GDenseMNet: Global Dense Multiscale Feature Learning Network for Efficient COVID-19 Detection in CT Images

Amogh Manoj Joshi
Dept. of Electronics and Telecom Engineering
V. E. S. Institute of Technology
Mumbai, India
joshiamogh9@gmail.com

Deepak Ranjan Nayak
Dept. of Computer Science and Engineering
Malaviya National Institute of Technology Jaipur
Jaipur, India
drnayak.cse@mnit.ac.in

***Abstract*—Accurate and rapid diagnosis of COVID-19 is crucial for curbing its fast spread across the globe, with constant mutations leading to newer variants. Recent studies have exhibited that chest CT scans manifest clear radiological findings for the COVID-19 infected patients. Convolutional neural networks (CNN) have been used considerably for COVID-19 diagnosis; however, most CNN architectures demand a huge amount of parameters, resulting in overfitting on limited training data and a slower inference. Further, residual and densely connected neural networks such as ResNet and DenseNet have been proven to strengthen feature extraction and feature propagation but fail to fully discover both local and global representations. Moreover, few linearly stacked networks fall short in capturing and preserving multiscaled features from various receptive fields. This paper proposes a new CNN architecture called global dense multiscale feature learning network (GDenseMNet) for COVID-19 detection from CT images that effectively incorporates global dense connections while capturing multiscaled features. The GDenseMNet model comprises multiscale local feature extraction (MLF) blocks that capture local features of various size receptive fields using multiple filters and residual skip connections. The global dense connections between these blocks further enable global feature learning capability. The proposed architecture is lightweight, end-to-end learnable, and validated using the SARS-CoV-2 CT-Scan dataset. Experimental results demonstrate that the GDenseMNet model achieves promising detection performance compared to state-of-the-art CNN approaches and hence, it can be utilized as an effective tool real-time COVID-19 diagnosis.**

***Index Terms*—CNN, GDenseMNet, MLF block, COVID-19, CT Scan**

I. Introduction

The COVID-19 virus which was declared a pandemic by the World Health Organization on 11th March 2020 has been infecting large masses of people since the last two years. This virus has infected around 521 million people and caused 6.27 million deaths across the globe. The major symptoms of COVID-19 are dry cough, fever, sore throat and fatigue. But recently, the rise in the number of asymptomatic cases has put a heavy strain on the medical infrastructure and government administration, making it difficult to detect and trace the COVID positive patients in time. The highly mutating nature of this virus has resulted in the development of various variants of the virus till date. Among these, Beta, Gamma, Delta and Omicron are the four variants of concern declared by WHO. The Omicron variant is currently spreading at a faster rate and is known to be much more transmissible than the earlier variants. Hence, an accurate and rapid diagnosis is required to cease the further spread of this transmissible variant. The current testing standard used is based on reverse transcription-polymerase chain reaction (RT-PCR) which is performed on swab samples obtained from the patients [1]. This test takes 4-6 hours to provide results. Moreover, the accuracy of this test depends on several factors like the time elapsed from the point when the person was infected.

Recent studies have shown that chest computed tomography (CT) scans have the potential to be used as an alternative and efficient testing method since they manifest clear radiological findings of COVID-19 positive patients [1], [2]. The CT scans exhibit a clear blend of multifocal peripheral lung changes of ground-glass opacity (GGO) and consolidation which distinctly highlights COVID-19 infections in lungs even at an early stage of infection, thus helping in the timely detection of the virus [3], [4]. Thus, the CT scans can be used along with RT-PCR tests as an early screening measure for preliminary diagnosis of the virus which would enable instant detection of the virus in the patients and can be further verified by an RT-PCR test. However, the medical experts require time to deeply analyze the CT scans and diagnose every case and hence can't be burdened with the heavy load of test CT scans. Therefore, an automated system that can accurately and quickly detect COVID-19 virus from the CT scans needs be designed to handle the current outbreak of the virus.

Deep Learning techniques, particularly convolutional neural networks (CNN) have been increasingly employed for COVID-19 detection from chest X-Rays and CT scans, owing to their superior feature extraction capability, resulting in a high performance [5]. Despite the superior performance, a major drawback of CNNs is that most of the models require huge amount of training data to learn the prominent features for classification. But the availability of limited amount of CT

This work is partially supported by the Science and Engineering Research Board, Department of Science and Technology, Govt. of India under project No. SRG/2020/001460.

