IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 11 July 2022, accepted 18 July 2022, date of publication 8 August 2022, date of current version 12 August 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3197189

RESEARCH ARTICLE

A Convolutional Neural Network-Based Framework for Classification of Protein Localization Using Confocal Microscopy Images

SONAM AGGARWAL[1], SHEIFALI GUPTA[1], RAMANI KANNAN[2], (Senior Member, IEEE), RAKESH AHUJA[1], DEEPALI GUPTA[1], SAPNA JUNEJA[3], AND SAMIR BRAHIM BELHAOUARI[4], (Senior Member, IEEE)

[1]Chitkara University Institute of Engineering and Technology, Chitkara University, Rajpura, Punjab 140401, India
[2]Department of Electrical and Electronics Engineering, Universiti Teknologi PETRONAS, Seri Iskandar 32610, Malaysia
[3]KIET Group of Institutions, Delhi NCR, Ghaziabad 201206, India
[4]Division of Information and Computing Technology, College of Science and Engineering, Hamad Bin Khalifa University, Doha, Qatar

Corresponding authors: Samir Brahim Belhaouari (sbelhaouari@hbku.edu.qa) and Sapna Juneja (sapnajuneja1983@gmail.com)

This work was supported by the Qatar National Library.

ABSTRACT Understanding protein subcellular localization is vital and indispensable in proteomics research. Molecular biology and computer science developments have enabled the use of computational approaches to identify proteins in cells. An excellent method for locating proteins is confocal microscopy, used by the Human Protein Atlas (HPA). By categorizing human proteins, it can assist researchers in better comprehending human pathophysiology and assist doctors in automating medical image interpretation. Human protein Atlas comprises millions of images annotated with single or multiple labels. However, only a few methods for automated prediction of protein localization have been developed, and they mostly concentrate on single-label classification. Therefore, a recognition system for multi-label classification of HPA with acceptable performance should be developed. Hence, this study aims to develop a deep learning-based system for the multi-label classification of HPA. Specifically, two architectures have been proposed in this work for automatically extracting features from the images and predicting the localization of the proteins in 28 subcellular compartments. First, a convolutional neural network has been proposed, which has been trained from scratch and second an ensemble-based model using transfer learning architectures has been proposed. The results demonstrate that both models are effective in classifying proteins according to their location in the major cellular organelles. Yet, in this study, the proposed convolutional network outperforms the ensemble model in classification of images with multiple simultaneous protein localizations. Three performance metrics standards—recall, accuracy, and f1-score—were used to assess the models. The proposed convolutional neural network beats the ensemble model by achieving recall of 0.75, precision of 0.75 and f1-score of 0.74.

INDEX TERMS Biomedical image analysis, convolutional neural network, deep learning, human protein atlas image classification, protein subcellular localization prediction.

I. INTRODUCTION

Proteins are the primary building block of cells, accounting for most of the cell's dry mass. Within the cell, proteins are segregated into various subcellular compartments. Each compartment has a different physiochemical environment required for the proteins to function correctly. Protein synthesis occurs in one of the subcellular compartments called the cytoplasm, and freshly generated proteins are transferred to their appropriate compartment to carry out their function.

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

Thus, proteins should be transported to the correct subcellular compartment for its proper functioning [1]. The mislocalization of proteins results in functional loss or disruption of cells, contributing to various diseases [2], including cardiovascular, neurodegenerative [3], [4], [5], and cancer [6], [7]. Thus, identifying protein localization within cellular compartments is often essential in proteomics research.

The Human Protein Atlas (HPA) initiative is currently dedicated to annotating the location of every human protein within a cell utilizing a diverse set of biotechnologies and techniques [8]. HPA captures images using fluorescence-based microscopy techniques. They are the most extensively utilized and recognized tool for predicting the protein's cellular location. [9]. These methods improve the visibility of intracellular proteins, either by producing fluorescent fusion proteins or by detecting target proteins with fluorophore-detected antibodies. Recent advancements have made it possible to visualize all human proteins in cells in a systematic and high-throughput manner. However, analyzing and visualizing such a vast number of protein localizations currently necessitates the development of more robust, high-throughput approaches. Regardless of the advantages of the present studies, there is still a need for more significant research in identifying protein localization in specific cell types and states and how localization varies over time and across disease states. Furthermore, the amount of protein localization imaging data to be collected in future research is expected to rise considerably. Thus, high-throughput analysis methods for automatically categorizing protein locations in microscopic images are required.

Deep Learning breakthroughs have enabled a slew of successful real-world applications. These includes image identification [10], gamification [11], and autonomous cars [12]. Deep neural architectures, specifically Convolutional Neural Network (CNNs), have already been commonly used in classification of images [13], [14], [15] and segmentation applications [16], [17], [18]. CNN frequently starts with raw images and learns end-to-end hierarchical feature representations, allowing the model to extract cellular localization patterns effectively. Several cutting-edge CNN networks with outstanding performance have been developed, including VGG16 [19], Resnet [20], Xception [21], ShuffleNet [22], MobileNet [23], and others. The effectiveness of deep learning techniques has also been proved useful in various public competitions that rely on crowdsourcing to find successful computational solutions for specific objectives [24], [25], [26]. The phenomenal achievements in deep learning have resulted in making the conventional feature-based machine learning technique practically obsolete in huge data requiring application. Deep learning approaches are likely to perform well in protein localization since imaging data, together with annotation from the HPA project, provides a rich source of training data.

Protein localization through microscopic images presents a unique machine learning challenge, specifically how to deal with weakly annotated data. The issue is that rather than labelling individual instances (in this case cells), a group of instances have been labelled. Each of those instances may shed light on the proper classification. This contrasts with the object recognition issue settings in MNIST, CIFAR, and ImageNet, where an image often clearly represents a class. The ideal solution to protein localization would be a general, resilient, and fully automated workflow that is as accurate as human experts, if not more so. Even more ideal would be a platform for uploading and automatically annotating high throughput images. Whereas the state-of-the-art in protein localization entails professionals fine-tuning the segmentation algorithm, cell crop extraction, and then applying Deep Learning to those with allegedly inferior performance to human experts. In this research, a general approach has been developed that accepts inputs of any size, does not require segmentation, but instead works on complete images, and performs at the level of human specialists or even higher. We hypothesize that by increasing the amount and diversity of training sets, the suggested architectures might be utilized to localize proteins in images from various biotechnological devices, cell lines, and laboratories. Therefore, this research helps understand an automated classification approach for the multi-label HPA problem. Following are the contributions of the author in the study:

i. A Convolutional Neural Network-based architecture was proposed and trained from scratch to classify protein's mixed patterns in subcellular compartments.
ii. An ensembled learning-based model was also proposed to compare the performance of proposed CNN with ensemble model.
iii. The proposed CNN architecture was rigorously tested with different input shapes, filter sizes and different hyperparameters.
iv. The proposed model's performance was evaluated using precision, recall and f1-score and compared with the proposed ensemble model and the state-of-art.

The remainder of the paper will be delivered in the following manner: Section 2 emphasized the contributions of various researchers in the prediction of protein subcellular location. In section 3, the dataset used for this study has been described, followed by proposed model's architecture and the experimental setup used for training the proposed CNN model. The results of the experiments performed on the suggested model are detailed in Section 4. Lastly, in section 5, limitations of the work and future work have been discussed.

II. RELATED WORK

The researchers have employed various approaches using machine and deep learning techniques for identification of subcellular location and patterns of proteins within human cells. Several machine learning-based methods integrating microscopy with pattern recognition have been created over the past decade to analyze proteins' location in cultured cells [27], [28]. These methods extract the subcellular location features from the microscopic images, which are then analyzed to find patterns [29]. In addition to morphological aspects,

SLFs include features like Haralick, Wavelet, and Zernike features, which quantitatively describes the subcellular location of proteins. Thus, it is possible to utilize SLFs to train classifiers to discriminate distinct protein patterns once they have been retrieved from the images.

Although machine learning approaches have effectively addressed the protein localization problem, extracting discriminant features from images is time-consuming. Deep learning eliminates the need for feature extraction by allowing the system to learn visual features independently. Recently, CNN-based approaches for categorizing protein subcellular distribution have been successfully applied.

Sullivan *et al.* [30] utilized a neural network to recognize and classify the protein patterns in microscopic imaged of Hela cells. Huang and Murphy [31] introduced a classifier based on SVM and ensemble method for recognizing protein in 2-dimensional and 3-dimensional microscopic images. Newberg *et al.* [32] proposed two new classifiers based on SVM and random forest, which significantly improved the accuracy of detecting subcellular locations of proteins. Two new datasets were created from microscopic images by Coelho *et al.* [33] to automate the prediction of protein's subcellular location through CD-tagging. Additionally, they devised a novel technique integrating K-means and SVM, which helped in better feature extraction and classification. Lu *et al.* developed a successful self-supervised technique for learning image representations in individual cells from microscope images, attaining a 55% accuracy [34]. Liimatainen *et al.* [35] examined two widely used deep learning-based techniques, CNN and FCN (Fully Connected Network), for identifying the protein locations in 13 subcellular compartments using microscopic images. FCN outperformed CNN, achieving an F1-score of 0.696 to CNN's 0.676. Li *et al.* [36] suggested a model with a macro F1-score of 0.706 that was built using pre-trained model Inception V3. Shwetha *et al.* [37] used two distinct strategies to classify images from Human Protein Atlas database. The first strategy entails feature extraction and classification using the Random Forests classifier. The second approach utilized two distinct CNN-based architectures – Xception and ResNet 50 – to extract features and classify them into 15 distinct categories. The Hybrid Xception model achieved a higher F1-score of 0.69 than with the standard approach's 0.61. Pärnamaa and Parts [38] classified microscopic fluorescent images collected from the HPA dataset using a combination of two approaches. First, they hosted an online video game image classification competition, resulting in 33 million protein location annotations. Then an automated program called Loc-CAT was developed that utilized the results obtained from online video game competition to classify the protein locations in 29 sub-compartments. Both techniques were then integrated using transfer learning to produce a model with an F1 score of 0.72 for classifying protein patterns. Kraus *et al.* [39] developed an 11-layer CNN (DeepLoc) to evaluate protein localization of yeast protein, reaching a prediction accuracy of 72.3% for proteins located in ten subcellular sites inside the cell.

While the approaches developed for automatic determination of the subcellular protein localization using microscopic images is accurate, present techniques struggle to match the same level as humans. This work investigates and addresses the problem of predicting the localization of proteins in human subcellular compartments. In this work, images acquired from HPA have been classified in 28 distinct cell phenotypes for the presence of protein. In addition, a fast and accurate approach for classifying human protein labels is described without the need for tedious feature selection and extraction.

III. MATERIALS AND METHODS

A. DATASET DESCRIPTION

The database utilized in this study was obtained through a crowdsourcing competition, titled 'Human Protein Atlas Image Classification', held on Kaggle in 2019 [41]. This dataset was acquired using a highly uniform imaging technique known as confocal microscopy. There are 31,072 samples in the database. Each sample contains 4 high-resolution images marked with a different fluorescent protein, including the targeted protein (green) and 3 cellular landmarks (red, yellow, and blue), as illustrated in Figure 1. In this study, three channels (red, blue, and green) were used. Due to diverse range of cell forms, types, spatial relationships in the data set, identifying proteins in specific cellular compartments becomes difficult.

The protein distribution in cells is broken down into 28 major organelles in this dataset. Hence there are 28 different labels, and each image may belong to a single label or multiple labels. Therefore, this is a multi-label classification problem, and proteins may localize in more than one subcellar compartment or organelles in a single sample image. The name of the labels are given in Table 1. Dataset was prepared by combining the three channels, red, green, and blue, for each sample. Sample images from the dataset belonging to single or multiple labels are shown in Figure 2. Data was split into 3 parts: Train (80%), Validation (10%) and Test (10%). Distribution of number of samples for each label in training, test and validation set is given in Table 1.

B. DATASET PRE-PROCESSING

Resizing and normalization of images have been done in data pre-processing. Original images of size 512 $\times$ 512 pixels were fed as an input to the proposed CNN model. Images were also resized to128 $\times$ 128 pixels and 256 $\times$ 256 pixels to test the model's performance with varied input shapes of the image. The dataset also had the version of images with resolution 2048 $\times$ 2048 and 3072 $\times$ 3072 pixels, but due to the massive size of the dataset and the resulting memory shortage, images of size 512 $\times$ 512 pixels were used. For ensemble learning model, the images were resized to 224 $\times$ 224 pixels

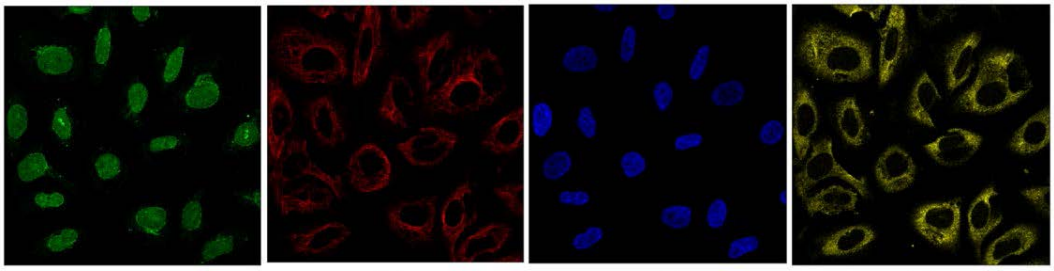

FIGURE 1. 4 filters of an image sample: (a) Green channel (Protein of interest) (b) Red channel (Microtubules) (c) Blue channel (Nucleus) (d) Yellow channel. (Endoplasmic reticulum.)

