

The Influence of Parallel Computing on Building Deep Learning Model for the Classification of Bean Diseases

¹Jean Bosco Gashugi ; ²Dr. Emmanuel Bugingo (PhD)
Master of Science with Honors in Information Technology, at the University of Kigali, Rwanda

Abstract:- In recent years, the utilization of deep learning techniques for image classification has made significant strides in the field of agriculture. One of the key areas of interest in agriculture is the early detection and classification of diseases in crops, as this can have an insightful impact on crop revenue and quality. This research has investigated the influence of parallel computing on the performance of a deep learning-based classification model for diagnosing bean diseases. Specifically, we have explored the use of parallel computing frameworks to accelerate model training and inference, thereby enhancing the efficiency and effectiveness of disease classification. Our findings demonstrated the potential for parallel computing to accelerate model training. When training a bean disease classification model, we achieved an accuracy of 0.93 using parallel computing, compared to 0.83 with serial computing. Moreover, parallel computing significantly reduced training time, taking only 3 minutes compared to 51 minutes with serial computing.

I. INTRODUCTION

Parallel computing has emerged as a transformative technology in various domains, revolutionizing the efficiency and scalability of computational tasks (Wu, 2021). In the realm of agriculture, early and accurate disease detection in crops is crucial for maximizing yield and minimizing losses. Deep learning particularly convolutional neural networks (CNNs), has developed as a powerful tool for building disease classification models. However, these models often require significant computational resources, especially when dealing with large datasets of images.

This paper explores the profound influence of parallel computing on the development of deep learning models for beans disease classification. By harnessing the computational power of parallel architectures, such as Graphics Processing Units (GPUs) and distributed computing frameworks, where we overcome the computational bottlenecks inherent in training deep neural networks on large-scale datasets (Geng, 2020).

II. METHODOLOGY

A. Data Collection Methods and Instruments/ Tools

Data collection in deep learning for image classification is a critical step that significantly impacts the performance of the model. Based on that while conducting Data collection for this research, we have chosen to use a dataset provided by **AI-Lab Makerere**, because the leaf images had high resolution and were annotated, standardized size, format, and color balance, which enhanced the quality and consistency of the dataset. Additionally, these leaf images are totally similar with the local bean leaf here in Rwanda (AI-Lab-Makerere, 2020).

➤ Jupyter Notebook

During this research, we have used Jupyter Notebook as an interactive computing environment that is specifically designed for developing, training, and evaluating deep learning models. It is integrated with popular deep learning frameworks called TensorFlow and many libraries installed (Brian, 2014).

➤ Python Language

Python language serves as a powerful and flexible tool for implementing algorithms and building models. It enables developers to easily express complex mathematical computations and build intricate neural networks, its simplicity, readability, and robust ecosystem (python, 2024).

➤ Parallel Computing

Parallel computing offers a compelling solution to address the computational challenges associated with deep learning for beans disease classification. (Hegde Vishakh; Sheema Usmani, 2017) investigated the use of parallel computing frameworks like TensorFlow or PyTorch to distribute the training workload across multiple GPUs, achieving significant speedups (Wang, 2019). SLURM as a job scheduler was used to efficiently distribute and manage the computational workload across multiple nodes of a cluster (Slurm & Deep Learning, n.d.).

➤ Ray Core

Ray Core is the heart of Ray, an open-source framework for parallel and distributed Python applications. It offers a small set of powerful primitives that enable you to easily leverage the processing power of multiple cores or even a cluster of machines (Team, 2024).

B. Data Analysis

To enhance deep learning model performance in distinguishing between healthy and diseased bean plants, robust datasets comprising images of both conditions are essential. Implementing data preprocessing techniques, including image resizing, normalization, and augmentation, is crucial to refine the data quality and augment model accuracy. Furthermore, leveraging parallel computing facilitates swift and efficient execution of tasks like parallel image resizing and data augmentation across multiple processors, thereby expediting the preprocessing phase and overall model development process (Elhoucine Elfatimi, 2023).