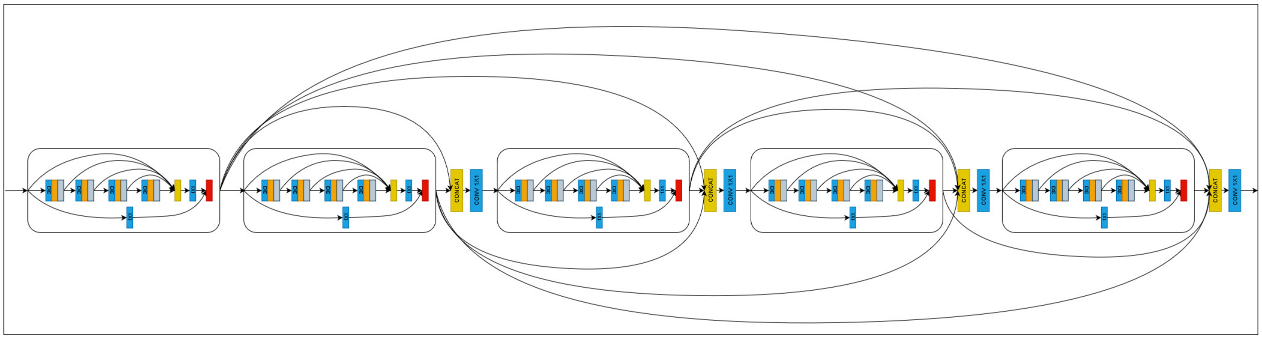

Fig. 1. Illustration of proposed multiscale local feature learning (MLF) blocks with global dense connections. The inner rectangular boxes indicate MLF blocks and the outer rectangular box represents global dense feature extraction (GDF) block. The MLF blocks facilitate learning local multiscaled features of various size receptive fields, while GDF block effectively uses these features through dense connections to learn global representations from chest CT images.

image data poses a challenge in achieving a good classification performance. Moreover, a majority of the CNN models require large amounts of parameters to achieve a satisfactory performance which results in a slower inference time while testing. Such heavy models are not suitable for real-time COVID-19 diagnosis in the scenario where obtaining quick and accurate results is highly essential. Further, it has been observed that many studies employed ImageNet pretrained CNN models such as ResNet and DenseNet which fail to capture both local and global representations of CT images [3], [6].

To address the above issues, in this paper, we present a lightweight global dense multiscale feature learning CNN which effectively extracts both local and global properties even in the presence of limited CT data as shown in Fig. 1. Despite a lightweight architecture, the proposed model outperforms most of the contemporary pretrained CNN models and state-of-the-art COVID-19 detection methods. The important contributions of this paper can be summarized as:

- We propose a lightweight end-to-end trainable CNN called GDenseMNet which has multiscale local feature extraction (MLF) blocks and global dense connections for effectively learning local and global features.
- In GDenseMNet, we propose MLF blocks which extract local multiscaled features from receptive fields of various sizes and incorporate residual skip connections within the blocks to strengthen the feature flow.
- We also propose global dense feature extraction (GDF) block which forms the core component of our proposed model architecture. The GDF block consists of five MLF blocks densely connected with each other effectively utilizing the extracted local features by MLF blocks to learn the global representations from CT data.
- We test the efficacy of GDenseMNet through an extensive set of experiments on a publicly available SARS-CoV-2 CT-Scan dataset.
- We compare the performance of the proposed GDenseMNet with recent ImageNet pretrained CNN models and other existing methods in terms of classification accuracy, computational cost and memory.

II. Related Works

A lot of studies have shown that chest CT scans manifest clear radiological findings of COVID-19 [1], [7]. This led to numerous investigations for automated diagnosis of COVID-19 from medical images like X-rays and CT Scans. A majority of these studies used deep learning for classifying COVID-19 infected patients. Xu et al. [8] developed a deep learning based early diagnosis system for distinguishing COVID-19 from pneumonia and normal cases with an accuracy of 86.7%. Few similar studies have been made in [3] and [9] to diagnose COVID from CT scans. Hasan et al. [6] used ResNet50 and Xception to design a multi encoder ensemble network for COVID-19 diagnosis in CT scans. Kaur et al. [10] fused the last layer of a pretrained MobileNetv2 and ResNet50 for generating the feature vector and then fed it to SVM for COVID-19 classification. Pathak et al. [11] proposed a deep bidirectional long short-term memory (LSTM) network with mixture density network for COVID-19 CT scan classification. Recently, Kaur et al. [12] extracted features from pretrained MobileNetv2 architecture and proposed a parameter free BAT optimization based fuzzy K-nearest neighbor classifier (PF-FKNN) for improved detection performance. Foysal et al. [13] proposed an ensemble network of three CNN models and used ensemble hard voting for COVID-19 detection from Chest CT scans. Wang et al. [14] proposed a redesigned COVID-Net architecture and implemented it with a contrastive cross-site learning strategy to tackle the data heterogenity across different datasets. Madan et al. [15] designed a triplet network using few-shot learning for COVID-19 diagnosis using only few CT images. The aim of this work was to obtain the best performance from a very limited training data.

