Skin Cancer Diagnosis using Deep Learning

CS-485 Computer Vision & Image Processing

Session: 2021-2025

Project Supervisor

Samyan Qayyum Wahla

Presented By

Muhammad Tahir 2021-CS-34

Department of Computer Science
University of Engineering and Technology, Lahore
Pakistan

Contents

List of Figures

Abstract

Skin cancer is one of the most dangerous forms of cancer. Generally, skin cancer is of two types: melanoma and non-melanoma. Melanoma also called as Malignant Melanoma is the 19th most frequently occurring cancer in women and men[1]. It is the deadliest form of skin cancer. Skin cancer tends to gradually spread over other body parts, so it is more curable in initial stages, which is why it is best detected at early stages. In the year 2015, the global occurrence of melanoma was approximated to be over 350,000 cases, with around 60,000 deaths. Non-melanoma skin cancer is the 5th most frequently occurring cancer, with over 1 million diagnoses worldwide in 2018[2]. As of 2019, greater than 1.7 Million new cases are expected to be diagnosed. Even though the mortality is significantly high, but when detected early, survival rate exceeds 95%. The increasing rate of skin cancer cases, high mortality rate, and expensive medical treatment require that its symptoms be diagnosed early. This motivates me to come up with a solution to save millions of lives by early detection of skin cancer. This project aims to develop a skin cancer detection system using CNN model which can classify the skin cancer types and help in early detection.

1 Introduction

1.1 Background

High occurrence of skin cancer compared to other cancer types is a dominant factor in making it one of the most severe health issues in the world. Historically, melanoma is a rare cancer, but in the past five decades, the worldwide occurrence of melanoma has drastically risen. In fact, it is one of the prominent cancers in average years of life lost per death. For prevention and successful treatment, early detection of skin cancer is essential. The five-year survival rate of patients with early stage diagnosis of melanoma is around 99%. Therefore, timely detection of skin cancer is the key factor in reducing the mortality rate.

1.2 Problem Statement

Doctors ordinarily use the biopsy method for skin cancer detection. This procedure removes a sample from a suspected skin lesion for medical examination to determine whether it is cancerous or not. This process is painful, slow, and time-consuming. Computer-based technology provides a comfortable, less expensive, and speedy diagnosis of skin cancer symptoms. Also there is small difference between melanoma and benign skin lesions Figure 1 , making it difficult to distinguish the two cases even for trained medical experts.

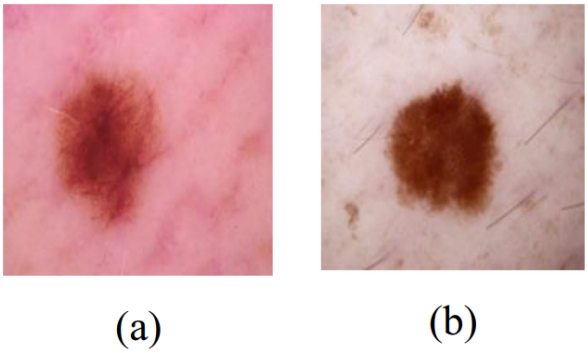

Figure 1: a) Benign Skin Cancer and b) Melanoma Skin Cancer

1.3 Proposed Solution

Various deep learning approaches have been used for computer-based skin cancer detection in recent years. From past research in this field, it is evident that CNN has an extraordinary ability to perform skin lesion classification in competition with professional dermatologists. CNNs can classify skin lesions in two ways. Firstly, a CNN can serve as a feature extractor for the images, with classification performed by another classifier. Alternatively, CNNs can perform end-to-end learning, divided into learning from scratch or learning from a pre-trained model. Training a CNN from scratch requires a large number of images to mitigate overfitting, which is often not feasible due to limited skin lesion data. Instead, training from a pre-trained model, known as Transfer Learning (TL), is preferred. TL enables effective learning with limited data and introduces generalization properties to the model.

2 Literature Review

2.1 Skin Cancer Classification using Deep Learning and Transfer Learning

The paper proposes an automated skin lesion classification method using a pre-trained deep learning network and transfer learning. Specifically, AlexNet is utilized with transfer learning, fine-tuning, and data augmentation techniques. The method aims to classify skin lesions into three categories: melanoma, common nevus, and atypical nevus. The proposed model is trained and tested using the PH2 dataset. The proposed approach addresses the challenges of insufficient data and the complexity of skin lesion classification, offering a reliable solution for early detection and diagnosis of skin cancer. The achieved rates are 98.61%, 98.33%, 98.93% and 97.73% for accuracy, sensitivity, specificity, and precision respectively.

