IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 5 April 2023, accepted 6 May 2023, date of publication 10 May 2023, date of current version 30 May 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3274848

RESEARCH ARTICLE

DeepSkin: A Deep Learning Approach for Skin Cancer Classification

H. L. GURURAJ[1], (Senior Member, IEEE), N. MANJU[2], A. NAGARJUN[2], V. N. MANJUNATH ARADHYA[3], AND FRANCESCO FLAMMINI[4], (Senior Member, IEEE)

[1]Department of Information Technology, Manipal Institute of Technology Bengaluru, Manipal Academy of Higher Education, Manipal 576104, India
[2]Department of Information Science and Engineering, JSS Science and Technology University, Mysuru 570006, India
[3]Department of Computer Application, JSS Science and Technology University, Mysuru 570006, India
[4]IDSIA USI-SUPSI, University of Applied Sciences and Arts of Southern Switzerland, 6928 Manno, Switzerland

Corresponding author: Francesco Flammini (francesco.flammini@supsi.ch)

ABSTRACT Skin cancer is one of the most rapidly spreading illnesses in the world and because of the limited resources available. Early detection of skin cancer is crucial accurate diagnosis of skin cancer identification for preventive approach in general. Detecting skin cancer at an early stage is challenging for dermatologists, as well in recent years, both supervised and unsupervised learning tasks have made extensive use of deep learning. One of these models, Convolutional Neural Networks (CNN), has surpassed all others in object detection and classification tests. The dataset is screened from MNIST: HAM10000 which consists of seven different types of skin lesions with the sample size of 10015 is used for the experimentation. The data pre-processing techniques like sampling, dull razor and segmentation using autoencoder and decoder is employed. Transfer learning techniques like DenseNet169 and Resnet 50 were used to train the model to obtain the results.

INDEX TERMS Skin cancer, segmentation, deep learning, CNN, Densenet169, Resnet50.

I. INTRODUCTION

A tumor is formed when healthy cells begin to change and grow out of control. Both cancerous and noncancerous tumors are conceivable. Malignant tumors are those that have the potential to grow and spread to other areas of the body [1]. A benign tumor may form, but it does not usually spread. Skin cancer is the result of abnormal skin cell growth. It is the most prevalent cancer nowadays and occurs everywhere. Every year, various forms of melanomas are thought to cause more than 3.5 million cases to be discovered [2], [3]. This number exceeds the sum of cases of lung, bone, and colon cancers. In reality, a person with melanoma dies every 57 seconds. When cancer is detected in dermoscopy images in advance, the survival percentage is significantly boosted. Therefore, accurate automatic skin excrescence discovery will undoubtedly help pathologists become more skilled and productive. The purpose of the dermoscopy technique is to improve each melanoma patient's performance. Noninvasive skin imaging technique dermoscopy uses a magnified and lighted picture of the affected skin area to increase visibility of the spots, therefore reducing facial reflection [4]. Skin cancer early detection is still a prized possession. It's difficult to tell if a skin lesion is benign or malignant because they all seem similar. The sun's harmful ultraviolet (UV) rays and the usage of UV tanning beds are the two most common causes of skin cancer. It is particularly difficult for dermatologists to distinguish between melanoma and non-melanoma lesions because of the low degree of difference between lesions and skin [5]. The main problem of similar opinion is largely dependent on private judgment and is scarcely reproducible. With the help of robotization using operation of deep literacy helps the case to get the early opinion report and grounded on the report case can consult dermatologists for treatment [6]. An early diagnosis of skin cancer is crucial and has limited number of available treatment options. Accurate evaluation and the capacity to accurately identify skin cancer are critical components of a skin cancer prevention approach. Even in literacy tasks that are unsupervised, deep literacy has been widely adopted [7]. Object detection and bracket tasks have

The associate editor coordinating the review of this manuscript and approving it for publication was Huiyan Zhang.

been dominated by Convolutional Neural Networks (CNN). As a result, trained end-to-end in a controlled environment, CNNs eliminate the need for humans to manually create feature sets. The use of Convolutional Neural Networks (CNNs) to categorize lesions in skin cancer in recent years has outperformed skilled mortal specialists.

II. LITERATURE REVIEW

The methodology for classification of skin lesions with fine-tuned neural networks is proposed in [8]. For the purpose of balancing the dataset, the skin lesion photos are resampled. A combination of DenseNet and U net is trained for segmentation and then used to fine-tune the following classifiers. The encoder element of the segmentation model's extracted architecture is then trained to categorize the seven different skin disorders. The classification model's average balanced accuracy was 0.836 in the test set and 0.840 in the validation set. An innovative strategy is put forth in [9] that can pre-process the image automatically before segmenting the lesion. Hair, gel, bubbles, and specular reflection are just a few of the undesirable artefacts that the system filters. An innovative method for identifying and painting the hairs visible in cancer images is given using the wavelet idea. Utilizing an adaptive sigmoidal function that manages the localised intensity distribution within the pictures of a specific lesion, the contrast between the lesion and the skin is improved. We then provide a segmentation method to precisely separate the lesion from the surrounding tissue. On the European dermoscopic image database, the proposed method is tested. The proposed system is focusing on classifying skin lesions in deep learning with specific implementation of CNN approach [10]. The methods in this paper includes the screening of a dataset from MNIST: HAM10000 which consists of seven different types of skin lesions with the sample size of 10015. The methods in this paper included training the model with the help of CNN and obtained an accuracy of 78%. In the study, a deep fully convolutional neural network is presented for semantic pixel-by-pixel segmentation. A pixel- wise classification layer, a corresponding decoder network, and an encoder network make up the trainable network [1]. Convolution layers are identical to those in VGG-16's 13 convolution layers. As its name implies, the decoder network's primary goal is to convert encoder feature maps into full input resolution feature maps. The method is focusing on classifying skin lesions in deep learning with specific implementation of CNN approach [2]. The methods in this paper includes the screening of a dataset from MNIST: HAM10000 which consists of seven different types of skin lesions with the sample size of 10015. The methods in this paper included training the model with the help of CNN and obtained an accuracy of 88% and used transfer learning methods like the Resnet model. An application of machine learning with a focus on skin cancer categorization has been developed [11]. Pre-processing, segmentation, feature extraction, and classification are all included in the research. The ABCD rule, GLCM, and HOG were used to extract features. Some of the most commonly used machine learning techniques were employed. 328 images of benign melanoma and 672 images of melanoma were retrieved from the ISIC collection. With SVM classifiers, a 97.8% accuracy and 0.94 area under the curve resulted in a classification [2], [12]. Furthermore, the KNN results showed that the Sensitivity was 86.2% and the Specificity was 85%. The proposed strategy in [13] focuses on the identification and categorization of skin cancer through the specialised use of machine learning methodology. The classification rate for unsupervised learning using the k-means algorithm is 52.63%. The k-means algorithm divides the input data into n data points and k clusters. Two clusters are produced when melanoma skin cancer is detected; one cluster is for cancer detection and the other is for non-cancer detection. Back Propagation Neural Network classification accuracy has been found to range from 60% to 75%, whereas Support Vector Machine (SVM) accuracy ranges from 80% to 90%. As a result, Support Vector Machine performs better than K-means clustering and Back Propagation Neural Network classification. Unsupervised learning is used as the foundation for the K-means method. The proposed approach in [14] concentrates on classification issues. The main goal of this study is to categorise skin lesions using deep learning, especially using the CNN method. The dataset was compiled using ISIC. The techniques include the employment of transfer learning algorithms like Inception V3, Resnet and VGG-16 and Mobilenet, as well as data augmentation, image normalisation, and image normalizing. The approaches utilised in [15] are focused on employing the supervised learning method to categorise skin lesions. With the specialised implementation of computer aided diagnosis, MAP estimate can carry out numerous routine activities in automated skin lesion diagnosis (CAD). Lesion segmentation, hair detection, and pigment network detection are some of the techniques. The developed model has an accuracy rate of 86%. The techniques include computer-aided diagnostics and MAP estimate for categorising skin lesions. The methods proposed in [16] is focusing on detection and classification of skin cancer with specific implementation of deep learning approach. The methods in this paper includes the screening of a dataset from MNIST: HAM10000 which consists of seven different types of skin lesions with the sample size of 10015 and PH2 dataset which contains 200 images of skin lesions. The methods include data augmentation and the model is trained using deep learning architectures like mobilenet, and VGG-16 [17]. The accuracy obtained is 81.52% with mobilenet and 80.07% with VGG-16.