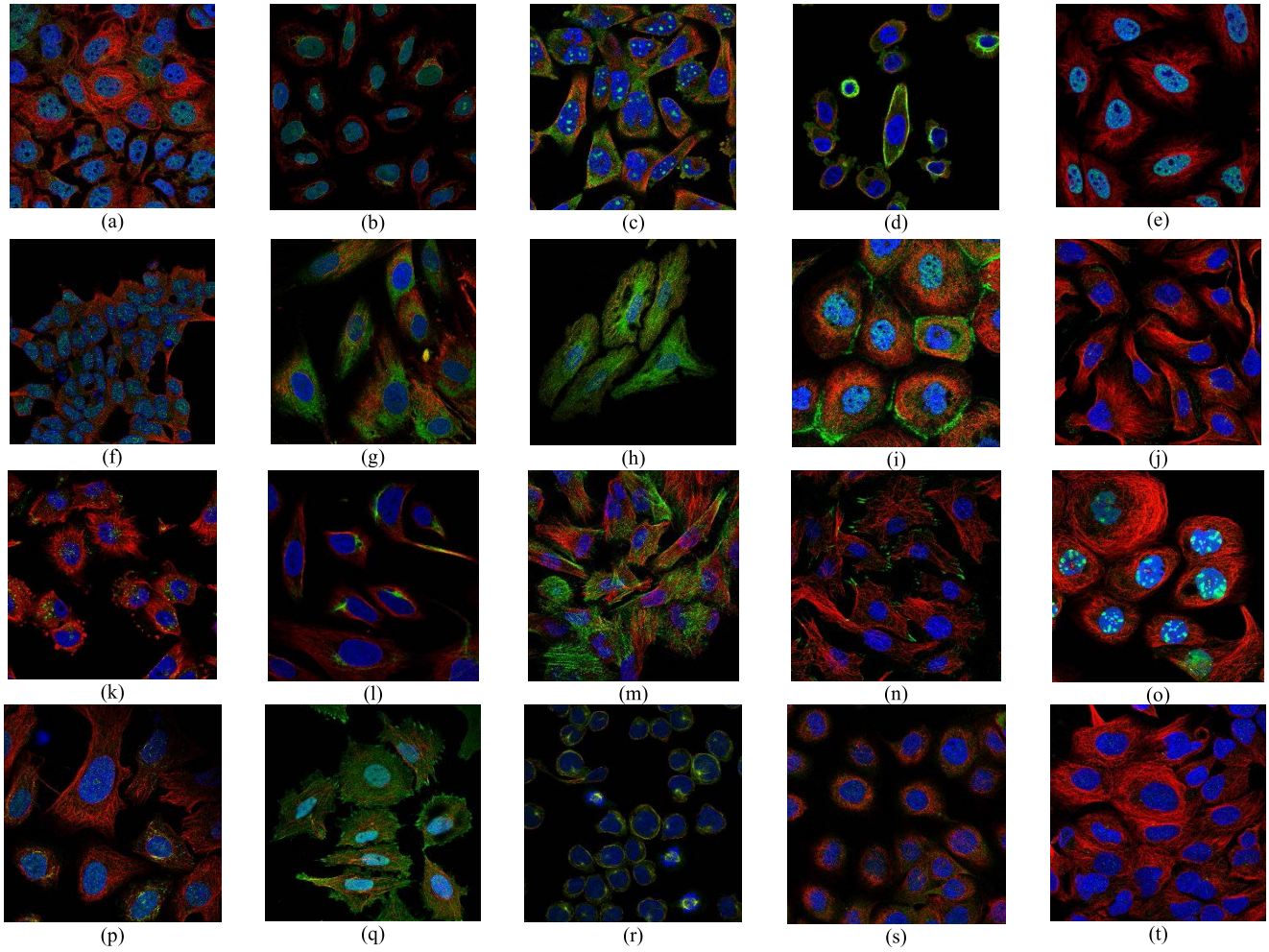

FIGURE 2. Sample images from the dataset belonging to single or multiple labels [41]: (a) Nucleoplasm, (b) Golgi Apparatus, Nucleoplasm, Nuclear membrane, Nucleoli, (c) Cytosol, Nucleoli, (d) Intermediate Filaments, Cytosol, (e) Nuclear Speckles, (f) Nuclear Bodies, (g) Endoplasmic reticulum, (h) Cytokinetic bridge, Endoplasmic reticulum, (i) Cell Junctions, Nucleoplasm, (j) Peroxisomes, Endosomes (k) Lysosomes, (l) Intermediate filaments, (m) Actin filaments, (n) Focal adhesion sites, (o) Nucleoli, Nucleoplasm, Centrosome (p) Microtubule End, (q) Nucleoplasm, Plasma Membrane, Focal adhesion sites, (r) Mitotic Spindle, (s) Cytosol, Microtubule organizing center, (t) Centrosome.

since the ensemble consist of pre-trained architectures whose maximum acceptable input image size is 224 × 224 pixlels.

The pixel values of the image were rescaled from 0 to 1 using normalization. Data normalization is an important step that assures that each input parameter (or pixel) has a consistent data distribution. As a result, the network gets trained faster and more efficiently. Each pixel value was normalized by multiplying it by 1/255.

C. DATA AUGMENTATION

Data augmentation is the technique used to increase the size of the dataset. Here, increasing the size of the dataset does not mean increasing the number of samples but to diversify the images to make our model more generalizable. Generalizability means the mismatch tween the output of the model based on previously seen data and unseen data. In data augmentation, there is a wide range of transformations that

TABLE 1. Distribution of samples for each label in training, test and validation set.

Label No.	Label Name	Total	Train	Validation	Test
0	Nucleoplasm	12,885	10,436	1160	1289
1	Nuclear Membrane	1,254	1007	112	135
2	Nucleoli	3,621	2913	324	384
3	Nucleoli Fibrillar Centre	1,561	1,275	142	144
4	Nuclear Speckles	1,858	1848	205	195
5	Nuclear Bodies	2,513	2034	226	253
6	Endoplasmic Reticulum	1,008	823	91	94
7	Golgi Apparatus	2,822	2,282	253	287
8	Peroxisomes	53	42	5	6
9	Endosomes	45	36	4	5
10	Lysosomes	28	22	3	3
11	Intermediate Filaments	1,093	892	99	102
12	Actin Filaments	688	562	63	63
13	Focal Adhesion Sites	537	442	49	46
14	Microtubules	1,066	856	95	115
15	Microtubule End	21	17	2	2
16	Cytokinetic Bridge	530	423	47	60
17	Mitotic Spindle	210	165	18	27
18	Microtubule Organizing Centre	902	721	80	101
19	Centrosome	1,482	1,196	133	153
20	Lipid Droplets	172	140	16	16
21	Plasma Membrane	3,777	3,073	341	363
22	Cell Junctions	802	658	73	71
23	Mitochondria	2,965	2,396	266	303
24	Aggresome	322	256	28	38
25	Cytosol	8,228	6636	737	855
26	Cytoplasmic Bodies	328	265	29	34
27	Rods and Rings	11	9	1	1

can be applied to the original images to create additional training samples while maintaining their original labels. The transformations applied to the images in this research include random rotation of the image in the range of 30 to 60 degrees, zoom in the range of 0.2 and vertical flip and horizontal flip. The sample images obtained after applying these transformations are shown in Figure 3.

D. THE PROPOSED CNN ARCHITECTURE

This research proposes an extensive Convolutional Neural Network with outstanding recognition performance for HPA images. The block diagram of the developed model is presented in Figure 4. The proposed model's details and its related configurations are summarized in Table 2.

The base architecture of proposed CNN model consists of 4 convolution blocks. Each block is created using a series of convolutional layers followed by a max-pooling layer. The network's depth was determined by experimenting with depths ranging from four to twelve convolutional layers. Following each convolution block, the number of filters was doubled. For instance, the first convolution block contains four convolution layers, each with 32 filters; the second block contains 64 convolution layers; the third convolution block contains 128 filters, and the fourth convolution block contains 256 filters.

Additionally, normalization of batch was performed between each layer of convolutional and activation. It is a technique which improves the neural network's performance by normalizing the inputs in each layer such that the mean activation at the output is 0 and deviation is 1.

Following the final convolutional block, a fully connected network was constructed consisting of two Dense Layers, the first with 128 nodes and the second with 28 nodes predicting the output values. Each convolution layer was activated with rectified linear unit (ReLU) function, except last layer, which used sigmoid activation to get the output between 0 and 1.

E. THE PROPOSED ENSEMBLE LEARNING MODEL

Ensemble learning refers to the process of generating and combining several models, such as classifiers or experts, to address specific computational intelligence challenges. Ensemble learning is typically employed to boost a model's efficiency or lessen the probability of selecting a poor one. One common strategy of ensembling different models is called the stacked ensemble approach. A stacking model's architecture consists of two or more base models, also known as level-0 models, and a meta-model that aggregates the predictions of the base models, also known as a level-1 model. The base models are the models that are fitted to the training data and whose predictions are compiled, while the meta-models are the models that learn how to integrate the base models' predictions in the best way possible. The meta model is trained on the predictions made by the base models on the validation data set. That means, the data samples which have not been used to train the base models are fed to the base models, predictions are made, and these predictions, along with the predicted outputs, form the input and output pairs of the training dataset used to fit the meta-model.

Three transfer learning models, ResNet102 [20], DenseNet201 [43], and VGG19 [19], were utilized as base learners in this study. The chosen transfer learning models

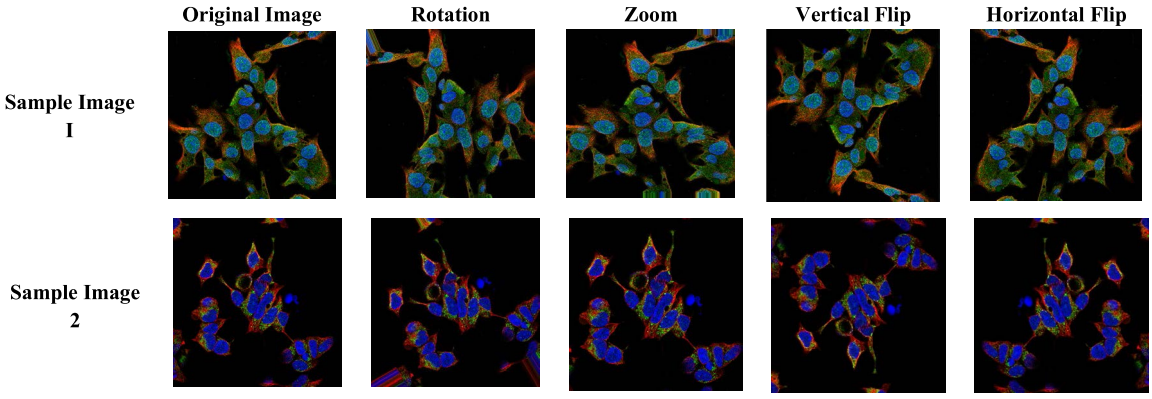

FIGURE 3. Sample images after applying data augmentation transformations.

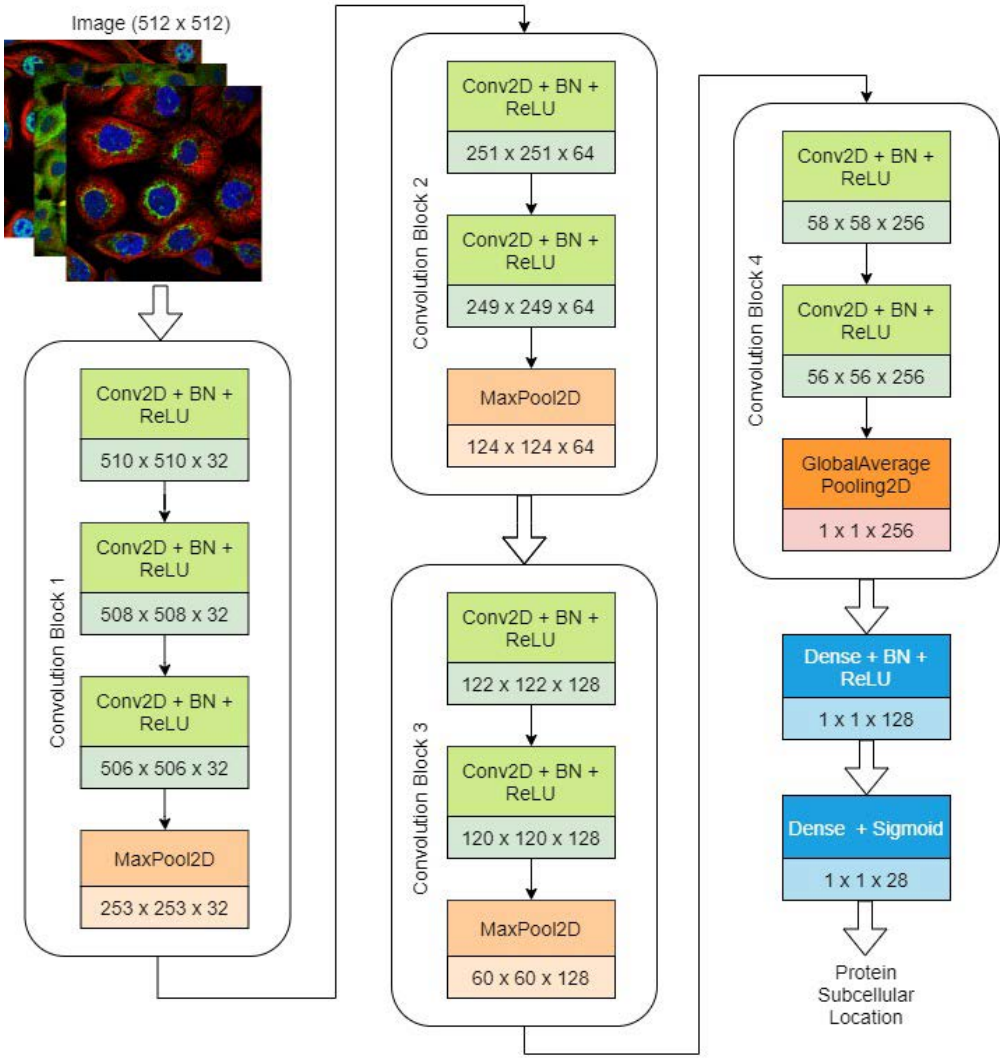

FIGURE 4. Block diagram of proposed CNN architecture.

were fine-tuned and trained separately on the training dataset. Fine-tuning of the models was done by replacing the original fully connected layers with a new set of layers comprising of Global Average Pooling Layer followed by a Dense layer of 1024 neurons. The models were saved and utilized to form a stack ensemble once they were trained.

