Pneumonia Detection in Chest X-Rays using Transfer Learning and TPUs

Niranjan C. Kundur
Department of Computer Science and Engineering, JSS Academy of Technical Education, Bengaluru, India
niranjanckundur@jssateb.ac.in

Bellary Chiterki Anil
Department of Artificial Intelligence and Machine Learning, JSS Academy of Technical Education, Bengaluru, India
anilbc@jssateb.ac.in (corresponding author)

Praveen M. Dhulavvagol
School of Computer Science and Engineering, KLE Technological University, India
praveen.md@kletech.ac.in

Renuka Ganiger
Electrical and Electronics Department, KLE Technological University, India
renuka.ganiger@kletech.ac.in

Balakrishnan Ramadoss
Department of Computer Applications, National Institute of Technology, India
brama@nitt.edu

Received: 27 August 2023 | Revised: 6 September 2023 | Accepted: 9 September 2023

 DOI: https://doi.org/10.48084/etasr.6335

ABSTRACT

Pneumonia is a severe respiratory disease with potentially life-threatening consequences if not promptly diagnosed and treated. Chest X-rays are commonly employed for pneumonia detection, but interpreting the images can pose challenges. This study explores the efficacy of four popular transfer learning models, namely VGG16, ResNet, InceptionNet, and DenseNet, alongside a custom CNN model for this task. The model performance is evaluated using Mean Absolute Error (MAE) as the performance metric. The findings reveal that VGG16 outperforms the other transfer learning models, achieving the lowest MAE (66.19). To optimize the model training process, a distributed training strategy utilizing TensorFlow's TPU (Tensor Processing Unit) strategy is implemented. The custom CNN model is parallelized using TPU's multiple instances available over the cloud, enabling efficient computation parallelization and significantly reducing model training times. The experimental results demonstrate a remarkable decrease of 68.36% and 54.74% in model training times for the CNN model when trained using TPU compared to training on a CPU and GPU, respectively.

Keywords-pneumonia detection; chest X-ray images; deep learning; transfer learning; TPU; distributed training

I. INTRODUCTION

Early and accurate diagnosis of pneumonia is essential for better patient outcomes and reducing morbidity and mortality rates [1]. Identifying pneumonia from medical imaging, such as chest X-rays, can be challenging for medical professionals. Machine Learning (ML) techniques, particularly deep learning, offer promising solutions to automate pneumonia detection, enhancing diagnostic accuracy and reducing radiologists' workload. ML models can learn patterns from large datasets, providing objective and consistent results. However, developing accurate ML models for pneumonia detection presents its own challenges. The complexity and variability of chest X-ray images makes robust performance difficult. Convolution Neural Networks (CNNs) are powerful for computer vision tasks, but they often require large amounts of data. Acquiring labeled biomedical images is costly and time-consuming, requiring medical expertise for accurate

classification. To address the limitation of limited labeled data, this study employs transfer learning. Transfer learning reuses pre-trained models on large datasets, leveraging their established network weights for smaller datasets. In this research, we explore the efficacy of a custom CNN model and four widely used transfer learning models (VGG16, DenseNet, ResNet, and InceptionNet) for pneumonia detection. The models are trained and tested on a binary image dataset with two categories: Pneumonia and Normal. The objective is to develop an accurate and reliable automated pneumonia detection system using transfer learning to aid timely diagnosis and improve patient care.

Deep learning, specifically transfer learning, is emerging as a vital tool for pneumonia detection in chest X-ray images, primarily due to the inherent challenges in manual interpretation. Chest X-rays are intricate, with the lung's overlapping structures such as bronchi, vessels, and other tissues potentially obscuring or mimicking pneumonia's radiological signs. Patient positioning during the scan can hide or distort certain features, complicating diagnosis. Other pre-existing lung conditions, like COPD or tumors, can also mask or resemble pneumonia, leading to diagnostic ambiguity. Additionally, the quality of the X-ray image, the variability in radiologists' experience, and the presence of early, mild, or atypical cases further introduce complexities. Such challenges necessitate advanced automated techniques like deep learning to augment the diagnostic process, ensuring more accurate and timely detection. The main contributions of the current work are:

- The implementation of a transfer learning approach for automated pneumonia detection in chest X-ray images.
- A custom CNN model and 4 widely used transfer learning models (VGG16, DenseNet, ResNet, and InceptionNet) were considered, allowing for a comparative analysis of the models' effectiveness in pneumonia detection.
- To optimize model training, a distributed training strategy with Tensor Processing Units (TPU) is used by parallelizing the CNN model using TPUs available over the cloud. Significant reductions in training times are achieved, contributing to improved efficiency in pneumonia detection.