III. METHODOLOGY

Detecting skin cancer at an early stage is challenging for dermatologists. With the extensive use of deep learning procedures as shown in Figure1 helps to classify the seven types of skin cancer images. Deep learning methods like Convolutional Neural Networks (CNN), has surpassed all others in object detection and classification tests.

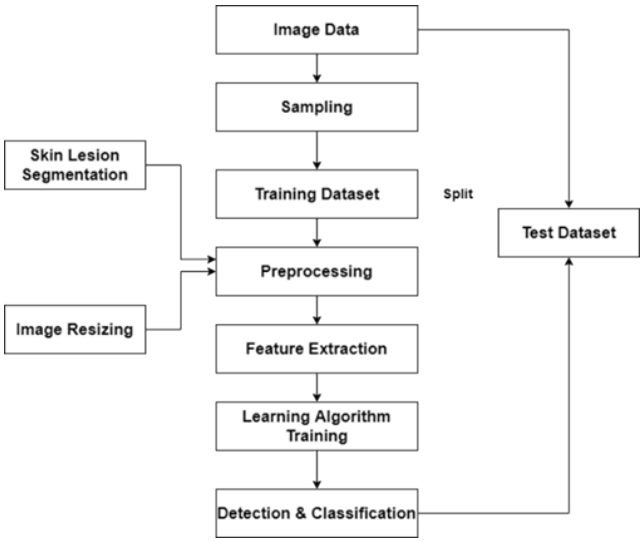

FIGURE 1. Flow of methodology.

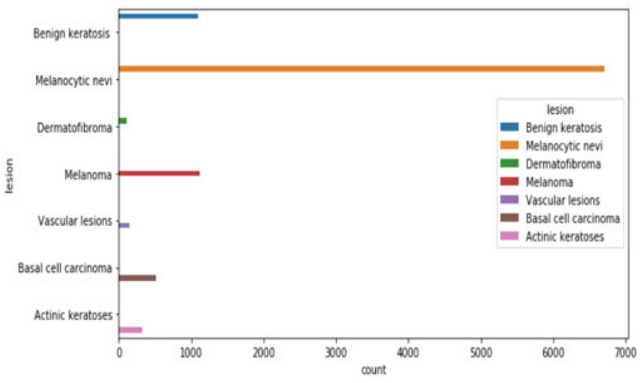

FIGURE 2. Data visualization.

A. DATASET DESCRIPTION

The dataset contains images of skin lesions from seven different classes. HAM10000 and ISIC from the National Institutes of Standards and Technology (NIST). It includes 10015 dermatoscopic images of all significant diagnostic categories of pigmented skin lesions. Which includes actinic keratoses and intraepithelial carcinoma/disease Bowen's (akiec), basal cell carcinoma (bcc), benign keratosis-like lesions (solar lentigines/seborrheic keratoses and lichen-plan (angiomas, angiokeratomas, pyogenic granulomas and hemorrhage, vasc).

B. DATA VISUALIZATION

The changes we make to our data before submitting it to the algorithm are referred to as pre-processing. Raw data may be transformed into a clean data collection via pre-processing. In other words, data obtained from various sources is often obtained in a raw format that is inappropriate for analysis. The visual presentation of information and data is known as data visualization. Figure 2. shows the graphical representation of data visualization of the skin lesions.

The changes we make to our data before submitting it to the algorithm are referred to as pre-processing. Raw data may be transformed into a clean data collection via pre-processing. In other words, data obtained from various sources is often obtained in a raw format that is inappropriate for analysis. The visual presentation of information and data is known as data visualization. Figure 2. shows the graphical representation of data visualization of the skin lesions.

The dataset is unbalanced in this case. Hence, we can use a technique called sampling to balance the dataset. Data sampling is a set of techniques for reshaping a training dataset so that classes are more evenly distributed. Once the dataset has been balanced, standard machine learning algorithms may be used to train on the adjusted dataset without any adjustments.

Oversampling (up sampling): Balanced datasets can be created by altering data classes that are not equal in value. To compensate when there's not enough data, the oversampling approach increases uncommon samples' sizes. Minorities are more prevalent in the training set when oversampling is used. Oversampling has the benefit of retaining every observation from the original training set for both the minority and majority classes. Oversampling works very well for small datasets. Undersampling (down sampling): This approach reduces the size of the overabundant class in order to balance the dataset. It aims to reduce the number of majority samples in order to get a more even distribution of classes. As a result of eliminating data from the original, it may lose out on important information. The bulk of the population can benefit from this strategy. It decreases the number of observations from the dominating class in order to make the data set more equal. We should use this method only when the dataset is big and the amount of training samples helps decrease run time and storage difficulties. Figure 3. shows the balanced dataset of the skin lesions to the respective classes they belong to.

