**DFU\_EfficientNet: Classification and Localization of Diabetic Foot Ulcer Using Convolutional Neural Network**

**Abstract**

Diabetic foot ulcer (DFU) is a serious complication of diabetes that can lead to amputation if left untreated. Current DFU diagnosis and treatment strategies are time consuming and very expensive. In this study, we proposed an efficient convolutional neural network model called DFU\_EffectiveNet to accurately classify images with and without ulcers from the DFU dataset. DFU\_EffectiveNet consists of five parallel convolutional modules, each with four convolutional layers. To extract better features, all evolutionary layers with different kernel numbers and sizes located in parallel convolutional modules are concatenated. For comparison purposes, we retrained and evaluated four previously trained deep learning networks (AlexNet, DenseNet 121, VGG16, and GoogleNet) for the same task. Our proposed DFU\_EffectiveNet model is designed with gradient propagation, depth and width in mind. Obtaining 96.9% accuracy, specificity 99.0%, and 98.3% precision, the proposed DFU EffectiveNet model outperforms the contemporary CNN network.

In this paper, we also evaluated the performance of deep learning algorithms by classifying as well as localizing diabetic foot ulcer tissue using well-known interpretable AI algorithms. Our findings indicate that the incorporation of interpretable AI into deep learning workflows can help in medical problems, choosing the best AI training techniques, and human-machine interaction.

**1 Introduction**

The term DIABETES Mellitus (DM) is known as Diabetes. It is a lifelong condition caused by hyperglycemia (high amounts of blood sugar), which causes severe, potentially fatal complications such as lower limb amputation, cardiovascular diseases, blindness, and kidney failure which is led by Diabetic Foot Ulcers (DFU) [1].

Current clinical practice includes several important procedures for early diagnosis of diabetes as part of the DFU test. These are 1) the patient's disease background is assessed; 2) a wound or diabetic foot specialist thoroughly examines the DFU; 3) Additional tests such as CT scan, MRI, and X-ray.

To prevent this harmful disease, a diabetic needs regular doctor consultations, ongoing expensive medication, and self-care. Consequently, this can impose a heavy financial burden on DFU sufferers and their families, especially in underdeveloped countries where the cost of treating this illness can exceed annual income [2]. More than a million diabetic people lose some of their legs each year [3] as a result of delayed diagnosis and inadequate treatment. For proper medical diagnosis and treatment of DFU needs Traditional medical procedures including blood tests, X-rays, CT scans and MRI others that are costly and time-consuming.

Several categorization techniques for diabetic foot ulcers (DFUs) have been developed recently. The first automatic solution, based solely on convolutional neural networks, to distinguish between DFU skin and healthy skin was presented by Goyal et al. [4] in 2017.

His proposed CNN model, DFUNet, consists of parallel convolution modules with various filter sizes that concatenate features at different network depths. The outcomes were considerable and demonstrated the CNN algorithm's potential for independent medical image analysis. The main weakness of this paper is the relatively small number of parallel convolutional blocks and filters, resulting in low accuracy, precision, and specificity of DFUNet. Adding parallel convolutional blocks and filters can increase its accuracy.

In another paper, a newly developed deep convolutional neural network by Alzubaidi et al. [5] is called DFU\_QUTNet for DFU image classification. The architecture of the network was designed to increase the breadth while keeping the depth unchanged, in line with modern networks. Support Vector Machine (SVM) and K-Nearest Neighbors (KNN) classifiers are trained using the features that are extracted from the proposed DFU\_QUTNet. As a result, the F1-score value has significantly improved. A drawback of this method is that each parallel convolutional layer has the same kernel size (3 x 3). Accuracy scores can be improved by using convolutional kernels of different sizes instead of the same size.

K. Das et al. [6] discussed recent developments. Their findings suggest a novel CNN design, DFU\_SPNet, that dramatically boosts DFU classification's cutting-edge performance metrics. This network's architecture consists of three parallel convolution layers that are stacked one on top of the other and have various sizes of the kernel. In this architecture, a transition layer is made up of a batch normalization layer and leakyRELU activation in between each convolution module. Although the DFU\_SPNet model performed admirably, a drawback of this paper is that the same filter (32Ꚛ64Ꚛ128) is used in each parallel convolution block. The precision score could be improved by employing varying numbers of filters rather than the same number.

For DFU classification, Alzubaidi et al. [7] developed a hybrid CNN model. In their suggested paradigm, four deep networks were constructed by combining multi-branched parallel layers and traditional convolutional layers, where each network consists of six parallel convolutional modules. The networks were divided based on parallel convolutional layer branches, where the number of branches varied between two and five. To extract a better feature map, the branches are combined using different filter sizes. The number of parallel convolution blocks used in this paper is huge, which leads to significant computing costs and time and makes the model quite complex even though it achieves remarkable performance.

    Motivated by all the above paper limitations, we proposed an innovative and reliable deep convolutional neural network called DFU\_EffectiveNet. Our network is composed of a small number of parallel convolution modules, each using different filters and convolution kernels of different sizes. Since our model is relatively small, the computation cost and time are low. It also improves our model accuracy and precision compared to other models.

Below is a summary of our intriguing findings:

1. We have collected a brand-new dataset of 754 feet of images from various individuals, both with and without ulcers. A labeling tool has been used to label the dataset. Then the image has been resized with a height of 224 pixels and a width of 224 pixels.
2. Our network contains varying degrees of features at each stage of the network, which shows that it is more beneficial and efficient.
3. Our newly designed CNN model divided photos of the skin of the feet into two categories: ulcer and not an ulcer. We enhanced the DFU classification's performance, which now outperforms modern CNN networks.
4. Traditional CNNs need a lot of data to produce correct results, but DFUEffectiveNet’s high filter size combined with parallel convolution blocks can generate better results on a limited amount of input.
5. For the DFU classification challenge, we have retrained four previously trained deep learning networks (such that AlexNet [8], VGGNet [9], GoogleNet [10], and DenseNet [11]) to be compared with our model.
6. As stated in Section 2, we have examined a few cutting-edge deep learning techniques as well as more conventional techniques for classifying DFU.

The remainder of the paper is structured as follows: The related work is reviewed in Section 2. In Section 3, the proposed methodology is described. The outcomes are shown in Section 4. Finally, section 5 presents conclusions.

**2 Related work**

In recent years have been created the development of many telecommunication systems for medical services, including systems for monitoring diabetes [12, 13]. These systems' goals are to provide automated solutions, improve patient access to medical facilities, improve the quality of the existing healthcare systems, and lower the price of medical facilities [14].

**2.1 Telemedicine systems in DFU**

Now remote communication has become very easy due to new technologies like laptops, internet media, and smartphones. These devices can capture and communicate video, music, and high-quality images with the help of advanced mobile internet [15, 16]. Non-automated medical communication systems such as digital cameras [17], video conferencing [18], optical scanners [19], and 3D lesion imaging [20] are set up remotely to assess the patient's body condition. However, a well-rounded assessment of the patient is still needed by experienced clinicians. Despite giving robust results, it is essential to have an automated system for DFU detection.

The first automated telemedicine system developed for DFU is still in its infancy. Notably, Liu et al. [21], [22] created an intelligent telecommunication and medicine system in 2015 using 3D surface reconstruction, spectrum imaging, and infrared thermal pictures to detect diabetic foot problems.