➤ **Model Evaluation**

Model evaluation is a very crucial part of building an effective deep learning model since it gives an explanation on the model performance. To evaluate a model, evaluation metrics are employed to discriminate among the model results. This research adopted 3 model evaluation metrics that are classification accuracy, confusion matrix, and classification report (OCHIENG, 2022).

C. Research Design

For this study, due to stratified sampling is a technique in which the population is divided into smaller, more homogeneous subgroups called strata. Furthermore, it ensures that the sample is representative of the entire population, and it is especially important when the population is not homogeneous. It is in this regard that we used this research method to collect the target population which corresponds to the research questions we intended to address. Therefore, we developed a model to be used by farmers and others to detect diseases in bean plants in real-time.

The following block diagram shows how parallel computing empowered by Ray Core and SLURM speeds up the development of a deep learning model for classifying bean diseases. Data is collected and prepared, then augmented to create a richer dataset. Ray Core and GPUs are used for parallel computing to train the model more efficiently. Once trained, the model is tested, evaluated, and potentially compressed (quantization) for deployment on smartphones.

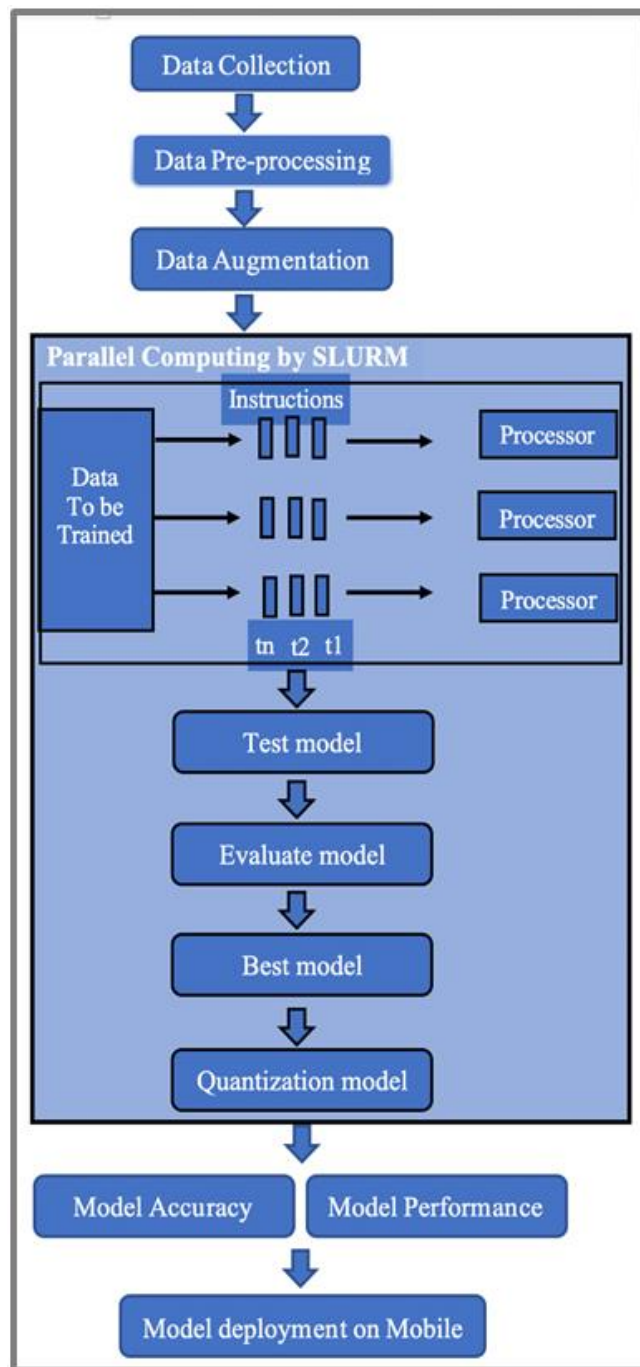


Fig 1 Parallel Computing Block Diagram

III. CONCEPTUAL FRAMEWORK

The conceptual framework illustrates independent and dependent variables to take into consideration while studying the influence of parallel computing on bean disease classification model building using deep learning. The parallel computing techniques were implemented to speed up computations, where many calculations are carried out simultaneously by taking advantage of modern multi-core processors and distributed computing environments, enabling efficient handling of complex and large-scale computational tasks. As a result, we obtained the best model to be deployed in real-world applications for agricultural contexts.