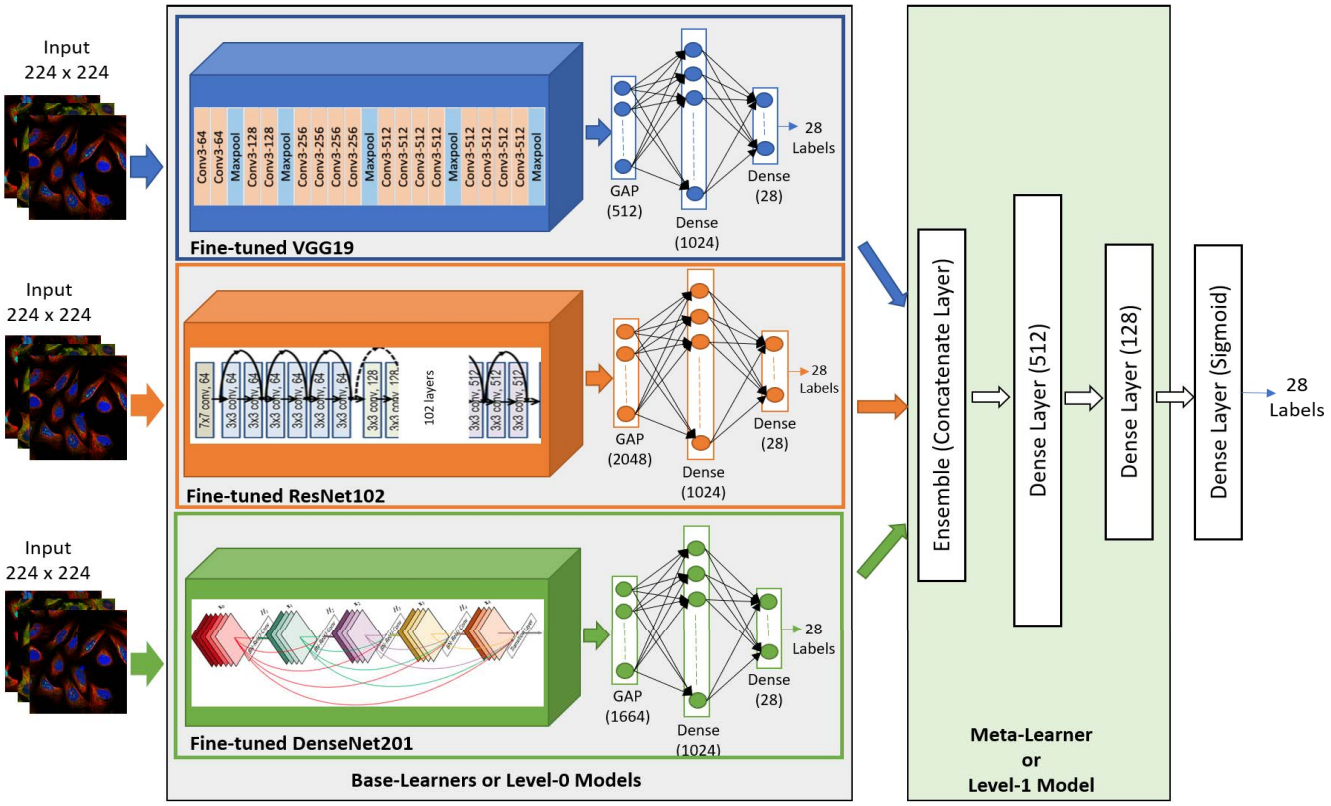

FIGURE 5. Proposed ensemble learning model.

The initial layer of input data is duplicated and distributed across the three base learners. After the images have been delivered through the pipeline of each base learner, a prediction vector for the input data is formed. The prediction vectors of all three base learners are then concatenated and used as a feature in the meta learner. Finally, the meta-learner attempts to better classify the input data. The meta-learner chosen has a significant impact on the ensemble model's performance. In our research, we used a fully connected network as a meta-learner. To reduce generalization error and enhance predictions, the base models were frozen, and only the meta-learner is trained on the predictions made by the base models on the validation dataset. The proposed stacked ensemble model is shown in Figure 6. The fully connected meta-learner is made up of two dense layers with 512 and 128 neurons, respectively, followed by a prediction layer with 28 neurons and sigmoid function for classification.

F. EXPERIMENTAL SETUP

Additional to the architecture of network, it is essential to understand the model's implementation and performance, which includes the intricacies of the training process itself. When training deep neural networks, several parameter choices were made. The network was created from scratch without the use of pre-trained models. Experimentation led to the selection of training hyperparameters. The model was

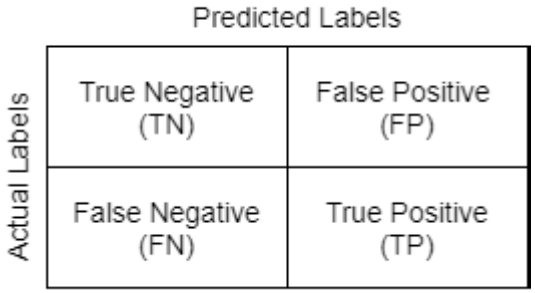

FIGURE 6. Confusion matrix.

run for 30 epochs. Initially, weights of the model were established using Golort initialization [44]. Next, the Adam optimizer was used, with the default parameters provided in the original Adam paper [44]. 0.001 was used as an initial rate of learning. After five consecutive epochs, if the loss did not improve, the learning rate was adjusted to decrease by a factor of 1/10 dynamically. Binary cross-entropy loss function was chosen based on the multi-label structure of the data. Batch size of 32 was selected because it was the maximum batch size that could be implemented with the available GPU memory. A TensorFlow backend was utilized to implement the models in Python with the Keras package [45], and GPUs were employed to speed up training. The sigmoid activation function after the final layer ensures that

TABLE 2. Configuration of training parameters.

Hyperparameters	Values
Mini Batch size	32
Initial Learning Rate	0.001
Weight Decay	10^{-8}
Beta	0.9, 0.999
Adam	True

output values are constantly between zero and one. Table 2 shows the training parameters, as well as the settings for the optimizer. Hyperparameters were chosen by tuning the model by applying different optimizers, batch size and learning rate. Results obtained for hyperparameter tuning are discussed in Section IV.

In case of ensembled approach, hyperparameter tuning was done with similar parameters for the ensembled network. But for the independent base-learners, first the only the new fully connected layers of the model were trained for 15 epochs at a learning rate of 0.001 by keeping the base layers frozen, and then for fine-tuning of the models, the complete model was trained for another 30 epochs at a learning rate of 0.00001.

IV. RESULTS AND DISCUSSION

This section discusses the performance metrics employed to evaluate the model. The results obtained from the various experiments done on the proposed models are presented in detail in this section.

A. PERFORMANCE METRICS

Precision, f1 Score and recall were used from the scikit-learn to assess the model's performance on multi-label classification. Proposed model's performance is quantified using four metrics: true positive = TP, true negative = TN, false-positive = FP, and false negative = FN. These four metrics are obtained using confusion matrix. Confusion matrix is shown in Figure 5.

Three criterions have been provided to evaluate theave been provided to evaluate th proposed model in comparison to existing methods:

Precision: It is a measure that describes true positive predicted labels from all the positive labels.

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

Recall: It is also called sensitivity and it determines the true positive labels out of all the predicted positive labels.

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

F1-Score: F1-score considers both recall and precision and is given by:

$$F1 - Score = \frac{2 * Precision * Recall}{Precision + Recall} \tag{3}$$

In a multi-label classification problem, precision, recall, and F1-score can be obtained for each label independently, also combined results across labels can also be calculated.

Since the dataset used in this study is highly imbalanced, hence it is crucial to consider the number of samples for each label while calculating the above-mentioned performance criterions. Hence weighted average of precision, recall and F1-score across all labels has also been calculated as shown in Figures 6 and 8.

Although the major performance parameters to evaluate the proposed models are precision, recall and f1-score. But further the performance of the proposed models can also be evaluated on other metrics like Specificity, True Positive Rate, False Positive Rate and Receiving Operating Characteristic (ROC) curve with Area under the ROC curve i.e., AUC score.

Specificity is also called True Negative Rate and it gives the proportion of negatives that have been predicted as negatives from the true negatives.

$$Specificity = \frac{TN}{FP + TN} \tag{4}$$

False Positive Rate (FPR) gives the proportion of how many true negatives were predicted as positives. It is given by the formula:

$$FPR = \frac{FP}{FP + TN} \tag{5}$$

Similarly, False Negative Rate (FNR) defines how many true positives have been predicted as negatives. Its is given by the formula:

$$FNR = \frac{FN}{TP + FN} \tag{6}$$

Specificity, FPR and FNR have been calculated using confusion matrix. Another important metric that can be used to evaluate a model is the AUC-ROC curve. ROC curve gives the performance of the model at different thresholds. It is the plot between False Positive Rate (FPR) and True Positive Rate (TPR) also called Sensitivity. The formula for FPR and True Positive Rate (TPR) is given in Equation (5) and (2). Threshold values have been taken from 0 to 1 at an equal interval of 0.1. AUC gives the area under the ROC curve.

B. PERFORMANCE OF PROPOSED ENSEMBLE LEARNING MODEL

The proposed ensemble learning model consists of 3 independent pre-trained models, namely, VGG19, ResNet102 and DenseNet201. Firstly, the performance parameters namely, precision. Recall and f1-score were obtained for these 3 pre-trained models as shown in Figure 7. It can be inferred from Figure 7 that VGG19 performed best among all the three models with an f1-score of 0.65. When the ensemble of three models was made, then it can be seen from Figure 7, that the ensembled model outperformed the three individual pre-trained models by achieving an f1 score of 0.68. Precision, recall and f1-score for each label are given in Table 3 and their weighted average value of precision, recall and f1-score is given in Figure 7. Apart from the precision, recall and f1-score, other parameters like specificity, FPR and FNR have

TABLE 3. Performance metrics of proposed ensemble model for individual labels.