But for the system to be properly implemented, expensive equipment and plenty of specialized training are needed. Another method developed by Wang, et al. [23] used a special capture box to capture images and a support vector machine to split the image into two parts to identify areas with DFU. Another significant study by Manu, et al. [24] used complete foot images for DFU segmentation.

**2.2 Computer Vision**

Numerous modern computer vision tasks and pattern recognition including image classification [25], age categorization [26], nucleus detection [27], and sketch recognition [28] have been powered by deep learning in recent years. Machine learning methods and traditional computer vision represent data with different levels of abstraction, the ability to handle large amounts of image data, and require a lot of manual tuning for each input image. Currently, such computer vision problems can be solved using deep convolutional networks, another area of modern machine learning [29], [30].

**2.3** **Deep Learning**

Deep convolutional networks achieve multiple levels of representation through non-linear modules that transform the representation into a more advanced representation for classification. A deep convolutional network takes an image dataset as input and starts learning features from an array of different values of pixels in the image. At the deeper level, it begins to learn by combining elements of features from previous layers and finally, combines these elements to form a complete object [31].

Nowadays the most popular type of learning is supervised learning which is a part of machine learning. This is very important for training the network as the network learns to classify images from a large collection of datasets. Without training, the machine cannot identify the required class using the best score [32], [33], [34].

**3 Methodology**

There are five components to this section: (i) dataset (ii) labeling process (iii) data augmentation) (iv) fine-tuned CNNs architectures of pre-trained models (v) our proposed CNN model called DFUEffectiveNet, which will improve DFU classification performance.

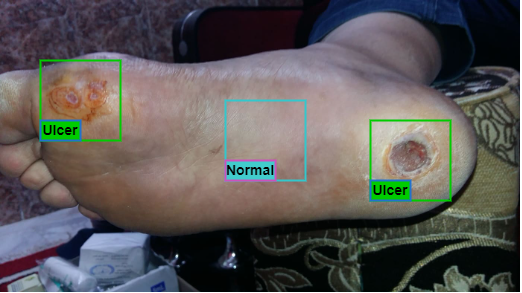
**3.1 Dataset:**

The datasets required for model training and testing were taken from datasets listed in the online platform https://www.kaggle.com/laithjj/diabetic-foot-ulcer-dfu. Whereas the dataset contained images of both feet with and without diabetic ulcers. Whereas the patch folder of the dataset contains 512 images of diabetic ulcer skin and 543 images of healthy skin. A labeling approach was used to increase the size of the dataset, which is shown in Figures 1 and 2 and is described in detail in Section 3.2.

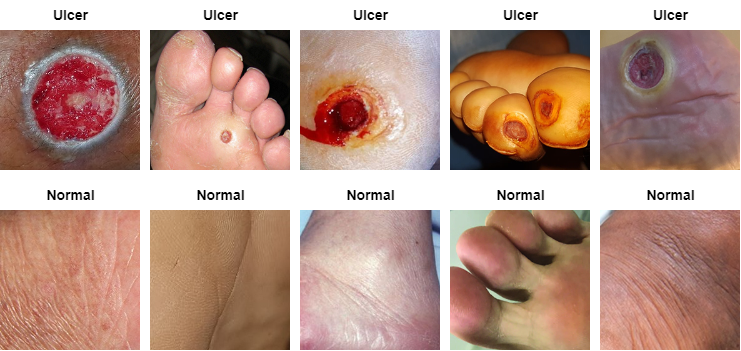
The images were reduced in size to 224 x 224 pixels to better illustrate the two categories of ulcer and not-ulcer skin patches. We divided the dataset into three parts: train (85%), test (10%), and validation (5%). The pre-processed images are then used to train and test the proposed model as well as deep learning models that have already been trained for DFU classification.

**3.2 Expert image labeling**

The Region of Interest (ROI) was initially cut at 224 pixels in height and 224 pixels in width. The specialist then assigned labels to the clipped patches in this region that contains significant tissues from both ulcer and normal skin classes. A total of 1655 skin patches, 1032 of which were ulcers and 623 normal. The data set was then split into 10% of patches for a testing set of images and 90% of patches for a training set of images. Before the cropping operation, **FIG.** 1 displays samples of both ulcer and normal patches. **FIG.** 2 shows a few sample images from patches.



**FIG.** 1 Samples of Ulcer and Normal patches

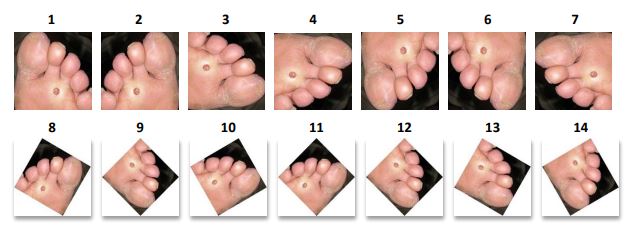


**FIG.** 2 Samples images from patches

**3.3 Data Augmentation of Training Patches**

Since deep networks have many parameters, it requires a lot of training image data. So, to enhance the effectiveness of deep learning techniques, we applied data augmentation techniques. To accomplish this, we combined some image processing methods, including random scaling and flipping, rotation, and contrast enhancement using new color spaces. Then used different types of flipping (horizontal, vertical, or both horizontal/vertical).

The rotation process is done by rotating the image from a dataset to different angles like 90, 180, or 270 degrees, etc. **FIG.** 3 shows some samples of enlarged images.

To work properly, CNNs need a large number of labeled training datasets. The larger the training dataset, the more accurate the training parameters of the CNN will be, but for a smaller dataset, this leads to additional overfitting problems. Each patch is iterated 13 times in an image using a data augmentation method for better training. Figure 3 shows different types of data augmentation techniques.

**FIG. 3** Samples of some augmented image created from original image

**3.4 Pre-trained CNN architectures**

In this section, large image datasets were used for training and testing CNN's networks. In numerous research studies, transfer learning of previously trained models are efficient and effective for classifying medical image datasets [35, 36, 37]. Performance is improved when pre-trained models are used in a transfer learning technique. The four popular CNNs models that we used are (AlexNet [8], VGGNet [9], GoogleNet [10], and DenseNet [11]). We enhanced these CNNs models for categorizing ulcer and not-ulcer classes because they have demonstrated great accuracy across various domains.

**(i) AlexNet:**

The AlexNet was first introduced by Alex Krizhevsky, [8], who also won the 2012 ILSVRC. It consists of three fully connected layers at the beginning and a softmax layer after five convolutional layers, some of which are followed by max-pooling layers. The input image, which has a dimension of 227x227x3, is filtered by the first convolutional layer, whose image size is 11x11x3 pixels and kernel size 96. The output of the first convolutional layer is filtered by the second convolutional layer, whose image size is 5x5x48 pixels and kernel size 256. The output of the second convolutional layer is filtered by the third convolutional layer, whose image size is 3x3x256 pixels and kernel size 384. The fourth convolutional layer has kernel size 384 and image size 3x3x192, whereas the fifth convolutional layer contains 256 kernels of size 3x3x192 as its final layer. Notably, the third, fourth, and fifth convolutional layers do not contain any normalizing or pooling layers connected. Finally, the fully connected layers include 4096 neurons. One thousand classes of images were classified using AlexNet. To accommodate AlexNet's input size, we downsized the dataset's images.