All these studies either used customized CNN models with large model size or used pretrained models with transfer learning which are computationally expensive and need large memory space. Further, many models do not focus on learning multiscale features which are essential for effectively capturing the minute COVID-19 lesions. Hence, in this study, we aim to design a CNN model which learns multiscale features from the limited chest CT images while retaining the light model size. Further, residual and densely connected CNN models

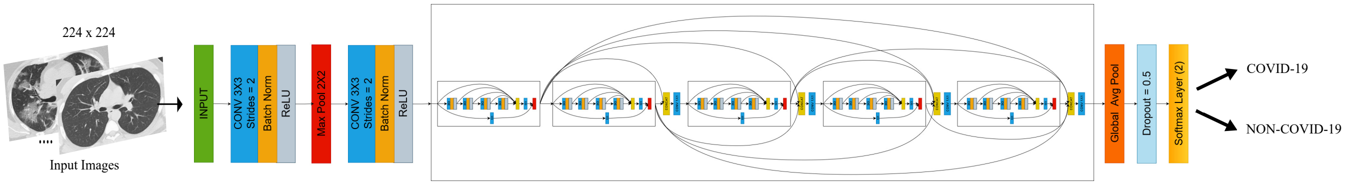

Fig. 2. Architecture of the proposed global dense multiscale feature learning network (GDenseMNet). It consists of two main blocks: MLF and GDF block. A global average pooling (GAP) layer followed by a classification layer is finally used to classify the image as COVID-19 or Non-COVID-19.

like ResNet and DenseNet have been extensively investigated due to their capability of extracting hierarchical features, however, these models fail to fully explore both local and global representations of the input data. Thus, to handle the above issues, this paper proposes a lightweight global dense multiscale feature learning network which effectively explores both local and global features from the CT images.

III. Proposed Methodology

In this section, we first discuss the need for multiscale feature learning in CNN and then outline the advantages of lightweight architecture. We further explain each component of our proposed model and its architectural layout.

A. Need for MultiScale Feature Learning and Lightweight Architecture

ImageNet pretrained models like VGGNet [16], ResNet [17], etc., and a majority of the CNN based state-of-the-art methods incorporate linearly connected single-chained convolution layers, stacked one below the other. Such networks are only capable of extracting homogeneously scaled features and fall short in extracting and preserving multiscale features which are highly effective for complex image classification tasks like COVID-19 classification, where the lesions vary in size [18]. In such cases, the vital discriminative features are ignored which leads to a poor generalization performance. Some recent works have proven the effectiveness of having multiscaled convolutional filters in the CNN architecture [18], [19]. Multiscaled filters enables the model to learn discriminant features, thereby improving the performance of the model to a large extent. Hence, we incorporate multiscale feature learning in the proposed GDenseMNet model.

Although CNN models with large parameters have shown their success in many image classification tasks, they are computationally expensive and demand quite a large memory. Moreover, such models often face an issue of overfitting when trained on small training data like COVID-19 CT data. They also take larger inference time during testing. Hence, these models may not be feasible as far as real-time deployment is concerned. Therefore, we aim at developing a fairly lightweight CNN architecture which learns discriminable features from the limited chest CT dataset.

B. Proposed GDenseMNet Architecture

The GDenseMNet is an end-to-end CNN model which uses multiscale local feature learning blocks and global dense connections, thereby preserving both the local and global features. The model is inspired by DenseNet [20] and ResNet [17] architecture and attempts to overcome some of their shortcomings. In ResNet, the residual skip connections are used mainly to solve the vanishing gradient problem and also to preserve the features throughout each block, resulting in a better local feature extraction. But, it fails to capture multiscale features. In DenseNet, the dense blocks facilitate short connections between all the layers which strengthens the feature flow and minimizes information loss between layers. But, in both these networks, the local features extracted by individual blocks have not been utilized effectively to capture and preserve the global properties. Hence, they fail to learn important global representations. Further, DenseNet captures redundant multiscale features due to multiple fusion of feature maps through concatenation, thereby leading to high computational cost. To solve these issues, our model incorporates the multiscale local feature learning with skip connections and global dense connections and further utilizes them effectively to fully explore local and global features from CT images. The model architecture comprises of two major blocks: multiscale local feature extraction (MLF) block and global dense feature extraction (GDF) block as shown in Fig. 2. Each block along with the overall architecture are explained below.