II. RELATED WORK

This section explores recent research on pneumonia detection using transfer learning and distributed training techniques applied to chest X-ray images. Various studies have investigated different transfer learning models and distributed training methods. Authors in [1] highlight the significance of rapid and accurate pneumonia detection. Authors in [2, 15] used CNN models for precise pneumonic lung detection from chest X-rays. Authors in [3] addressed the challenges of examining chest X-rays and proposed a computer-aided diagnosis system for automated pneumonia detection. Their approach involves deep transfer learning with an ensemble of GoogLeNet, ResNet-18, and DenseNet-121 models, achieving high accuracy rates of 98.81% and 86.85%. The need for an automated method for pneumonia detection using CNN is highlighted in [4]. The VGG16-based learning model has accuracy of 97.28%, showcasing satisfactory performance accuracy. In, [5, 6], the Efficient Net model was used for image processing and achieved an accuracy of 97%. Authors in [7] used ML models to obtain accuracy of 83.67%. Authors in [8] focused on diagnosing lung inflammation severity in COVID-19 patients using CNN, KNN, and other classification algorithms achieving 92.80% testing accuracy. Authors in [9] developed a mobile app utilizing deep learning techniques to classify pneumonia in patients. They achieved accuracies between 78% and 85% using Create ML. Authors in [10] studied CNN acceleration on various platforms (CPU, GPU, TPU) and compared their performance for pneumonia detection, face mask detection, and virus detection in plants. Disease detection using CNN models was used in [11]. Image segmentation techniques were used in [12] to locate ulcers in images. In [13], faster RCNN with Efficient Net model was used for image detection and classification with an accuracy of 98%. Authors in [14] proposed a CNN model for pneumonia detection from chest X-ray images using tensor flow. The model achieved an impressive accuracy of 99.46% with just 1000 training images and 10 epochs. Overall, the mentioned studies demonstrate the effectiveness and potential of transfer learning in pneumonia detection from chest X-ray images.

III. ML PIPELINE AND ARCHITECTURE DESIGN

The ML pipeline for pneumonia detection involves a series of steps that start from data preprocessing and end with the deployment and evaluation of the trained model, as shown in Figure 1. The first step is to collect a large dataset of chest X-ray images, consisting of pneumonia-positive and pneumonia-negative cases. The dataset should be diverse and representative of the target population to ensure the model's generalizability. In data preprocessing, the collected images are preprocessed to ensure consistency and quality. This may involve resizing the images to a standard size, converting them to grayscale or RGB format, and normalizing pixel values to a specific range [0, 1]. Preprocessing also includes handling missing data, if any. For Feature Extraction, CNNs are widely used for image-based tasks like pneumonia detection. In this step, the CNN is used to automatically extract relevant features from the preprocessed images. CNN layers detect different patterns and characteristics present in the images, which are important for distinguishing between pneumonia-positive and pneumonia-negative cases. For Model Building, the extracted features are used as input, and a classification model is constructed. This model can be a custom-designed CNN or a pre-trained CNN using transfer learning. Transfer learning involves using a pre-trained CNN on a large dataset (e.g. ImageNet) and fine-tuning it on the pneumonia detection dataset. The model is trained using the training dataset. During training, the model learns to map the extracted features to the correct classification labels (pneumonia-positive or pneumonia-negative). The optimization process adjusts the model's weights to minimize the error or loss between the predicted and the actual labels. To optimize the model's performance and prevent over fitting, hyper parameter tuning is conducted using the validation dataset. Hyper parameters are variables that control the learning process, such as the learning rate, batch size, number of layers, and the size of filters in the CNN.

