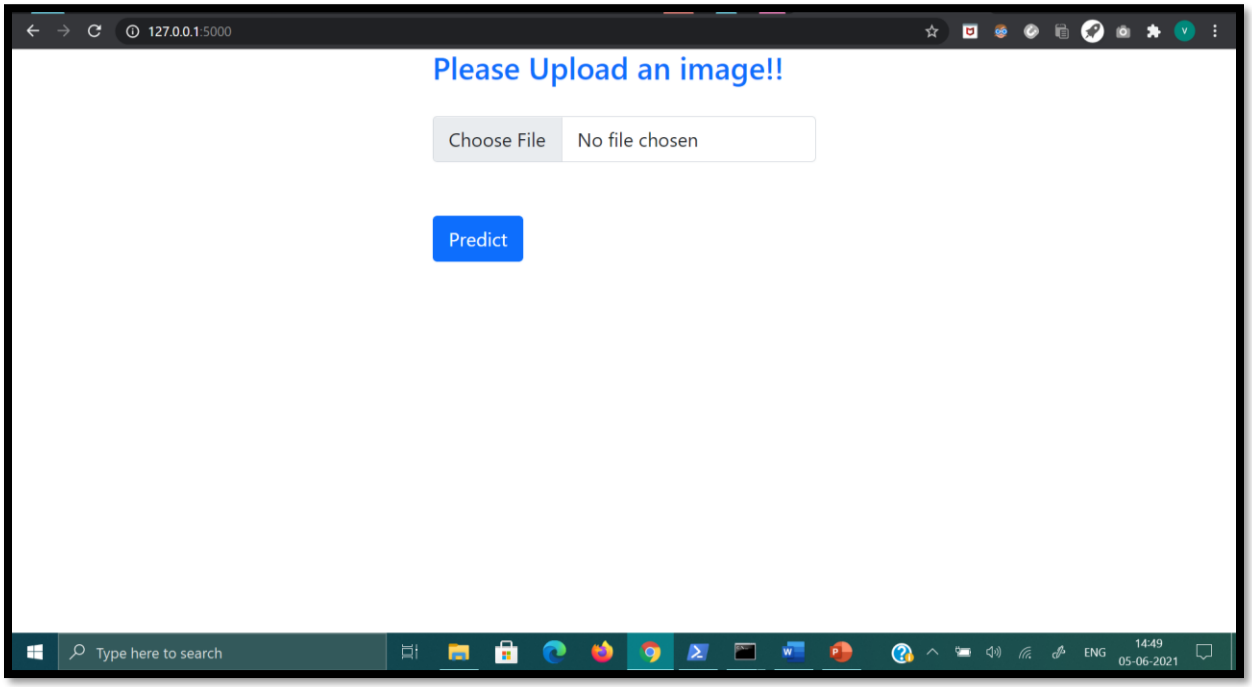


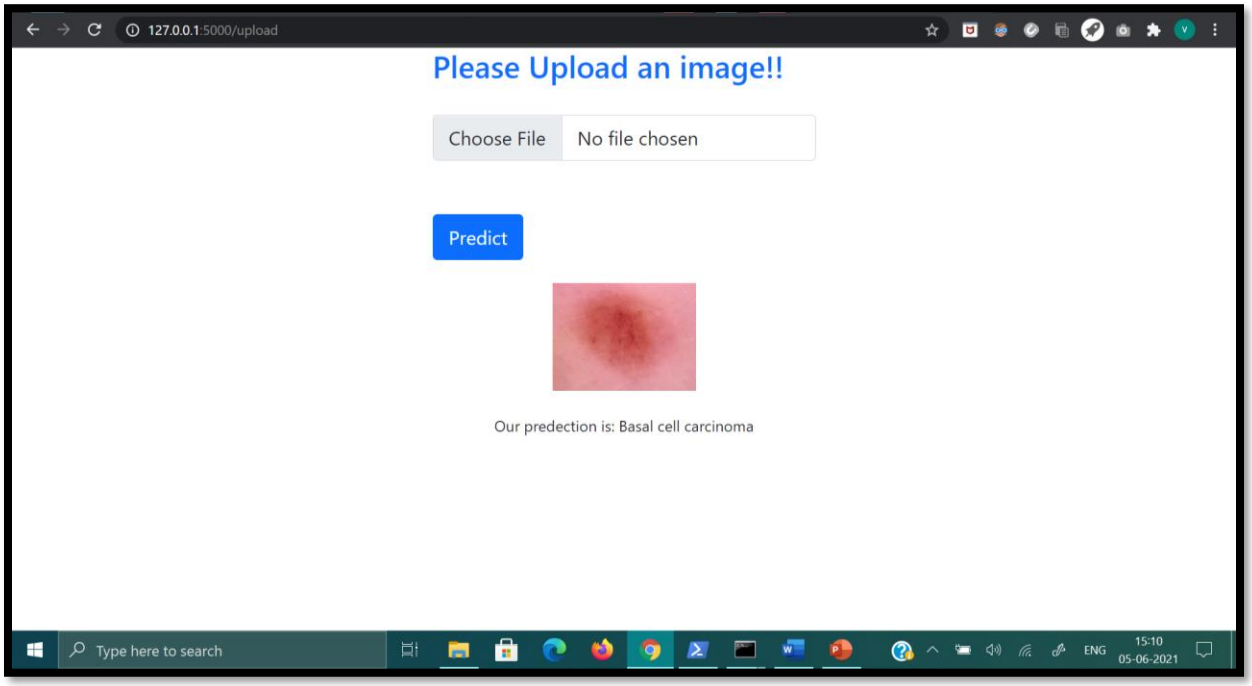
Actinic keratoses

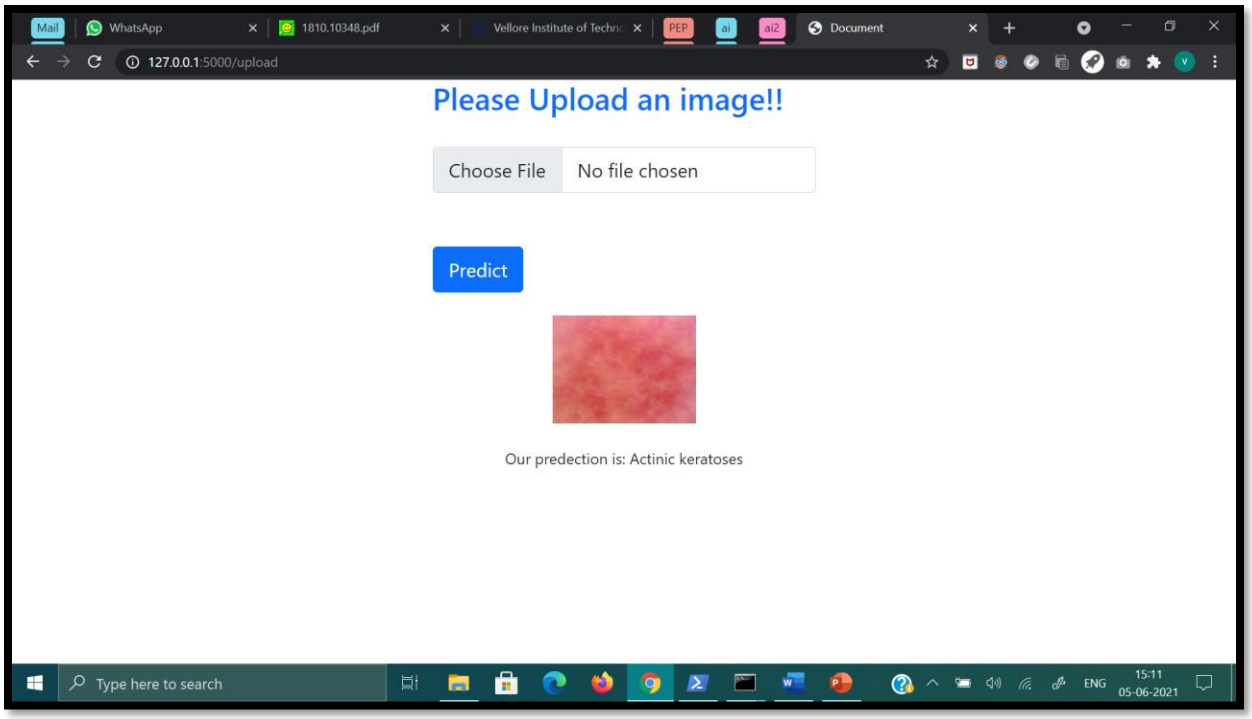


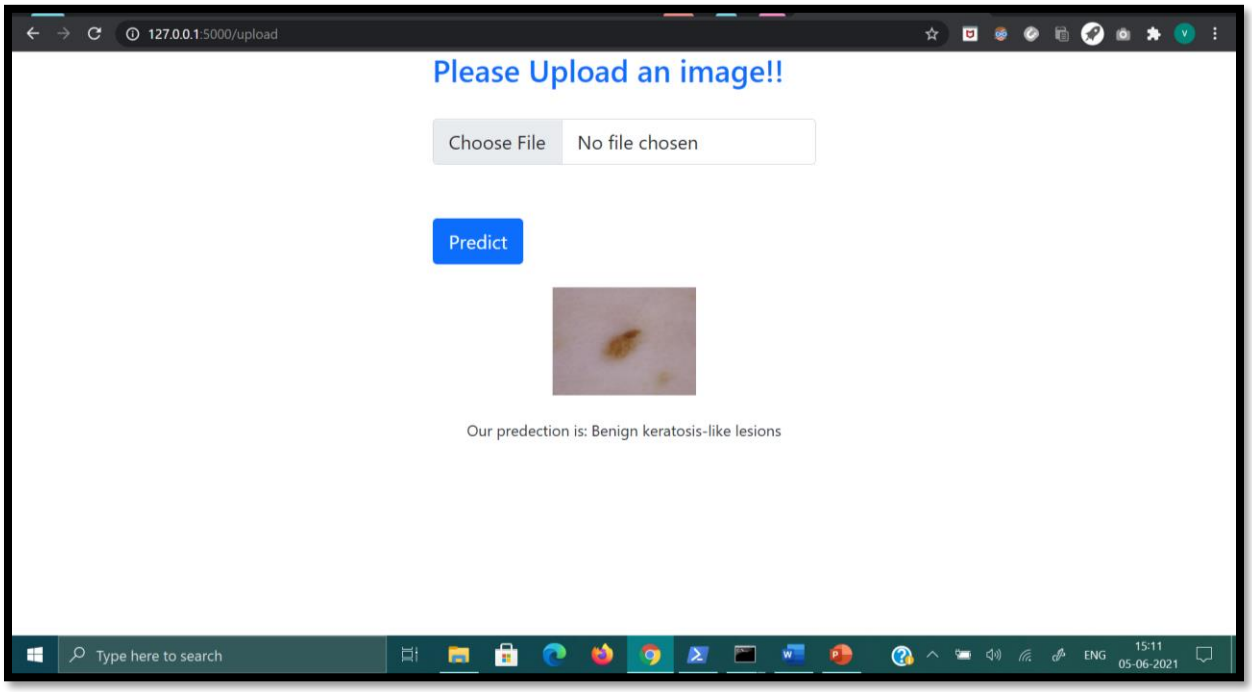
Benign keratosis-like lesions

Web Version


127.0.0.1:5000
Please Upload an image!!
Choose File
No file chosen
Predict
Type here to search
ENG
14:49
05-06-2021


127.0.0.1:5000/upload
Please Upload an image!!
Choose File
No file chosen
Predict
Our predection is: Basal cell carcinoma
Type here to search
ENG
15:10
05-06-2021


127.0.0.1:5000/upload
Please Upload an image!!
Choose File
No file chosen
Predict
Our predection is: Actinic keratoses
Type here to search
ENG
15:11
05-06-2021


127.0.0.1:5000/upload
Please Upload an image!!
Choose File
No file chosen
Predict
Our predection is: Benign keratosis-like lesions
Type here to search
ENG
15:11
05-06-2021

RESULTS AND CONCLUSION

```
[[0.10908436 0.13261303 0.7145127 0.11747606 0.18175167 0.12038327 0.16724032]]

Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
mobilenetv2_1.00_224 (Functi (None, 5, 5, 1280)        2257984
_________________________________________________________________
flatten (Flatten)            (None, 32000)             0
_________________________________________________________________
dense (Dense)                (None, 512)               16384512
_________________________________________________________________
dropout (Dropout)            (None, 512)               0
_________________________________________________________________
dense_1 (Dense)              (None, 256)               131328
_________________________________________________________________
dense_2 (Dense)              (None, 128)               32896
_________________________________________________________________
dropout_1 (Dropout)          (None, 128)               0
_________________________________________________________________
dense_3 (Dense)              (None, 64)                8256
_________________________________________________________________
dense_4 (Dense)              (None, 7)                 455
=================================================================
Total params: 18,815,431
Trainable params: 16,557,447
Non-trainable params: 2,257,984
```