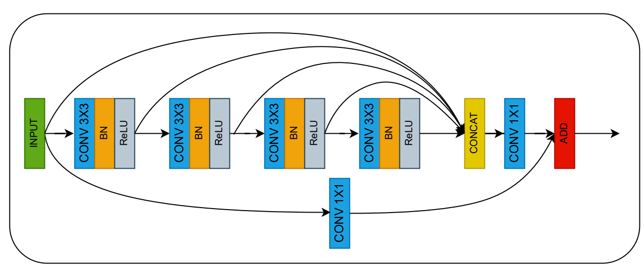

Fig. 3. Proposed multiscale local feature extraction (MLF) block architecture.

1) Multiscale Local Feature Extraction (MLF) Block: Fig. 3 shows the architectural layout of the proposed MLF block. The MLF block is a multiscale feature extraction module which is repeated five times throughout the GDenseMNet architecture. The purpose of multiscale feature extraction is to enable the model to learn different features captured from various receptive fields (RFs), thereby enhancing its local feature learning capability. These multiscale features can be captured using filters of different sizes. However, larger filter size leads to higher parameters and the objective of MLF block is to enable multiscale feature learning along with maintaining a light size in terms of parameters.

TABLE I
PARAMETERS REQUIRED FOR A CONVOLUTION LAYER BY VARYING NUMBER OF FILTERS WITH DIFFERENT SIZES

Kernel Size	Number of Filters			
	16	32	64	128
3×3	448	896	1792	3584
5×5	1216	2432	4864	9728
7×7	2368	4736	9472	18944

Generally, the parameters required by a convolutional (CONV) layer by varying number of filters and kernel sizes are shown in Table I. It can be seen that a CONV layer of 7×7 kernel size takes more than four times the parameters taken by a CONV layer of 3×3 kernel size. So, in order to maintain a lightweight architecture, we used CONV layers of 3×3 kernel size in such a way that it captures the same features of 5×5, 7×7 and 9×9 kernels. For instance, two CONV layers of 3×3 used one after another to collectively extract the same features which a single CONV layer of 5×5 extracts. Moreover, using CONV layers with 3×3 kernel size multiple times reduces the parameters drastically than using a single CONV layer of kernel size 5×5, 7×7 or 9×9. Hence, we used CONV layers with only 3×3 kernel size throughout the MLF block in a way which enabled efficient multiscale feature extraction along with maintaining minimal parameters.

The MLF block consists of four CONV layers connected sequentially, each followed by a batch normalization (BN) and a ReLU activation layer. The feature map obtained after the first CONV layer has a receptive field (RF) of 3×3, whereas the feature map after second CONV layer has a RF of 5×5 as explained earlier and so on. The outputs from all the four CONV layers are concatanated along with the input which results in feature maps containing features extracted from an RF of size 3×3 to 9×9. The output of this concatanation is then passed to a CONV layer of 1×1 kernel size for feature map depth reduction. Finally, a skip connection from the input via a 1×1 CONV layer is added with the stream to reinforce the feature flow and reduce vanishing gradient problem. The functioning of MLF Block can be mathematically explained as follows:

Let $\varphi^{f,f,N}(I)$ represent convolution operation with N number of kernels of size $f \times f$, η denote the BN function and α denote the ReLU activation function. Let $\delta_N(I)$ represent an operation of CONV layer followed by a BN and ReLU layer. It can be mathematically represented as

$$\delta_N(I) = \alpha(\eta(\varphi^{3,3,N}(I))) \tag{1}$$

Let a chain of such k $\delta_N(I)$ operations be represented by $\delta_N^k(I)$ such that

$$\delta_N^k(I) = \delta_N\left(\delta_N ...(\delta_N(I))\right) \quad [k\ times] \tag{2}$$

Let $\odot$ denote concatenation operation and $\oplus$ denote addition operation. The output of MLF block ($\beta_{MLF}^N(I)$) on input I with N filters and can be expressed as

$$\beta_{MLF}^N(I) = \varphi^{1,1,N}(I \odot \gamma(I)) \oplus \varphi^{1,1,N}(I) \tag{3}$$

where,

$$\gamma(I) = \delta_N^1(I) \ \odot \ \delta_N^2(I) \ \odot \ \delta_N^3(I) \ \odot \ \delta_N^4(I) \tag{4}$$

Thus, the output of the MLF block contains a perfect amalgamation of local multiscale features using only 3×3 kernels along with maintaining a lightweight architecture.