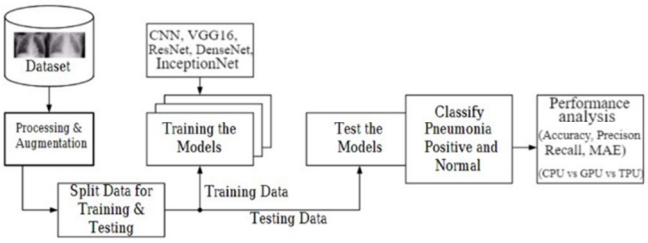

Fig. 1. ML pipeline for pneumonia detection.

Figure 2 shows the custom CNN architecture for pneumonia detection, consisting of five convolutional blocks, each comprising a convolutional layer, a max-pooling layer, and a batch normalization layer. The architecture is designed to extract relevant features from the chest X-ray images to distinguish between pneumonia-positive and pneumonia-negative cases. Dropout layers are added to prevent overfitting and improve generalization. The architecture includes a flatten layer to transform the output from the convolutional blocks into a 1D vector. This is followed by four fully connected layers, which further process the extracted features for classification. The activation function used throughout the model is ReLU, except for the last layer, where a sigmoid activation function is applied since the problem is a binary classification task (pneumonia or normal). The optimizer used for training the model is Adam, a popular optimization algorithm that efficiently updates the model weights during the training process. The loss function chosen is cross-entropy, which is suitable for binary classification tasks.

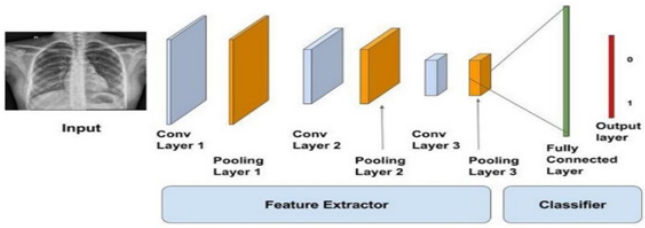

Fig. 2. Overview of the CNN architecture.

Before training the model, two callbacks are defined: Model Checkpoint and Early Stopping. Model Checkpoint is used to save the best model weights during training, ensuring that the model with the highest validation accuracy is retained. Early Stopping is used to monitor the validation loss and stop training if the loss starts to increase, indicating over fitting. These callbacks help in achieving better results and preventing overfitting during the training process. The model is trained for 100 epochs, which represents the number of times the entire training dataset is passed through the model during training. The image size used for training is (180, 180), indicating the resolution at which the chest X-ray images are processed by the model.

IV. IMPLEMENTATION

The procedure for using transfer learning in pneumonia detection through chest X-rays follows a detailed and methodical workflow. Initially, the images are processed to guarantee uniformity and boost model efficiency. This involves adjusting image dimensions, equalizing pixel intensities, and employing data enhancement methods to diversify the training dataset. Subsequently, a suitable pre-trained architecture, such as VGG16, ResNet, InceptionNet, or DenseNet, is chosen. Tailoring the latter sections of the chosen model is crucial for the pneumonia detection objective. The final layer is adjusted or swapped out to cater to the binary nature of the task of distinguishing pneumonia cases from normal ones. During the primary training, the pre-trained model's weights remain static to preserve its acquired knowledge. Only the appended layers are trained using the pneumonia-centric dataset, enabling the model to recognize pneumonia-related patterns, enriched by the foundational knowledge. Evaluation metrics like accuracy, precision, recall, F1 score, and the ROC curve's area are pivotal in measuring a model's proficiency. For managing expansive datasets and hastening training durations, TPUs, purpose-built for resource-intensive deep learning tasks, are utilized. Subsequent to the primary training, the entire model, encompassing the pre-trained layers, undergoes refinement at a reduced learning rate. This refinement fine-tunes the model for pneumonia detection, leveraging the insights of the pre-trained model. It's essential to fine-tune hyperparameters, such as learning rate, batch size, and regularization force. Techniques like grid or random search assist in this, ensuring peak validation set performance. The model's adaptability is then scrutinized against a distinct validation set, and techniques like cross-validation might be incorporated for sturdier performance insights. Ultimately, the polished model faces a standalone test dataset to gauge its efficiency on unfamiliar data. The concluding step involves making predictions on fresh chest X-ray images.