Label No.	Precision	Recall	F1-score	Specificity	FPR	FNR	AUC Score
0	0.81	0.87	0.84	0.85	0.148	0.13	0.93
1	0.77	0.65	0.70	0.99	0.007	0.35	0.91
2	0.68	0.77	0.72	0.95	0.050	0.23	0.93
3	0.46	0.42	0.44	0.98	0.025	0.58	0.87
4	0.80	0.62	0.70	0.99	0.010	0.38	0.91
5	0.62	0.43	0.51	0.98	0.023	0.57	0.84
6	0.47	0.61	0.53	0.98	0.021	0.39	0.92
7	0.66	0.62	0.64	0.97	0.030	0.38	0.90
8	0.03	0.17	0.05	0.99	0.012	0.83	0.89
9	0.04	0.40	0.08	0.99	0.015	0.60	0.95
10	0.25	0.33	0.29	1.00	0.001	0.67	0.99
11	0.77	0.52	0.62	0.99	0.005	0.48	0.89
12	0.71	0.48	0.57	1.00	0.002	0.52	0.91
13	0.71	0.48	0.57	1.00	0.003	0.52	0.92
14	0.85	0.77	0.81	0.99	0.005	0.23	0.96
15	0.00	0.00	0.00	1.00	0.005	1.00	0.61
16	0.08	0.27	0.12	0.94	0.061	0.90	0.71
17	0.17	0.15	0.16	0.99	0.006	0.85	0.83
18	0.41	0.41	0.41	0.98	0.020	0.59	0.82
19	0.45	0.44	0.45	0.97	0.027	0.56	0.84
20	0.68	0.77	0.72	1.00	0.000	0.88	0.87
21	0.58	0.73	0.65	0.93	0.070	0.27	0.91
22	0.51	0.45	0.48	0.99	0.010	0.55	0.85
23	0.72	0.72	0.72	0.97	0.030	0.28	0.94
24	0.74	0.61	0.67	1.00	0.003	0.40	0.89
25	0.63	0.78	0.70	0.83	0.175	0.22	0.88
26	0.32	0.26	0.29	0.99	0.006	0.74	0.79
27	0.00	0.00	0.00	1.00	0.002	1.00	0.99

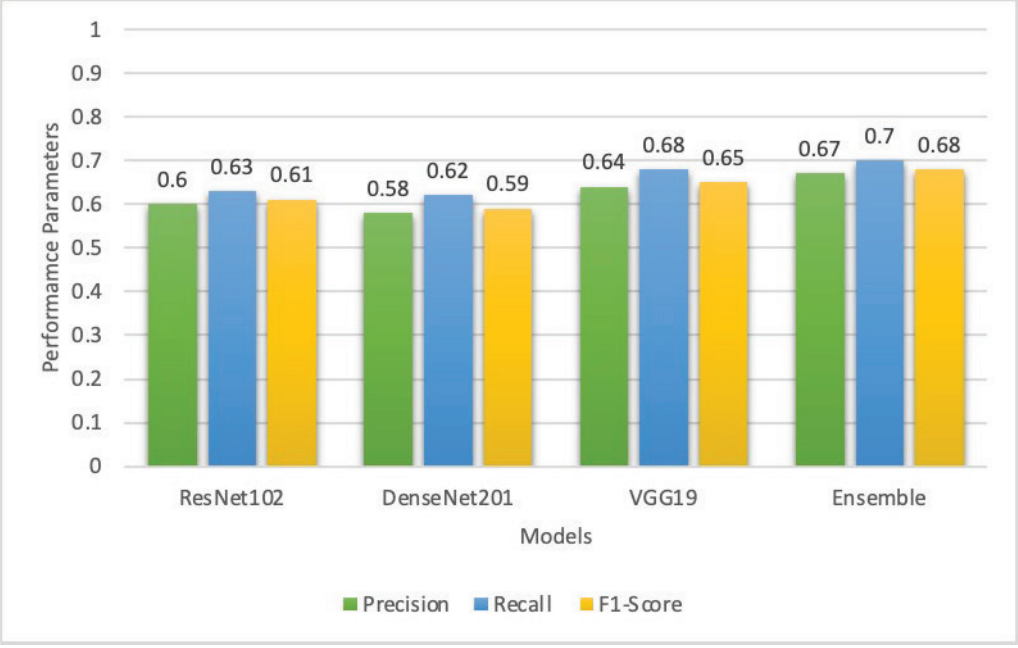

FIGURE 7. Performance of proposed ensemble learning model.

also been calculated using a confusion matrix for each label as shown in Table 3. The confusion matrix obtained for each label for the proposed ensemble learning model is shown in Figure 8. AUC-ROC curve at different thresholds is shown in Figure 9. AUC score obtained for each label is also given in Table 3.

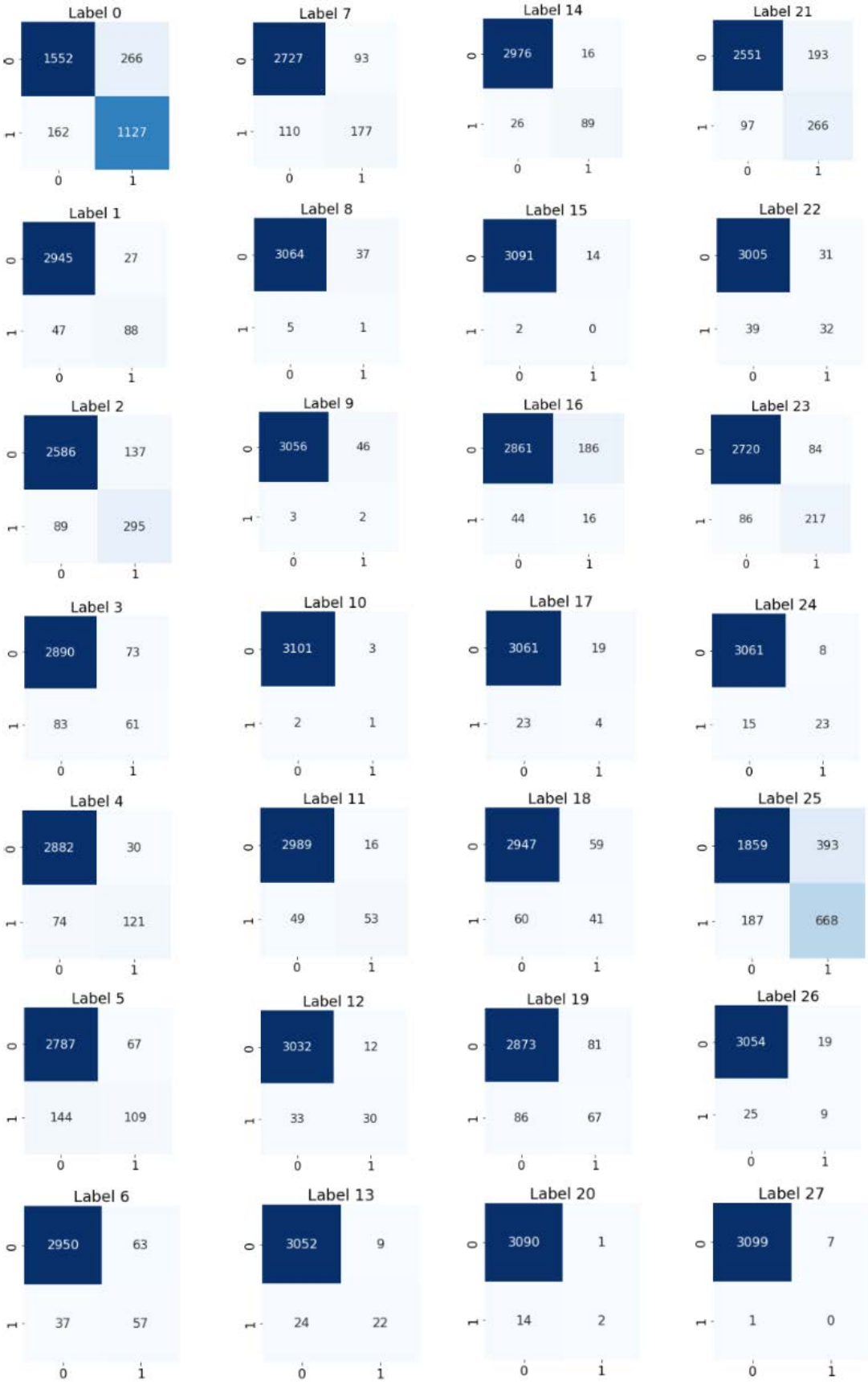

FIGURE 8. Confusion matrix of proposed ensemble learning model.

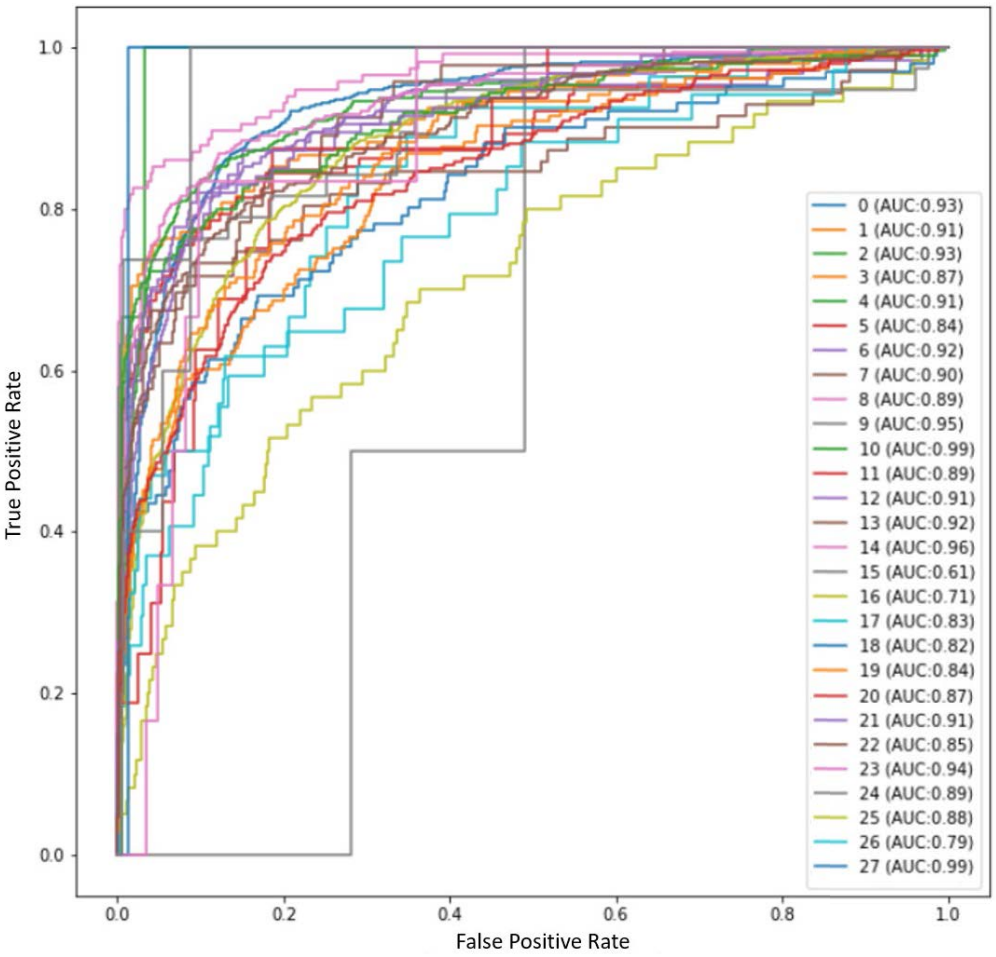

FIGURE 9. ROC curve for the proposed ensemble model.

C. PERFORMANCE OF PROPOSED CNN MODEL

The proposed CNN model was tested rigorously with different input shapes and filter sizes. The hyperparameter tuning was performed to select the set of optimal hyperparameters for the training of the model. The results obtained have been discussed one by one in this section.

1) PERFORMANCE ANALYSIS BASED ON INPUT IMAGE SIZE

A comparative test with three different scales of the input image (512×512, 256×256, and 128×128) was conducted to evaluate the model's performance. Loss and accuracy plots obtained from the performed experiments are presented in Figure 10. Highest training accuracy of 97.7% and validation accuracy of 97% was obtained using an input image of shape 512×512. Decrease in training and validation loss to 0.06 and 0.08, respectively, was observed for the model fed with an input shape of size 512×512 as shown in Figure 10(b) and 10(d).

Average recall, Precision and F1-score obtained for different input image sizes are shown in Figure 11. From Figure 11 it can be inferred that the best results were attained when model was fed with an input image of shape 512×512 with recall, precision, and F1-score of 0.75, 0.75 and 0.74, respectively.

2) PERFORMANCE ANALYSIS BASED ON FILTER SIZE

Experiments with different filter sizes were also conducted to present the proposed CNN model's effectiveness and analyze the model's performance in extracting features from protein atlas images. These experiments were conducted on images of size 512×512. The performance of the model was evaluated based on 3 different sizes of the filters (3×3, 5×5 and 7×7). Figure 13 presents accuracy and loss plots obtained with different filter sizes. Clearly, from Figure 13, it can be inferred that the maximum accuracy and minimum loss for both train and validation sets were obtained with the filter size of 3×3. With increasing filter size, the model's performance decreases.

Average precision, recall and F1-score obtained using different filter sizes has been compared in Figure 13. Figure 13 confirms that the best results were achieved with a 3×3 filter size. With the increase in filter size to 7×7, F1-score reached 0.68 as compared to 0.74 obtained using a 3×3 filter size.