C. REMOVING THE NOISE FROM THE SKIN IMAGE

The Dull Razor method can be used to remove hair from skin images. Dull Razor takes the following actions:

1. The dark hair locations are identified by the use of a generalized grayscale morphological closing technique.
2. A bilinear interpolation is used to replace the hair pixels that have been verified as having a thin and long structure.
3. To smooth out the hair pixels that have been replaced, it uses an adaptive median filter.

Let G_t represent the original band's generalized grayscale closure pictures, and O_t, S_1, S_2, and S_3 represent the horizontal, vertical, and diagonal structure elements. G_t can be written as given in equation (1)

$$G_t = |O_t - max\{O_t \cdot S_1 O_t \cdot S_2 O_t \cdot S_1\}| \quad (1)$$

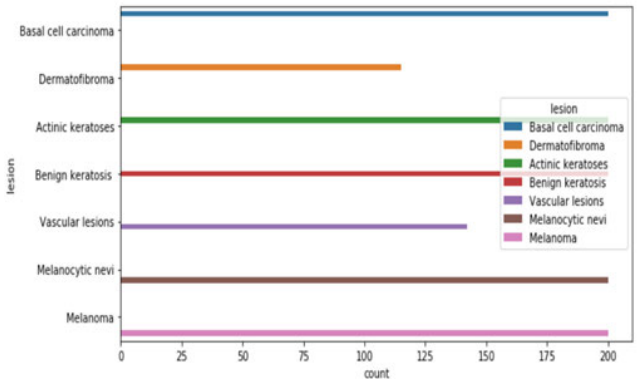

FIGURE 3. Data visualization.

where · indicates that the operation is in grayscale. Furthermore, for the coordinates (x, y), Mr (x, y), the binary hair mask pixel is computed as shown in equation (2)

$$\begin{aligned} \text{Mr}\,(\text{x}, \text{y}) &= 1, \quad \text{if Gr}\,(\text{x}, \text{y}) > \text{T} \\ &= 0, \quad \text{otherwise} \end{aligned} \tag{2}$$

where T is a predefined threshold value.

For green and blue channels, we may use a phrase like that. It's the aggregate of the three-color channels' hair masks that makes up the final mask for the original image, M as given in equation (3).

$$M = M_r \cup M_a \cup M_b \tag{3}$$

The hair masks for the appropriate color channels are M_r, M_a, and M_b. Due to the thresholding procedure and image noise, it is difficult to determine exactly where the borders are in the image. It is common to produce tiny lines around the hair region's boundary. The thin lines are smoothed out using an adaptive median filter in this step. The reduction of intensity shifts from one pixel to the next is its main goal. In this filter, the pixel value is switched out for the median value. By arranging all the pixel values in ascending order, the median is calculated by substituting the calculated pixel with the middle pixel value. Image smoothing with mean filtering minimizes the intensity variation across pixels in a plain, intuitive, and straightforward manner. It is often used to lessen the amount of noise in images. The median filter compares each pixel in the picture to its immediate surroundings to see if it is representative of those in the near vicinity. Instead of just replacing the pixel value with the average of surrounding pixel values, the median of those values is utilized. The outcome of subtracting the closed picture and the original image is black hat filtering. This is used to extract minute details from the structures in the original image, highlighting brilliant items in dark backgrounds. Figure 4. shows the removal of hair from the skin lesion.

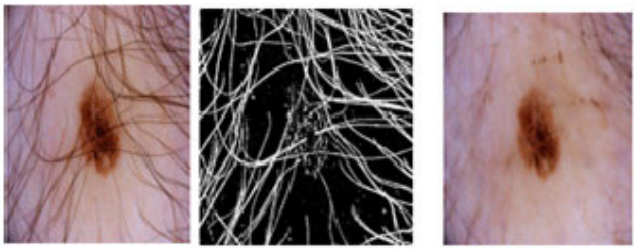

FIGURE 4. Removal of hair from the skin lesions.

D. SEGMENTATION OF IMAGE USING ENCODER AND DECODER

The technique of segmenting an image involves dividing the foreground from the background or grouping pixels according to how similar they are in terms of color or shape. The encoder and decoder technology are used to segment images for biomedical image segmentation.

Encoder: Convolutions and max pooling are done at the encoder. From the original fully connected layers, 13 convolutional layers were eliminated. The associated max pooling indices (locations) are stored while executing 2 × 2 max pooling.

$$\begin{aligned} \phi &: X \to F \\ \psi &: F \to X \\ \phi, \psi &= argmin\|X - (\psi \circ \phi)X\|^2 \end{aligned} \tag{4}$$

In the encoder function, indicated by ϕ, a latent space F is used to map the original data X. To get the data out of the bottleneck, we use the decoder function, which we indicate by ψ. Here, the result is the same as what you got when you put it in. After some generic nonlinear compression, we are essentially attempting to restore the original image which is given in equation (4). Standard neural network function is transferred via an activation function, which represents the latent dimension z in the encoding network is given in equation (5).

$$Z = \sigma(Wx + b) \tag{5}$$

It is also possible to describe the decoding network this way, but with a wide range of weights, biased and activation functions. The autoencoder's purpose is to pick our encoder and decoder algorithms such that we only use the lowest quantity of data to encode the image for it to be regenerated on the other side.

Decoder: Upsampling and convolutions are done at the decoder. Each pixel has a softmax classifier at the end. Prior to sampling, encoder layers remember their highest possible maximum pooling indices. In the end, a K-class softmax classifier is used to predict each pixel's class. Image thresholding**:**Images are resized when they are scaled down. A wide range of image processing tasks benefit from scaling. As a result, the number of pixels in an image may be reduced, which has several benefits. Since the more pixels in an image

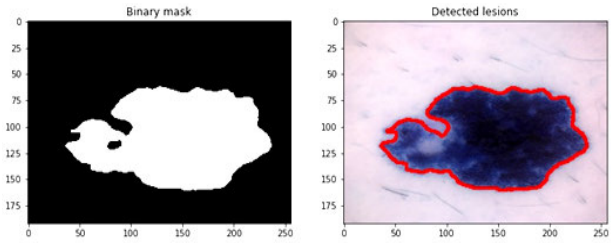

FIGURE 5. Output of skin lesion thresholding.

there are, the more input nodes there are, which increases the model's complexity, training a neural network may be shortened. It also helps with zooming in on images. We frequently need to resize an image, either to reduce it or to scale it up to suit the size requirements. Figure5 shows the output for image thresholding for skin lesion image. a neural network may be shortened. It also helps with zooming in on images. We frequently need to resize an image, either to reduce it or to scale it up to suit the size requirements. Figure 5. shows the output for image thresholding for skin lesion image.