2) Global Dense Feature Extraction (GDF) Block: GDF block forms the main component of our GDenseMNet architecture which enables the effective local and global feature extraction. The GDF block uses five MLF blocks densely connected with each other for global feature learning using the locally extracted features from individual MLF blocks. The output of the MLF block is concatenated with the outputs of all the previous MLF blocks similar to dense connections in DenseNet architecture, but with the difference that we densely connect MLF blocks with each other instead of single CONV layers in DenseNet as shown in Fig. 1.

3) Overall Architecture: The input image is first passed through two CONV layers with stride (s) 2 followed in between by a max pooling layer of kernel size 2×2. This setup initiates the feature extraction process along with subsequently reducing the feature map size to a fair extent. The output of these blocks are given as input to the GDF block. The output feature maps of GDF block contains a fusion of local and global features which is then followed by a global average pooling (GAP) layer and a dropout layer with a value of 0.5. At the end, a final classification layer of two neurons is introduced with softmax activation for classifying the images as COVID-19 or Non-COVID-19. The architecture of GDenseMNet is summarized in Table II.

TABLE II
GDENSEMNET ARCHITECTURE SUMMARY

Layers	Description	Output Size
CONV Layer	$[3 \times 3 \text{ conv}] \times 1$ s=2	$112 \times 112 \times 8$
BN Layer		$112 \times 112 \times 8$
ReLU Layer		$112 \times 112 \times 8$
Pooling	2 max pool, $s = 2$	$56 \times 56 \times 8$
CONV Layer	$[3 \times 3 \text{ conv}] \times 1$ $s = 2$	$28 \times 28 \times 16$
BN Layer		$28 \times 28 \times 16$
ReLU Layer		$28 \times 28 \times 16$
GDF Block (5 MLF blocks)	$\begin{bmatrix} [1 \times 1 \text{ conv}] \times 2 \\ [3 \times 3 \text{ conv}] \times 4 \\ [1 \times 1 \text{ conv}] \times 4 \end{bmatrix} \times 5$	$28 \times 28 \times 256$
Pooling	Global Average Pool	$1 \times 1 \times 256$
Final Classification Layer	2-d fc, softmax	$1 \times 1 \times 2$

IV. EXPERIMENTS AND RESULTS

In this section, we present the implementation details, dataset and performance evaluation metrics used, and the experimental results of the proposed model. Further, we compare the proposed model with ImageNet pretrained models along with the existing approaches.

A. Dataset

To verify the effectiveness of the proposed GDenseMNet, we used a publicly available SARS-CoV-2 CT-Scan dataset [21] that contains 1252 CT scans from 60 COVID-19 positive patients and 1230 CT scans obtained from 60 patients non-infected by COVID-19. In this dataset, the images vary in height and width from 104-484 and 153-416 pixels respectively. All the images were hence resized to 224×224 pixels. For a fair comparison, we followed the same data split in our experiments as reported in [21]. We further divided the Train set into train and validation sets in 80:20 ratio, respectively. The dataset description is summarized in Table III.

TABLE III
DESCRIPTION OF SARS-COV-2 CT-SCAN DATASET

Set	COVID-19	NON-COVID-19
Train	802	788
Validation	200	196
Test	250	246
Total Images	1252	1230

Though this dataset has been widely used for COVID-19 detection, its major drawback is the limited number of samples. To deal with this issue and prevent overfitting, each image in the training set was augmented with four types of transformations such as Gaussian noise (with mean and standard deviation 0 and 0.01, respectively), horizontal flip, anticlockwise rotation of degree 5^o, and clockwise rotation of degree 5^o.

B. Implementation Details and Evaluation Metrics

The proposed GDenseMNet takes input CT images of resolution 224×224. The final classification layer of the model has two neurons with softmax activation. The categorical cross-entropy was used as the loss function during the entire training process. The mini-batch size was set to 32. The model was trained for 50 epochs with a learning rate of 0.001 decayed by a factor 0.5. All experiments were conducted in Kaggle Notebooks with NVIDIA K80 GPU with 12 GB RAM. The methods were implemented using Keras framework with Tensorflow as backend.

To evaluate the proposed model as well as other existing models, we used several evaluation metrics such as Accuracy (Acc), F1-score, AUC, Sensitivity (Sens), Specificity (Spec) and Precision (Prec).