V. DATASET DESCRIPTION AND VISUALIZATION

Figure 3 shows an overview of the dataset considered for the experimentation purpose. Pixel distribution is plotted using a bar plot for a given chest X-ray image. Figure 4 shows the number of pneumonia and normal samples of the dataset which are used for training the model. The training data are imbalanced with more pneumonic images. There are 3931 pneumonic images and 1341 Normal images. The data were balanced by using various augmentation techniques.

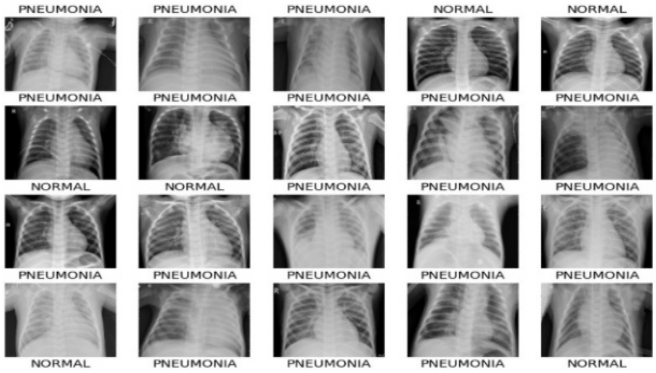

Fig. 3. Dataset overview.

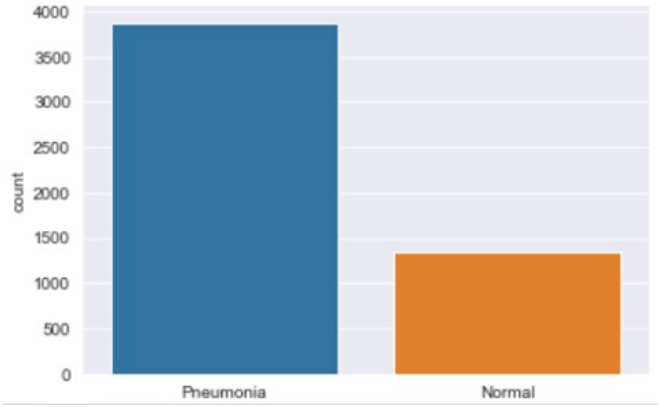

Fig. 4. Number of pneumonia and normal samples in the train dataset.

VI. RESULTS AND DICUSSION

In this section, we present the results and discussion of our experiments on pneumonia detection using different deep learning models, including VGG16 (proposed), DenseNet (State-of-the-Art (SotA)), ResNet-50 (SotA), Custom CNN (baseline).

A. Performance Comparison with State-of-the Art Methods

Table I shows the performance analysis of the proposed model with the SotA. The comparison table offers a detailed look into the performance of various deep learning models for pneumonia detection from chest X-rays. The proposed VGG16 model showcases the highest accuracy of 87.5%. In contrast, the SotA models DenseNet and ResNet-50 follow closely with 82.3% and 73.4% accuracy, respectively, as shown in Figure 5.