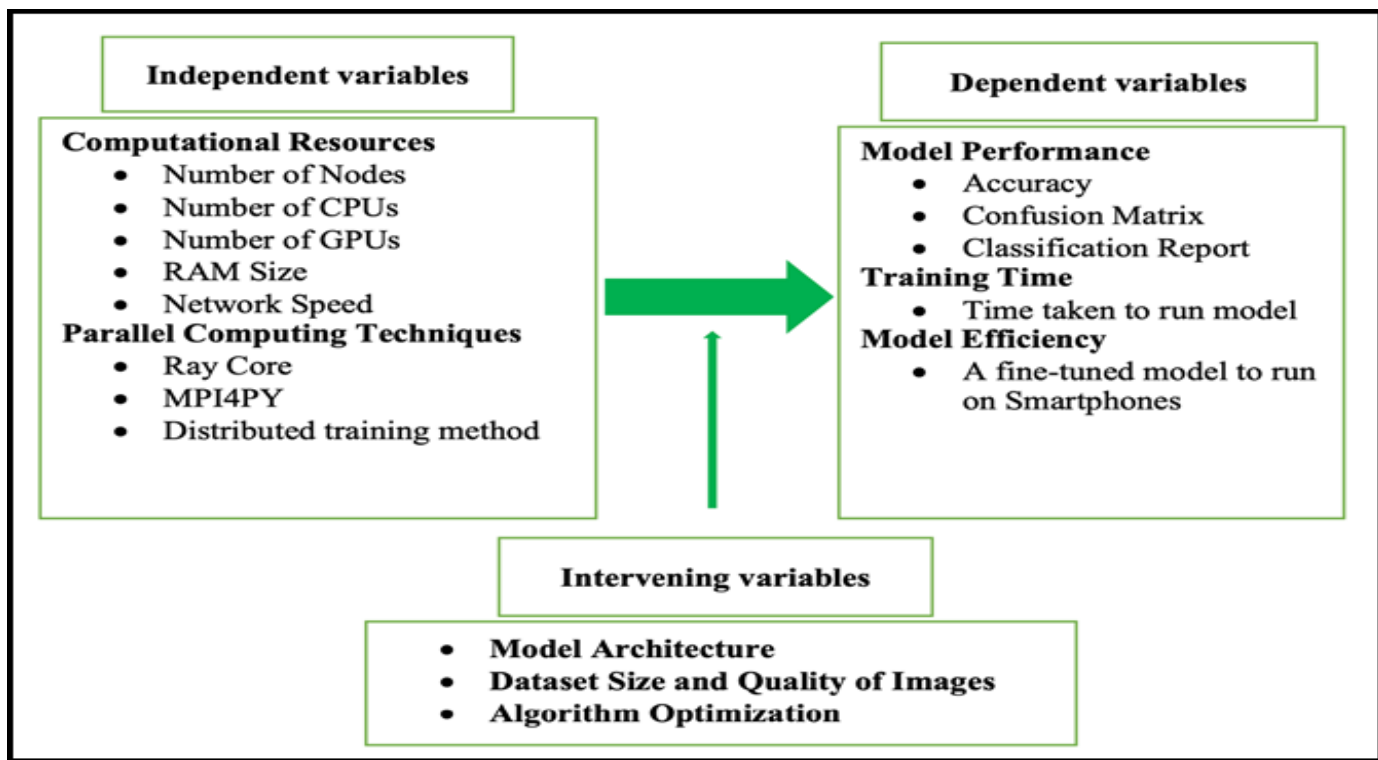


Fig 2 Conceptual Framework

IV. DATA PRESENTATION, ANALYSIS AND INTERPRETATION OF FINDINGS

A. Introduction of the Contents of the Chapter

This chapter presents the findings from the study on the influence of parallel computing on building deep learning models for the classification of bean diseases. It includes a detailed analysis of the data and the results obtained from various experiments. The objective was to provide a comprehensive demonstration of how parallel

computing influences the efficiency and accuracy of deep learning models in this specific context.

B. Parallel vs Serial Computing

Parallel computing leverages multiple processors to perform computations simultaneously, significantly reducing training times for models. We have compared the performance of parallel and serial computing approaches in the context of training deep learning models for the classification of bean diseases, then the following findings were observed.