**(ii) VGGNet 16:**

VGG16 network was employed to win the 2014 ILSVR competition and was developed by Simonyan, et al., [9]. VGG-16 model takes color images as input whose size is 224x224x3. This model is composed of three main components. These components are thirteen convolutional layers, five max-pooling layers, and three fully connected (FC) layers [9]. Each convolutional layer has one filter whose size is 3x3, one stride whose length is 1, and the same padding.

**(iii) GoogLeNet:**

Szegedi, et al. [10] first introduced and trained the GoogleNet model on the ImageNet dataset in 2015. It has nine Inception components spread across 22 convolutional layers and 27 pooling layers, which greatly increases model speed. Each Inception module consists of three convolution layers and one max pooling layer. The filter size of the first convolution layer is 1x1, the second convolution layer is 3x3, the third convolution layer is 5x5, and the max polling layer. This model takes the color image as input whose size is 224x224x3. It also goes by the name Inception V1 CNN model. Finally, the global average-pooling operation is used at the ends of the inception components.

**(iv) DenseNet 121:**

This section will discuss a popular CNN model with a densely connected. Whose name is the Densely Connected CNN model (Densenet 121). The 121 here shows that there are 121 deep layers in this well-known CNN model. Since DenseNet-121 [11] CNN is a profound network, this model uses many small connections between all layers for more precise and effective training. The layers of DenseNet-121 are as follows: four AvgPool, one Fully Connected Layer, one 7x7 Convolution, fifty-eight 3x3 Convolution, sixty-one 1x1 Convolution.

On the other hand, each max-pooling layer has one filter whose size is 2x2, and one stride whose length is 2. VGG-16 takes small convolutional kernels to improve the network performance. The 16 in VGG-16 indicates that there are 16 layers with weights. This network has over 138 million parameters, making it a sizable network.

All of these traditional networks have been adjusted to categorize the skin on the foot into two classes, ulcer and not-ulcer, and have been retrained using our dataset. With transfer learning, such that AlexNet [8], VGGNet [9], GoogleNet [10], and DenseNet [11] Networks are frequently used for medical image classification.

GoogLeNet uses pooling and multiple kernel-size convolution layers inside a single module. Considered to be CNN's first advancement is AlexNet. Last but not least, the VGG network is organized similarly to a standard CNN, with a series of convolutional, max-pooling, activation functions (ReLU), and full-connected layers.

**3.5 Proposed approach DFUEffectiveNet model**

We designed a concurrent DFUEffectiveNet model, as shown in Fig. 4, with details of each layer, which increases the extraction of key features for diabetic foot ulcer (DFU) classification. Our model combines two crucial components of CNNs' architecture: (i) the depth of the model and (ii) the parallel convolution layer.

To learn additional feature mappings and improve the network performance, we reduced the network's depth, enhanced the number of parallel convolutional modules, and enhanced the number of filters inside these modules. In our model, traditional convolution layers are combined with parallel convolutional layers, whereas traditional convolution layers employ a single convolutional layer, and parallel convolutional layers employ many convolutional layers. The DFU\_EffectiveNet is built to detect and classify which foot skin has an ulcer and which does not.

The DFU\_EffectiveNet architecture is split into three main stages: (i) the initial layers; (ii) the intermediate layers; and lastly, (iii) the final layers. The detailed layers of the general DFUEffectiveNet architecture are provided in Table I.

The parameters used for training DFUEffectiveNet are 50 epochs, an 85% validation dataset, a 5% validation dataset, a batch size of 32, and an Adam optimizer with a learning rate of 0.001. To minimize the learning rate, factor = 0.4, patience = 2, min\_delta = 0.001 is set.

**FIG.** 6 depicts the general pipeline of the suggested methodology. As previously indicated, we separated our model dataset into three phases: training (85%), validation (5%), and testing (10%).

**Input image:** The DFU input images contain training and validation images. There are three channels in it. The image size of each channel is 224x224. The input images cropped into 224×224 sizes are labeled into two classes: normal and ulcer, as shown in **FIG.** 2, and then these labeled images are given as input to the DFU\_EffectiveNet network for training.

**(i) The initial layers:**This layer is also called the traditional convolutional layer, which contains different types of convolutional and max-pooling layers. The length of this layer ranges from layer 1 to layer 5, as shown in Table I. Three convolutional layers with three different kernel sizes—7x7, 1x1, and 3x3—make up this stage. Batch normalization and rectified linear unit (ReLU) layers are placed after each convolution layer. This stage is crucial to ensuring that any huge crude input images are reduced in dimension before moving on to the next stage.

**Convolutional layer:** This layer is coupled with a filter set [39] that learns the results of the previous layer since the weights are capable of identifying each convolutional filter. To create feature maps of the appropriate filters, all filters are moved across the input images' height and width. The depth of each filter is the same [40]. The output of each layer can be combined by four hyper-parameters: zero-padding, stride, kernels, and kernel size.

The first convolution layer of the DFU\_EffectiveNet network has a kernel size of 7x7, a stride size of 2x2, and 64 kernels. The second convolution layer has a kernel size of 1x1, a stride size of 1x1, and 64 kernels. Finally, the third convolution layer has a kernel size of 3x3, a stride size of 1x1, and 92 kernels.

**Batch normalization (BN) layer:** This layer applies a mini-batch to normalize each input channel. This layer plays an effective role in reducing the sensitivity of the CNN algorithm and increasing the training speed [41].

This stage has 3 batch normalization layers, which are placed after each convolution layer but before the ReLU layers.

**Rectified Linear Unit (ReLU):**ReLU is the non-linear activation function, which filters data at this level using the max (0, x) function [42], where x denotes the input to the neuron.

**Max pooling layer:**This layer increases the efficiency of the CNN model by selecting the largest elements from the activation map area covered by the filter. These layers reduce the dimensions of the activation maps and the number of learning parameters. In this stage, two max-pooling layers with 2x2 kernel size and 2x2 stride size are used, where the first max-pooling layer reduces the input image size from 112x112 to 56x56 and the second max-pooling layer reduces the input image size from 56x56 to 28x28.

**(ii) Intermediate layers:**This layer, also called the parallel convolutional layer, consists of multiple convolution layers and three max pooling layers. A parallel convolutional layer is a special type of layer that combines multiple convolutional filters. This special type of layer is used to extract multi-level features from a single input and combine sparse clusters. Parallel convolutional layers can also be called the core of the DFU\_EffectiveNet model.

The parallel convolution block can extract features associated with different convolution layers with filters of different sizes, which discriminates the DFU\_EffectiveNet more efficiently than previous network layers. The length of this intermediate layer ranges from layer 6 to layer 12, as shown in Table I, where layers 6, layer 8, layer 9, layer 10, and layer 12 are parallel convolutional layers. Layers 7 and 11 are two max-pooling layers, where the first max-pooling layer is located between the first and second parallel convolutional layers, and the second max-pooling layer is located between the fourth and fifth parallel convolutional layers.

