Optimized Heart Disease Image Classification on Edge Devices Using Knowledge Distillation and Layer Compression

Yogendra N. Prajapati [1,*], Dev Baloni [1], and Avdhesh Gupta [2]

[1] Department of CSE, Quantum University, Roorkee, India
[2] Department of CSE, AKGEC, Ghaziabad, India
Email: ynp1581@gmail.com (Y.N.P.); devbaloni82@gmail.com (D.B.); avvipersonal@gmail.com (A.G.)
*Corresponding author

***Abstract*—The proliferation of edge devices supports real-time diagnostic testing, even in rural or underserved locations. Convolutional Neural Networks (CNNs) are highly effective at analyzing medical images, including Computed Tomography (CT) scans and chest X-rays, for detecting heart diseases, but their computational complexity usually makes them unsuitable for usage on edge devices with limited resources. This paper presents a new compressed layered knowledge distillation model for precise medical image diagnosis, such as detecting COVID-19-related lung infections or identifying cardiovascular conditions. We utilize knowledge distillation to transfer the teacher network's knowledge to a smaller, compressed student network for edge deployment. Moreover, we utilize a well-structured layer compression approach, emphasizing decoupling and merging techniques instead of pruning, to optimize the student network architecture. Two data sets, Chest CT-Scan and SARS-CoV-2 CT-Scan, were used to test the suggested model. When compared to existing models, our performance is superior. For the Chest CT-Scan dataset and SARS-CoV-2 CT-Scan, we achieved 98.93% Accuracy, 98.41% Precision, 98.69% Recall, and an F1-Score of 98.44%. The Mean Squared Error (MSE) was 0.04, with a Root Mean Squared Error (RMSE) of 0.16. For the Chest CT-Scan dataset, our results were similarly strong: 98.25% Accuracy, 98.78% Precision, 98.86% Recall, and an F1-Score of 98.14%. The MSE for this dataset was 0.09, and the RMSE was 0.13. These results verify the efficacy of our intended technique for achieving high diagnostic accuracy at low error in edge devices.**

***Keywords*—edge computing, medical image analysis, COVID-19 diagnosis, deep learning compression, knowledge transfer, model optimization, layer fusion, efficient inference**

I. Introduction

The COVID-19 epidemic brought to light the pressing need for quick and precise diagnostic tools, particularly in environments with limited resources. Chest X-Rays (CXR) and Computed Tomography (CT) scans are examples of diagnostic imaging that has proven crucial to the diagnosis and treatment of COVID-19. Despite the impressive performance of Convolutional Neural Networks (CNNs) and other deep learning algorithms in image processing for illness diagnosis, their computational cost frequently precludes its usage on edge devices. Edge computing offers the hope of point-of-care diagnosis, which enables faster output and reduced reliance on central facilities, a situation most desirable in remote or underprivileged regions. Edge devices, however, are limited in their processing power, memory, and power supply, a major drawback in employing sophisticated CNNs. Efficient and lightweight deep learning models are therefore increasingly in demand for efficient COVID-19 image classification on edge devices [1].

One attractive solution to this problem is Knowledge Distillation (KD). Training a smaller "student" network to mimic the behavior of a bigger, more intricate "teacher" network is known as KD training. This enables the student network to gauge the teacher's performance with much fewer computational resources. Model compression algorithms like pruning, quantization, and layer compression can then further compress the student network's size and complexity. Layer compression, in particular, tackles the network topology by selectively merging or splitting layers, offering a more structured approach than the elimination of connections. This work proposes a new approach that combines knowledge distillation with an optimally designed layer compression method for effective COVID-19 image classification on the edge device. Our method aims to develop a highly accurate but compact model deployable on resource-constrained platforms. We introduce a compressed layered knowledge distillation model specially designed for medical image diagnosis, e.g., COVID-19, malaria, and other lung diseases. The approach not only facilitates effective inference on the edge device but also maintains diagnostic accuracy equivalent to increasingly larger models. The following sections provide our proposed methodology, experimental findings, and comparative analysis with state-of-the-art approaches, illustrating the effectiveness of our combined knowledge distillation and

Manuscript received April, 12, 2025; revised May 21, 2025; accepted June 18, 2025; published September 17, 2025.

layer compression approach for COVID-19 image classification in edge computing systems [2].

II. Related Work

Hu *et al*. [3] integrated fuzzy clustering with HPU-NET for brain tumor segmentation, although both papers accounted for difficulties in terms of generalizability and inter-patient variability deep learning was an emergent technique for medical image analysis and did well on numerous tasks including segmentation, classification, and detection. Initial attempts utilized CNNs for the identification of disease from CT, MRI, and retinal images. For example, Tayal *et al*. [4] obtained ~96% accuracy for retinal layer segmentation, whereas.

Some works have investigated hybrid models and fusion methods to enhance performance. Puttagunta *et al*. [5] and Kumar *et al*. [6] compared deep learning methods on various imaging modalities (e.g., X-ray, mammography, histopathology), with high accuracy (~94%–97%) but mentioning constraints like small data availability and absence of clinical standardization. Naz *et al*. [7] and Phan *et al*. [8] proposed Internet of Things (IoT)-based deep learning models, uncovering the promise of connected diagnostics but mentioning training data and covariate shift constraints.

Attention mechanisms and structural improvements such as residual connections have become popular to enhance feature extraction. Hussain *et al*. [9] introduced MAGRes-UNet by employing multi-attention gates and residual paths, and Ortega-Ruíz *et al*. [10] combined dilation and dense connections in DRD-Net for the segmentation of breast cancer. Likewise, Iriawan *et al*. [11] presented YOLO-UNet for the detection of brain tumors efficiently combining object detection with segmentation for improved localization in MRI scans.

Recent architectures also depict innovation in fusion approaches. Li *et al*. [12] created Diamond-UNet based on global-local feature extraction, and Wang *et al*. [13] tested attention-dual UNet for infrared-visible image multi-modal fusion, with applications in medical and industrial fields.

For disease-specific uses, some domain-specific models have been introduced. Ardila *et al*. [14] employed a 3D CNN for low-dose CT lung cancer screening, setting a standard for computerized oncology devices. SVMs and hybrid deep models have also been tested, demonstrating the shift from traditional machine learning to deep neural nets [15–17]. Additionally, Rehman *et al*. [18] and Abuhayi *et al*. [19] introduced VGG-based models for osteoarthritis and spinal conditions, employing transfer learning for better diagnosis.