C. Evaluation of Proposed Model

The training curves obtained by our model are shown in Fig. 4. It can be observed that the model is converged well within 50 epochs. As shown in Table IV, the model achieved an F1-score of 98.99%, AUC score of 99.00%, specificity of 100.00%, sensitivity of 98.00% and precision of 100.00%. Fig. 5 depicts the confusion matrix obtained by our model. It can be seen that five COVID-19 positive images out of 250 were wrongly classified, while all COVID-19 negative images were correctly classified.

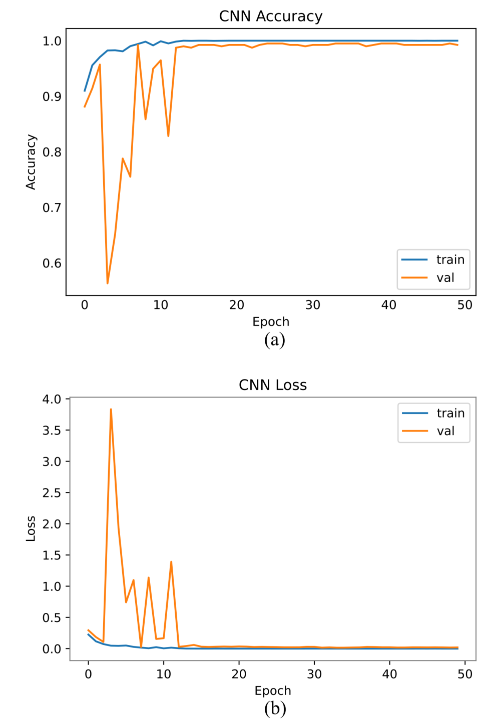

Fig. 4. Training curves on SARS-CoV-2 CT-Scan Dataset: (a) Accuracy vs Epoch and (b) Loss vs Epoch

TABLE IV
RESULTS OF PROPOSED MODEL ON SARS-COV-2-CT-SCAN DATASET

Metric	Value
Accuracy	98.99
F1-score	98.99
AUC Score	99.00
Specificity	100.00
Sensitivity	98.00
Precision	100.00

D. Comparison with ImageNet Pretrained CNN Models

To demonstrate the efficacy of GDenseMNet, we compared its performance with ImageNet pretrained CNN architectures by finetuning them on the dataset used in our study. The pretrained models included VGG-16 [16], ResNet-101 [17], DenseNet-121 [20], MobileNet [23], and Inception-V3 [24].

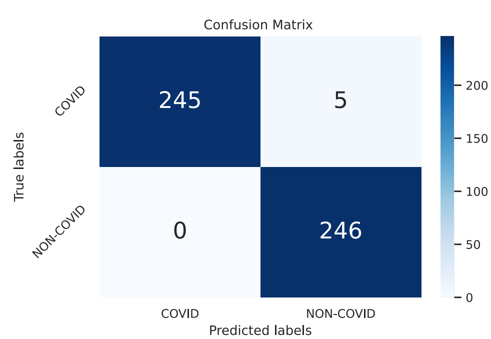

Fig. 5. Confusion matrix obtained by our GDenseMNet

TABLE V
COMPARISON OF GDENSEMNET WITH STATE-OF-THE-ART CNN ARCHITECTURES IN TERMS OF PARAMETERS AND SIZES

Model	**Parameter (M)**	**Size (MB)**
VGG-16	134.26	512.27
ResNet-101	42.66	163.68
ResNet-50	23.53	90.48
Inception-V3	21.77	84.00
DenseNet-121	6.95	27.94
MobileNet	3.23	13.44
Our GDenseMNet	2.1	9.33

TABLE VI
CLASSIFICATION PERFORMANCE COMPARISON WITH IMAGENET PRETRAINED CNN MODELS ON SARS-COV-2-CT-SCAN DATASET

Model	**Acc**	**F1**	**AUC**	**Spec**	**Sens**	**Prec**
VGG-16	98.79	98.78	98.79	99.19	98.40	99.19
ResNet-101	97.78	97.76	97.78	97.97	97.60	97.99
ResNet-50	98.79	98.79	98.80	100	97.60	100
Inception-V3	97.98	98.00	98.00	100	96.00	100
DenseNet-121	98.79	98.78	98.79	99.19	98.40	99.19
MobileNet	98.79	98.78	98.79	97.63	100	97.56
Our GDenseMNet	**98.99**	**98.99**	**99.00**	**100**	**98.00**	**100**