Parallel convolutional blocks are considered the basic units of the DFU\_EffectiveNet architecture. We are trying to find the best architecture for DFU\_EffectiveNet by testing DFU\_EffectiveNet with different variations of these parallel parts. In Table II, we construct four different variations of DFU\_EffectiveNet by adding different size filters in parallel convolution. We also tested whether increasing the filter size would result in the highest performance of DFU\_EffectiveNet. The results of these variations are shown in Table III below.

Each version of DFU\_EffectiveNet consists of five parallel convolution modules that operate in parallel, and the filter concatenation layer combines the outputs and provides them as input to the next layer.

In the first version, each parallel convolution module consists of two convolution layers with two filter sizes—1x1 and 3x3—and several different filters. In the second version, each parallel convolution module consists of three convolution layers with several different filters, each with filter sizes of 1x1, 3x3, and 5x5. In the third version, each parallel convolution module consists of four convolution layers with several different filters, each with filter sizes of 1x1, 3x3, 5x5, and 7x7. In the fourth version, each parallel convolution module consists of five convolution layers with several different filters, each with filter sizes of 1x1, 3x3, 5x5, 7x7, and 9x9.

Each convolution layer located in the parallel convolution module of each version has a ReLU activation function.

On the other hand, the three max-pooling layers are located after the first parallel convolution, fourth parallel convolution, and fifth parallel convolution modules, respectively.

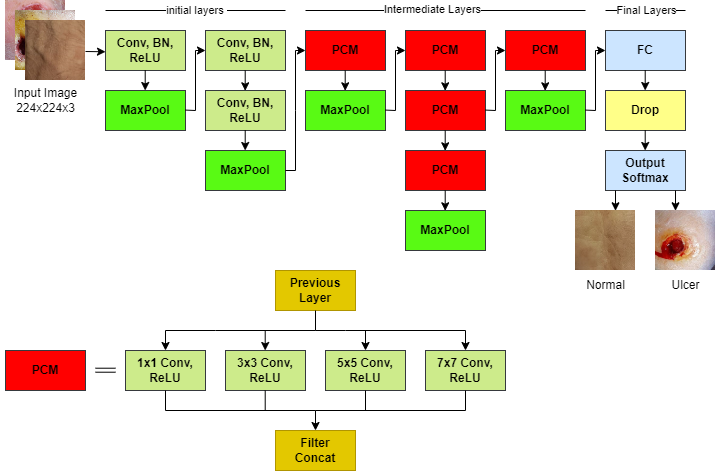
Thus, by increasing the version, we can see that the third version gives the best results, as shown in Table 3. The complete architecture of this version is shown in Table 1.

**(iii) Final layer:**This layer is also called the fully connected layer (FC). This layer consists of two fully connected layers and a drop-out layer.

**Fully connected layer (FC):** All of the neurons in the preceding layer are connected to this fully connected layer [38]. The output of the first FC layer is 100 and the output of the second FC layer is 2. The main function of this layer is to classify the skin into two categories, normal and ulcerated, by combining the features of the leg skin. The second FC layer has one softmax activation function.

**Dropout Layer:** After the first FC layer, a dropout layer was used to increase model performance and avoid overfitting. The probability of dropout level in our model was 0.5.

**Softmax**: It is an output classifier that predicts what the model's class labels will be. It determines how close the evidence data labels are to the training data.



**FIG. 4** DFUEffectiveNet architecture

More discriminative features are present in each convolution layer. Different levels of activation maps for both normal and ulcer images of DFUEffectiveNet are shown in FIG. 7 and FIG. 8.

**Table I.**

**Proposed Model Architecture with five modules. Here, LN =Layer No., S = Stride, C2d = Convolutional Layer, PC2d = Parallel Convolutional Layer, BN = Batch Normalization Layer, RL = Rectified Linear Unit, Drop = dropout layer, FC = Fully Connected Layer, MP2d = Max Pooling Layer, SM = Softmax.**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **LN** | **Layer type** | **Kernel Size** | **S** | **No. of kernels** | **FC** | **Input** | **Output** |
| Layer 1 | **C2d, BN, R** | 7x7 | 2x2 | 64 | - | 224x224x3 | 112x112x64 |
| Layer 2 | **MP2d** | 2x2 | 2x2 | - | - | 112x112x64 | 56x56x64 |
| Layer 3 | **C2d\_1, BN\_1, RL\_1** | 1x1 | 1x1 | 64 | - | 56x56x64 | 56x56x64 |
| Layer 4 | **C2d\_2, BN\_2, RL \_2** | 3x3 | 1x1 | 92 | - | 56x56x64 | 56x56x192 |
| Layer 5 | **MP2d\_1** | 2x2 | 2x2 | - | - | 56x56x192 | 28x28x192 |
| Layer 6 | **PC2d** | 1x1, 3x3, 5x5, 7x7 | 1x1 | 128Ꚛ128Ꚛ128Ꚛ128 | - | 28x28x192 | 28x28x512 |
| Layer 7 | **MP2d\_2** | 2x2 | 2x2 | - | - | 28x28x512 | 14x14x512 |
| Layer 8 | **PC2d\_1** | 1x1, 3x3, 5x5, 7x7 | 1x1 | 192Ꚛ192Ꚛ192Ꚛ192 | - | 14x14x512 | 14x14x768 |
| Layer 9 | **PC2d\_2** | 1x1, 3x3, 5x5, 7x7 | 1x1 | 256Ꚛ256Ꚛ256Ꚛ256 | - | 14x14x768 | 14x14x1024 |
| Layer 10 | **PC2d\_3** | 1x1, 3x3, 5x5, 7x7 | 1x1 | 256Ꚛ256Ꚛ256Ꚛ256 | - | 14x14x1024 | 14x14x1024 |
| Layer 11 | **MP2d\_3** | 2x2 | 2x2 | - | - | 14x14x1024 | 7x7x1024 |
| Layer 12 | **PC2d\_4** | 1 x1, 3x3, 5x5, 7x7 | 1x1 | 512Ꚛ512Ꚛ512Ꚛ512 | - | 7x7x1024 | 7x7x2048 |
| Layer 13 | **MP2d\_4** | 7x7 | 1x1 | - | - | 7x7x2048 | 1x1x2048 |
| Layer 14 | **FC, R \_3** | - | - | - | 1000 |  |  |
| Layer 15 | **FC\_1, SM** | - | - | - | 2 |  |  |

**3.6 Explainable AI technique for localization of DFU Images**

Although our DFU\_Effectivenet model showed impressive results compared to other existing models for DFU classification, we could have achieved better results if we could distinguish images with and without ulcers based on the exact location of the ulcer instead of other features.

The development of artificial intelligence (AI) has opened doors for human life in various fields including education, business, and healthcare [43, 44].

Deep-learning-derived algorithms help classify medical images in the medical arena as part of AI [43]. Explainable AI algorithms provide interpretable tools for understanding deep learning results by inferring properties at each level [45]. Long before the training and testing phases, biases can develop in the models [45]. The data used to train the model may have their own biases. As a result, any bias in the training dataset must be identified and managed within any AI method. The goal of explainable AI should be such that the model created is reliable, open, and free of bias.

Using AI techniques, it has been possible to accurately analyze various medical image datasets such as distinguishing between healthy and ulcer-affected skin. However, AI models are also termed black-box models, as these models take a necessary step to integrate AI imaging techniques into daily clinical practice while hiding logical explanations. Explainable AI is a method that displays top-level features of a trained model or aggregates them during the training process.