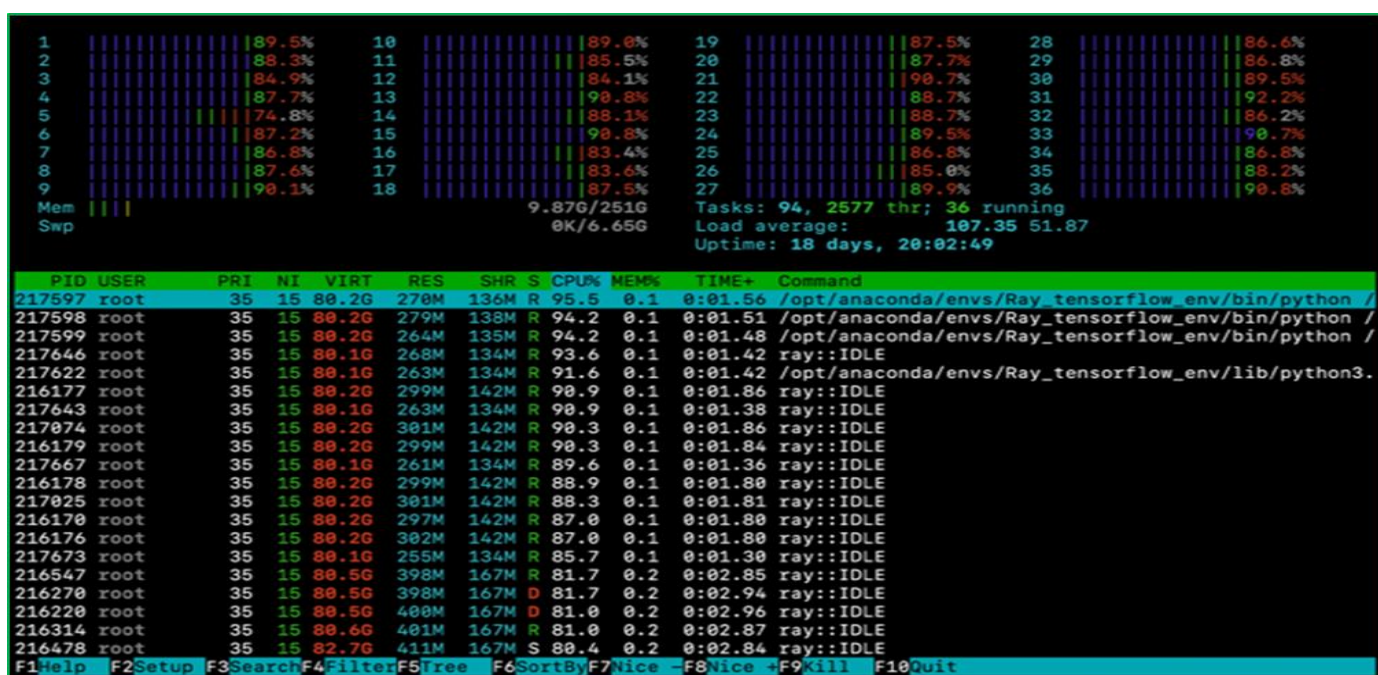


Fig 3 Model Training with Parallel Processing

In contrast, Serial Computing executes one instruction at a time in a sequential manner. The process data were in a step-by-step fashion and the performance was generally slow due to the sequential of tasks. The speed was limited by a single core and could not exploit modern multi-core architectures effectively. It was taking a long training time for large datasets and complex models because serial computing was limited to the capabilities of a single core.

C. Analysis of Findings

➤ Training Time Findings

To evaluate the impact of parallel computing on training time, we trained the same deep learning model using both serial and parallel computing approaches. The training times were recorded and compared as follows:

- With Parallel Computing **Training Time: 201.33 (s) ≈ 3 mins**
- With Serial Computing **Training Time: 3049.52 (s) ≈ 51mins**

We observed that, when employing parallel computing, the training time of the deep learning model was taking approximately **3 minutes**. This efficiency was achieved by distributing computations across multiple processors, allowing simultaneous processing of data and tasks. In contrast, serial computing, which processes tasks sequentially, took extremely longtime around **51 minutes** to complete the same training. This stark difference highlights the advantage of parallel computing in handling the intensive computations typical in deep learning tasks, thereby accelerating the training process and improving overall efficiency.

➤ Model Performance Findings

Our model was trained using parallel computing approach and achieved higher accuracy due to the ability to process larger datasets and reduce the risk of overfitting by mitigating the use of larger datasets and regularization techniques as shown on below figures.