TABLE VII
PERFORMANCE COMPARISON WITH EXISTING COVID-19 DETECTION METHODS ON SARS-COV-2 CT-SCAN DATASET

Reference	**Method**	**Acc (%)**	**F1 (%)**
Wang et al. [14]	Redesigned COVID-Net	90.83	90.87
Angelov et al. [21]	XDNN	97.38	97.31
Angelov et al. [21]	VGG-16	94.96	94.97
Foysal et al. [13]	Ensemble CNN	96.00	95.60
Pathak et al. [11]	DBM	97.23	97.89
Pathak et al. [11]	DBM + MADE	98.37	98.14
Kaur et al. [10]	MobileNetv2 + ResNet50 and SVM	98.35	98.41
Sen et al. [22]	CNN + SVM	98.39	98.00
Our model	GDenseMNet	**98.99**	**98.99**

At first, a comparison was made on model size, specifically in the context of model parameters (in millions) and memory space required (in MB) as listed in Table V. It can be seen that few ImageNet pretrained models are extremely heavy like VGG-16 with 134.26M and ResNet-101 with 42.66M parameters as compared to our proposed model which requires just 2.1M parameters. Our model occupies a minute amount of memory space as compared to others and thus, is more suitable for real-time COVID-19 diagnosis using CT images. Table VI shows the classification results of the ImageNet pretrained models compared with our proposed model. It can be observed that our model achieves comparable or better performance than most models on SARS-CoV-2 CT-Scan dataset.

Although our model was inspired by ResNet and DenseNet, we also consider other ImageNet models for comparison. It can be seen that GDenseMNet outperformed all the models. Specifically comparing with ResNet and DenseNet, our model achieves better performance inspite of having a lightweight architecture (3 times lighter than DenseNet121 and 21 times lighter than ResNet101). Also comparing with Inception-V3 which learns multiscale features, our model achieves better performance indicating a better feature learning capability. Also, comparison was made with MobileNet since it is known specifically for its lightweight architecture. Our model inspite being lighter than MobileNet, achieved better results. It is worth mentioning here that all the ImageNet models were implemented under similar experimental set up for fair comparison purposes.

E. Comparison with Existing Methods

Table VII shows a comparison with the existing approaches [10]–[14], [22], [25] on SARS-CoV-2 CT-Scan dataset. It can be seen that the proposed model outperforms all the existing methods. Some works though used multiple models and ensemble approaches [10], [11], but could not able to achieve higher performance. Moreover, these models led to more computational cost. Further, the methods proposed by Sen et al. [22] and Kaur et al. [10] are not end-to-end trainable. The superior performance of our proposed model on both the datasets can be majorly attributed to its multiscale feature learning capability coupled with effective global dense connections, thus proving its robustness and efficacy.

V. CONCLUSION

In this paper, we proposed a global dense multiscale feature learning network called GDenseMNet for COVID-19 detection from CT Scans. The proposed model uses residual skip connections and dense connections to effectively learn both the local and global features. Further, GDenseMNet facilitates multi-scale feature learning by effectively capturing and preserving feature maps from various receptive fields without inducing redundancy in the feature flow. Moreover, the proposed model is lightweight and end-to-end trainable. The GDenseMNet was compared with several ImageNet pretrained models along with existing DL based state-of-the-art COVID-19 detection methods on the SARS-CoV-2 CT-Scan dataset. Experimental results indicated that our model achieved comparable or better classification performance with much less number of model parameters and memory space which makes it suitable for real-time COVID-19 diagnosis. However, in future, some studies are yet to be explored like effect of number of MLF blocks used in GDF block, type of convolutions used, etc. The effectiveness of GDenseMNet could be verified using a large-scale medical dataset. In addition, the concept of MLF and GDF blocks can be explored in vision transformer based models for further performance improvement. The proposed model, having achieved promising performance can prove to be an effective aid to the healthcare system even in case of any further outbreaks of the COVID-19 virus.

REFERENCES

[1] D. Dong, Z. Tang, S. Wang, H. Hui, L. Gong, Y. Lu, Z. Xue, H. Liao, F. Chen, F. Yang *et al.*, "The role of imaging in the detection and management of COVID-19: a review," *IEEE Reviews in Biomedical Engineering*, 2020.

[2] X. Li, W. Zeng, X. Li, H. Chen, L. Shi, X. Li, H. Xiang, Y. Cao, H. Chen, C. Liu *et al.*, "CT imaging changes of corona virus disease 2019 (COVID-19): a multi-center study in southwest china," *Journal of Translational Medicine*, vol. 18, no. 1, pp. 1–8, 2020.

[3] X. He, X. Yang, S. Zhang, J. Zhao, Y. Zhang, E. Xing, and P. Xie, "Sample-efficient deep learning for COVID-19 diagnosis based on CT scans," *MedRxiv*, 2020.