2.2 Skin cancer detection using ensemble of machine learning and deep learning techniques

The paper proposes a novel approach to skin cancer detection by combining machine learning (ML) techniques and deep learning models for feature extraction. The authors employ contourlet transform, Local Binary Patterns (LBP) histograms, and the VGG19 deep learning model to extract features from images. Despite achieving high performance, the model is not intended to replace dermatologists and radiologists but rather to assist them in diagnosis by reducing false negatives.The proposed model offers overall F1-Score of 93% with an impressive individual recall score of 89% and 99% for the benign and malignant class, respectively Moreover, the test set accuracy was found to be 93%.

3 Methodology

3.1 Image Dataset

The International Skin Imaging Collaboration (ISIC): Melanoma Project is a partnership between industry and academia in order to facilitate the application of digital skin imaging to help curtail skin cancer.. Starting from 2015, ISIC started to organize global challenges for skin lesion analysis for melanoma diagnosis and in 2018 through a challenge.Sample images from the dataset are shown in Figure 2.
URL of the dataset `https://challenge.isic-archive.com/data/#2018`.

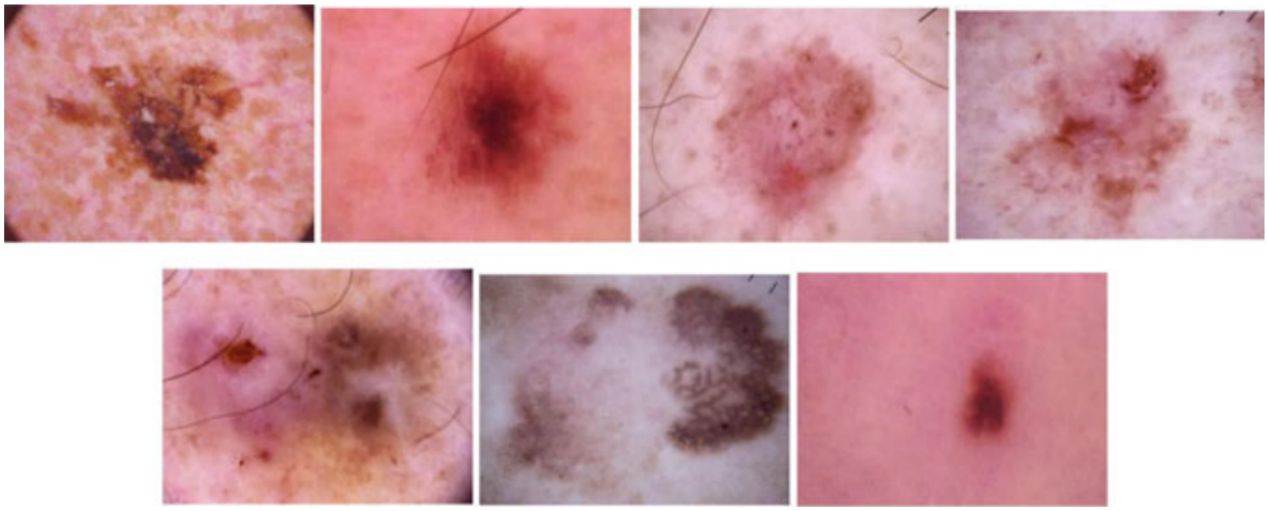

Figure 2: Sample images from the ISIC dataset (arranged in ascending order of lesion type: melanoma, melanocytic nevus, basal cell carcinoma, actinic keratosis, benign keratosis, dermatofibroma and vascular lesion)

Table 1 summarizes the dataset used for this project.

Table 1 Dataset information for ISIC 2018

Dataset	ISIC challenge 2018
Type	Dermoscopic
Image size	600 pixels × 450 pixels
Number of images	10,015
Image type	JPEG (RGB)
Class labels	0: Melanoma 1: Melanocytic Nevus 2: Basal Cell Carcinoma 3: Actinic Keratosis 4: Benign Keratosis 5: Dermatofibroma 6: Vascular Lesion

3.2 Data Pre-processing

The dataset undergoes pre-processing steps to ensure its suitability for training skin cancer disease detection model.