Our DFU\_Effectivenet model implements various interpretable algorithms such as backpropagation saliency map, gradcamp (GradCAM), and gradcamp++ (GradCAM++), to help smooth convolutional neural networks. Each XAI method is unique and may be helpful in a different situation with its inherent advantages and limitations.

GradCAM helps analyze the convolutional layer of any CNN by highlighting the discriminative regions of the image and qualitatively and quantitatively understanding the inner workings. GradCAM++ was used to solve the low-resolution heatmap problem in GradCAM.

Our experiment showed that the explainable GradCAM++ and GradCAM methods could improve the network's performance, separating the input images with and without ulcers based on precise localization rather than other attributes.

The use of these explanation techniques for ulcer analysis tasks is thoroughly discussed. Gradient-based algorithms, such as BP saliency map, GradCAM, and GradCAM++, are used to highlight all features of ulcerated skin in this application, as given in Figure 7.

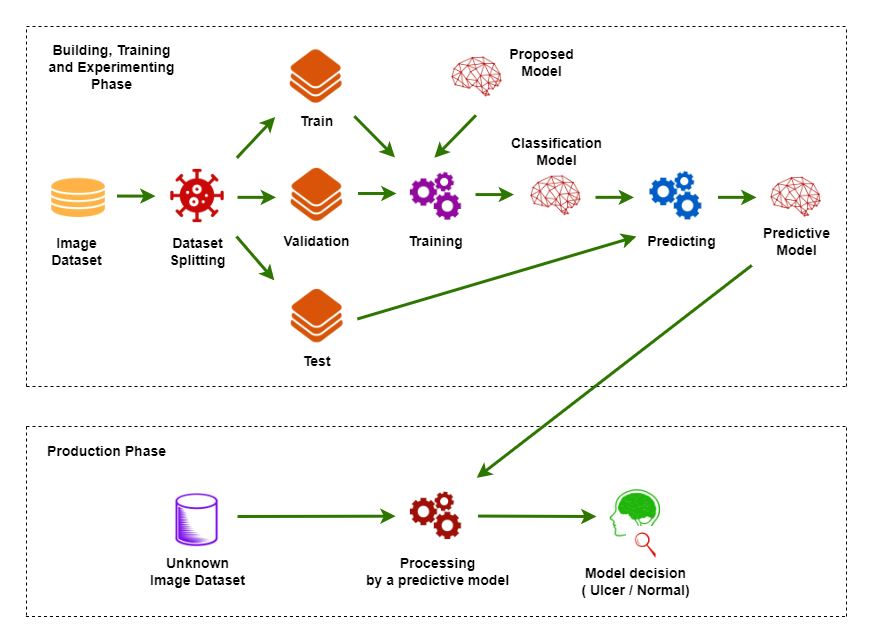
**4. Experimental Results**

**4.1. Evaluation metrics**

On our highly qualified dataset, we ran many experiments to gauge how well our network and specially pre-trained networks performed at classifying data.

The different evaluation metrics are defined as follows:

Where, SEN = Sensitivity, SPE = Specificity, PRE = Precision, REC = Recall, ACC = Accuracy, FS = F1-Score, = True Positive, = True Negative, = False Positive, = False Negative



**FIG.** 6 The suggested approach workflow for distinguishing normal and the ulcer

**4.2. Performance measures of several DFUEffectiveNET versions**

The performance metrics of several DFUEffectiveNet versions with varied parameters are described in Table II.

**Table II**

The explanations of several DFUEffectiveNET versions that use parallel convolutional modules with varied kernel and kernel sizes. Here, PC2d = parallel convolutional, and V = Version.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PC2d types** | **DFUEffectiveNET V1** | **DFUEffectiveNET V2** | **DFUEffectiveNET V3** | **DFUEffectiveNET V4** |
| PC2d | 1x1, 3x3  128Ꚛ128 | 1x1, 3x3, 5x5  128Ꚛ128Ꚛ128 | 1x1, 3x3, 5x5, 7x7  128Ꚛ128Ꚛ128Ꚛ128 | 1x1, 3x3, 5x5, 7x7, 9x9  128Ꚛ128Ꚛ128Ꚛ128Ꚛ128 |
| PC2d\_1 | 192Ꚛ192 | 192Ꚛ192Ꚛ192 | 192Ꚛ192Ꚛ192Ꚛ192 | 192Ꚛ192Ꚛ192Ꚛ192Ꚛ192 |
| PC2d\_2 | 256Ꚛ256 | 256Ꚛ256Ꚛ256 | 256Ꚛ256Ꚛ256Ꚛ256 | 256Ꚛ256Ꚛ256Ꚛ256Ꚛ256 |
| PC2d\_3 | 256Ꚛ256 | 256Ꚛ256Ꚛ256 | 256Ꚛ256Ꚛ256Ꚛ256 | 256Ꚛ256Ꚛ256Ꚛ256Ꚛ256 |
| PC2d\_4 | 512Ꚛ512 | 512Ꚛ512Ꚛ512 | 512Ꚛ512Ꚛ512Ꚛ512 | 512Ꚛ512Ꚛ512Ꚛ512Ꚛ512 |

**TABLE III**

The explanations of performance measurements of several DFUEffectiveNET versions. Here, V = Version, SPE = Specificity, PRE = Precision, REC = Recall, ACC = Accuracy, REC = Recall, and FS = F1-Score.

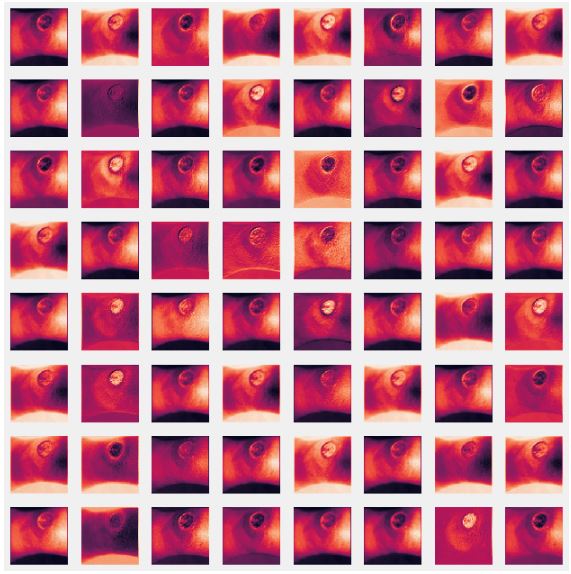
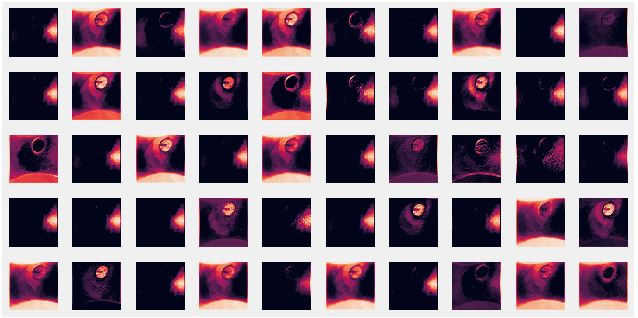
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Model** | **ACC** | **PRE** | **REC** | **FS** | **SPE** |
| DFUEffectiveNET V1 | 0.937 | 0.941 | 0.895 | 0.917 | 0.964 |
| DFUEffectiveNET V2 | 0.959 | 0.959 | 0.935 | 0.947 | 0.974 |
| **DFUEffectiveNET** **V3** | **0.969** | **0.983** | **0.936** | **0.959** | **0.990** |
| DFUEffectiveNET V4 | 0.934 | 0.926 | 0.903 | 0.903 | 0.954 |

The measurements of the four versions of our newly proposed DFUEffectiveNET are shown in Table **III**. DFUEffectiveNET version 3 has the maximum accuracy, precision, recall, specificity, and F1-score scores, which were 96.9%, 98.3%, 93.6%, 99.0%, and 95.9%, respectively.