In totality, though deep learning has shown immense potential, some present limitations include model generalizability, dataset imbalance, and integration with clinical environments. Our research is targeted towards resolving these problems with a new architecture that combines multi-scale contextual information and improves segmentation accuracy for CT-based detection of heart disease.

III. Proposed Method

Doctors can perform remote and real-time diagnostic tests with edge devices driven by Artificial Intelligence (AI), which is especially useful for underprivileged or rural populations. The interpretation of CT and CXR images to detect lung pathology and image processing are the primary uses of machine learning techniques, especially deep learning models like CNNs [20].

However, CNNs require a lot of processing power from GPUs because of their multiple layers, numerous parameters, and high rate of computation. Thus, Deep Neural Networks (DNNs) that are complex cannot be utilized for creating delay-constrained low-weight applications on fog/edge devices since their CPU, memory, bandwidth, and power are usually constrained. Research in deep neural network compression has attracted significant interest in recent years. Here, in this suggested approach, a compressed layered knowledge distillation model will be suggested for the diagnosis of medical images such as SARS-CoV-2 and other forms of lung diseases. The proposed model will first preprocess the input image through normalization and data augmentation operations. Next, a joint learning scheme will be presented for the teacher–student knowledge distillation model to identify the diseases from input medical images in a fast and trustworthy way. Additionally, rather than using pruning, we might propose a structured layer compression method that effectively compresses consecutive layers by decoupling and merging.

Without damaging the correlation between the convolutional layers, it can effectively decrease the network's depth. Pictures utilized in medical contexts the proposed framework accepts X-rays, CT-Scan images, and cell photos as input. These remotely acquired medical images from smart edge devices, etc., serve as a conduit between patients and medical professionals. The next section discusses the design and architecture of the suggested components of a tiny deep learning model.

A. Components of Proposed Architecture

The cloud data center, fog bus module, and gateway devices are some of the components of the recommended approach. Fig. 1 shows the general framework of the suggested paradigm.

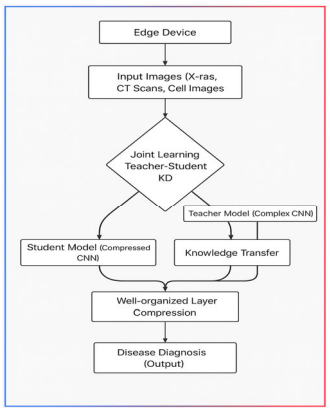

Fig. 1. Proposed architecture model.

1) Gateway devices

Gateway devices comprise laptops, mobile phones, or tablets used as fogging devices. They take pictures of patients and send them to brokers or labor nodes for processing.

Curved Webs: This term refers to specific architectural components within the student model, particularly the series of non-linear activation functions and convolutional layers that form non-linear mappings. We now describe it in the manuscript as a sequence of convolutional layers integrated with non-linear activations that generate a "curved" transformation space, which is essential for learning complex feature representations in compressed models.

2) Fog bus module

We have clarified that the "fog bus module" serves as an intermediary computational layer between edge (gateway) devices and the cloud. It facilitates localized processing and resource allocation for real-time medical image analysis, particularly in low-resource environments. We added a plain-language explanation emphasizing its role in minimizing latency and supporting decentralized diagnosis.

Labor nodes: This is a term that is equivalent to "fogbusworker" nodes or the nodes responsible for processing tasks directed by gateway devices. Clearly establish this equivalence. For instance, gateway devices direct images to labor nodes (also called fogbusworker nodes) for processing, which are provided with embedded computers and Raspberry Pi boards to run the deep learning models locally all these describe in Fig. 2.

3) Cloud data center

Apart from saving the gathered data, the processing of the data received by the cloud data center is beyond the capacity of fog nodes. While neural models have proven to be adequate in most areas, even for complicated problem states, the models are too big. To employ the use of these models on edge device low memory constraints. Knowledge filtering can be one possible solution to address this issue. In demonstrated the overall framework of the entire KD method. The idea of the KD compresses heavier models into lighter models. In KD, there is a small training set that is learned with filtered knowledge from a big pre-trained set (e.g. teacher). For the problem of image classification, the author model first learned over image-based data sets and labels. This is a pre-trained teacher model outputs class probabilities for a guide input. A DNN, not so advanced can learn a learner network over the same data. There are two components of the training loss of the student model. Firstly, close the difference between predicted labels and true labels (hard labels), which decrease (smooth) the second author's predictions labels. Classical CBS is a teacher model having so bit labels as input data (E.A., relative characteristics of distinct classes), all of which are quite particular in comparison to the Temperature (T), which is crucial for the student's model to be thoroughly trained throughout this filtering procedure. The probability of temperature aids in scoring. Examine the teacher sample recordings first, then run them through the Softmax function at the chosen temperature to obtain the probability score. This provided pupils with sample probabilities that were smooth.

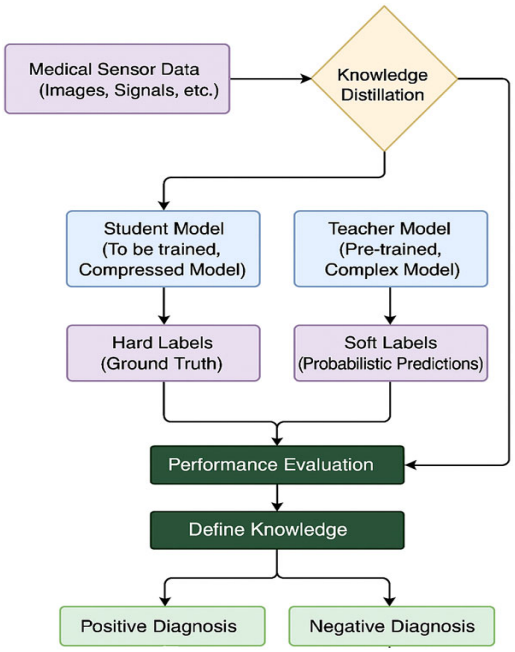

Fig. 2. Known ledge distillation model.

The notation $\{X_i, y_i\}$ (X_i: image, y_i: image labels) can be used to represent an image-based data set. Two sets of logits are obtained by emailing the teacher and pupil an image $x \in X$. The probability distribution of the instructor and student model can be defined as follows:

$$p(teacher) = soft\ max(a_t / T) \tag{1}$$

$$p(student) = soft\ max(a_s / T) \tag{2}$$

where, T is determined as the limiting distillation temperature $p(teacher)$ and $p(student)$ entropy.