IV. SYSTEM IMPLEMENTATION

Convolutional Neural Networks (CNNs) are built in such a way that they can account for the input's spatial structure. They were originally developed to work with images and were inspired by the visual system of the mouse. CNNs feature fewer parameters than ordinary neural networks, allowing for the efficient training of very deep designs (usually more than 5 layers which is almost impossible for fully-connected network. An input layer, an output layer, and several hidden layers make up a convolutional neural network. Convolutional, RELU (activation function), pooling, fully connected, and normalization layers are often seen in a CNN's hidden layers. Pre-trained models like Densenet169 and Resnet50 were used to make a comparison study between oversampling and undersampling technique. Densenet169 architecture didn't supported for Oversampling technique. Table 1 shows training the model for 30 epochs.

Learning graph: At each iteration, the training accuracy improves but the validation loss is detected more often, resulting in a reduction in the training accuracy at each iteration of the plot and Figure 8 shows the learning graph of training and validation accuracy plotted for 30 epochs. Figure 6 shows the learning graph of training and validation accuracy for 30 epochs with the batch size of 32 and dropout layer of 0.5 to avoid overfitting of the model. Figure 8 shows the learning graph of training and validation loss for 30 epochs.

The accuracy of the model obtained is 91.2%. The RELU activation functions is used for training Relu (Rectified linear Unit) shown in equation (6)

It is defined as

$$y = max(0, x) \tag{6}$$

Rectified linear is a more is more interesting transform that activates a node only if the input is above a certain quantity.

TABLE 1. Training the model for 30 epochs.

Epoch	Train_loss	Valid_loss	Accuracy	F_beta	Time
1	2.085684	1.64159	0.350598	0.340776	00:39
2	1.723471	1.072841	0.609562	0.562534	00:27
3	1.357395	0.835934	0.709163	0.683892	00:26
3	1.075796	0.664414	0.749004	0.726332	00:27
5	0.878755	0.620203	0.788845	0.787139	00:27
6	0.773527	0.565302	0.776892	0.783491	00:26
7	0.662125	0.582520	0.788845	0.824112	00:27
8	0.593850	0.550581	0.816733	0.820410	00:27
9	0.532479	0.537552	0.824701	0.841533	00:27
10	0.483688	0.431171	0.844622	0.810738	00:27
11	0.425181	0.518207	0.796813	0.860285	00:26
12	0.375488	0.408174	0.864542	0.874379	00:27
13	0.368176	0.415641	0.828685	0.839563	00:27
14	0.307774	0.472929	0.884462	0.887070	00:27
15	0.306202	0.366155	0.892430	0.893602	00:26
16	0.280250	0.338538	0.884462	0.884261	00:27
17	0.242262	0.340482	0.884462	0.887629	00:27
18	0.232738	0.349039	0.892430	0.895714	00:26
19	0.196570	0.363468	0.912351	0.917444	00:26
20	0.176028	0.304878	0.904382	0.95152	00:27
21	0.185741	0.348560	0.904382	0.910507	00:26
22	0.152473	0.300359	0.884462	0.906855	00:27
23	0.157553	0.326813	0.908367	0.889172	00:27
24	0.128807	0.347411	0.904382	0.911148	00:26
25	0.130462	0.308695	0.900398	0.906195	00:39
26	0.100807	0.322980	0.912351	0.904567	00:27
27	0.103093	0.325383	0.904382	0.914000	00:26
28	0.093458	0.318765	0.9048367	0.904804	00:27
29	0.119412	0.344834	0.904382	0.909710	00:27
30	0.099323	0.321925	0.9048367	0.909713	00:27

While the input is below zero, the output is zero, but when the input rises above a certain threshold, it has a linear relationship dependent variable f(x) = max (0, x) as shown in Figure 8.

Softmax is a function that takes the output values from the model and turns it into a probability function. The softmax function is given by the formula in equation (7).

$$\frac{e^{z_i}}{\sum_{j=1}^{K} e^{z_j}} \tag{7}$$

Performance on issues involving sparse gradients is enhanced by the use of the adaptive gradient algorithm (AdaGrad) (e.g., natural language and computer vision problems). Additionally, using Root Mean Square Propagation (RMSProp), per-parameter learning rates are adjusted to correspond to the mean of recent weight gradient magnitudes (e.g., how quickly

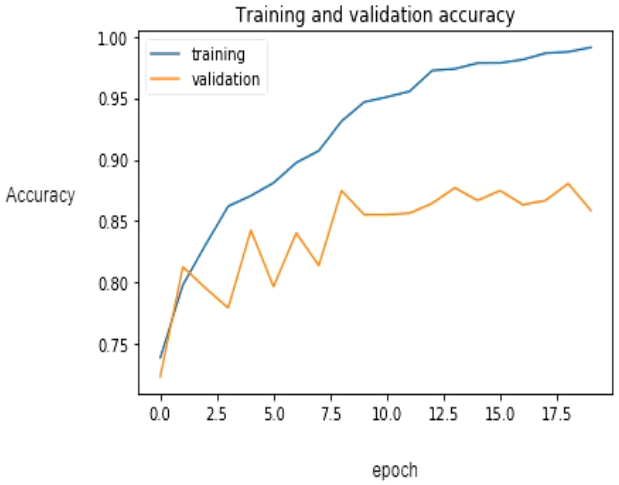

FIGURE 6. Training and validation accuracy graph for 30 epochs.

TABLE 2. Hyperparameters used for denset 169.

Split Ratio	Learning Rate	Batch Size
80:20	0.0001	32
70:30	0.000001	
40:60	0.0001	

it is changing). This implies that the method performs well for online and non-stationary situations (e.g., noisy). Adam is aware of the advantages of RMSProp and AdaGrad. Binary cross entropy is the chosen loss function. The definition of the loss function is given in equation (8).

$$H_p(q) = -\frac{1}{N}\sum_{i=1}^{N} y_i \times log(p(y_i)) + (1\text{-}y_i) \times \log(1 - p(y_i)) \tag{8}$$

The accuracy of the model obtained is 91.2%. The training and testing ratio were set to different ranges like 80:20, 70:30 and 40:60. For 70:30 ratios, the model is trained for 20 epochs with a learning rate of 0.000001. The accuracy of the model obtained is 87.7%. Figure 9 shows the training and validation accuracy graph plot and Figure 10 shows the training and the validation loss graph plot.

It takes the model 30 epochs to train at 0. 0001 learning rate for 40:60 ratios. The model's accuracy was determined to be 82.8%. Figure 11 shows the training and the validation accuracy and Figure 12 shows the loss graph plot. Table 1 shows the hyperparameters used for densenet 169 model.