The above image shows an array (top) which gives the probabilistic analysis according to this order : Actinic keratoses and intraepithelial carcinoma / Bowen's disease (akiec), basal cell carcinoma (bcc), benign keratosis-like lesions (solar lentigines / seborrheic keratoses and lichen-planus like keratosis, bkl), dermatofibroma (df), melanoma (mel), melanocytic nevi (nv) and vascular lesions (angiomas, angiokeratomas, pyogenic granulomas and hemorrhage, vasc).

Out[52]:

	pred_0	pred_1	pred_2	pred_3	pred_4	pred_5	pred_6
true_0	3	4	16	0	0	0	5
true_1	2	15	9	0	1	0	1
true_2	0	3	19	0	4	0	2
true_3	0	2	7	8	9	0	2
true_4	0	0	3	0	25	0	0
true_5	0	0	2	0	2	21	3
true_6	0	1	10	1	3	1	12

	precision	recall	f1-score	support
0	0.60	0.11	0.18	28
1	0.60	0.54	0.57	28
2	0.29	0.68	0.40	28
3	0.89	0.29	0.43	28
4	0.57	0.89	0.69	28
5	0.95	0.75	0.84	28
6	0.48	0.43	0.45	28
accuracy			0.53	196
macro avg	0.63	0.53	0.51	196
weighted avg	0.63	0.53	0.51	196

In this project, we were able to produce the highest training accuracy of 65% and testing accuracy of 47%. On printing the classification however it was found that a high number of images were detected as class 5-type cancer and most of the class-2, class-6 type cancer images were not predicted correctly. This is the best that a model has performed so far with the balanced validation data. There is clear overfitting and we are not sure why the model is not learning past a certain point. Slowing the learning rate down seems only to have helped the training data, not the testing data. After reading about batch size influence, manipulating the batch size to be larger than 32 definitely increased the accuracy. In the future, we would like to use a platform like AWS to run my more time consuming models so that we can use a larger input image and run more tests.

REFERENCES

1. Kanimozhi, T., & Murthi, A. (2016). Computer aided melanoma skin cancer detection using artificial neural network classifiers. Singaporean Journal of Scientific Research (SJSR), Journal of Selected Areas in Microelectronics (JSAM), 8(2), 35-42.
2. Kawahara, J., BenTaieb, A., & Hamarneh, G. (2016, April). Deep features to classify skin lesions. In 2016 IEEE 13th International Symposium on Biomedical Imaging (ISBI) (pp. 1397-1400). IEEE.
3. Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems (pp. 1097-1105).
4. Nasr-Esfahani, E., Samavi, S., Karimi, N., Soroushmehr, S. M. R., Jafari, M. H., Ward, K., & Najarian, K. (2016, August). Melanoma detection by analysis of clinical images using convolutional neural networks. In 2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC) (pp. 1373-1376). IEEE.
5. Oquab, M., Bottou, L., Laptev, I., & Sivic, J. (2014). Learning and transferring mid-level image representations using convolutional neural networks. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1717-1724).
6. Rajab, M. I., Woolfson, M. S., & Morgan, S. P. (2004). Application of region-based segmentation and neural network edge detection to skin lesions. Computerized Medical Imaging and Graphics, 28(1-2), 61-68.
7. Lau, H. T., & Al-Jumaily, A. (2009, December). Automatically early detection of skin cancer: Study based on neural network classification. In 2009 International Conference of Soft Computing and Pattern Recognition (pp. 375-380). IEEE
8. S. Sasikala, S. Arun Kumar, S.N. Shivappriya & Priyadharshini T (2020). Towards Improving Skin Cancer Detection Using Transfer Learning (pp-55-60). BBRC.
9. M. Afzal Ismail, Nazia Hameed & Jeremie Clos. Deep Learning-Based Algorithm for Skin Cancer Classification (pp 709-719). PICTCCE.
10. A.Cheim, Adel Al - Jumaily, R.Khushaba (2007). A Novel Hybrid System for Skin Lesion Detection. IEEE.