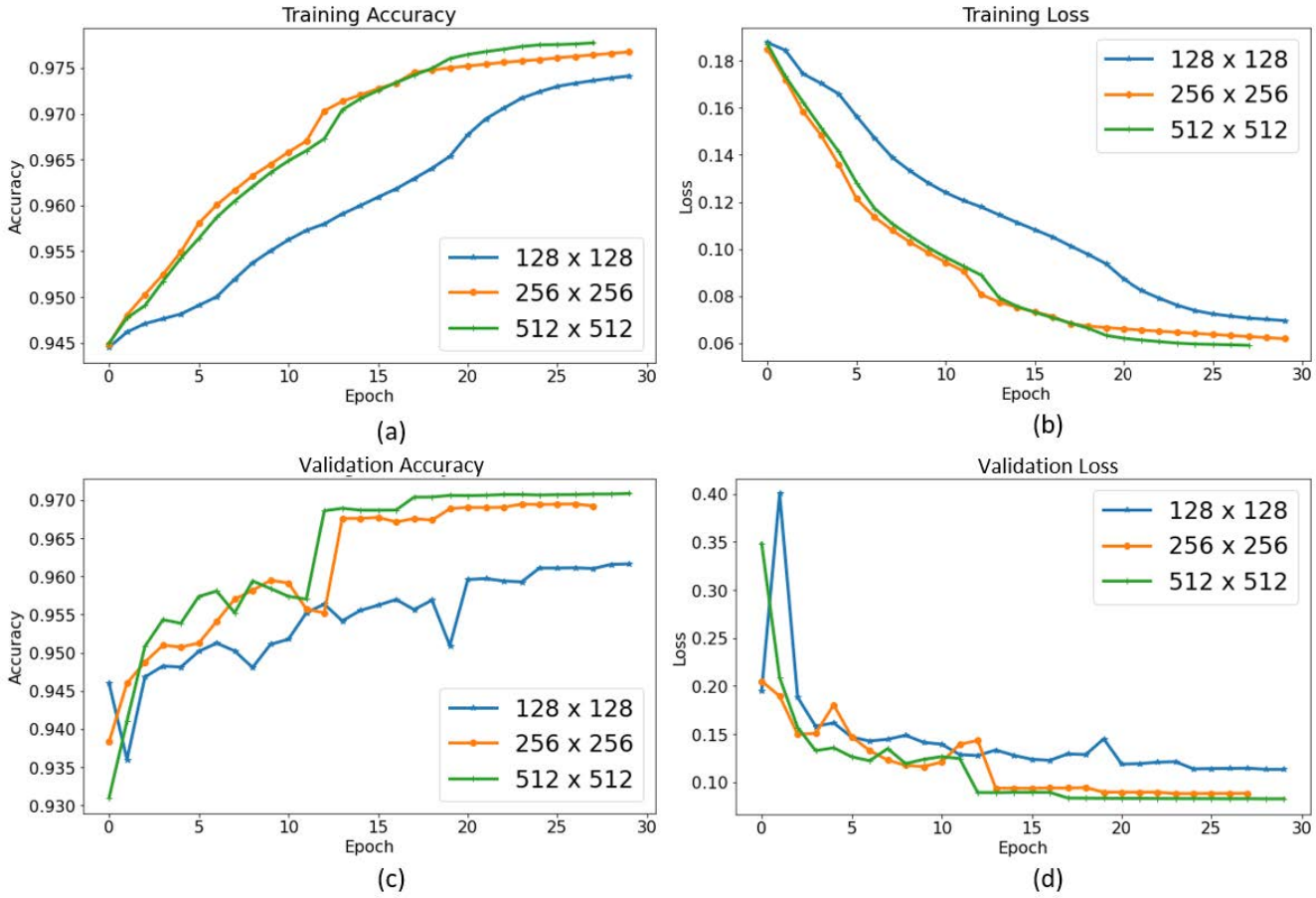

FIGURE 10. Plots of accuracy and loss on the basis of different input image size: (a) Training accuracy vs epochs, (b) Training loss vs epochs, (c) Testing accuracy vs epochs and (d) Testing loss vs epochs.

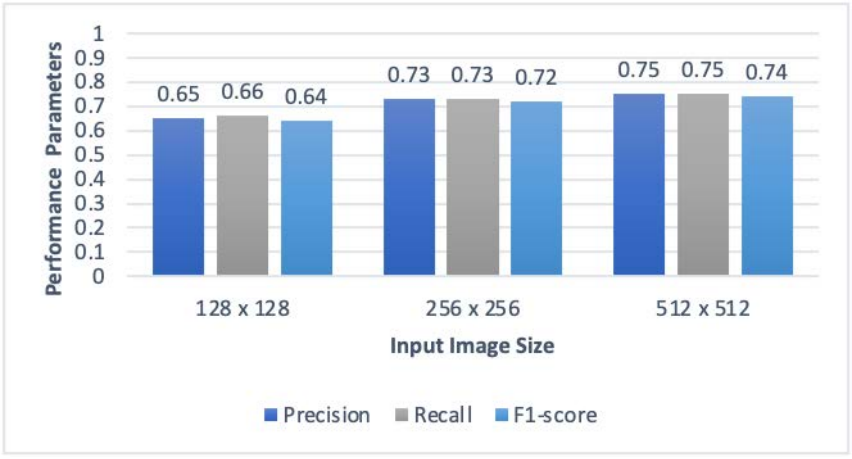

FIGURE 11. Comparison in the performance of the model based on different input image size.

3) PERFORMANCE ANALYSIS WITH DATA AUGMETATION

The performance metrics of the proposed CNN model was also obtained with and without data augmentation. Transformations applied for data augmentation have been shown in Figure 3. The precision, recall and f1-score obtained by the model with and without data augmentation is shown in Figure 14(a). It can be concluded from Figure 14(a) that the proposed CNN model performed better with data augmentation with an f1-score of 0.74.

4) HYPERPARAMETER TUNING

Hyperparameter tuning was done to select the set of optimum parameters for the training of the model. The performance of the model was measured by tuning vatrious hyperparameters

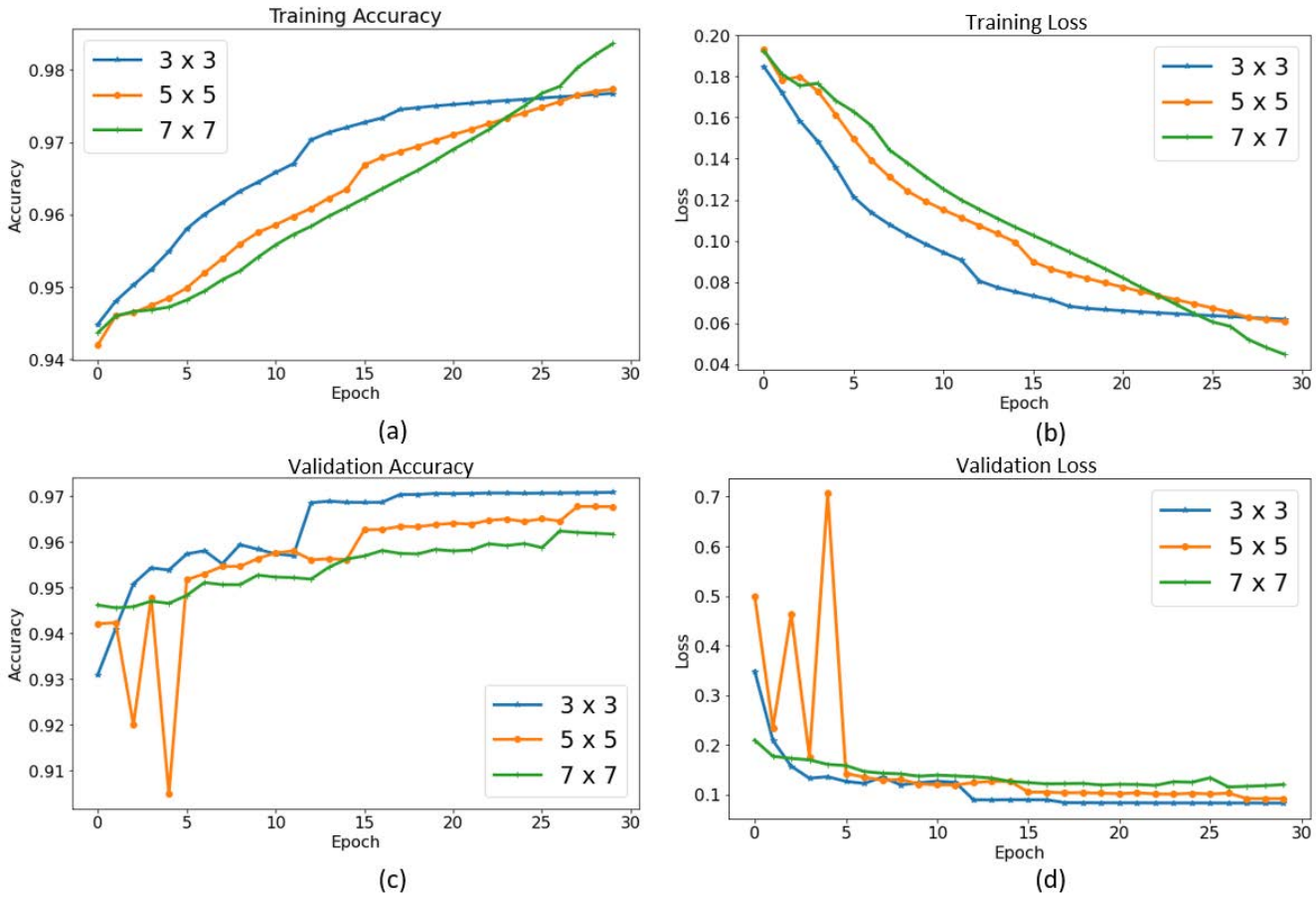

FIGURE 12. Plots of accuracy and loss on the basis of different filter size: (a) Training accuracy vs epochs, (b) Training loss vs epochs, (c) Testing accuracy vs epochs and (d) Testing loss vs epochs.

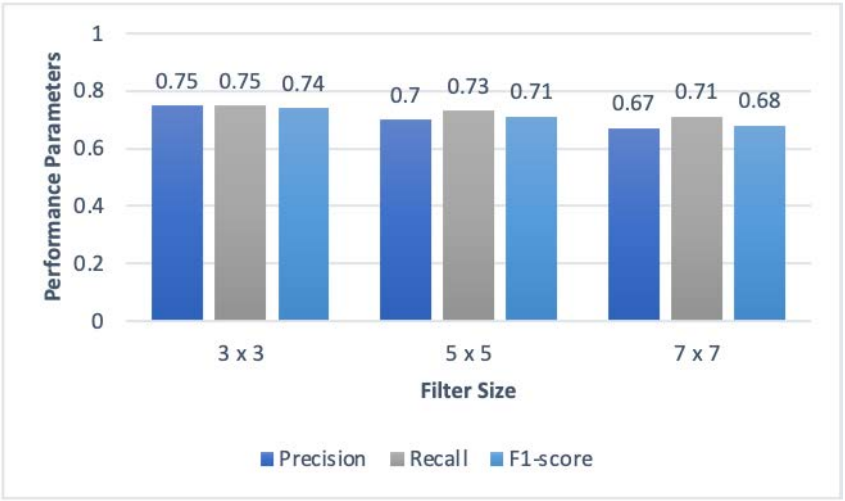

FIGURE 13. Comparison in the performance of the model based on different filter size.

as shown in Figure 14(b)-(c). The hyperparameter tuning was done by varying the following parameters:

(a) Batch Size: Batch size is the number of training samples used in one iteration. Results of the proposed CNN were obtained on batch size of 8, 16 and 32. Batch size couldn't be increased after 32 because of the low memory space. As shown in Figure 14(d), with increase in batchsize the performance of the proposed model increased.

(b) Optimizers: In artificial neural networks, an optimizer is an algorithm that adjusts parameters like weights and

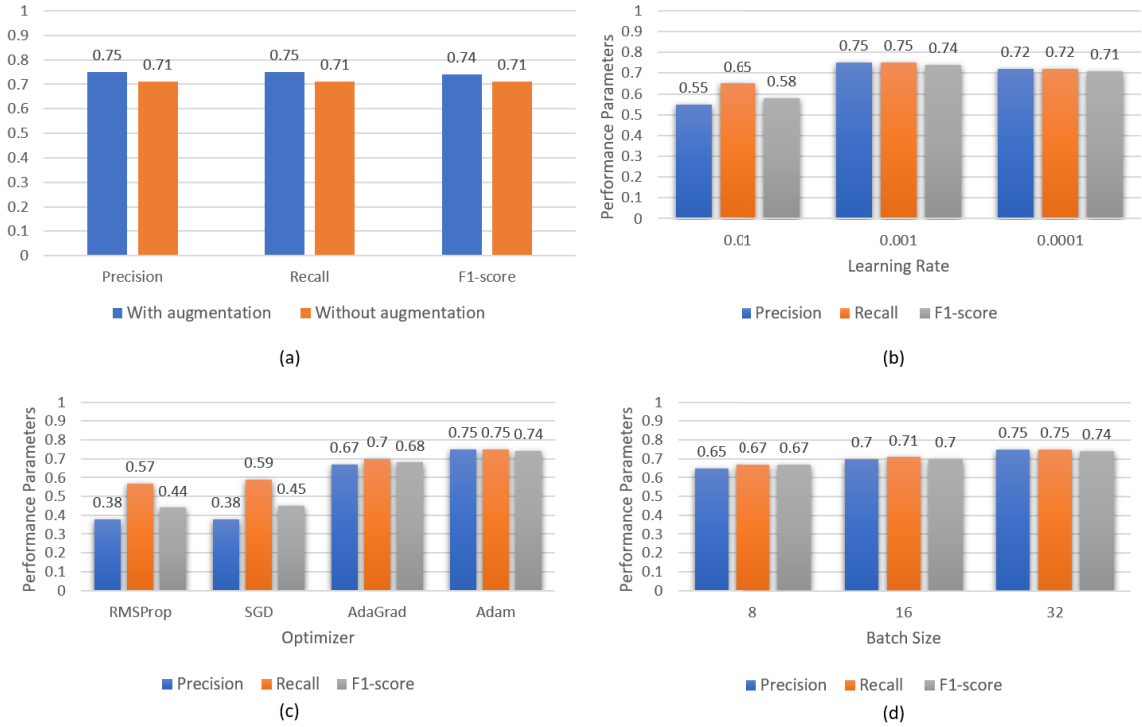

FIGURE 14. Performance of the proposed model on different parameters (a) With and without data augmentation (b) Learning rate (c) Optimizer (d) Batch size.

TABLE 4. Performance metrics of proposed CNN model for individual labels.