DFUEffectiveNET with version 4 obtained the minimum scores of 93.4% for accuracy, 92.6% for precision, 90.3% for recall, 95.4% for specificity, and 90.3% for F1-score.

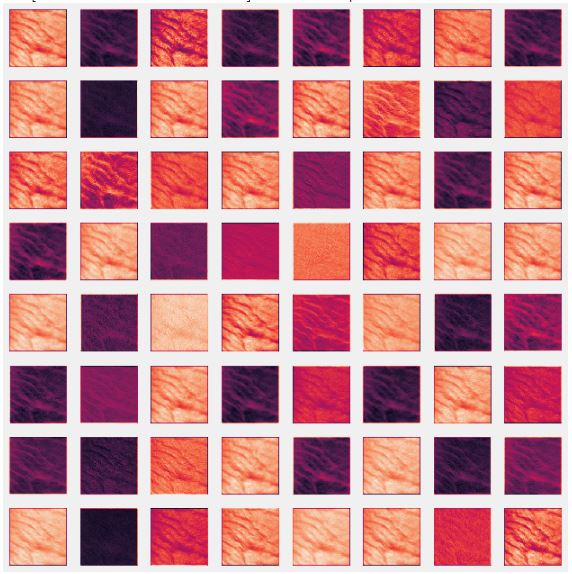
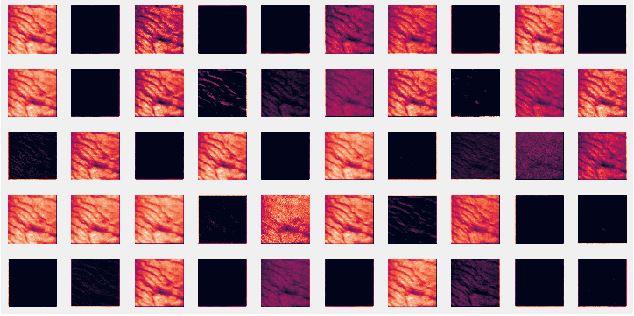
On the other hand, DFUEffectiveNET version 1 achieved 93.7% accuracy, 94.1% precision, 96.4% specificity, 89.5% recall, and 91.7% F1-score, respectively.

DFUEffectiveNET version 2 achieved 95.9% accuracy, 93.5% recall, 97.4% specificity, 95.9% precision, and 94.7% F1-score, respectively.

Convolution2dmax\_pooling

**FIG.** 7 first Convolution2d and max\_pooling2d layers for ulcer image

Convolution2dmax\_pooling

**FIG. 8** first Convolution2d and max\_pooling2d layers for normal image

To compare our proposed DFUEffectiveNET with four ultra-modern models, we trained, evaluated, and compared our DFU dataset. The reports give in Table IV. AlexNet [8] achieved the minimum accuracy, precision, specificity, F1-score, and recall scores of 90.9%, 89.3%, 93.3%, 88.2%, and 87.1. VGG16 [9] has lower precision, accuracy, specificity, recall, and F1-score than our DFUEffectiveNET and DenseNet121 [11] but is higher than AlexNet [8]. The precision, accuracy, specificity, recall and F1-score of VGG 16 [9] are respectively 92.3%, 92.5%, 95.4%, 87.8%, and 90.0%. Although Densenet121 [11] has the lowest score from our recommended DFUEffectiveNET, its recall performance is higher than our DFUEffectiveNET and other ultra-modern models. In Table IV, a comparison of our proposed DFUEffectiveNET model with the state-of-the-art model is shown, where our model shows remarkable results.

The results in Table V show that, without recall, our DFUEffectiveNET model outperforms all existing models such as DFUNet [4], 4-branch model [7], DFU\_QUTNet [5], and DFU\_SPNet [6]. The DFUEffectiveNET model has a lower recall score due to a slightly higher value of false negatives (FN). A deeper look at table V reveals that DFUEffectiveNet has very high precision and accuracy value.

So this high precision and accuracy value makes DFUEffectiveNet a very potent model for DFU abnormal prediction. DFUEffectiveNet is called an effective prediction system because of its higher precision (0.983) and accuracy (0.969) score.

Our model can detect subtle changes in DFU skin and simultaneously classify both normal and abnormal skin.

**4.3. Results comparison between DFUEffectiveNET and the fine-tuned CNNs architecture**

**TABLE IV**

Comparison of DFUEffectiveNET with ultra-modern Networks. Here, V = Version, SPE = Specificity, PRE = Precision, REC = Recall, ACC = Accuracy, REC = Recall, and FS = F1-Score.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Model** | **ACC** | **PRE** | **REC** | **FS** | **SPE** |
| AlexNet [8] | 0.909 | 0.893 | 0.871 | 0.882 | 0.933 |
| VGG16 [9] | 0.925 | 0.923 | 0.878 | 0.900 | 0.954 |
| GoogLeNet [10] | 0.944 | 0.927 | 0.927 | 0.927 | 0.954 |
| DenseNet121 [11] | 0.956 | 0.944 | **0.944** | 0.944 | 0.964 |
| **DFUEffectiveNET V3** | **0.969** | **0.983** | 0.936 | **0.959** | **0.990** |

From the above comparison table, we showed that our proposed DFUEffectiveNET obtained better accuracy.

After the final epoch, the final training accuracies of DFUEffectiveNET, AlexNet [8], GooLeNet [10], and VGG16 [9] models are respectively 86.67%, 86.67%, 86.67%, 86.67%, 86.67%. Similarly, after the final epoch, the final validation accuracies of DFUEffectiveNET, AlexNet [8], GooLeNet [10], and VGG16 [9] models are respectively 86.67%, 86.67%, 86.67%, 86.67%, 86.67%. For each epoch, the validity loss and precision are displayed. After training the above models, these models are tested using an unseen test dataset.

**4.4. Results comparison between DFUEffectiveNET and existing models**

**Table v**

We contrast the outcomes of our proposed DFUEffectiveNET with those of other existing models.