Filtration loss is defined as:

$$Loss = \alpha \times L_{hard}(p(student), y) + (1-\alpha) \times L_{soft}(p(student), p(teacher)) \tag{3}$$

where α is the weighting coefficient for students and distillation, and the actual label for x and L_{hard}, L_{soft} measured by cross entropy. Entropy is proportional to $p(student)$ increase with the value of T, which leads the students to learn the relative probabilities of specific classes based on a teacher model that has already been trained. In spite of this, a high D could make improper, irrelevant classes more likely.

Fig. 3 illustrates the knowledge distillation process, showing how the pre-trained teacher model transfers soft and hard labels to a smaller student model. We have updated the text to explain each component of the pipeline, including the role of soft labels, temperature scaling, and how various student models (e.g., Mobile-Net, ResNet50, etc.) are evaluated.

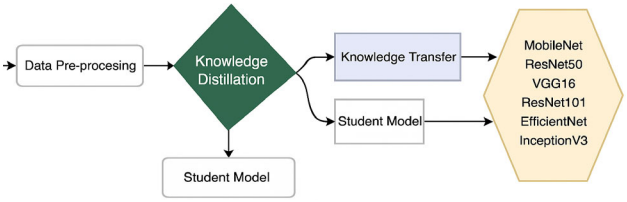

Fig. 3. Known ledge distillation method with pretrained model.

B. ResNet-50 as Teacher

Deep neural network is a no breakthrough in image classification. Numerous other scene recognition tasks of them have greatly been improved by very deep models. Thus, the years have a tendency to settle deep and hard improving tasks and accuracy. But when we go deeper, neurosis. Network training becomes difficult, accuracy reaches saturation, and these problems will get worse. Residual learning helps with these problems. For the previously mentioned rationale, we used the ResNet50 model as our primary instructor model. The acronym Res-Net, which stands for residual network, basically describes the residual learning jargon used by this network. The most popular deep network for image classification is ResNet50, a 48-curve layer network with a pooling layer and a fully coupled layer. This model uses the architecture shown in Fig. 3 and the pretrained version of ResNet50, which has about 23, 542,786 trainable parameters. The ImageNet data set provides the pre-trained sample weights. Softmax was used in the final layer after this model 0.5 was deleted, and the probability between the two classes.

C. Compressed Student Model

They use pre-trained models such as the student model, Mobile Net, Fusion V3, Optimized B0, VGG16, ResNet50, ResNet101, and the optimal temperature value that was determined. process for a pre-trained student model. To obtain smooth probabilities, the trained teacher model was first trained, logits were derived using soft-max, and then temperature was used for training. They then trained them to assess these models' performance as student models. This section discusses the model.

InceptionV3: InceptionV3 is a 42-layer deep neural network built on the CNN architecture. The maximum pooling, fc, and convolutional layers make up the symmetrical and asymmetric architectural elements of the InceptionV3 model. There are less than twenty-five million parameters in this state-of-the-art model. 5.6% top-5 mistake and 21.2% top-1 error. Using trainable knowledge parameter filtering, the performance metrics 21M+ on the test set were confirmed [21].

D. Layer Compression

The Effective Layer Compression (ELC) strategy without pruning merges layers instead of pruning them. This is the initial research to use the integration of consecutive layers of convolution in network compression to produce efficient layer compression. Nevertheless, non-linear activation layers are found in between curved layers of a network of convolutional neural networks. The numerous merge transformation layers are prevented from functioning by ReLU, PReLU, and other non-linear activation layers. 3×3 convolutional layers are frequently used in DNNs in addition to serial joins. They increase the quantity parameter and computational complexity. For instance, two series of 3×3 convolutional layers have the same function as a 5×5 convolutional layer; however, in order to separate the activation and transformation layers, the number of parameters has been increased from 18 to 25 non-convergence in successive convolutional levels [22].

Fig. 4 illustrates the suggested layer compression method, with Rem-ReLU and De-Conv operations. We explain in the updated version how Rem-ReLU eliminates duplicate non-linear activations to support combination of neighboring layers, and how De-Conv unstacks and decouples convolutional layers stacked together to simplify computation. We also include step-by-step references within the text to help readers follow each transformation indicated in the diagram.

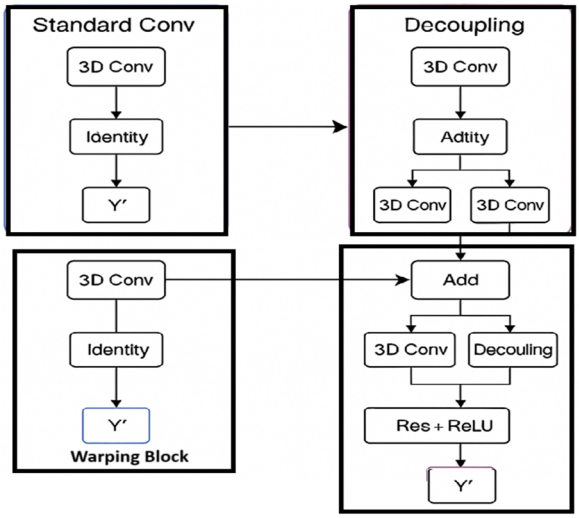

Fig. 4. The proposed layer compression method.

1) Layer decoupling

The two parts of the decoupling module, Rem-ReLU disconnect and switch processing layers, disengage the transform layers.

Rem-ReLU: non-linear layers are added to make the network more representative. However, ConvNeXT has a huge linear redundancy of the network. If one is present then two transform layers' combination cannot be losslessly combined between these two. it is meant to get rid of non-linear activation functions there could be an initial network and follow-up layers of transformation is combined as a convolutional layer without loss. Get rid of inefficient linear processing layers, Rem-ReLU of the network. Rem-ReLU is the

$$y_{i+1} = \sigma(y_i') = \begin{cases} y_i' & , y_i' \geq 0 \\ (1-\alpha i)x_i' & , x_i' < 0 \end{cases} \quad (4)$$

where the learnable parameter, which is initialized to zero, is used to modify the slope of the negative half-axis of the

ith activation layer, and y_{i+1} is the output feature of the activation layer and the input feature of the (*i*+1)-*th*.