- Accuracy

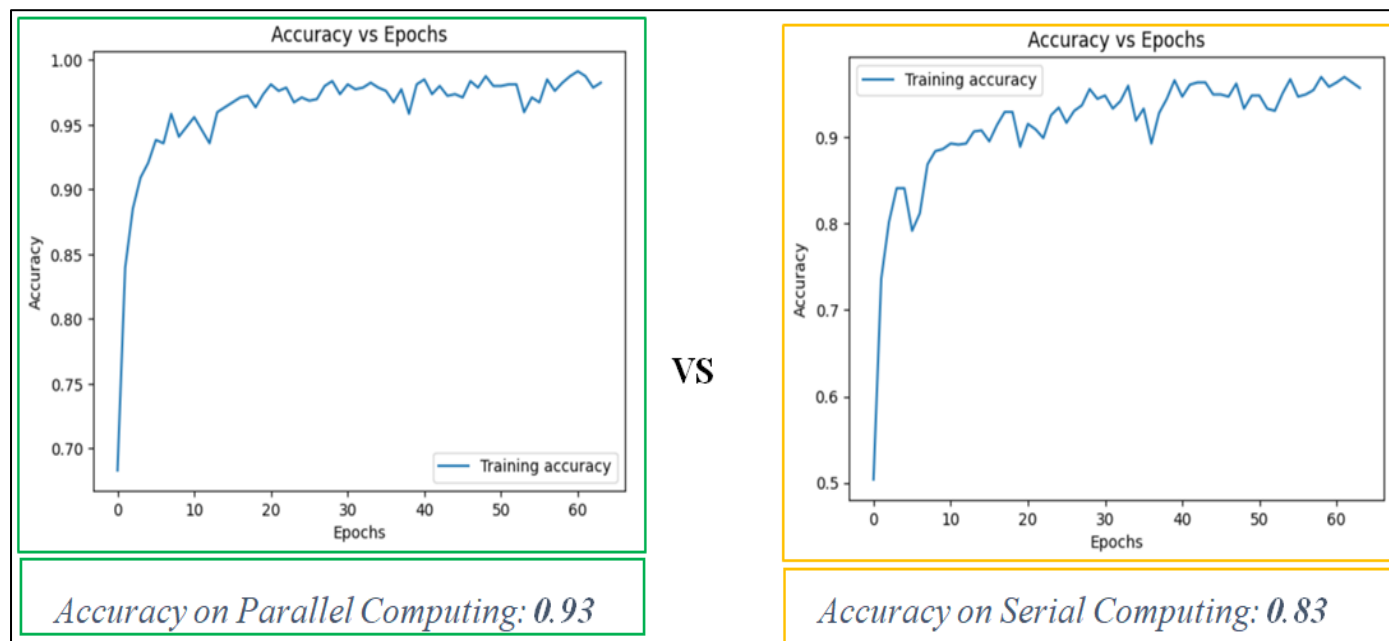


Fig 4 Accuracy Obtained on Parallel vs Serial Computing

Using parallel computing, the deep learning model achieved an accuracy of approximately **0.93**. This is due to the ability to process larger batches of data simultaneously and utilize more complex architectures efficiently. In contrast, with serial computing, the model reached an accuracy of around **0.83**, likely due to limitations in processing power and the inability to handle larger datasets effectively. This demonstrates the significant benefits of parallel computing in both speed and accuracy for deep learning tasks.

➤ Model in Real-World Application Findings

Deploying the model on mobile devices allows farmers and agricultural workers to use their smartphones to quickly and accurately diagnose bean diseases in the field. This has led to timely interventions and better crop management practices in bean disease detection.