[4] S.-H. Wang, D. R. Nayak, D. S. Guttery, X. Zhang, and Y.-D. Zhang, "COVID-19 classification by ccshnet with deep fusion using transfer learning and discriminant correlation analysis," *Information Fusion*, vol. 68, pp. 131–148, 2021.

[5] H.-C. Shin, H. R. Roth, M. Gao, L. Lu, Z. Xu, I. Nogues, J. Yao, D. Mollura, and R. M. Summers, "Deep convolutional neural networks for computer-aided detection: Cnn architectures, dataset characteristics and transfer learning," *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1285–1298, 2016.

[6] M. Hasan, M. Alam, M. Elahi, E. Toufick, S. Roy, S. R. Wahid *et al.*, "CVR-Net: A deep convolutional neural network for coronavirus recognition from chest radiography images," *arXiv preprint arXiv:2007.11993*, 2020.

[7] F. Shi, J. Wang, J. Shi, Z. Wu, Q. Wang, Z. Tang, K. He, Y. Shi, and D. Shen, "Review of artificial intelligence techniques in imaging data acquisition, segmentation and diagnosis for COVID-19," *IEEE Reviews in Biomedical Engineering*, 2020.

[8] X. Xu, X. Jiang, C. Ma, P. Du, X. Li, S. Lv, L. Yu, Q. Ni, Y. Chen, J. Su *et al.*, "A deep learning system to screen novel coronavirus disease 2019 pneumonia," *Engineering*, vol. 6, no. 10, pp. 1122–1129, 2020.

[9] S. Wang, B. Kang, J. Ma, X. Zeng, M. Xiao, J. Guo, M. Cai, J. Yang, Y. Li, X. Meng *et al.*, "A deep learning algorithm using CT images to screen for corona virus disease (COVID-19)," *European Radiology*, pp. 1–9, 2021.

[10] T. Kaur and T. K. Gandhi, "Automated diagnosis of COVID-19 from CT scans based on concatenation of mobilenetv2 and resnet50 features," in *5th International Conference on Computer Vision and Image Processing, CVIP 2020*, 2021, pp. 149–160.

[11] Y. Pathak, P. K. Shukla, and K. Arya, "Deep bidirectional classification model for COVID-19 disease infected patients," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 2020.

[12] T. Kaur, T. K. Gandhi, and B. K. Panigrahi, "Automated diagnosis of COVID-19 using deep features and parameter free bat optimization," *IEEE Journal of Translational Engineering in Health and Medicine*, vol. 9, pp. 1–9, 2021.

[13] M. Foysal and A. A. Hossain, "COVID-19 detection from chest CT images using ensemble deep convolutional neural network," in *2021 2nd International Conference for Emerging Technology (INCET)*. IEEE, 2021, pp. 1–6.

[14] Z. Wang, Q. Liu, and Q. Dou, "Contrastive cross-site learning with redesigned net for COVID-19 CT classification," *IEEE Journal of Biomedical and Health Informatics*, vol. 24, pp. 2806–2813, 2020.

[15] S. Madan, S. Chaudhury, and T. K. Gandhi, "Automated detection of COVID-19 on a small dataset of chest CT images using metric learning," in *2021 International Joint Conference on Neural Networks (IJCNN)*. IEEE, 2021, pp. 1–8.

[16] S. Liu and W. Deng, "Very deep convolutional neural network based image classification using small training sample size," in *2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR)*. IEEE, 2015, pp. 730–734.

[17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[18] S. Li, Y. Liu, X. Sui, C. Chen, G. Tjio, D. S. W. Ting, and R. S. M. Goh, "Multi-instance multi-scale CNN for medical image classification," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2019, pp. 531–539.

[19] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 1–9.

[20] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4700–4708.

[21] P. Angelov and E. Almeida Soares, "SARS-CoV-2 CT-scan dataset: A large dataset of real patients CT scans for SARS-CoV-2 identification," *MedRxiv*, 2020.

[22] S. Sen, S. Saha, S. Chatterjee, S. Mirjalili, and R. Sarkar, "A bi-stage feature selection approach for COVID-19 prediction using chest CT images," *Applied Intelligence*, pp. 1–16, 2021.

[23] 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," *arXiv preprint arXiv:1704.04861*, 2017.

[24] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2818–2826, 2016.

[25] P. Silva, E. Luz, G. Silva, G. Moreira, R. Silva, D. Lucio, and D. Menotti, "COVID-19 detection in CT images with deep learning: A voting-based scheme and cross-datasets analysis," *Informatics in Medicine Unlocked*, vol. 20, p. 100427, 2020.