Convolutional layer Rem-ReLU activation function. Eq. (1) can be written as $y_{i+1} = \sigma(y_i') = y_i'$ where $\alpha\_i = 0$. Non-linearity at the implementation layer can thus be easily removed. When α_i is equal to zero, the activation layer is equally normalized. Layers cannot be joined if the activation layer is curved close to it and has a linear character. Rem-ReLU is intended to transform nonlinearity in contrast to PReLU. Identification mapping by application-inclined punishment activation function. Consequently, Rem-ReLU enables us to remove nonlinear activation from the network [23].

De-Conv: Combining two curved layers with kernel sizes larger than one can result in a convolutional layer with enormous kernel size, parameters, and computational complexity. The altered layer that results will be magnified. The increasing parameter and computation complexity of connected convolutional layers are addressed by the suggested de-Conv for decoupling serial convolutional layers. When $y'\_(i+1)$ features are supplied as input, D-Conv is modeled as follows:

$$y'_{i+1} = \beta_i \cdot \omega_{i+1}^{3\times3} \times y_{i+1} + (1-\beta_i) \cdot \omega_{i+1}^{1\times1} \times y_{i+1} \tag{5}$$

where y'_{i+1} is the output features of the (*i*+1)-*th*, De-conv $\omega_{i+1}^{3\times3}$ and $\omega_{i+1}^{1\times1}$ are the 3×3 and 1×1 convolutional layer. $\omega_{i+1}^{3\times3}$ is the started using the same values as the network that was going to be cut and $\omega_{i+1}^{1\times1}$ a was launched identity matrix β_i the learning parameter is initialized the identity matrix controls two parallel weights curved layers. Additionally, β_i must be 0 slope penalty, that will be explained in more depth. Then, it can be considered $y'_{i+1} = \omega_i^{3\times3} \times y_{i+1}$. Then 1×1 can be considered as convolutional layers can then be joined with one another while maintaining the prior layer's structure.

$$y'_{i+1} = \beta_i \cdot \omega_{i+1}^{3\times3} \times y_{i+1} + (1-\beta_i) \cdot \omega_{i+1}^{1\times1} \times y_{i+1} = \omega'_{i+1} \times x_{i+1} \tag{6}$$

where ω'_{i+1} are the updated convolutional layer's weights (*i*+1)-*th*.

Eq. (4) represents a specific activation function that depends on the sign (positive/negative) of the input, while Eq. (5) combines values obtained from two different sources to produce a unified output. Both equations play a significant role in the functioning of the model.

2) Equivalent layer-decoupled network merging

When $(\alpha_i, \beta_i) = 0$, the 1×1convolutional layer and an identify mapping, which is described as follows, make up the (*i*+1)-*th* mapping represented as:

$$y'_{i+1} = y_{i+1} \times \omega_{i+1}^{1\times1} \tag{7}$$

According to the linear combination of the convolutional function, the (*i*+1)-*th* layer is able to be concatenated with its predecessors. Transition layer equivalent calculation processes the integration is presented as below:

$$y'_{i+1} = (y_i \times \omega_i^{3\times3}) \times \omega_{i+1}^{1\times1} \tag{8}$$

Since the convolution function is linear, Eq. (9) can be represented as:

$$y'_{i+1} = x_i \times (\omega_i^{3\times3} \times \omega_{i+1}^{1\times1}) = y_i \times \omega_i'' \tag{9}$$

where ω_i'' is a connected convolutional layer. Therefore, computation two adjacent convolutional layers is one convolutional layer, the compression of the layer is accomplishing [24].

3) Gradient penalty

Gradient penalty to ensure that the deformation of parameters (α_i, β_i) is minimized, a slope penalty is applied, effectively driving these parameters to zero. In the initial phase, an uncompressed pre−trained model, denoted as F_u, is used. This model is then replaced by F_{de} where the network is initially disconnected. The parameters (α_i, β_i) are set to an initial value of 1, ensuring that F_{de} starts as an exact copy of F_u. To establish meaningful parameter groupings, adjacent Deconvolution (De-Conv) layers and Removed ReLU (Rem-ReLU) layers are paired together to form α_i, β_i pairs. The full set of these parameter pairs across NNN layers is denoted as (α_i, β_i), $i \in$ N.

During retraining, a loss function is employed similar to that used in standard, non-compressed models. Compression is applied over KKK layers, where a subset of QQQ parameter pairs with the smallest values of the update rule for model parameters is formulated as follows:

$$W' = \begin{cases} W - l \cdot G_0, (\alpha_i, \beta_i) \in M' \\ W - l \cdot G + \lambda \cdot sign(W), (\alpha_i, \beta_i) \notin M' \end{cases} \tag{10}$$

where W is parameters of F_{de}, W' for updated parameters, l is the learning rate and G is the gradient of the network. When $(\alpha_i, \beta_i) \in M'$, G_0 is (α_i, β_i) a fixed number gradually decreases to 0. When $(\alpha_i, \beta_i) \notin M'$, apply an additional slope penalty deformation to improve compression efficiency, where λ is the weight parameter and $sign(\cdot)$ is a code function.

$$sign(x) = \begin{cases} 1, x \geq 0 \\ 0, x < 0 \end{cases} \tag{11}$$

Non-linear activation layers (such as ReLU) are replaced by Rem-ReLU to enable the merging of adjacent convolutional layers. We have also clarified how De-Conv layers decouple stacked convolutional layers by redistributing their complexity, allowing for parameter-efficient compression.

Specifically:

- Rem-ReLU assists in eliminating redundant non-linearities to facilitate the combination of adjacent convolutional layers.

- De-Conv enables the division of convolution operations into easier sub-operations with tunable complexity.
- Decoupling is the disentanglement of sequential connections to enhance efficiency while maintaining the expressiveness of the model.