Label No.	Precision	Recall	F1-score	Specificity	FPR	FNR	AUC Score
0	0.83	0.87	0.85	0.88	0.123	0.13	0.94
1	0.90	0.81	0.86	1.00	0.004	0.19	0.99
2	0.90	0.76	0.82	0.99	0.012	0.24	0.95
3	0.84	0.64	0.73	0.99	0.006	0.36	0.97
4	0.81	0.82	0.81	0.99	0.013	0.18	0.97
5	0.68	0.67	0.68	0.97	0.027	0.33	0.94
6	0.62	0.60	0.61	0.99	0.011	0.40	0.94
7	0.84	0.66	0.74	0.99	0.012	0.34	0.94
8	0.44	0.67	0.53	1.00	0.002	0.33	0.99
9	0.80	0.80	0.80	1.00	0.000	0.20	1.00
10	0.50	1.00	0.67	1.00	0.001	0.00	1.00
11	0.77	0.61	0.68	0.99	0.006	0.39	0.91
12	0.76	0.54	0.63	1.00	0.004	0.46	0.96
13	0.55	0.57	0.56	0.99	0.007	0.43	0.94
14	0.90	0.82	0.85	1.00	0.004	0.18	0.98
15	1.00	1.00	1.00	1.00	0.00	0.00	1.00
16	0.41	0.28	0.34	0.99	0.008	0.72	085
17	0.28	0.44	0.34	0.99	0.010	0.56	0.94
18	0.53	0.47	0.49	0.99	0.014	0.53	0.92
19	0.56	0.52	0.54	0.98	0.021	0.48	0.90
20	0.19	0.38	0.25	1.00	0.003	0.88	0.96
21	0.61	0.72	0.66	0.93	0.070	0.27	0.93
22	0.62	0.39	0.48	0.99	0.010	0.55	0.94
23	0.84	0.78	0.81	0.97	0.030	0.28	0.97
24	0.86	0.63	0.73	1.00	0.002	0.39	0.93
25	0.62	0.82	0.71	0.83	0.175	0.22	0.89
26	0.48	0.44	0.46	0.99	0.006	0.74	0.92
27	1.00	1.00	1.00	1.00	0.002	1.00	1.00

learning rate. Therefore, it contributes to better accuracy and less overall loss. Different optimizers were used to check the model's performance like Stochastic Gradient Descent (SGD), Root Mean Squared Propagation (RMSProp), Adaptive Gradient (AdaGrad) and Adaptive Moment Esetimation (Adam). Average precision, recall and f1-score for the

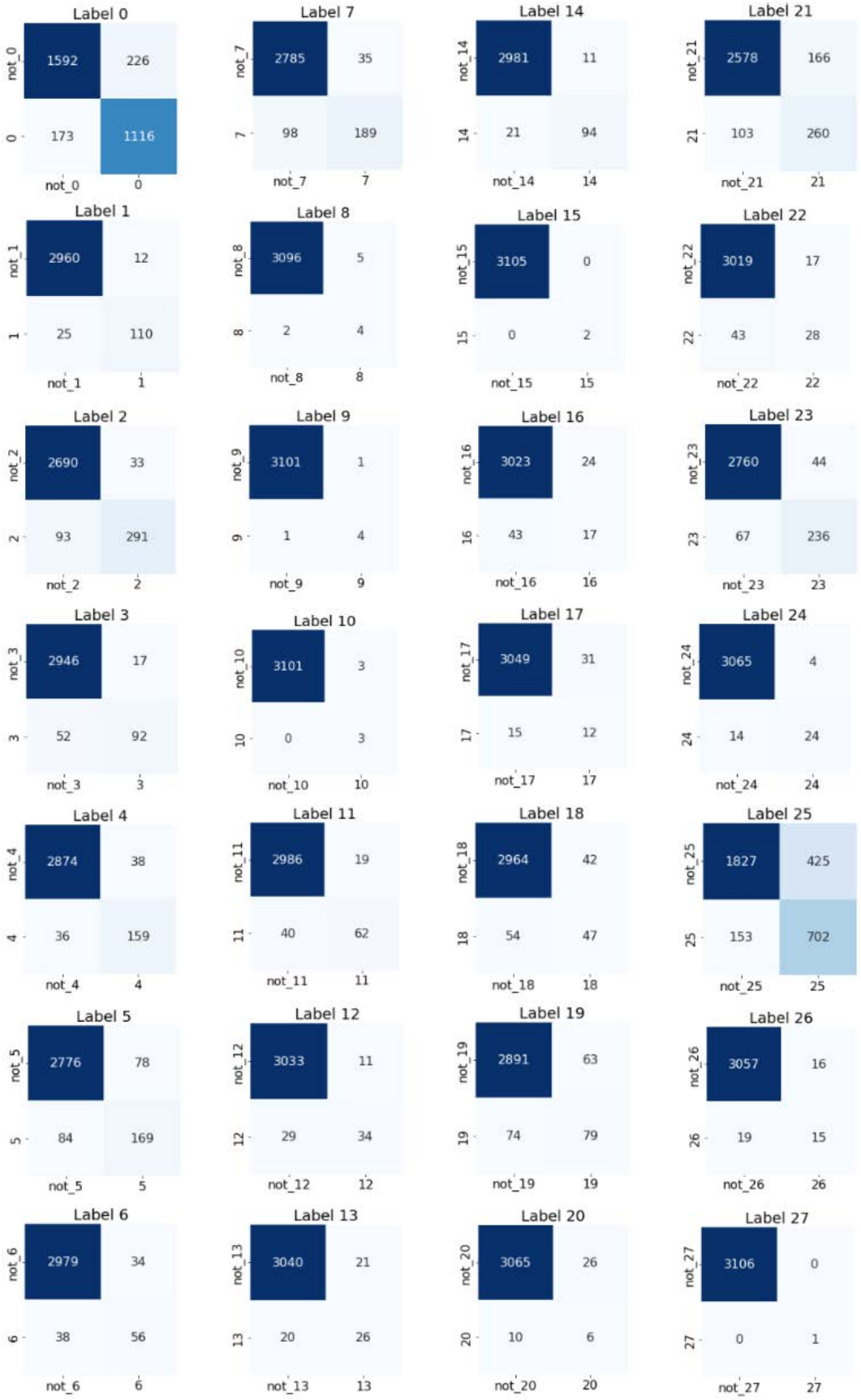

FIGURE 15. Confusion matrix of proposed CNN model.

TABLE 5. Comparison of the proposed model with the sate-of-the-art.

Reference No.	Technique	Dataset Used	F1-score
[35]	FCN	HPA Dataset	0.696
	CNN		0.676
[36]	Inception V3	HPA Dataset	0.706
[37]	Hybrid Xception	HPA Dataset	0.69
[38]	CNN based tool Loc-CAT	HPA Dataset	0.72
[39]	DeepLoc	Yeast FGP Dataset	0.72
[40]	ResNet	HPA Dataset	0.3459
Proposed CNN Model	CNN	HPA Dataset	**0.74**

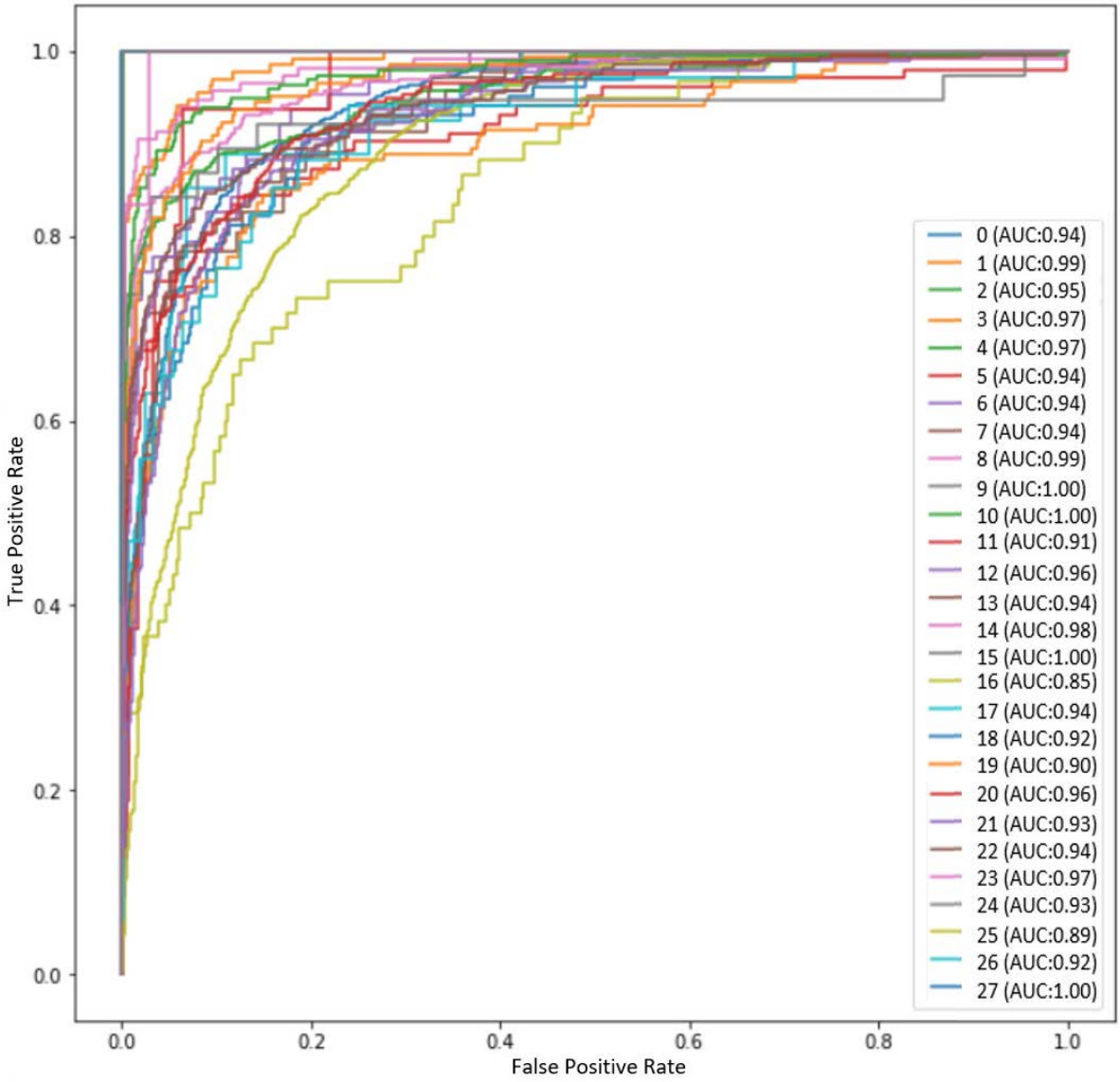

FIGURE 16. ROC curve for the proposed CNN model.

aforementioned optimizers is shown in Figure 14(c). It can be inferred from Figure 14(c) that the best results were obtained for Adam otimizer with an F1-score of 0.74.

(c) Learning Rate: In an optimization technique, the learning rate is a tunable parameter that affects the size of the step taken during each iteration on the path to the minimal of a loss function. Model's performcance was checked on different learning rates of 0.01, 0.001 and 0.0001 as shown in Figure 14(b). It can be seen from the figure that the best results were obtained for learning rate of 0.001.

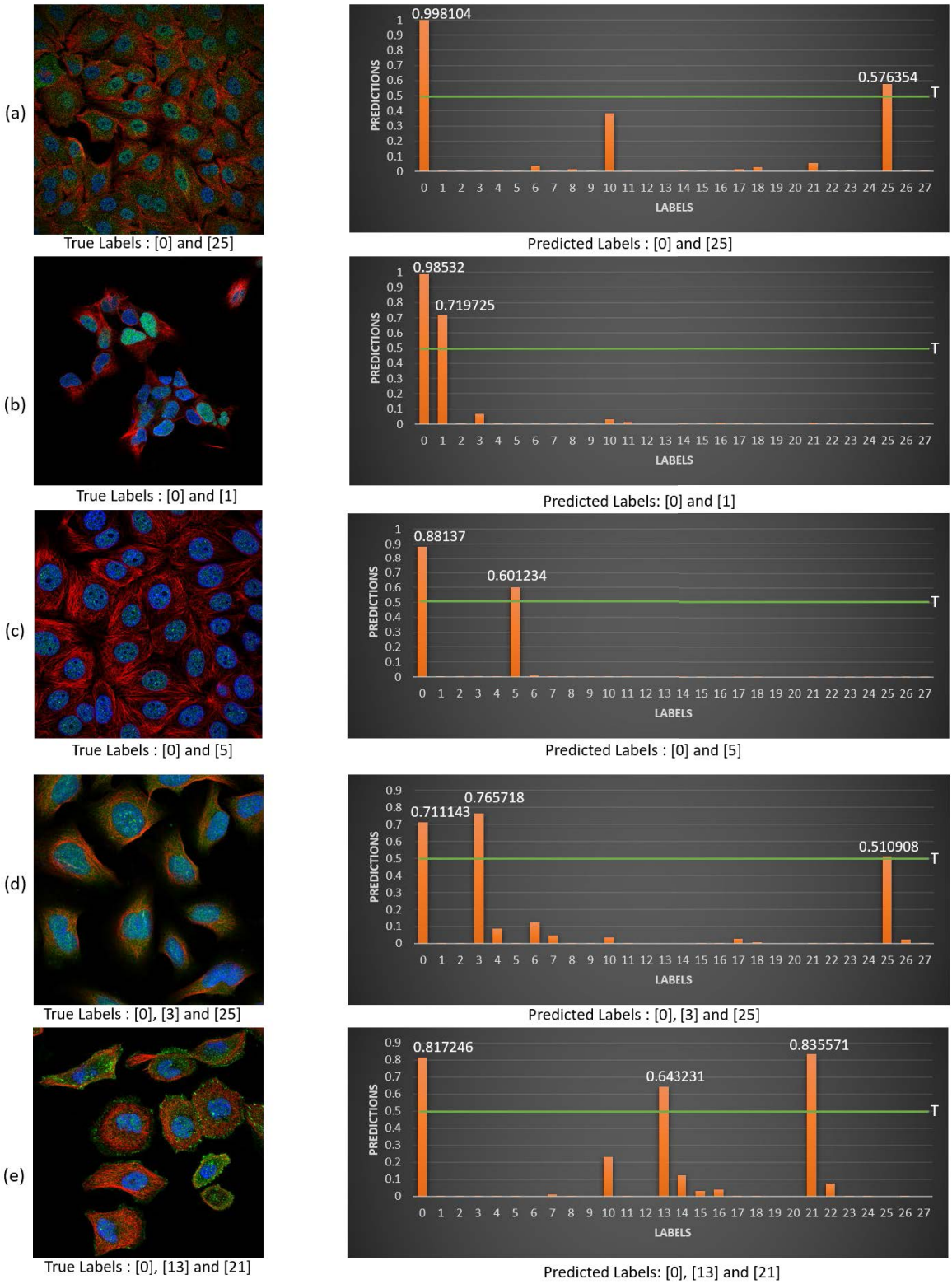

FIGURE 17. Examples of successful classification. Original images along with their actual labels are given in the left column, and the right column presents probabilities and the predicted labels obtained from the model.