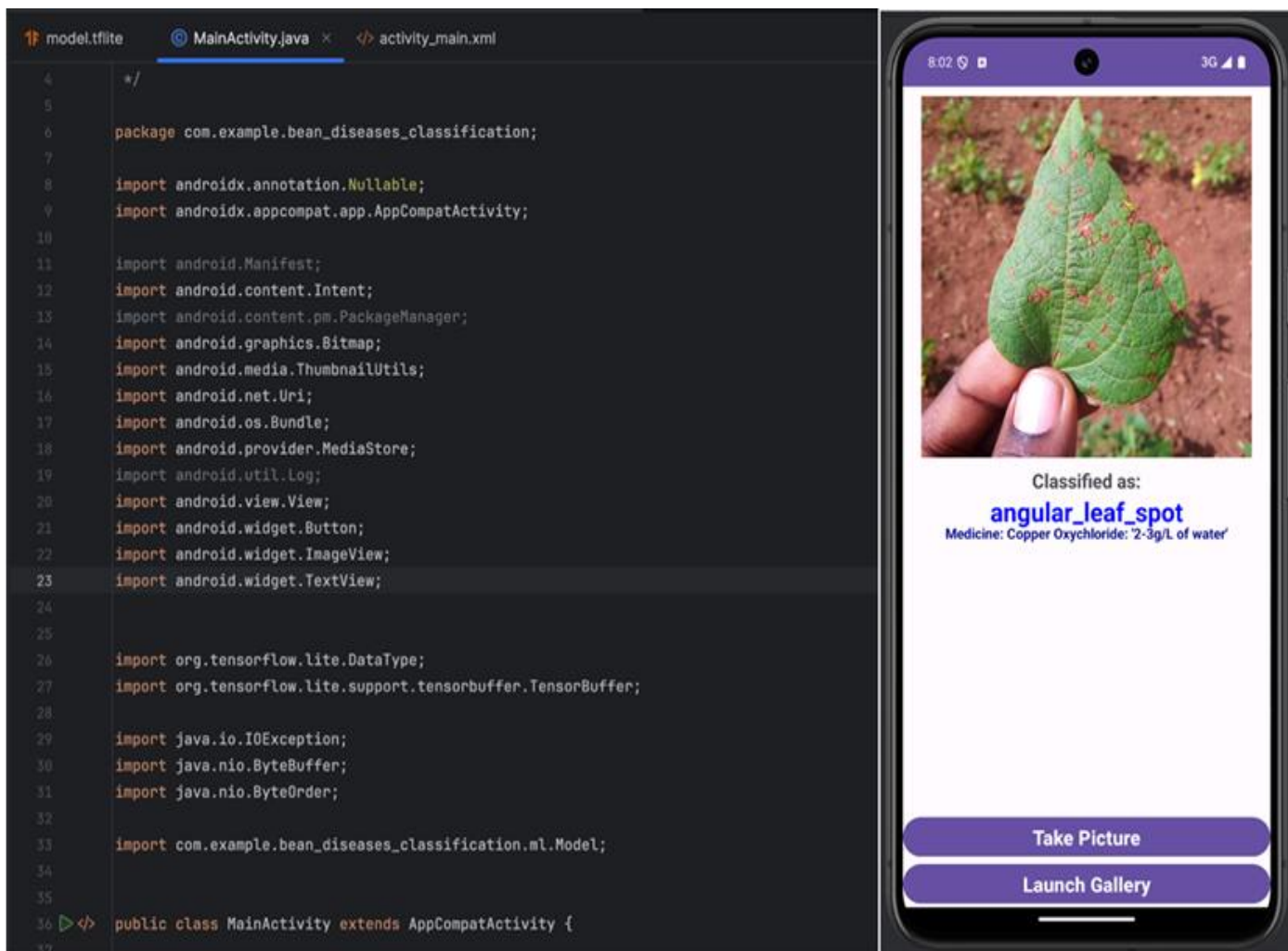


Fig 5 Model deployment on Android Smartphones in Real-World Application

As a result, the farmers will diagnose bean diseases directly in the field by capturing images with their smartphones. This eliminates the need for sending samples to labs for analysis, saving valuable time, etc. Moreover, early detection is crucial for containing outbreaks and preventing significant yield loss.

V. CONCLUSION

This study conclusively showed that parallel computing demonstrably accelerated the development and deployment of a deep-learning model for bean disease classification. Leveraging parallel computing yielded a significant improvement in model accuracy, achieving a classification accuracy of 0.93 compared to 0.83 obtained with serial computing. Furthermore, parallel computing drastically reduced training time, taking only 3 minutes compared to 51 minutes with serial processing. This enhanced efficiency facilitates faster model iterations, exploration of a broader range of hyperparameters, and ultimately, the development of more robust and accurate models. Additionally, the ability to deploy these models on mobile devices paves the way for real-time disease diagnosis in agricultural fields, empowering farmers and agricultural workers.

RECOMMENDATIONS

Based on the study findings, the following recommendations are made: First, we encourage individuals to create their own high-performance computing clusters using Raspberry Pi nodes, which offers a cost-effective solution for setting up HPC clusters and utilizing parallel computing for model training and heavy computational tasks. Second, researchers and practitioners should adopt parallel computing frameworks to accelerate the training process and enhance model performance. Third, leveraging transfer learning with pre-trained models can significantly reduce training time and improve accuracy, particularly when dealing with limited datasets. Fourth, implementing comprehensive data augmentation techniques consistently can enhance model robustness and generalization capabilities. Finally, optimizing models for mobile deployment is crucial, as it enables real-time, on-the-field diagnostics, greatly benefiting agricultural practices by providing valuable tools for farmers and agricultural workers.