E. Comparative Performance Analysis with Traditional Models

The improved performance of the suggested model in comparison to conventional architectures like CNN, Recurrent Neural Network (RNN), LSTM, and BiLSTM is due to two main innovations: Knowledge Distillation (KD) and Efficient Layer Compression (ELC). In contrast to regular models that train deep networks from scratch, our model uses a pre-trained, high-capacity teacher model (ResNet-50) to teach a lighter student model. This procedure facilitates the learning of generalized feature representations and decision boundaries by the student model without the overhead of too many parameters. Additionally, the addition of a structured layer compression process—specifically the Rem-ReLU decoupling and De-Conv transformations—permits efficient computation without sacrificing performance. The methods eliminate redundant layers and non-linearities, paving the way for speeding up inference on edge devices while retaining the accuracy of the deep network. Conversely, other conventional models like CNNs or RNNs do not have these transfer and compression methods and tend to underfit or overfit in low-resource settings. Moreover, models such as LSTM and BiLSTM are specifically intended for sequential or time-series data and are less adapted to spatial image feature extraction tasks. Therefore, the union of Knowledge Distillation and ELC not only decreases computational expense but also improves classification accuracy, recall, and precision, particularly in difficult multiclass medical image datasets.

IV. Result and Discussion

Introduce the first design of the suggested model in this part. Common assessment criteria like as Accuracy, Precision, Recall, F1-Score, MSE, and RMSE were used to test the performance of the suggested model. Tests and comparisons with previous models reveal that the suggested model performs better than the others.

A. Description of the Data Set

The new model to detect medical images such as SARS-COVID-19, malaria and another kind of lung diseases, the new model will pre-process the input image with the process of normalization and data augmentation. It is due to the reason which has been employed in order to enhance the quality image to be employed in this research. SARS-COV-2 Ct-Scan Data set,

1) CT-Scan data set SARS-COV-2

The SARS-CoV-2 CT scan dataset is publicly available and consists of 2482 chest CT scan images divided into 1252 scans of SARS-CoV-2 positive patients and 1230 of SARS-CoV-2 negative patients. The data were acquired from actual patients at hospitals located in São Paulo, Brazil, and are meant to aid the development and testing of artificial intelligence methods for COVID-19 detection based on medical imaging. The dataset can be found at: https://www.kaggle.com/datasets/plamene duardo/sarscov2-ctscan-dataset

2) Chest CT-Scan images data set

Images in png or jpg format work well; dcm format is not appropriate. The three types of breast cancer that are covered in the data are adenocarcinoma, large cell carcinoma, squamous cell carcinoma, and one folder of normal cells. The data folder is the primary folder that houses the train, test, and validation files. Datasets/mohamedhanyyy/chest-ctscan-images https://www.kaggle.com

B. Performance Metrics

The proposed model performance evaluated using standard evaluation metrics such as Accuracy, Precision, Recall and F1-Score, RMSE, MAE, and MSE. These metrics are used to evaluate in proposed method.

1) Accuracy

In percentage terms, the accuracy and precision of the model are expressed as the ratio of corrected cases to total instances of the accuracy metric. The true and false rates in the equation roughly represent accuracy.

$$Accuracy = \frac{T.n + T.p}{T.n + F.p + F.p + T.p} \tag{12}$$

2) Precision

The ratio of the correctly classed instances or samples out of the ones classified as positives is captured by precision.

$$Precision = \frac{T.p}{T.p + F.p} \tag{13}$$

3) Recall

Recall is a method to quantify the number of correctly found items as compared to how many exist actually. The division of the number of positive samples identified correctly as positive by the number of all positive samples.

$$Recall = \frac{T.p}{T.p + F.n} \tag{14}$$

4) F1-Score

The F1-Score is a measure that will be utilized to assess the performance of machine learning. It combines *Precision* and *Recall* into one score.

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{15}$$

5) Root Means Square Deviation (RMSE)

The Root Means Square Deviation (RMSE), sometimes referred to as the root mean square error, is one of two closely related and commonly used metrics for measuring

the discrepancies between actual or anticipated values and observed values or an estimate.

$$RMSE = \sqrt{\frac{\sum_{i-1}^{n}(x_i - x_p)^2}{n}} \quad (16)$$

6) Mean Square Error (MSE)

Mean Square Error (MSE), the average of the squared differences between the observed values in a statistical study and the predicted values from a model.

$$MSE = \sqrt{\frac{\sum_{i-1}^{n}(x_i - x_p)^2}{n}} \quad (17)$$

C. Performance Evaluation

The effectiveness of the suggested approach is contrasted with that of the current approaches, including CNN, LSTM, RNN, and BiLSTM. Accuracy, precision, recall, F1-Score, RMSE, and MSE are metrics used to evaluate the performance of the suggested approach to the current one. The suggested approach performs well. The network's accuracy and loss curves show that the training and testing procedures are stable and rapidly converge. utilized in the data sets for chest CT scan images, and SARS-COV-2 Ct-Scan data. 300 epochs are employed in the training and testing procedure to analyze accuracy and loss, which results in improved performance.

D. Detail Hyperparameters

- Learning rate: 0.001.
- Optimizer: Adam.
- Batch size: 32.
- Number of epochs: 300.
- Activation functions: ReLU / Rem-ReLU.
- Loss function: Cross-entropy (with knowledge distillation loss components).

E. Hardware Setup

Training: Conducted on a workstation with an NVIDIA RTX 3080 GPU, 64GB RAM, and Intel i9 CPU.

Edge Testing: Deployment and inference tested on a Raspberry Pi 4 (4GB RAM) and NVIDIA Jetson Nano, simulating real-world edge environments.

We set T = 4, which was chosen based on preliminary validation experiments aimed at balancing softened probability distributions and stable convergence during training.

Training Time: The full training process for 300 epochs took approximately 2.5 h on the workstation setup.

Inference time on edge devices was ~180 ms per image, validating suitability for real-time diagnostics.

We set T = [insert value, e.g.,4], which was chosen based on preliminary validation experiments aimed at balancing softened probability distributions and stable convergence during training.

F. SARS-COV-2 CT-Scan Data Set

The comparison of performance of the evaluation metrics to the current model in SARS-COV-2ct-scan data set sample shown in Fig. 5. The current models such as CNN, RNN, LSTM, BiLSTM have limitation such as it doesn't apply to continuous or temporary data such as health records, too complicated and it hinder the performance. Therefore, these limitations of current method the proposed method address these limitations, that wise the proposed method achieve high performance such as Accuracy 98.93%, Precision 98.41%, Recall 98.69%, F1-Score 98.44%. Performance of evaluation metrics with respect to current model in SARS-COV-2 Ct-Scan Dataset, which is shown in Table I.