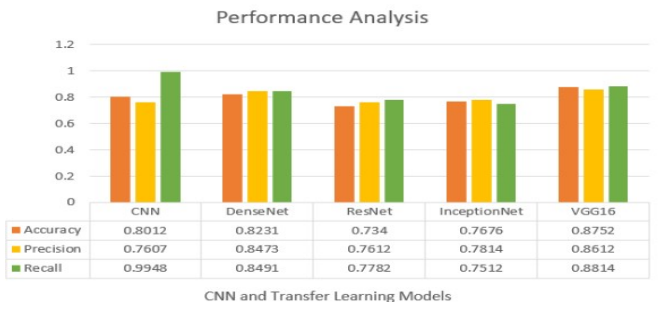

Fig. 5. Performance comparison of the DL models

The Custom CNN, used as a baseline, stays behind with a 80.3% accuracy. While VGG16 excels in precision and recall metrics, suggesting its superior capability in correctly identifying and capturing positive pneumonia cases, it does come at the cost of longer training times. Specifically, VGG16 takes 10 hours, nearly double the time required for ResNet-50. All models were trained on a consistent dataset size, ensuring a level playing field for comparison. Table I provides a clear illustration of the trade-offs involved, such as accuracy versus training time, helping researchers make informed decisions based on their specific needs and constraints. A lower Mean Absolute Error (MAE) value corresponds to better model performance. Among the evaluated models, VGG16 obtained the lowest MAE.

TABLE I. PERFORMANCE COMPARISON

Model	Accuracy (%)	Precision	Recall	F1-score
VGG16 (proposed)	87.5	86.4	88.1	88.4
DenseNet (SotA)	82.3	84.7	84.9	84.2
ResNet-50 (SotA)	73.4	76.2	77.2	77.1
Inception Net (SotA)	76.6	78.4	75.2	76.3
Custom CNN (baseline)	80.3	76.0	79.4	77.1

Figure 6 shows the different deep learning models examined for their effectiveness in detecting pneumonia from chest X-rays, especially focusing on their accuracy over training epochs. The initial results suggest that models like VGG16 can achieve high accuracy rapidly in early epochs. In contrast, deeper architectures such as DenseNet and ResNet-50, although initially slower, could potentially excel in performance during prolonged training due to their intricate structures. Meanwhile, simpler custom CNN models might show a consistent but more gradual increase in accuracy, plateauing sooner than their complex counterparts. The analysis underscores the importance of choosing the right model based on both the desired computational efficiency and detection accuracy.

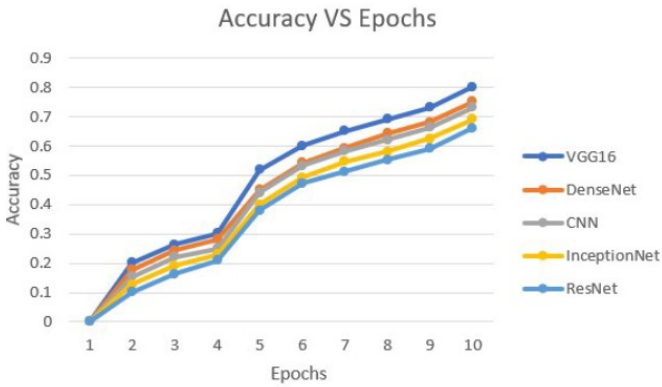

Fig. 6. Accuracy vs epochs.

B. Distributed training using TPU's

Distributed training with TPUs represents a potent technique that involves parallel training of large neural networks across multiple processors. TPUs are specialized Application-Specific Integrated Circuits (ASICs) developed by Google, designed to optimize deep learning model performance. Their exceptional efficiency in matrix multiplication, a crucial operation in deep learning, surpasses that of traditional CPUs or GPUs, resulting in significantly faster training times. In this paper, we harnessed the power of TPUs as hardware accelerators to parallelize and train a custom CNN model. The training time for the CNN model using TPUs was remarkably low, taking only 3 minutes and 6 seconds. This distributed training approach utilized the cloud TPUs provided by the Google Colab platform. By employing the TPU strategy in the TensorFlow framework, we achieved a substantial improvement in model training time.

In comparison, training the CNN model with CPUs extended the training time to 9 minutes and 48 seconds. CPUs,

especially when dealing with large batches of data, exhibit the slowest neural network training speeds. On the other hand, utilizing GPUs to train the CNN model reduced the training time to 6 minutes and 51 seconds. GPUs are known for their parallel processing capabilities, which enable notable reductions in model training time. Furthermore, increasing the batch size when using GPUs can further enhance training speed. However, the most efficient approach among the three options (TPU, GPU, and CPU) for training the CNN model was undoubtedly the use of TPUs. Distributed training with TPUs provided the lowest model training time, underscoring the advantage of TPUs' specialized architecture for deep learning tasks. Overall, the adoption of distributed training with TPUs showcased a remarkable performance boost in training deep learning models, specifically in the context of pneumonia detection on chest X-ray images. Leveraging TPUs' acceleration capabilities was proved to be a critical asset in advancing medical image analysis and diagnosis applications.