It is important to select an optimal learning rate bcause if learning rate is too high then model may show an undesirable divergent behaviour during gradient descent, and if learning rate is too low, then model will train very slowly and model may fail to reach optimum minimal of the loss function.

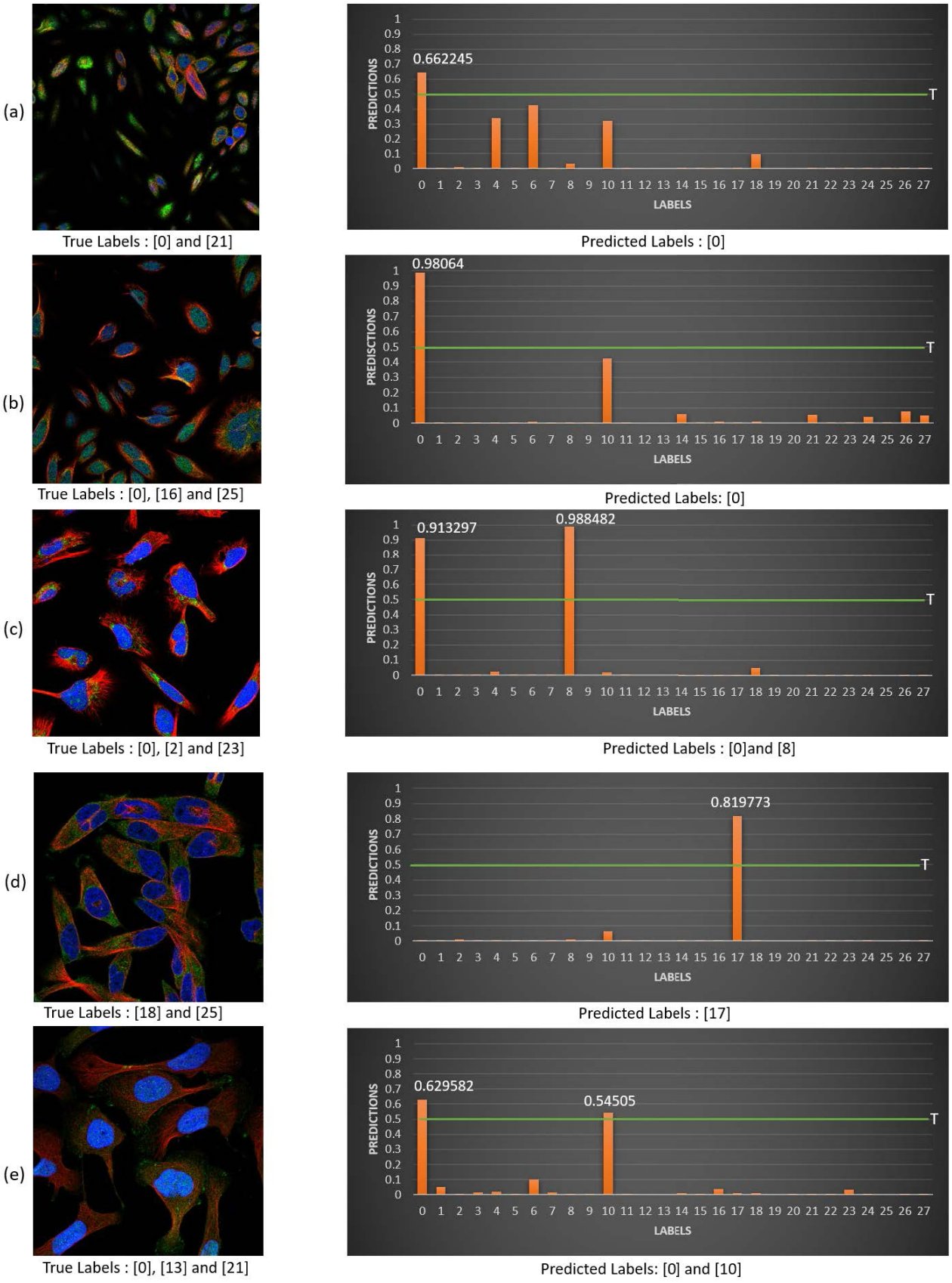

FIGURE 18. Examples of unsuccessful classification. Left column's image denotes original images along with their actual labels, and the right column presents probabilities and the predicted labels obtained from the model.

5) PERFORMANCE METRICS OF THE PROPOSED CNN MODEL

After selecting the appropriate input image size, filter size and optimum hyperparameters, the performance of the proposed CNN model was obtained. Precision, recall and f1-score obtained across each label on the test dataset are shown in Table 4. Average precision, recall and f1-score obtained from the model is 0.75, 0.75 and 0.74 respectively. Other performance parameters like specificity, FPR and FNR have also been calculated using confusion matrix shown in Figure 15. The value of these parameters across each label are shown in Table 4. From confusion matrix in Figure 15 and Table 4, it can be seen that for some labels classification was done very accurately. Like, for label 15 and 27, all the samples in test dataset were classified accurately with zero false positive and false negative rate and precision of 1.00. While some labels despite of having sufficient number of samples were not classified accurately. The reason of misclassification can be due to the overlapping location of proteins in subcellular compartments. AUC-ROC curve for each class obtained using test dataset is shown in Figure 16.

D. VISUALIZATION OF CORRECTLY CLASSIFIED SAMPLES

To demonstrate the proposed CNN architecture's performance, Figure 17 shows examples of successful classifications made by the model. Since the last layer of our model is a sigmoid layer, a threshold value was required to obtain the output probabilities from this layer. The threshold was changed from 0 to 1, with the 0.5 threshold yielding the best results. If the output probabilities surpass the threshold, the predicted value was assigned to the corresponding label (or several labels). Notably, the proposed model produces a higher likelihood for the target protein. Furthermore, as illustrated in Figure 17(a)-(e), more confident prediction results have been obtained for the majority classes like class-0 and class-25. Also, the model gives a more confident prediction for class-25, as shown in Figure 17(a) and (d).

E. VISUALIZATION OF MIS-CLASSIFIED SAMPLES

Figure 18 shows some examples of unsuccessful classification done by the model. Majority classes like class-0 was usually classified correctly by the model as shown in Figure 18(a)-(e). However, there were many sample images for which the model predicted new labels which were not present in their true labels. For example, in Figure 18(d), the true labels are 18 and 25, but the model predicted presence of protein in class 17. Similarly, in Figure 18 (e), proteins are present in subcellular compartments with classes 0, 2 and 23, but the models predicted presence of protein under class 0 and class 18. More examples of incorrect classification are given in Figure 18.

F. COMPARISON WITH THE STATE-OF-ART

The proposed model was compared with the present state of the art, as shown in Table 5. In addition, the proposed model was compared with the existing models based on F1-score, considering both precision and Recall. Table 5 shows that the technique which is proposed performed better in comparison with the other techniques. It achieved an F1- score of 0.74, which is 0.2 more than the existing methods.

V. CONCLUSION

A deep neural network has been presented for automatically classifying human proteins in the Human Protein Atlas in this research. Specifically, a CNN-based model has been proposed to classify the patterns of protein in subcellular compartments. To illustrate the efficacy of the proposed system, various experiments were done on the proposed CNN model. The model was tested with different input shapes, filter sizes, and hyperparameter combinations to get the best results. Furthermore, F1-score of 0.74 was achieved with the proposed model for the multi-label classification problem, while most of the work in the literature focused on single-label classification. Only a few research teams have demonstrated an exceptional strategy for classifying proteins with multiple labels. Unlike prior classification algorithms, the suggested method automatically extracts features from the Human Protein Atlas images and performs multi-label HPA classification.

Furthermore, in order to compare the performance of proposed CNN model with more competitive emerging models, we have proposed another model using ensembling technique. A stacked ensemble of three fine-tuned transfer learning models, VGG19, DenseNet121 and ReNet102 has been presented. Performance metrics was obtained for both the proposed models and the results shows that the CNN made from scratch outperformed the ensemble model.

Class imbalance is one of the critical difficulties encountered during the experiment, causing significant complications, and impairing the model's effectiveness. As a result, future work will focus on applying data balancing strategies to enhance model performance and mitigate the effect of imbalanced sample sizes in each class. More number of samples could be collected for minority classes to obtain more accurate prediction results.

ACKNOWLEDGMENT

The authors would like to thank the QNL for supporting collaborative research work.

REFERENCES

[1] B. Alberts, A. Johnson, J. Lewis, D. Morgan, M. Raff, K. Roberts, P. Walter, J. Wilson, and T. Hunt, *Molecular Biology of the Cell*, 6th ed. Boca Raton, FL, USA: Norton, 2017, pp. 256–290.

[2] M.-C. Hung and W. Link, "Protein localization in disease and therapy," *J. Cell Sci.*, vol. 124, no. 20, pp. 3381–3392, Oct. 2011.

[3] A. S. Payne, E. J. Kelly, and J. D. Gitlin, "Functional expression of the Wilson disease protein reveals mislocalization and impaired copper-dependent trafficking of the common H1069Q mutation," *Proc. Nat. Acad. Sci. USA*, vol. 95, no. 18, pp. 10854–10859, Sep. 1998.

[4] H. Parfrey, R. Mahadeva, and D. A. Lomas, "α1-Antitrypsin deficiency, liver disease and emphysema," *Int. J. Biochem. Cell Biol.*, vol. 35, no. 7, pp. 1009–1014, 2003.