3.2.1 Image Normalization

Image Normalization is a technique used to normalize the pixel values of the image in a similar distribution. It is beneficial to normalize images before feeding into the neural network as this helps in approaching the global minima at error surface at a faster rate while performing gradient descent. In a way, it helps the network to converge faster. Also, the computations become significantly less intensive for the machine to perform as all the pixel values are scaled.

```
train_datagen = ImageDataGenerator(
        rescale=1. / 255,
        shear_range=0.2,
        zoom_range=0.2,
        horizontal_flip=True)
```

Figure 3: Image Normalization

3.2.2 Resizing

All images in the dataset are resized to a consistent dimension by resizing them to 128x128 pixels. This step is crucial for ensuring uniformity in the input data and facilitating the learning process of the neural network.

```
img_width, img_height = 128, 128
train_generator = train_datagen.flow_from_directory(
        train_data_dir,
        target_size=(img_width, img_height),
        batch_size=batch_size,
        class_mode='binary')
```

Figure 4: Image Resizing

3.3 Data Augmentation

Having large data helps crucially in improving the performance of machine learning models. But obtaining such vast quantities of data is cost-intensive and tedious. Hence, we use the technique of Data Augmentation. It is a technique that enables us to considerably increment the diversity and quantity of data available, without actually aggregating new data. In order

to generate new data through augmentation of images, various techniques such as cropping, padding, adding noise, brightness changing and horizontal flipping are commonly used to train large neural networks.

```
import Augmentor
p = Augmentor.Pipeline(train_path)
p.rotate(probability=0.7, max_left_rotation=10, max_right_rotation=10)
output_dir = os.path.join(train_path, "output")
os.makedirs(output_dir, exist_ok=True)
p.sample(1530)
```

Figure 5: Data Augmentation

3.4 Transfer Learning

Transfer Learning is a learning method in which a model trained for a particular task is reiterated as the origin for another model on a similar task. This approach is very mainstream in deep learning due to the vast computation resources and time consumed to train neural network models. In the case of problems in the computer vision domain, low-level features, such as shapes, corners, edges and intensity, can be shared across tasks, and thus enable knowledge transfer.

3.5 VGG-16 Architecture

The VGG-16 architecture is a deep convolutional neural network (CNN) designed for image classification tasks. The VGG-16 configuration typically consists of 16 layers, including 13 convolutional layers and 3 fully connected layers. These layers are organized into blocks, with each block containing multiple convolutional layers followed by a max-pooling layer for downsampling.

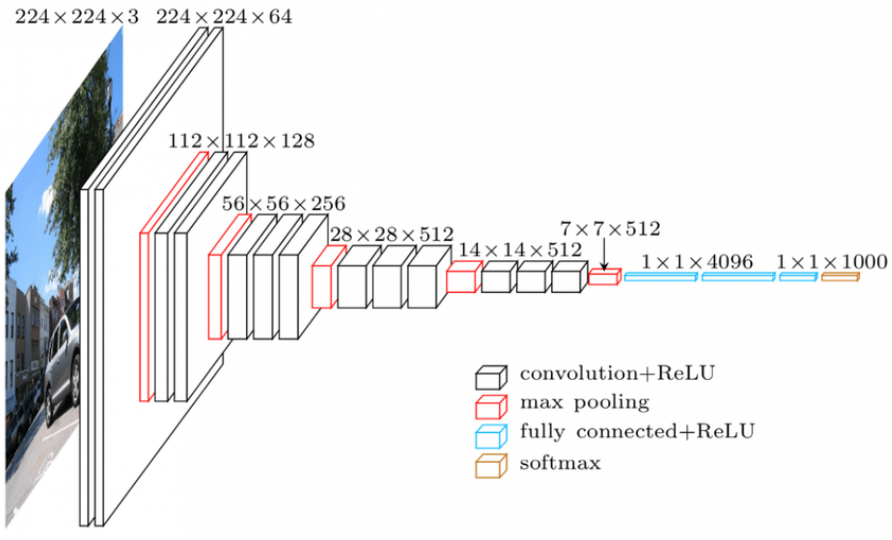

Figure 6: VGG-16 Architecture

3.6 Model Training

Even though there are many DCNNs model achieving better result on ImageNet than VGG16, I choose to fine-tune VGG16 given its simplicity. Figure 6 is the schema for VGG16. This mode is trained on 10704 images out of which 9220 were of non-melanoma and 1485 were of melanoma.

```
base_model = VGG16(weights='imagenet', include_top=False, input_shape=(128, 128, 3))
x = Flatten()(base_model.output)
x = Dense(256, activation='relu')(x)
x = Dense(1, activation='sigmoid')(x)

VGG16_model = Model(inputs=base_model.input, outputs=x)
for layer in base_model.layers:
    layer.trainable = False
VGG16_model.compile(optimizer="Adam", loss='binary_crossentropy', metrics=['accuracy'])
earlystop = EarlyStopping(monitor="val_accuracy",patience=5,mode="auto",verbose=1)

VGG16_model.summary()
history = VGG16_model.fit(train_generator, validation_data=validation_generator, epochs=nb_epochs,
                          callbacks=[earlystop,best_model_val_acc,best_model_val_loss])
```