Resnet50 architecture did not supported for undersampling technique. The model is trained for 10 epochs with learning rate of 0.01 as given in Table 2.

At each iteration, the training accuracy improves but the validation loss is detected more often, resulting in a reduction in the training accuracy at each iteration of the plot as shown in Figure 13.

The accuracy of the model obtained is 81%. To increase training accuracy, the model is trained across three epochs

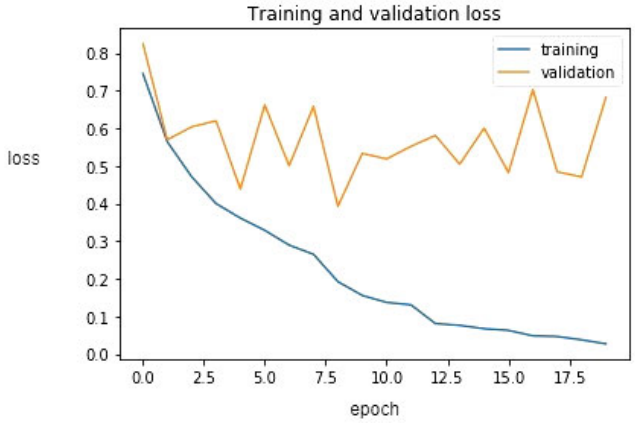

FIGURE 7. Training and validation a graph for 30 epochs.

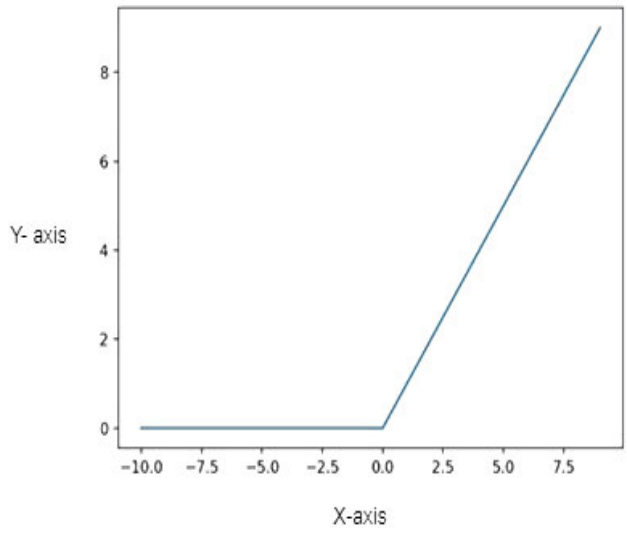

FIGURE 8. Relu Activation function graph.

FIGURE 9. Training and validation accuracy graph for 20 epochs.

with a learning rate of 0.0001. Table 3 shows training the model for 3 epochs.

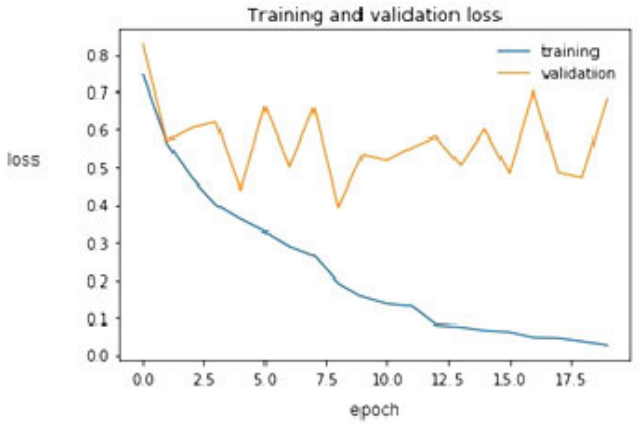

FIGURE 10. Training and validation loss graph for 20 epochs.

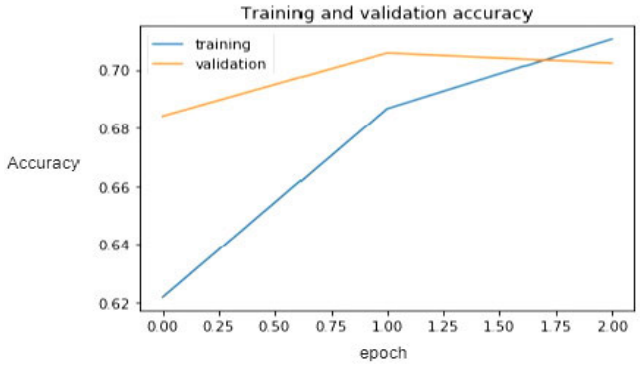

FIGURE 11. Training and validation accuracy graph for 20 epochs.

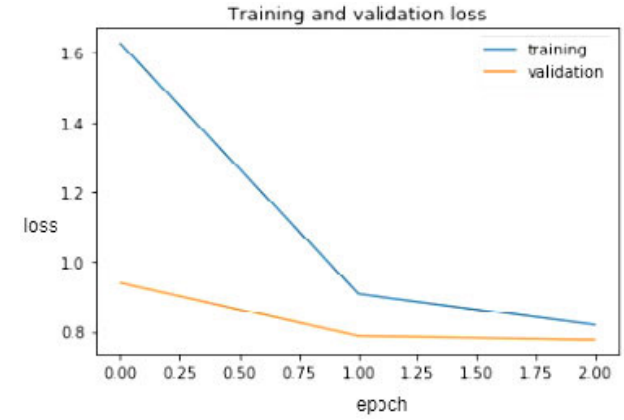

FIGURE 12. Training and validation loss graph for 20 epochs.

TABLE 3. Training the model for 10 epochs.

Epoch	Train_loss	Valid_loss	Accuracy	Time
1	1.957333	1.304661	0.678750	02:09
2	1.273871	0.743658	0.748125	02:05
3	0.956650	0.597204	0.750625	02:05
3	0.797674	0.582422	0.789375	02:04
5	0.696314	0.559734	0.791875	02:094
6	0.614487	0.533615	0.803125	02:04
7	0.555274	0.522421	0.809375	02:04
8	0.497298	0.522421	0.805625	02:04
9	0.464781	0.500112	0.815625	02:05
10	0.442373	0.498877	0.815000	02:05

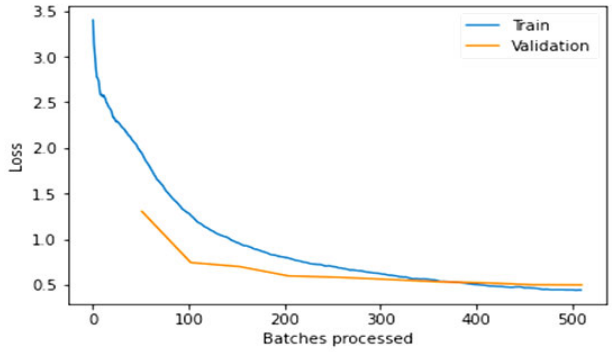

FIGURE 13. Learning graph of training for 10 epochs.