Here, V = Version, SPE = Specificity, PRE = Precision, REC = Recall, ACC = Accuracy, REC = Recall, and FS = F1-Score.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Model** | **ACC** | **PRE** | **FS** | **REC** | **SPE** |
| DFUNet [4] | 0.925 | 0.945 | 0.939 | - | 0.911 |
| DFU\_QUTNet [5] | - | 0.954 | 0.945 | 0.936 | - |
| DFU\_SPNet[6] | 0.964 | 0.926 | 0.954 | **0.984** | 0.951 |
| Model with 4 branches [7] | - | 0.973 | 0.958 | 0.945 | - |
| **DFUEffectiveNET** **V3** | **0.969** | **0.983** | **0.959** | 0.936 | **0.990** |

**5 Discussion:**

The automatic classification of normal and ulcer skin samples using a novel approach is discussed in this paper. As we use a more generalizable Adam optimizer and fewer epochs for the network to converge, that reduces the training process.

Additionally, a few samples of predictions obtained from DFUEffectiveNet for both classes are provided in Fig. 9. Due to DFUEffectiveNet's great precision and specificity, it performed exceptionally well in correctly predicting both classes.

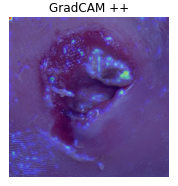
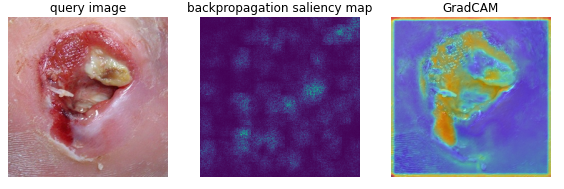
The proposed strategy is superior for many reasons, some of which are listed below:

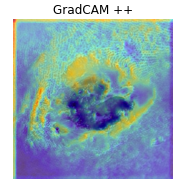
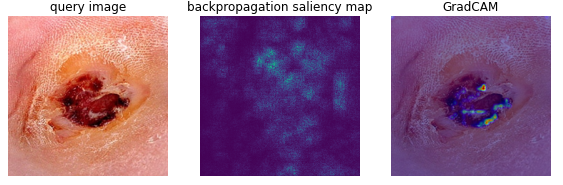
1) Using appropriate depth and width as well as parallel convolutional modules, gives the network a fairly good receptive field.

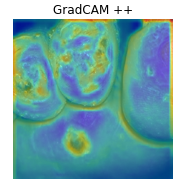
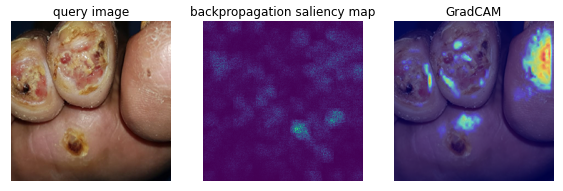
2) The detection of the DFU at various scales is further improved by the use of many distinct kernel numbers and sizes.

3) In addition, the more generalizable Adam optimizer and the use of fewer epochs strengthen the predictive efficiency of DFU\_EfficientNet.

According to the findings in Fig. 7, the DFUEffecNet can identify diabetic foot features at various phases of the DFU disease. The suggested method can therefore be used to automatically and objectively diagnose DFU. In addition, the maximum number of feature maps in parallel modules is simultaneously detected using new interpretable AI techniques to detect only ulcerated skin locations.







**FIG. 7:** Comparing different XAI visualization methods for DFU classification. Left to right: original DFU image, backpropagation saliency map, Grad-CAM, and Grad-CAM ++.

**6 Conclusion and Feature work:**

DFU\_EffectiveNet is a lightweight CNN model that classifies the DFU dataset into two classes (normal and ulcer skin). Therefore, we have shown how the custom DFU\_EffectiveNet architecture achieves improved accuracy, precision, F1-score, and specificity by reducing processing time and the number of neurons and layers in fully connected layers. These findings imply that the DFU\_EffectiveNet can assist DFU specialists in reaching a judgment more quickly. We compared our model's performance with that of four cutting-edge CNN networks (Densenet 121, GoogleNet, VGG16, and AlexNet), where our model surpassed all others.

This research has the potential to develop new technologies that could revolutionize the diagnosis and therapy of diabetic foot ulcers and change a paradigm in diabetic foot care. This work is presented to achieve better results in the future, including

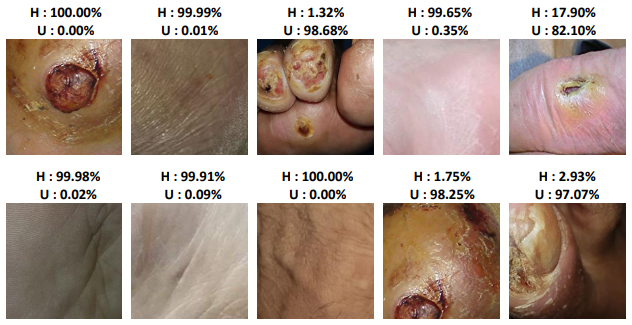
1) Automatically delineating and classifying images of foot ulcers by generating an automatic annotation without any intervention by clinicians;

2) Developing an automatic ulcer segmentation, localization, and detection system using these classifiers;

3) To practice the method of identifying different pathologies of DFU as a multi-category classification;

4) Facilitating the development of mobile applications based on user-friendly ulcer detection software.

The purpose of this study is to evaluate how well deep-learning techniques perform in localizing skin with ulcers and distinguishing those ulcers from healthy skin regions. This study provided an explainability framework for understanding the behavior of deep learning algorithms. This interpretable AI framework helps analyze the performance of CNNs using the characteristics of each internal filter in the CNN network.



**FIG. 9**. DFUEffectiveNet based some prediction samples.

**References**

[1] S. Wild, G. Roglic, A. Green, R. Sicree, and H. King, “Global prevalence of diabetes estimates for the year 2000 and projections for 2030,” Diabetes Care, vol. 27, no. 5, pp. 1047–1053, 2004.

[2] Cavanagh P, Attinger C, Abbas Z, Bal A, Rojas N, Xu Z-R (2012) Cost of treating diabetic foot ulcers in five different countries. Diabetes / Metabolism Research and Reviews 28(S1):107–111

[3] P. Cavanagh, C. Attinger, Z. Abbas, A. Bal, N. Rojas, and Z.-R. Xu, “Cost of treating diabetic foot ulcers in ﬁve different countries,” Diabetes/Metabolism Res. Rev., vol. 28, no. S1, pp. 107–111, 2012.

[4] Goyal, M.; Reeves, N.D.; Davison, A.K.; Rajbhandari, S.; Spragg, J.; and Yap, M.H. (2018). DFUNet: Convolutional neural networks for diabetic foot ulcer classification. IEEE Transactions on Emerging Topics in Computational Intelligence, 4(5), 728 - 739.

[5] Alzubaidi, L.; Fadhel, M. A.; Oleiwi; S. R.; Al-Shamma, O.; and Zhang,J. (2020). DFU\_QUTNet: diabetic foot ulcer classification using novel deep convolutional neural network. Multimedia Tools Applications, 79, 15655-15677

[6] Das, Sujit Kumar, Pinki Roy, and Arnab Kumar Mishra. "DFU\_SPNet: A stacked parallel convolution layers based CNN to improve Diabetic Foot Ulcer classification." *ICT Express* 8.2 (2022): 271-275.

[7] Alzubaidi, L. A. I. T. H., et al. "Comparison of hybrid convolutional neural networks models for diabetic foot ulcer classification." *J. Eng. Sci. Technol* 16 (2021): 2001-2017.