Figure 7: Model Training

3.7 Model Summary

- **Total params:** 16,812,353 (64.13 MB)
- **Trainable params:** 2,097,665 (8.00 MB)
- **Non-trainable params:** 14,714,688 (56.13 MB)

3.8 Model Testing

This model in then tested on test dataset containing 16 melanoma and 102 non-melanoma images. The results are:

- **Test Accuracy:** 89%
- **Test Loss:** 40%

```
# Load the trained model
model = load_model('D:/6th Semester/Computer Vision and Image Processing/Skin Cancer Detection/models/VGG16_Updated.keras')
test_data_dir = 'D:/6th Semester/Computer Vision and Image Processing/Skin Cancer Detection/Dataset2/Test/'
batch_size = 32
test_datagen = ImageDataGenerator(rescale=1./255)  # Normalize pixel values

test_generator = test_datagen.flow_from_directory(
    test_data_dir,
    target_size=(128, 128),  # Resize images to match input size of the model
    batch_size=batch_size,
    class_mode='binary',  # Assuming binary classification (change if needed)
    shuffle=False  # Do not shuffle test data to maintain order
)
evaluation = model.evaluate(test_generator)
```

Figure 8: Model Testing

4 Evaluation Metrics

The model is evaluated using various metrics which were such as accuracy, precision, recall, and F1-score. The statistical evaluation measures used to analyze the quantitative performance can be calculated as follows:

```
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score

true_labels = validation_generator.classes
predictions = VGG16_model.predict(validation_generator)
binary_predictions = (predictions > 0.5).astype(int)

accuracy = accuracy_score(true_labels, binary_predictions)
precision = precision_score(true_labels, binary_predictions)
recall = recall_score(true_labels, binary_predictions)

f1 = f1_score(true_labels, binary_predictions)
```

Figure 9: Model Evaluation

This model achieved F1-Score of 87%.
T_P: represents the number of true positive image samples that are perfectly identified as infected.
F_P: is the number of false-positive image samples that are incorrectly classified as infected.
T_N: is the number of true-negative image samples that are correctly classified as healthy.
F_N: is the number of false-negative image samples that are incorrectly identified as uninfected.

4.1 Accuracy

Accuracy is a fundamental evaluation metric that measures the proportion of correctly classified samples to the total number of classified samples. This measure is employed to assess the overall performance of a suggested model. It is calculated as the ratio of the number of correctly predicted instances to the total number of instances in the dataset.

$$Accuracy = \frac{T_p + T_n}{T_p + F_p + T_n + F_n} \tag{1}$$

This model achieved accuracy of 77%.

4.2 Prescision

The precision ratio describes the performance of our model at predicting the positive class. It is calculated by dividing the number of true positives by the sum of the true positives and false positives. For multi-class classification problems, Precision is averaged across the classes.

$$Precision = \frac{T_p}{T_p + F_p} \tag{2}$$

This model achieved precision of 81%.

4.3 Recall

The recall ratio is calculated as the ratio of the number of true positives divided by the sum of the true positives and the false negatives, as follows:

$$Recall = \frac{T_p}{T_p + F_n} \tag{3}$$

This model achieved precision of 93%.

4.4 F1-Score

F1-score is the harmonic average of both precision and recall. For multi-class classification problems, F1 is averaged across all classes, where:

$$F1 - Score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{4}$$

5 Conclusion

By the art of transfer learning , I was able to create an fine tuned VGG16 model and achieved 89% accuracy on test set and 84% on validation set for ISIC 2018 dataset. By using large datasets and techniques like transfer learning and data augmentation, we can improve accuracy. Research shows that models like AlexNet, VGG-16, and ResNet50 are effective in classifying skin lesions. These models, along with machine learning methods, help doctors in diagnosis. In conclusion, by continuing to improve these techniques, we can make early detection of skin cancer easier, potentially saving lives.

References

[1] Gery P. Guy, Claudine C. Thomas, Trevor Thompson, Meg Watson, Greta M. Massetti, and Lisa C. Richardson. Vital signs: Melanoma incidence and mortality trends and projections—united states, 1982–2030. *Morbidity and Mortality Weekly Report*, 64(21):591–596, 2015.

[2] World Cancer Research Fund. Skin Cancer Statistics, 2018. Accessed: 28-Oct-2019.