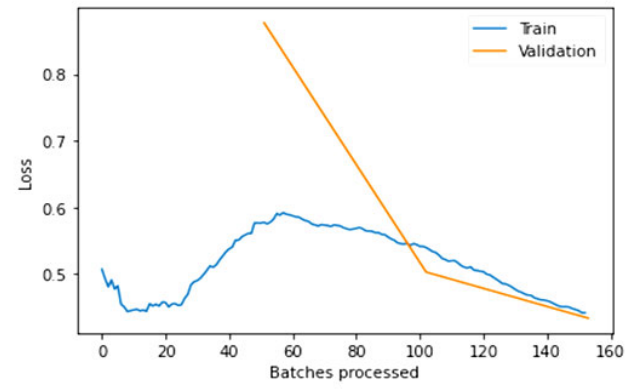

FIGURE 14. Learning graph of training for 3 epochs.

The accuracy of the model obtained is 83%. Training loss versus validation loss and training accuracy vs validation accuracy are shown in the graph given below as shown in Figure 14.

The training and testing ratio were conducted on different ranges like 80:20, 70:30 and 40:60. Using a learning rate of 0.00001 and 30 training epochs, the model is trained to learn 70:30 with the batch size of 16 dropout layer of 0.5 to avoid overfitting of the model. The model's accuracy is 80.9%. Figure 15 shows the model accuracy graph plot and Figure 16 shows the model loss learning graph plot.

For 40:60 the model is trained for 30 epochs with a learning rate of 0.0001. The accuracy of the model obtained is 81.6%. At each iteration, the training accuracy is shown below the validation accuracy displayed in the plot of training loss versus validation loss. Figure 17 shows the model accuracy graph plot and Figure 18 shows the model loss learning graph plot.

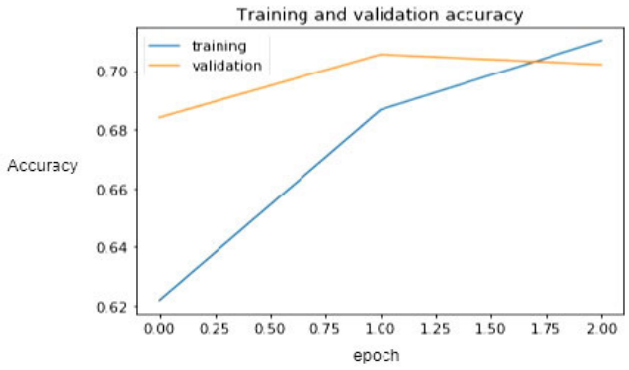

FIGURE 15. Training and validation accuracy graph for 30 epochs.

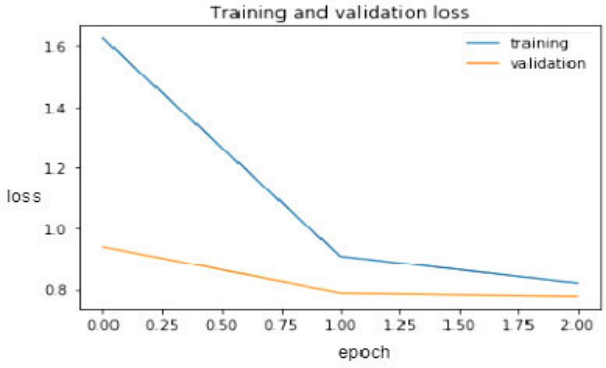

FIGURE 16. Training and validation loss graph for 30 epochs.

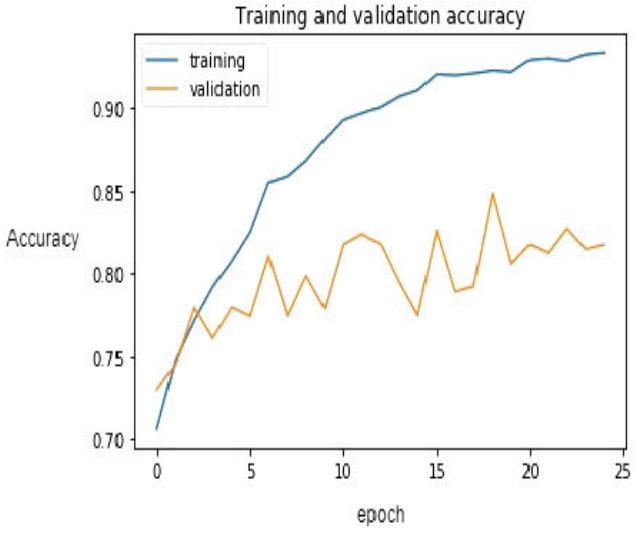

FIGURE 17. Training and validation accuracy graph for 30 epochs.

V. RESULT AND DISCUSSION

Confusion matrix can be used to produce the F1 score as an evaluation metric for the model. The performance of a classification model (or ''classifier'') on a set of test data for which the real values are known can be represented using a table called a confusion matrix. The model is trained with both undersampling and oversampling techniques and observations are made by using confusion matrix. The accuracy obtained by using undersampling technique is 91.2% and f1 measure of 91.7%.

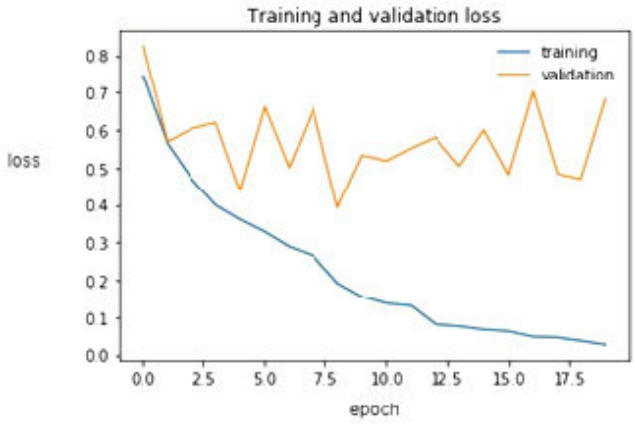

FIGURE 18. Training and validation loss graph for 30 epochs.