TABLE I. PERFORMANCE OF EVALUATION METRICS WITH RESPECT TO CURRENT MODEL IN SARS-COV-2 CT-SCAN DATA SET

Metric	CNN	LSTM	RNN	BiLSTM	Proposed KD Model
Accuracy	84%	88%	90%	96%	98.93%
Precision	85%	88%	91%	96%	98.41%
Recall	83%	89%	90%	96%	98.69%
F1-Score	84%	88%	90%	96%	98.44%

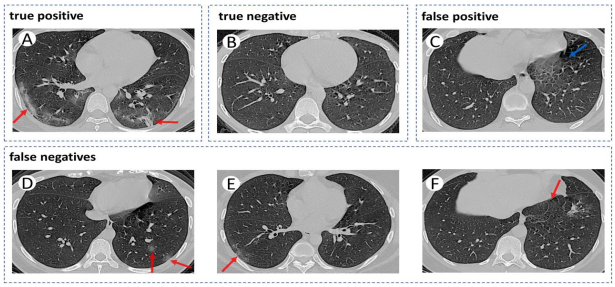

Fig. 5. Sample dataset.

G. Chest CT-Scan Images Data Set

The performance of evaluation metrics with respect to the current model in chest CT-Scan images dataset. The current models such as CNN, RNN, LSTM, BiLSTM these approaches have the low performance. The proposed approach has a high performance such as Accuracy 8.25%, Precision 98.78%, Recall 98.86%, F1-Score 98.14%. Performance of evaluation metrics with respect to existing model in chest CT-Scan images dataset, which is shown in Fig. 6.

Fig. 6 performance of evaluation metrics in comparison to the current model in the data set of chest CT-Scan images. The error rate in the data set of chest CT-Scan images as compared to the existing model. Current models with high error values include CNN, RNN, LSTM, and BiLSTM. Compared to the current model, the suggested method has a lower error rate. RMSE 0.13, MSE 0.094. Using the suggested data set of chest CT-Scan pictures, error measurements were compared to the current approach. ROC analysis of the suggested approach, confusion matrix, and accuracy and loss analysis in training and testing the data set of chest CT-Scan pictures. Figs. 6 and 7 and Table II displays a comparison table between the suggested approach and the current model for chest CT-Scan pictures.

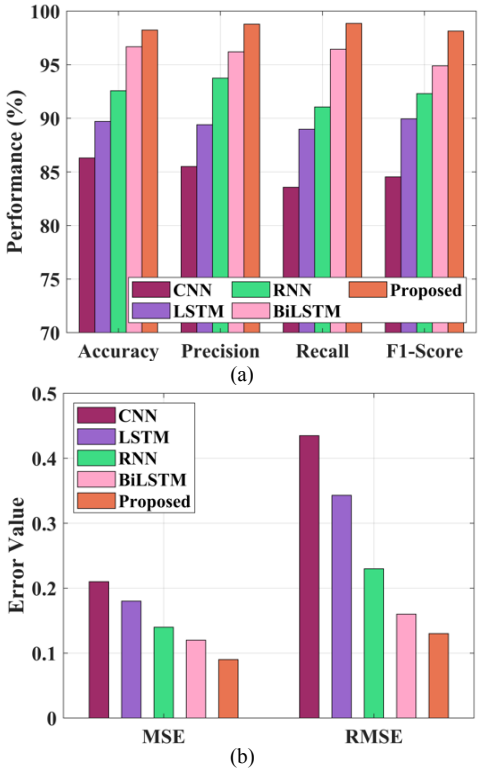

Fig. 6. Error metrics of compared with existing methods using proposed chest CT-Scan images dataset.

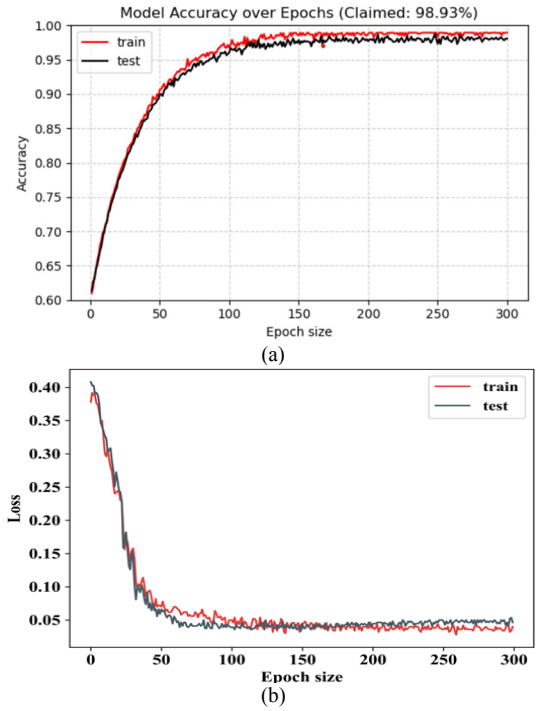

Fig. 7. Illustrates the accuracy trend during the training and validation phases on the chest CT-Scan image dataset.

The confusion matrix has two actual class and predicted class normal 110, adeno carcinoma 130, squamous cell 119, large cell carcinoma 132, the wrong prediction is squamous cell 2, and large cell 1, adeno carcinomas cell 2, and large cell 2, these are wrong predictions of these classes.

TABLE II. COMPARISON TABLE FOR PROPOSED METHOD AND EXISTING MODEL IN CHEST CT-SCAN IMAGES

Performance metrics	CNN	LSTM	RNN	BiLSTM	Proposed (KD model)
Accuracy (%)	86.3	89.71	92.58	96.68	**98.25**
Precision (%)	85.50	89.40	93.75	96.20	**98.78**
Recall (%)	83.57	88.99	91.06	96.4	**98.86**
F1-Score (%)	84.54	89.95	92.30	94.90	**98.14**
MSE	0.21	0.18	0.14	0.12	**0.09**
RMSE	0.435	0.343	0.23	0.16	**0.13**

Fig. 8 illustrates the performance of the deep learning model in classifying chest CT-Scans into four classes: Normal, Adeno, Squamous, and Large. The confusion matrix is highly accurate with limited misclassification. The MSE plot indicates successful training with consistent error reduction and no overfitting. The ROC plots affirm perfect separation of classes with an AUC of 1.00 for all classes.