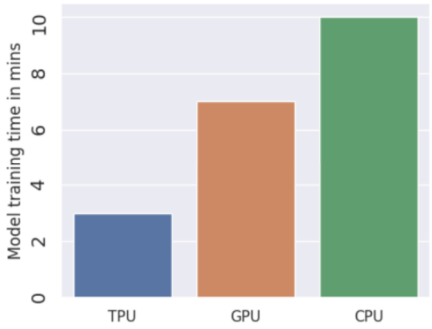

Fig. 7. Performance comparison of CPU, GPU, and TPU.

Figure 7 illustrates a comprehensive performance comparison of CPU, GPU, and, TPU for training the CNN model. The goal was to evaluate the efficiency of each hardware option in terms of model training times. When utilizing TPU for training the CNN model, a remarkable percentage decrease of 54.74% in model training time was observed compared to using GPU. This clearly showcases the significant advantage of TPUs, which excel in speeding up the training process due to their parallel processing capabilities. Moreover, when training the CNN model with TPU, an even more substantial decrease percentage of 68.36% in model training time was observed compared to using CPU. This underscores the immense performance gain achieved by leveraging TPU's specialized architecture for matrix multiplications, making them highly efficient for training large neural networks. The experimental results unequivocally demonstrate that TPUs have a profound impact on reducing model training times, thanks to their specialized optimization for matrix multiplications. Their superiority over GPU and CPU is evident not only in terms of performance but also in power consumption.

VII. CONCLUSION

Transfer learning has emerged as a highly effective approach for pneumonia detection in chest X-ray images. By leveraging pre-trained CNN models and fine-tuning them on a smaller dataset of chest X-ray images, transfer learning significantly enhances the accuracy and efficiency of pneumonia detection. In this research, we explored the performance of a custom CNN model and four transfer learning models (VGG16, ResNet-50, DenseNet-121, and Inception v3) trained on chest X-ray images. Among the transfer learning models, VGG16 stood out as the top-performing one, achieving the lowest Mean Absolute Error (MAE) value of 66.19. It outperformed the other models in accuracy, precision, and recall. Specifically, the VGG16 model demonstrated an impressive accuracy of 0.8752, precision of 0.8612, and recall of 0.8814, highlighting its exceptional ability to distinguish between pneumonia-positive and pneumonia-negative cases. The current advancements in leveraging deep learning for pneumonia detection in chest X-rays offer promising avenues, but there's still a spectrum of untapped potential. Future work could focus on integrating multimodal data sources, such as combining X-rays with CT scans or patient history, to refine detection accuracy. The use of more sophisticated neural architectures or the fusion of multiple models might also yield better diagnostic insights. Furthermore, as datasets grow and become more diverse, models will benefit from the inclusion of more varied and rare pneumonia presentations. The broader implications of this research are profound: enhancing diagnostic precision can lead to faster treatment interventions, potentially reducing hospitalizations and mortality rate. In a global context, such automated systems could be vital for resource-limited settings, democratizing access to high-quality diagnostic tools and revolutionizing patient care worldwide.

REFERENCES

[1] S. Kalgutkar *et al.*, "Pneumonia Detection from Chest X-ray using Transfer Learning," in *2021 6th International Conference for Convergence in Technology (I2CT)*, Maharashtra, India, Apr. 2021, https://doi.org/10.1109/I2CT51068.2021.9417872.

[2] S. V. Militante and B. G. Sibbaluca, "Pneumonia Detection Using Convolutional Neural Networks," *International Journal of Scientific & Technology Research*, vol. 9, no. 4, pp. 1332–1337, 2020.

[3] R. Kundu, R. Das, Z. W. Geem, G.-T. Han, and R. Sarkar, "Pneumonia detection in chest X-ray images using an ensemble of deep learning models," *PLOS ONE*, vol. 16, no. 9, 2021, Art. no. e0256630, https://doi.org/10.1371/journal.pone.0256630.