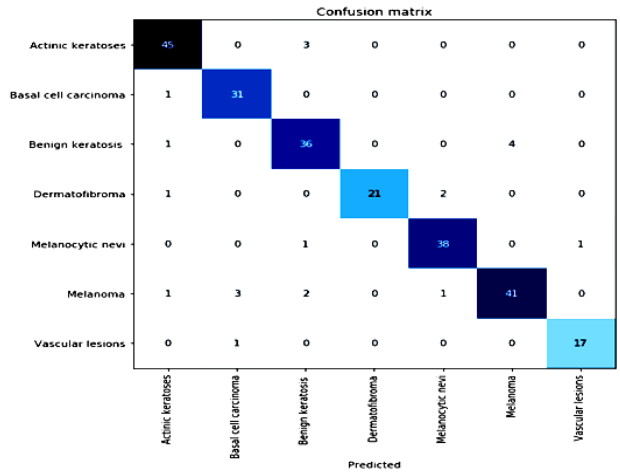

FIGURE 19. Confusion matrix for undersampling.

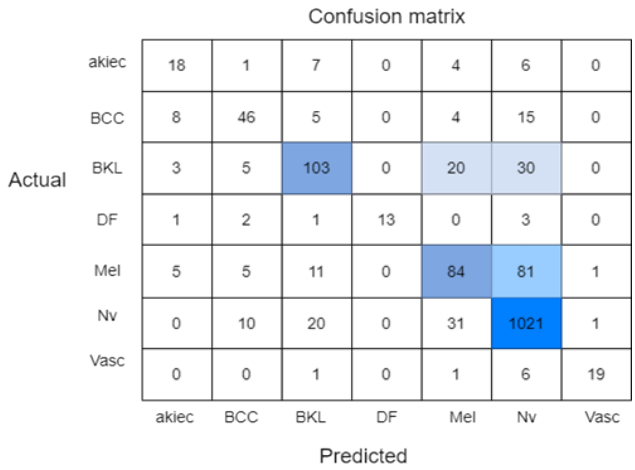

FIGURE 20. Confusion matrix for Oversampling.

TABLE 4. Training the model for 3 epochs.

Epoch	Train_loss	Valid_loss	Accuracy	Time
1	0.576541	0.876795	0.760625	02:12
2	0.541571	0.503058	0.823125	02:11
3	0.442065	0.433988	0.843750	02:11

TABLE 5. Classification report.

Class	Precision	Recall	F1-score	Support
Akiec	0.55	0.44	0.49	0036
BCC	0.71	0.64	0.68	0078
BKL	0.77	0.66	0.71	0169
Df	0.74	0.70	0.72	0020
Mel	0.66	0.55	0.60	0187
Nv	0.89	0.95	0.92	1083
Vasc	0.93	0.93	0.93	0027

TABLE 6. Comparison between undersampling and oversampling with different split.

Technique	Split Ratio	Accuracy in %
Under sampling Using Densenet169	80:20	91.20
	70:30	87.70
	40:60	82.80
Over sampling Using Resnet50	80:20	83.00
	70:30	80.90
	40:60	81.60

TABLE 7. Comparison between undersampling and oversampling.

Technique	Accuracy	F1 score in %
Under sampling Using Densenet169	91.20%	91.70
Over sampling Using Resnet50	83.00%	84.00

Figure 19 shows the confusion matrix for undersampling technique. The accuracy obtained by using undersampling technique is 83% and f1 measure of 84%. Figure 20 shows the confusion matrix for oversampling technique.

The training and testing ratio were conducted on different ranges like 80:20, 70:30 and 40:60. as shown in Table 5. Table 6 shows the comparison between undersampling and oversampling technique.

VI. COMPARISON BETWEEN EXISTING METHOD

Our proposed work performs better when compare with other CNN models that were recently published. We noticed that, the AUC score of 0.912 higher than the result in [17], 4% higher than the work carried out in [18] and 5% higher when compared with [19].

VII. CONCLUSION

Skin cancer is one of the illnesses that is spreading the quickest on the earth. Skin cancer is mostly brought on by a person's vulnerability to the sun's UV radiation. Given the limited resources available, early identification of skin cancer is essential. Accurate diagnosis and identification viability are generally essential for skin cancer prevention strategies. Additionally, dermatologists have trouble seeing skin cancer in its early stages. The use of deep learning for both supervised and unsupervised applications has increased significantly in recent years. Convolutional Neural Networks (CNNs) are one of these models that have excelled in object identification and classification tasks (CNN). The dataset is filtered from MNIST: HAM10000, which has a sample size of 10015 and includes seven different types of skin lesions. Data preprocessing methods include sampling, segmentation using an autoencoder and decoder, and dull razor. The model was trained using transfer learning methods like DenseNet169 and Resnet 50. Different ratios were used for the training and assessment, including 80:20, 70:30, and 40:60. When undersampling and oversampling were compared, DenseNet169's undersampling technique produced accuracy of 91.2% with a f1-measure of 91.7% and Resnet50's oversampling technique produced accuracy of 83% with a f1-measure of 84%. The future extension of this study includes increasing forecast accuracy through parameter tuning.

REFERENCES

[1] Y. C. Lee, S.-H. Jung, and H.-H. Won, "WonDerM: Skin lesion classification with fine-tuned neural networks," 2018, *arXiv:1808.03426*.

[2] U. Jamil, M. U. Akram, S. Khalid, S. Abbas, and K. Saleem, "Computer based melanocytic and nevus image enhancement and segmentation," *BioMed Res. Int.*, vol. 2016, pp. 1–13, Jan. 2016.

[3] A. Mahbod, G. Schaefer, C. Wang, R. Ecker, and I. Ellinge, "Skin lesion classification using hybrid deep neural networks," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, May 2019, pp. 1229–1233.

[4] K. Pai and A. Giridharan, "Convolutional neural networks for classifying skin lesions," in *Proc. TENCON IEEE Region 10 Conf. (TENCON)*, Oct. 2019, pp. 1794–1796.

[5] A. S. Shete, A. S. Rane, P. S. Gaikwad, and M. H. Patil, "Detection of skin cancer using CNN algorithm," *Int. J.*, vol. 6, no. 5, pp. 1–4, 2021.

[6] M. Vidya and M. V. Karki, "Skin cancer detection using machine learning techniques," in *Proc. IEEE Int. Conf. Electron., Comput. Commun. Technol. (CONECCT)*, Jul. 2020, pp. 1–5.

[7] H. Nahata and S. P. Singh, "Deep learning solutions for skin cancer detection and diagnosis," in *Machine Learning with Health Care Perspective*. Cham, Switzerland: Springer, 2020, pp. 159–182.

[8] P. Wighton, T. K. Lee, H. Lui, D. I. McLean, and M. S. Atkins, "Generalizing common tasks in automated skin lesion diagnosis," *IEEE Trans. Inf. Technol. Biomed.*, vol. 15, no. 4, pp. 622–629, Jul. 2011.