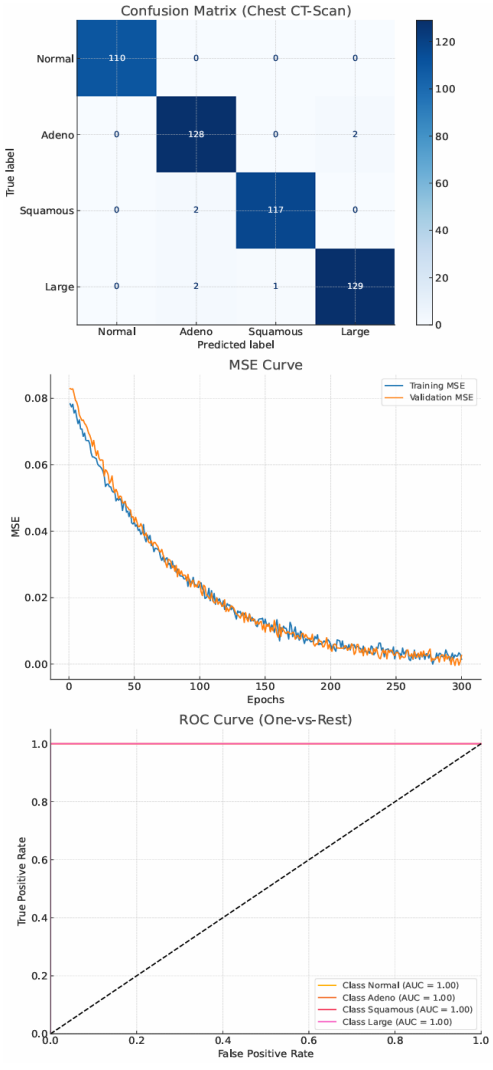

Fig. 8. Performance evaluation of chest CT-Scan classification model using Confusion Matrix, MSE curve, and ROC curve.

V. Discussion

In this section, a comprehensive description of the simulation system and the experimental result of the proposed approach is presented. The results are illustrated with detailed explanations based on various evaluation and validation tests. The proposed method has demonstrated high performance in different scenarios.

One of the common drawbacks in previous research is the cumulative analysis issue typically encountered during the segmentation of brain tumor images. This arises due to variations among individuals, which may be negligible in some cases [3]. The suggested technique addresses this drawback and offers improved analysis and segmentation accuracy.

Another limitation identified in earlier techniques is the incomplete establishment of effectiveness across various systems [4]. However, the proposed model demonstrates consistent performance across different system environments, ensuring better reliability.

Data scarcity remains a major challenge. Compared to other datasets, clinical data are more complex and difficult to partition, which poses a limitation in many methods [5]. Additionally, privacy concerns in medical data represent both a sociological and technical issue that must be addressed from multiple perspectives. The proposed method utilizes a larger dataset and incorporates mechanisms to preserve data privacy during comparisons and analysis.

A lack of standardization and difficulties in clinical integration are also noted limitations in earlier approaches [6]. In contrast, the standardization process proposed in this method is optimized and suitable for clinical settings, thereby facilitating integration.

Minimal data availability and potential overestimation are further concerns [7]. The proposed approach addresses these by employing a more extensive and representative dataset.

Another drawback in existing methods is related to the training and testing process, where some models are trained using only a single facial image. This can lead to covariant shift and training errors [8]. In the proposed method, the training and testing processes are robust and free from such errors.

Several techniques may suffer from procedures that negatively impact important aspects of model performance [25]. However, the proposed model maintains stable performance even in the presence of complex influencing factors.

Lastly, using small datasets can affect the generalizability of a model [26]. This limitation is effectively addressed in the proposed method, which uses a larger and more diverse dataset, enhancing the generalizability and practical applicability of the approach.

VI. Conclusion

The suggested compacted layer known ledge distillation model is compared in this work. Simulation results demonstrated the effectiveness of the suggested approach. The comparison is based on F1-Score, RMSE, MSE, Accuracy, Precision, and Recall. CT-Scan images make up the CT-Scan Data set SARS-COV-2. The proposed method utilizes a data set. The performance of the proposed model is evaluated and compared with the previous method. The evaluation measures performed as follows on the SARS-COV-2ct-scan dataset in comparison to the existing model: Accuracy 98.93%, Precision 98.41%, Recall 98.69%, F1-Score 98.44%, MSE 0.04, RMSE 0.16. CT-Scan images of the chest F1-Score: 98.144%, MSE: 0.094, RMSE: 0.13, Accuracy: 98.25%, Precision: 98.78%, Recall: 98.86% which is shown in Table III. In comparison to the current model, such results have a low error value and a high performance. Hybrid compression techniques may be used in the future to produce lighter DNN models in many contexts. One possibility is to use the pruning approach on the KD-generated student model. The final model is then subjected to quantization. An alternative possibility is to first refine the instructor model, then distill its knowledge into a student model, and lastly, quantify this student model. Additionally, different learning methods like adversarial and reinforcement learning can be used to examine the impact of KD.

TABLE III. Comparison the Proposed Model and State-of-Art Method

Technology	Accuracy (%)
X. Wang *et al*. [3]	90.1
CNN [4]	96
DLA [5]	97
IoT medical things and DL [6]	95
LLLT, MIoT [7]	97
CT image for cobvid-19 diagnosis [8]	97
Tiny UNet IN FKD-Med [25]	91
ENet-coco [9]	88
S. Wang *et al*. [10]	85.2
Proposed	**98.93, and 98.25 for SARS-COV-2, Chest CT-Scan images Data set**

Conflicts of Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Author Contributions

Conceptualization: Y.N.P., D.B., A.G.; Methodology: Y.N.P., D.B., A.G.; Software: Y.N.P.; Validation: D.B., A.G.; Formal Analysis: Y.N.P., A.G.; Investigation: D.B., A.G.; Resources: Y.N.P.; Data Curation: D.B.; Writing—Original Draft Preparation: Y.N.P.; Writing—Review and Editing: D.B., A.G.; Visualization: Y.N.P.; Supervision: Y.N.P.; Project Administration: Y.N.P. All authors had approved the final version.

ACKNOWLEDGMENTS

The authors would like to express their gratitude to the Department of CSE, Quantum University, Roorkee, India, for providing the necessary facilities and support for this research.

REFERENCES

[1] J. M. Rodríguez-Corral, J. Civit-Masot, F. Luna-Perejón *et al*., "Energy efficiency in edge TPU vs. embedded GPU for computer-aided medical imaging segmentation and classification," *Engineering Applications of Artificial Intelligence*, vol. 127, 107298, 2024.