[8] Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems (pp. 1097–1105)

[9] Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition.arXiv preprint arXiv: 1409.1556

[10] C. Szegedy et al., “Going deeper with convolutions,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2015, pp. 1–9.

[11] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, ‘‘Densely connected convolutional networks,’’ in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Honolulu, HI, USA, Jul. 2017, pp. 2261–2269, doi: 10.1109/CVPR.2017.243.

[12] El-Gayar O, Timsina P, Nawar N, Eid W (2013) A systematic review of it for diabetes self-management: are we there yet? Int J Med Inform 82(8):637–652

[13] Vilcahuaman L, Harba R, Canals R, Zequera M, Wilches C, Arista M, Torres L, Arbanil H (2015) Automatic analysis of plantar foot thermal images in at-risk type II diabetes by using an infrared camera. IFMBE Proc 51:228–231

[14] C. Liu, J. J. van Netten, J. G. van Baal, S. A. Bus, and F. van Der Heijden, “Automatic detection of diabetic foot complications with infrared thermography by asymmetric analysis,” J. Biomed. Opt., vol. 20, no. 2, 2015, Art. no. 026003.

[15] Fraiwan L, AlKhodari M, Ninan J, Mustafa B, Saleh A, Ghazal M (2017) Diabetic foot ulcer mobile detection system using smart phone thermal camera: a feasibility study. Biomed Eng Online 16(1):117 [16] Fraiwan L, Ninan J, Al-Khodari M (2018) Mobile application for ulcer detection. The Open Biomedical Engineering Journal 12:16

[17] Hazenberg CE, van Netten JJ, van Baal SG, Bus SA, Bus S (2014) Assessment of signs of foot infection in diabetes patients using photographic foot imaging and infrared thermography. Diabetes Technol Ther 16(6):370–377

[18] Clemensen J, Larsen SB, Kirkevold M, Ejskjaer N (2008) Treatment of diabetic foot ulcers in the home: video consultations as an alternative to outpatient hospital care. Int J Telemed Appl 2008:1

[19] Foltynski P, Wojcicki JM, Ladyzynski P, Migalska-Musial K, Rosinski G, Krzymien J, Karnafel W (2011) Monitoring of diabetic foot syndrome treatment: some new perspectives. Artif Organs 35(2):176–182

[20] Bowling FL, King L, Paterson JA, Hu J, Lipsky BA, Matthews DR, Boulton AJ (2011) Remote assessment of diabetic foot ulcers 11 using a novel wound imaging system. Wound Repair Regen 19(1):25–30

[21] C. Liu, J. J. van Netten, J. G. van Baal, S. A. Bus, and F. van Der Heijden, “Automatic detection of diabetic foot complications with infrared thermography by asymmetric analysis,” J. Biomed. Opt., vol. 20, no. 2, 2015, Art. no. 026003.

[22] J. J. van Netten, M. Prijs, J. G. van Baal, C. Liu, F. van Der Heijden, and S. A. Bus, “Diagnostic values for skin temperature assessment to detect diabetes-related foot complications,” Diabetes Technol. Therapeutics, vol. 16, no. 11, pp. 714–721, 2014.

[23] Wang L, Pedersen PC, Agu E, Strong DM, Tulu B (2017) Area determination of diabetic foot ulcer images using a cascaded two-stage SVM-based classification. IEEE Trans Biomed Eng 64(9):2098–2109

[24] Goyal M, Yap MH, Reeves ND, Rajbhandari S, Spragg J (2017) Fully convolutional networks for diabetic foot ulcer segmentation. In: 2017 IEEE international conference on systems, man, and cybernetics (SMC) (pp. 618–623). IEEE

[25] Zhu X, Li Z, Zhang X-Y, Li P, Xue Z, Wang L (2018) Deep convolutional representations and kernel extreme learning machines for image classification. Multimed Tools Appl:1–20

[26] Huang J, Li B, Zhu J, Chen J (2017) Age classification with deep learning face representation. Multimed Tools Appl:20231–20247

[27] Al-Zubaidi L (2016) Deep learning based nuclei detection for quantitative histopathology image analysis (Doctoral dissertation, University of Missouri–Columbia)

[28] Sert M, Boyacı E (2019) Sketch recognition using transfer learning. Multimedia Tools and Applications.

[29] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” Nature, vol. 521, no. 7553, pp. 436–444, 2015. [30] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “Imagenet classification with deep convolutional neural networks,” in Proc. Adv. Neural Inf. Process. Syst., 2012, pp. 1097–1105.

[31] Y. LeCun, Y. Bengio, and G. Hinton, “Deep learning,” Nature, vol. 521, no. 7553, pp. 436–444, 2015

[32] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” arXiv preprint arXiv:1409.1556, 2014.

[33] J. Long, E. Shelhamer, and T. Darrell, “Fully convolutional networks for semantic segmentation,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit., 2015, pp. 3431–3440.

[34] N. Tajbakhsh et al., “Convolutional neural networks for medical image analysis: Full training or fine tuning?” IEEE Trans. Med. Imag., vol. 35, no. 5, pp. 1299–1312, May 2016.

[35] Cheng PM, Malhi HS (2017) Transfer learning with convolutional neural networks for classification of abdominal ultrasound images. J Digit Imaging: 234–243.

[36] Ravishankar H, Sudhakar P, Venkataramani R, Thiruvenkadam S, Annangi P, Babu N, Vaidya V (2016) Understanding the mechanisms of deep transfer learning for medical images. In: Deep learning and data labeling for medical applications. Springer, Cham, pp 188–196

[37] Yu Y, Lin H, Meng J, Wei X, Guo H, Zhao Z (2017) Deep transfer learning for modality classification of medical images. Information 8(3):91

[38] Koushik J (2016) Understanding convolutional neural networks. arXiv preprint arXiv:1605.09081

[40] 18. Gibiansky A (2018) Convolutional neural networks. [http://andrew.gibiansky.com/blog/machine-learning / convolutional- neural-networks/. Accessed 26 December 2018](http://andrew.gibiansky.com/blog/machine-learning%20/%20convolutional-%20neural-networks/.%20Accessed%2026%20December%202018)

[39] Vedaldi A, Lenc K (2015) Matconvnet: convolutional neural networks for matlab. In: Proceedings of the 23rd ACM international conference on multimedia (pp. 689–692). ACM

[41] Ioffe S, Szegedy C (2015) Batch normalization: accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167

[42] Dahl GE, Sainath TN, Hinton GE (2013) Improving deep neural networks for LVCSR using rectified linear units and dropout. In: 2013 IEEE international conference on acoustics, speech and signal processing (pp. 8609–8613). IEEE

[43] Battineni, G.; Sagaro, G.G.; Chinatalapudi, N.; Amenta, F. Applications of Machine Learning Predictive Models in the Chronic Disease Diagnosis. J. Pers. Med. 2020, 10, 21.

[44] Topol, E.J. High-performance medicine: The convergence of human and artificial intelligence. Nat. Med. 2019, 25, 44–56.

[45] Antoniadi, A.; Du, Y.; Guendouz, Y.; Wei, L.; Mazo, C.; Becker, B.; Mooney, C. Current Challenges and Future Opportunities for XAI in Machine Learning-Based Clinical Decision Support Systems: A Systematic Review. Appl. Sci. 2021, 11, 5088.