[9] J. Saeed and S. Zeebaree, "Skin lesion classification based on deep convolutional neural networks architectures," *J. Appl. Sci. Technol. Trends*, vol. 2, no. 1, pp. 41–51, Mar. 2021.

[10] Y. Li, A. Esteva, B. Kuprel, R. Novoa, J. Ko, and S. Thrun, "Skin cancer detection and tracking using data synthesis and deep learning," 2016, *arXiv:1612.01074*.

[11] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional encoder–decoder architecture for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017.

[12] P. Tschandl, C. Rosendahl, and H. Kittler, "The HAM10000 dataset, a large collection of multi-source dermatoscopic images of common pigmented skin lesions," *Sci. Data*, vol. 5, no. 1, pp. 1–9, Aug. 2018.

[13] K. M. Hosny, M. A. Kassem, and M. M. Foaud, "Skin cancer classification using deep learning and transfer learning," in *Proc. 9th Cairo Int. Biomed. Eng. Conf. (CIBEC)*, Dec. 2018, pp. 90–93.

[14] A. Javaid, M. Sadiq, and F. Akram, "Skin cancer classification using image processing and machine learning," in *Proc. Int. Bhurban Conf. Appl. Sci. Technol. (IBCAST)*, Jan. 2021, pp. 439–444.

[15] R. Ashraf, I. Kiran, T. Mahmood, A. U. R. Butt, N. Razzaq, and Z. Farooq, "An efficient technique for skin cancer classification using deep learning," in *Proc. IEEE 23rd Int. Multitopic Conf. (INMIC)*, Nov. 2020, pp. 1–5.
[16] M. Uckuner and H. Erol, "A new deep learning model for skin cancer classification," in *Proc. 6th Int. Conf. Comput. Sci. Eng. (UBMK)*, Sep. 2021, pp. 27–31.
[17] Y. Filali, H. E. Khoukhi, M. A. Sabri, and A. Aarab, "Analysis and classification of skin cancer based on deep learning approach," in *Proc. Int. Conf. Intell. Syst. Comput. Vis. (ISCV)*, May 2022, pp. 1–6.
[18] D. Yousra, A. B. Abdelhakim, and B. A. Mohamed, "Transfer learning for automated melanoma classification system: Data augmentation," in *Proc. Int. Conf. Smart City Appl.* Cham, Switzerland: Springer, Mar. 2023, pp. 311–326.
[19] T. Mazhar, I. Haq, A. Ditta, S. A. H. Mohsan, F. Rehman, I. Zafar, J. A. Gansau, and L. P. W. Goh, "The role of machine learning and deep learning approaches for the detection of skin cancer," *Healthcare*, vol. 11, no. 3, p. 415, Feb. 2023.

H. L. GURURAJ (Senior Member, IEEE) received the Ph.D. degree in computer science and engineering from Visvesvaraya Technological University, Belagavi, in 2019. He has 13 years of teaching and research experience with UG and PG levels. He has published more than 75 research articles in peer-reviewed and reputed international journals. He has authored one book on network simulators and one edited book on blockchain by Springer Publishers. He has presented more than 30 papers at various international conferences. He is a Senior Member of ACM. He received a Young Scientist International Travel Support ITS-SERB, Department of Science and Technology, Government of India, in December 2016. He was a recipient of best paper awards at various national and international conferences. He was appointed as an ACM Distinguish Speaker (2018–2021) by the ACM U.S. Council, and he is one among 15 speakers across India. He was a reviewer for various journals and conferences. He has delivered more than 100 technical talks across the globe. He has been honored as the session chair, a keynote speaker, a TPC member, and an advisory committee member at national and international seminars, workshops, and conferences across the globe.

N. MANJU received the B.E. degree in computer science and engineering from the Sri Jayachamarajendra College of Engineering, Mysuru, the M.Tech. degree in computer network engineering from NIE, Mysuru, and the Ph.D. degree from Visvesvaraya Technological University. He is currently an Associate Professor with the Department of Information Science and Engineering, JSS Science and Technology University, Mysuru. He has published papers in peer-reviewed international journals and international conferences. His research interests include machine learning, computational intelligence, and computer networks. He is a member of BOE and BOS, and a Life Member of ISTE. He was a technical program committee member and the session chair at international conferences. He has organized many workshops and FDPs. He is a reviewer of international journals and international conferences.

A. NAGARJUN received the B.E. degree in computer science and engineering from the Sri Jayachamarajendra College of Engineering, Mysuru, and the M.Tech. degree in data science from the JSS Science and Technology University, Mysuru. He was a Project Intern with Vigyanlabs, Mysuru. He is currently an Assistant Professor with the Department of Information Science and Engineering, JSS Science and Technology University. He has published papers in international conferences. His research interests include machine learning, artificial intelligence, and image processing.

V. N. MANJUNATH ARADHYA received the M.S. and Ph.D. degrees in computer science from the University of Mysore, Mysuru, India, in 2004 and 2007, respectively. He was a Postdoctoral Research with the University of Genoa, Italy, from 2009 to 2010. He is currently a Professor with the Department of Computer Applications, JSS Science and Technology University, India. He is guiding four Ph.D. students and completed 12 Ph.D. students. He has published more than 180 research papers in reputed international journals, conference proceedings, and edited books. His research interests include pattern recognition, image processing, the Internet of Things (IoT), document image analysis, computer vision, machine intelligence, the applications of linear algebra for the solution of engineering problems, the bi-clustering of gene expression data, and web data analysis and understanding. He was a recipient of the Young Indian Research Scientist Award from the Italian Ministry of Education, University and Research, Italy, in 2009. He also received the prestigious Award for Research Publications (ARP) from the Government of Karnataka, in 2017.

FRANCESCO FLAMMINI (Senior Member, IEEE) received the master's (Cum Laude) and Ph.D. degrees (Hons.) in computer engineering from the University of Naples Federico II, Italy, in 2003 and 2006, respectively. He is currently a Professor of trustworthy autonomous systems at the University of Applied Sciences and Arts of Southern Switzerland, where he is affiliated with Dalle Molle Institute for Artificial Intelligence (IDSIA). He has worked for 15 years in industry (Ansaldo STS, IPZS), in verification and validation, cyber-security, and innovation management. Since 2018, he has been working as an Associate Professor at Linnaeus University, Sweden. Since 2020, he has been a Full Professor of computer science with a focus on cyber-physical systems at Mälardalen University, Sweden. He is an ACM Distinguished Speaker, an IEEE Distinguished Visitor, a member-at-large of the IEE SMC BoG, and the Chair of the IEEE SMC TC on Homeland Security.