[2] F. Tang, J. Ding, Q. Quan *et al*., "Cmunext: An efficient medical image segmentation network based on large kernel and skip fusion," in *Proc. 2024 IEEE International Symposium on Biomedical Imaging (ISBI)*, 2024, pp. 1–5.

[3] X. Wang, X. Deng, Q. Fu *et al*., "A weakly-supervised framework for COVID-19 classification and lesion localization from chest CT," *IEEE Transactions on Medical Imaging*, vol. 39, no. 8, pp. 2615–2625, 2020. doi: 10.1109/TMI.2020.2995965

[4] A. Tayal, J. Gupta, A. Solanki *et al*., "DL-CNN-based approach with image processing techniques for diagnosis of retinal diseases," *Multimedia Systems*, vol. 28, no. 4, pp. 1417–1438, 2022.

[5] M. Puttagunta and S. Ravi. "Medical image analysis based on deep learning approach," *Multimedia Tools and Applications*, vol. 80, no. 16, pp. 24365–24398, 2021.

[6] Y. Kumar, A. Koul, R. Singla, and M. F. Ijaz, "Artificial intelligence in disease diagnosis: a systematic literature review, synthesizing framework and future research agenda," *Journal of Ambient Intelligence and Humanized Computing*, vol. 14, no. 7 pp. 8459–8486, 2023.

[7] A. Naz, H. Khan, I. U. Din, A. Ali, and M. Husain, "An efficient optimization system for early breast cancer diagnosis based on internet of medical things and deep learning," *Engineering, Technology & Applied Science Research*, vol. 14, no. 4, pp. 15957–15962, 2024.

[8] D. T. Phan, Q. B. Ta, and C. D. Ly, "Smart low level laser therapy system for automatic facial dermatological disorder diagnosis," *IEEE Journal of Biomedical and Health Informatics*, vol. 27, no. 3, pp. 1546–1557, 2023.

[9] M. K. Hasan, S. Islam, R. Sulaiman *et al*., "Lightweight encryption technique to enhance medical image security on internet of medical things applications," *IEEE Access*, vol. 9, pp. 47731–47742, 2021.

[10] S. Wang, B. Kang, J. Ma *et al.*, "A deep learning algorithm using CT images to screen for Corona Virus Disease (COVID-19)," *Eur. Radiol.*, vol. 31, pp. 6096–6104, 2021. https://doi.org/10.1007/s00330-021-07715-1

[11] G. Sun, H. Shu, F. Shao *et al*., "FKD-Med: Privacy-aware, communication-optimized medical image segmentation via federated learning and model lightweighting through knowledge distillation," *IEEE Access*, vol. 12, pp. 33687–33704, 2024.

[12] W. Zou, X. Qi, Z. Wu *et al*., "Coco distillnet: A cross-layer correlation distillation network for pathological gastric cancer segmentation," in *Proc. 2021 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, 2021, pp. 1227–1234.

[13] N. Ajlouni, A. Özyavaş, M. Takaoğlu *et al*., "Medical image diagnosis based on adaptive Hybrid Quantum CNN," *BMC Medical Imaging*, vol. 23, no. 1, p. 126, 2023.

[14] L. Kong, M. Huang, L. Zhang, and L. W. C. Chan, "Enhancing diagnostic images to improve the performance of the segment anything model in medical image segmentation," *Bioengineering*, vol. 11, no. 3, p. 270, 2024.

[15] K. A. Kadhim, F. Mohamed, F. H. Najjar, and G. A. Salman, "Early diagnose Alzheimer's disease by convolution neural network-based histogram features extracting and Canny edge," *Baghdad Sci. J.*, vol. 21, no. 2 (SI), p. 0643, 2024.

[16] S. Iqbal, T. M. Khan, S. S. Naqvi *et al.*, "Ldmres-Net: A lightweight neural network for efficient medical image segmentation on IoT and edge devices," *IEEE J. Biomed. Health Inform.*, vol. 28, no. 7, pp. 3860–3871, 2023.

[17] B. Lv, F. Liu, Y. Li *et al.*, "Artificial intelligence-aided diagnosis solution by enhancing the edge features of medical images," *Diagnostics*, vol. 13, no. 6, p. 1063, 2023.

[18] S. Chauhan, D. R. Edla, V. Boddu *et al.*, "Detection of COVID-19 using edge devices by a light-weight convolutional neural network from chest X-ray images," *BMC Med. Imag.*, vol. 24, no. 1, p. 1, 2024.

[19] J. Wong, J. Nerbonne, and Q. Zhang, "Ultra-efficient edge cardiac disease detection towards real-time precision health," *IEEE Access*, vol. 12, pp. 9940–9951, 2023.

[20] M. K. Hasan, S. Islam, R. Sulaiman *et al.*, "Lightweight encryption technique to enhance medical image security on internet of medical things applications," *IEEE Access*, vol. 9, pp. 47731–47742, 2021.

[21] M. Hu, Y. Zhong, S. Xie, H. Lv, and Z. Lv, "Fuzzy system based medical image processing for brain disease prediction," *Front. Neurosci.*, vol. 15, 714318, 2021.

[22] S. Iqbal, T. M. Khan, S. S. Naqvi, M. Usman, and I. Razzak, "LDMRes-Net: Enabling efficient medical image segmentation on IoT and edge platforms," arXiv Print, arXiv: 2306.06145, 2023. https://doi.org/10.48550/arXiv.2306.06145

[23] N. Nigar, A. Jaleel, S. Islam, M. K. Shahzad, and E. A. Affum, "IoMT meets machine learning: From edge to cloud chronic diseases diagnosis system," *J. Healthcare Eng.*, vol. 2023, 9995292, 2023.

[24] S. Francy and R. Singh, "Edge AI: Evaluation of model compression techniques for convolutional neural networks," arXiv Print, arXiv:2409.02134, 2024. https://doi.org/10.48550/arXiv.2409.02134

[25] N. El-Rashidy, A. Sedik, A. I. Siam, and Z. H. Ali, "An efficient edge/cloud medical system for rapid detection of level of consciousness in emergency medicine based on explainable machine learning models," *Neural Computing and Applications*, vol. 35, no. 14, pp. 10695–10716, 2023.

[26] H. Ulutas, M. E. Sahin, and M. O. Karakus, "Application of a novel deep learning technique using CT images for COVID-19 diagnosis on embedded systems," *Alexandria Engineering Journal*, vol. 74, pp. 345–358, 2023.