REFERENCES

- [1]. Cheng, Y. e. (2019). Bean leaf disease detection and classification based on deep residual learning. *Computer and Electronics in Agriculture*.
- [2]. Deep Learning on Supercomputers. (n.d.). Retrieved from <https://towardsdatascience.com/>: <https://towardsdatascience.com/deep-learning-on-supercomputers-96319056c61f>
- [3]. Elhoucine Elfatimi, R. E. (2023, November). Impact of datasets on the effectiveness of MobileNet for beans leaf disease detection. Retrieved from SpringerLink: <https://link.springer.com/article/10.1007/s00521-023-09187-4>
- [4]. Geng. (2020). Parallel computing for training deep learning models for beans disease classification. *IEEE international Conference*.
- [5]. Hoang-Tu Vo, L.-D. Q. (2023). Ensemble of Deep Learning Models for Multi-plant Disease Classification in Smart Farming. Cantho City, Vietnam: Software Engineering Department, FPTUniversity.
- [6]. Jean B. Ristainoa, P. K. (2021). The persistent threat of emerging plant disease pandemics to global food security. Manhattan: Barbara Valent, Kansas State University, Manhattan, KS.
- [7]. Kahira, A. N. (2021). Convergence of Deep Learning and High Performance Computing: Challenges and Solutions. Barcelona: Universitat Politècnica de Catalunya.
- [8]. Kolodziejczak, K. P. (2020). The Role of Agriculture in Ensuring Food Security in Developing Countries. Considerations in the Context of the Problem of Sustainable Food Production. Poznan, Poland: Department of Economics and Economic Policy in Agribusinesses, Faculty of Economics and Social Sciences, Poznan University of Life Sciences, Wojska Polskiego 28, 60-637 Poznan, Poland.
- [9]. Michelle M. Nay, T. L.-V.-C. (2018). A Review of Angular Leaf Spot Resistance in Common Bean. Parana: Dep. Agronomia, Univ. Estadual de Maringá, Maringá, Paraná, Brazil.
- [10]. NVIDIA cuDNN. (n.d.). Retrieved from <https://docs.nvidia.com/cudnn/index.html>: <https://docs.nvidia.com/cudnn/index.html>
- [11]. NVIDIA Tesla P100 PCIe 16 GB. (n.d.). Retrieved from techpowerup: <https://www.techpowerup.com/gpu-specs/tesla-p100-pcie-16-gb.c2888>
- [12]. P. Pamela, D. M. (2014). Severity of angular leaf spot and rust diseases on common beans in Central Uganda. Kampala: National Crops Resources Research Institute, Namulonge.
- [13]. Paymode, A. S. (2021). Transfer Learning for Multi-Crop Leaf Disease Image Classification using Convolutional Neural Network VGG. MGM's Jawaharlal Nehru Engineering College, Aurangabad 431001, Maharashtra, India.
- [14]. Prathamesh Borhade, R. D. (2020). Image Classification using Parallel CPU and GPU Computing. *International Journal of Engineering and Advanced Technology(IJEAT)*, ISSN: 2249 - 8958, Volume-9 Issue-4.
- [15]. Seyed Hossein Nazer Kakhki, M. V. (2022). Predict bean production according to bean growth, root rots, fly and weed development under different planting dates and weed control treatments. Kermanshah: Plant Protection Research Department, Kermanshah Agricultural & Natural Resources Research & Education Center, AREEO, Kermanshah, Iran.
- [16]. Shouan Zhang, N. D. (2019). Disease Control for Snap Beans in Florida. IFAS Extension, University of Florida.
- [17]. Slurm & Deep Learning. (n.d.). Retrieved from run.ai: <https://www.run.ai/guides/slurm/slurm-deep-learning>
- [18]. Stratified Random Sampling. (n.d.). Retrieved from Questionpro: <https://www.questionpro.com/blog/stratified-random-sampling/>
- [19]. Wang, X. e. (2019). A GPU-accelerated deep learning framework for bean disease classification. *Computers and Electronics in Agriculture*.
- [20]. What Is CUDA? (n.d.). Retrieved from <https://blogs.nvidia.com/>: <https://blogs.nvidia.com/blog/what-is-cuda-2/>
- [21]. Wu, W. e. (2021). Parallel computing for training models for multi-plant disease classification. In *Proceedings of the 2021 International Conference on Artificial Intelligence and Machine Learning (ICAIML)*.
- [22]. Xidong Wu, P. B. (2023). Performance and Energy Consumption of Parallel Machine Learning Algorithms. *ECE* 2166.
- [23]. Yang, S. J. (2010). A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345-1359.