[4] A. Ranjan, C. Kumar, R. K. Gupta, and R. Misra, "Transfer Learning Based Approach for Pneumonia Detection Using Customized VGG16 Deep Learning Model," in *Internet of Things and Connected Technologies*, Cham, 2022, pp. 17–28, https://doi.org/10.1007/978-3-030-94507-7_2.

[5] T. Rahman *et al.*, "Transfer Learning with Deep Convolutional Neural Network (CNN) for Pneumonia Detection Using Chest X-ray," *Applied Sciences*, vol. 10, no. 9, Jan. 2020, Art. no. 3233, https://doi.org/10.3390/app10093233.

[6] N. C. Kundur and P. B. Mallikarjuna, "Deep Convolutional Neural Network Architecture for Plant Seedling Classification," *Engineering, Technology & Applied Science Research*, vol. 12, no. 6, pp. 9464–9470, Dec. 2022, https://doi.org/10.48084/etasr.5282.

[7] F. Mlawa, E. Mkoba, and N. Mduma, "A Machine Learning Model for detecting Covid-19 Misinformation in Swahili Language," *Engineering, Technology & Applied Science Research*, vol. 13, no. 3, pp. 10856–10860, Jun. 2023, https://doi.org/10.48084/etasr.5636.

[8] A. M. Ali, K. Ghafoor, A. Mulahuwaish, and H. Maghdid, "COVID-19 pneumonia level detection using deep learning algorithm and transfer learning," *Evolutionary Intelligence*, Sep. 2022, https://doi.org/10.1007/s12065-022-00777-0.

[9] D. Varshni, K. Thakral, L. Agarwal, R. Nijhawan, and A. Mittal, "Pneumonia Detection Using CNN based Feature Extraction," in *2019 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT)*, Coimbatore, India, Oct. 2019, https://doi.org/10.1109/ICECCT.2019.8869364.

[10] P. Pattrapisetwong and W. Chiracharit, "Automatic lung segmentation in chest radiographs using shadow filter and multilevel thresholding," in *2016 International Computer Science and Engineering Conference (ICSEC)*, Chiang Mai, Thailand, Sep. 2016, https://doi.org/10.1109/ICSEC.2016.7859887.

[11] S. Alqethami, B. Almtanni, W. Alzhrani, and M. Alghamdi, "Disease Detection in Apple Leaves Using Image Processing Techniques," *Engineering, Technology & Applied Science Research*, vol. 12, no. 2, pp. 8335–8341, Apr. 2022, https://doi.org/10.48084/etasr.4721.

[12] R. H. Mwawado, B. J. Maiseli, and M. A. Dida, "Robust Edge Detection Method for Segmentation of Diabetic Foot Ulcer Images," *Engineering, Technology & Applied Science Research*, vol. 10, no. 4, pp. 6034–6040, Aug. 2020, https://doi.org/10.48084/etasr.3495.

[13] N. C. Kundur and P. B. Mallikarjuna, "Insect Pest Image Detection and Classification using Deep Learning," *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 9, pp. 411–421, 2022, https://doi.org/10.14569/IJACSA.2022.0130947.

[14] P. Rajpurkar *et al.*, "CheXNet: Radiologist-Level Pneumonia Detection on Chest X-Rays with Deep Learning." arXiv, Dec. 25, 2017, https://doi.org/10.48550/arXiv.1711.05225.

[15] N. C. Kundur and P. B. Mallikarjuna, "Ensemble Efficient Net and ResNet model for Crop Disease Identification," *International Journal of Intelligent Systems and Applications in Engineering*, vol. 10, no. 4, pp. 378–390, Dec. 2022.

[16] V. Sirish Kaushik, A. Nayyar, G. Kataria, and R. Jain, "Pneumonia Detection Using Convolutional Neural Networks (CNNs)," in *Proceedings of First International Conference on Computing, Communications, and Cyber-Security (IC4S 2019)*, Singapore, 2020, pp. 471–483, https://doi.org/10.1007/978-981-15-3369-3_36.