[5] W. R. Skach, "Defects in processing and trafficking of the cystic fibrosis transmembrane conductance regulator," *Kidney Int.*, vol. 57, no. 3, pp. 825–831, Mar. 2000.
[6] Y.-Y. Xu, F. Yang, Y. Zhang, and H.-B. Shen, "An image-based multi-label human protein subcellular localization predictor (iLocator) reveals protein mislocalizations in cancer tissues," *Bioinformatics*, vol. 29, no. 16, pp. 2032–2040, Aug. 2013.
[7] K. Lee, K. Byun, W. Hong, H.-Y. Chuang, C.-G. Pack, E. Bayarsaikhan, S. H. Paek, H. Kim, H. Y. Shin, T. Ideker, and B. Lee, "Proteome-wide discovery of mislocated proteins in cancer," *Genome Res.*, vol. 23, no. 8, pp. 1283–1294, Aug. 2013.
[8] M. Uhlen, P. Oksvold, L. Fagerberg, E. Lundberg, K. Jonasson, M. Forsberg, M. Zwahlen, C. Kampf, K. Wester, S. Hober, H. Wernerus, L. Björling, and F. Ponten, "Towards a knowledge-based human protein atlas," *Nature Biotechnol.*, vol. 28, no. 12, pp. 1248–1250, Dec. 2010.
[9] R. Pepperkok and J. Ellenberg, "High-throughput fluorescence microscopy for systems biology," *Nature Rev. Mol. Cell Biol.*, vol. 7, no. 9, pp. 690–696, Sep. 2006.
[10] E. C. Too, L. Yujian, S. Njuki, and L. Yingchun, "A comparative study of fine-tuning deep learning models for plant disease identification," *Comput. Electron. Agricult.*, vol. 161, pp. 272–279, Jun. 2019.
[11] N. Vos, H. van der Meijden, and E. Denessen, "Effects of constructing versus playing an educational game on student motivation and deep learning strategy use," *Comput. Educ.*, vol. 56, no. 1, pp. 127–137, Jan. 2011.
[12] A. Gupta, A. Anpalagan, L. Guan, and A. S. Khwaja, "Deep learning for object detection and scene perception in self-driving cars: Survey, challenges, and open issues," *Array*, vol. 10, Jul. 2021, Art. no. 100057.
[13] C. Affonso, A. L. D. Rossi, F. H. A. Vieira, and A. C. P. F. de Carvalho, "Deep learning for biological image classification," *Expert Syst. Appl.*, vol. 85, pp. 114–122, Nov. 2017.
[14] J. Zhang, Y. Xie, Q. Wu, and Y. Xia, "Medical image classification using synergic deep learning," *Med. Image Anal.*, vol. 54, pp. 10–19, May 2019.
[15] A. M. Ismael and A. Şengür, "Deep learning approaches for COVID-19 detection based on chest X-ray images," *Expert Syst. Appl.*, vol. 164, Feb. 2021, Art. no. 114054.
[16] X. Liu, L. Song, S. Liu, and Y. Zhang, "A review of deep-learning-based medical image segmentation methods," *Sustainability*, vol. 13, no. 3, p. 1224, Jan. 2021.
[17] S. Devunooru, A. Alsadoon, P. W. C. Chandana, and A. Beg, "Deep learning neural networks for medical image segmentation of brain tumours for diagnosis: A recent review and taxonomy," *J. Ambient Intell. Hum. Comput.*, vol. 12, no. 1, pp. 455–483, Jan. 2021.
[18] R. F. Mansour and N. O. Aljehane, "An optimal segmentation with deep learning based inception network model for intracranial hemorrhage diagnosis," *Neural Comput. Appl.*, vol. 33, no. 20, pp. 13831–13843, Oct. 2021.
[19] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.
[20] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.
[21] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 1251–1258.
[22] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam, "MobileNets: Efficient convolutional neural networks for mobile vision applications," 2017, *arXiv:1704.04861*.
[23] X. Zhang, X. Zhou, M. Lin, and J. Sun, "ShuffleNet: An extremely efficient convolutional neural network for mobile devices," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 6848–6856.
[24] B. E. Bejnordi, M. Veta, P. J. Van Diest, B. Van Ginneken, N. Karssemeijer, G. Litjens, J. A. Van Der Laak, M. Hermsen, Q. F. Manson, M. Balkenhol, and O. Geessink, "Diagnostic assessment of deep learning algorithms for detection of lymph node metastases in women with breast cancer," *J. Amer. Med. Assoc.*, vol. 318, no. 22, pp. 2199–2210, 2017.
[25] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 25, 2012, pp. 1097–1105.
[26] W. Ouyang *et al.*, "Analysis of the human protein atlas image classification competition," *Nature Methods*, vol. 16, no. 12, pp. 1254–1261, 2019.
[27] E. Glory and R. F. Murphy, "Automated subcellular location determination and high-throughput microscopy," *Develop. Cell*, vol. 12, no. 1, pp. 7–16, Jan. 2007.
[28] X. Chen, M. Velliste, and R. F. Murphy, "Automated interpretation of subcellular patterns in fluorescence microscope images for location proteomics," *Cytometry*, vol. 69, no. 7, pp. 631–640, 2006.
[29] M. V. Boland and R. F. Murphy, "A neural network classifier capable of recognizing the patterns of all major subcellular structures in fluorescence microscope images of HeLa cells," *Bioinformatics*, vol. 17, no. 12, pp. 1213–1223, Dec. 2001.
[30] D. P. Sullivan, C. F. Winsnes, L. Åkesson, M. Hjelmare, M. Wiking, R. Schutten, L. Campbell, H. Leifsson, S. Rhodes, A. Nordgren, K. Smith, B. Revaz, B. Finnbogason, A. Szantner, and E. Lundberg, "Deep learning is combined with massive-scale citizen science to improve large-scale image classification," *Nature Biotechnol.*, vol. 36, no. 9, pp. 820–828, Oct. 2018.
[31] K. Huang and R. F. Murphy, "Boosting accuracy of automated classification of fluorescence microscope images for location proteomics," *BMC Bioinf.*, vol. 5, no. 1, pp. 1–19, 2004.
[32] J. Y. Newberg, J. Li, A. Rao, F. Pontèn, M. Uhlèn, E. Lundberg, and R. F. Murphy, "Automated analysis of human protein atlas immunofluorescence images," in *Proc. IEEE Int. Symp. Biomed. Imag., From Nano Macro*, Jun. 2009, pp. 1023–1026.
[33] L. P. Coelho, J. D. Kangas, A. W. Naik, E. Osuna-Highley, E. Glory-Afshar, M. Fuhrman, M. R. Simha, P. B. Berget, J. W. Jarvik, and R. F. Murphy, "Determining the subcellular location of new proteins from microscope images using local features," *Bioinformatics*, vol. 29, no. 18, pp. 2343–2349, 2013.
[34] A. X. Lu, O. Z. Kraus, S. Cooper, and A. M. Moses, "Learning unsupervised feature representations for single cell microscopy images with paired cell inpainting," *PLOS Comput. Biol.*, vol. 15, no. 9, Sep. 2019, Art. no. e1007348.
[35] K. Liimatainen, R. Huttunen, L. Latonen, and P. Ruusuvuori, "Convolutional neural network-based artificial intelligence for classification of protein localization patterns," *Biomolecules*, vol. 11, no. 2, p. 264, Feb. 2021.
[36] Z. Li, R. Togo, T. Ogawa, and M. Haseyama, "Classification of subcellular protein patterns in human cells with transfer learning," in *Proc. IEEE 1st Global Conf. Life Sci. Technol. (LifeTech)*, Mar. 2019, pp. 273–274.
[37] T. R. Shwetha, S. A. Thomas, V. Kamath, and B. N. Krupa, "Hybrid Xception model for human protein atlas image classification," in *Proc. IEEE 16th India Council Int. Conf. (INDICON)*, Dec. 2019, pp. 1–4.
[38] T. Pärnamaa and L. Parts, "Accurate classification of protein subcellular localization from high-throughput microscopy images using deep learning," *G3, Genes, Genomes, Genet.*, vol. 7, no. 5, pp. 1385–1392, 2017.
[39] O. Z. Kraus, B. T. Grys, J. Ba, Y. Chong, B. J. Frey, C. Boone, and B. J. Andrews, "Automated analysis of high-content microscopy data with deep learning," *Mol. Syst. Biol.*, vol. 13, no. 4, p. 924, Apr. 2017.
[40] H.-Y. Chang and C.-L. Wu, "Deep learning method to classification human protein atlas," in *Proc. IEEE Int. Conf. Consum. Electron. Taiwan (ICCE-TW)*, May 2019, pp. 1–2.
[41] *Human Protein Atlas Image Classification*. Accessed: Nov. 15, 2021. [Online]. Available: https://www.kaggle.com/c/human-protein-atlas-image-classification
[42] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proc. 13th Int. Conf. Artif. Intell. Statist.*, 2010, pp. 249–256.
[43] F. Iandola, M. Moskewicz, S. Karayev, R. Girshick, T. Darrell, and K. Keutzer, "DenseNet: Implementing efficient ConvNet descriptor pyramids," 2014, *arXiv:1404.1869*.
[44] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.
[45] M. Abadi, "TensorFlow: Large-scale machine learning on heterogeneous distributed systems," 2016, *arXiv:1603.04467*.

SONAM AGGARWAL received the bachelor's and master's degrees in electronics and communication engineering from Kurukshetra University, Haryana. She is currently pursuing the Ph.D. degree with Chitkara University, Punjab. Her research interests include artificial intelligence, biomedical signal processing, machine learning, deep learning, and computer vision.

SHEIFALI GUPTA is currently a Professor with the Chitkara University Research and Innovation Network (CURIN), Chitkara University, Punjab Campus, India. She specializes in the area of digital image processing, pattern recognition, machine intelligence, biomedical image processing, agriculture based image processing, and deep learning. She has published more than 100 research papers and articles in reputed national and international journals and conferences. She has filed 25 patents in the field of image processing and mechatronics. She has conducted different workshops based on Modeling Real-Time Video Processing System in Simulink using the Video and Image Processing Blockset. She has received a prestigious IRDP Award 2018 for remarkable achievements in Teaching, Research, and Publications.

RAMANI KANNAN (Senior Member, IEEE) received the B.E. degree from Bharathiar University, India, and the M.E. and Ph.D. degrees in power electronics and drives from Anna University. He is currently a Senior Lecturer and a PG Programmer Leader at Universiti Teknologi PETRONAS, Malaysia. He holds more than 175 publications in reputed international and national journals and conferences. His research interests include electrical engineering, power electronics and drives, renewable energy, electric motors and drives, electric vehicles, design of power inverters, modeling of induction motor and optimization techniques, and artificial intelligent techniques. He is a Chartered Engineer (CEng), U.K., a Professional Technologist certified from MBOT, Malaysia, an Active Senior Member in IEEE (USA), and a members in IET (U.K.), BEM (Malaysia), IE (India), ISTE (India), and the Institute of Advance Engineering and Science. He is recognized with many awards, including ''Career Award for Young Teacher'' from AICTE India, ''Highest Research publication Award'' ''Outstanding Researcher Award'' prestigious award from Universiti Teknologi PETRONAS, in 2019. He received Award for Outstanding Performance, Service, and Dedication at UTP, Malaysia, in 2019, and Award for BEST PRESENTER CENCON 2019 IEEE Conference on Energy Conversion (CENCON 2019) Indonesia. He is actively serving as a Secretary for the IEEE Power Electronics Society, Malaysia, from January 2020 to February 2022. He has been the Chair of the IEEE Power Electronics Society, Malaysia, since February 2022, and the Organizing Chair of International Virtual Conference on Artificial Intelligence for Smart Community 2020, IEEE PECON 2020, and IEACon. He is currently an Associate Editor of IEEE Access, USA, *IET Journal of Engineering*, U.K. He is an Editor of *Machines* (MDPI), an Academic Editor of *Journal of Electrical and Computer Engineering* and *Mathematical Problems in Engineering* (Hindawi). He is a guest editor of special issues of MDPI, Springer, Hindawi, Inderscience Journals, and Elsevier Journals. He is a Book Editor and a Series Editor of Springer Nature Singapore, CRC Press, Taylor and Francis Group, Bentham Science Publishers, and Wiley.

RAKESH AHUJA is currently pursuing the Ph.D. degree in computer science and engineering. He is having more than 25 years of experience in academic, research, and industries. His areas of expertise are information hiding, machine learning, digital right management, multimedia security, and pattern recognition. He is currently working as a Professor with Chitkara University, Punjab, India. He has supervised several Scholars in the areas of Information Hiding. His research interests include database management systems, real time systems, distributed systems, software engineering, and operating systems.

DEEPALI GUPTA is currently working as a Professor Research with the Chitkara University Research and Innovation Network (CURIN), Chitkara University, Punjab, India. She specializes in software engineering, cloud computing, and genetic algorithms. She has worked with undergraduate, postgraduate students as well as research scholars throughout her career and plans to continue to involve students in her research and is eager to participate in projects and guide independent student's research. She has published more than 50 research papers in national and international journals and conferences. Based on these areas she has guided many Ph.D. and M.E. scholars. She has worked at various administrative positions like a Principal, the Head (CSE), the Dean Academics, an IBM (Spoc), a Remote Centre Coordinator (IITB), a Coordinator for IITB Spoken Tutorial, an Executive Committee Member of Computer Science Division, Haryana State Centre (IEI), the President of Sangam Kala Group (Kurukshetra, Mohali, and Chandigarh Chapter), a member of Anti-Ragging Committee, an Academic Council, Faculty of Engineering and Technology, the Board of Management, the Chairman SC/ST Cell of MMU, Sadopur, and the Principal (MMGI, Sadopur). Her research interests include software engineering, cloud computing, machine learning, and blockchain. She is an active member of various professional bodies like IEI (India), IETE, and ISTE. She is an Editor-in-Chief of MMU journal. She is an editorial board member and a reviewer of various journals.

SAPNA JUNEJA received the master's and Ph.D. degrees in computer science and engineering from M.D. University, Rohtak, in 2018 and 2010, respectively. She is currently working as a Professor with the Department of Computer Science, KIET Group of Institutions, Ghaziabad, India. She has more than 17 years of teaching experience. Her topic of research is *Software Reliability of Embedded System*. She has published six patents. She has published various research articles in the renowned national and international journals. Her research interests include software engineering, computer networks, operating systems, database management systems, and artificial intelligence. She has guided several research thesis of UG and PG students in computer science and engineering. She is a reviewer of several international journals of repute.

SAMIR BRAHIM BELHAOUARI (Senior Member, IEEE) received the master's degree in telecommunications from the National Polytechnic Institute (ENSEEIHT) of Toulouse, France, in 2000, and the Ph.D. degree in applied mathematics from the Federal Polytechnic School of Lausanne (EPFL), in 2006. He is currently an Associate Professor with the Division of Information and Computing Technology, College of Science and Engineering, Hamad Bin Khalifa University (HBKU). He also holds and leads several academic and administrator positions, as the Vice Dean for Academic and Student Affairs at the College of Science and General Studies and the University Preparatory Program at ALFAISAL University (Saudi Arabia), University of Sharjah (United Arab Emirates), Innopolis University (Russia), Petronas University (Malaysia), and EPiFL Federal Swiss School (Switzerland). He is also actively working on developing algorithms in machine learning applied to visual surveillance, sensing technologies and biomedical data, with the support of several international fund for research in Russia, Malaysia, and GCC. His main research interests include stochastic processes, machine learning, and number theory.