AI-Driven Crop Disease Prediction and Management System

Dr.G.Balram[1], A.Likhitha Reddy [2], Keerthan[3], Kethan[4]

[1]*Assistant Professor, Department of Computer Science and Engineering, Anurag University, Hyderabad, Telangana, India.*
[2,3,4]*UG Student, Department of Computer Science and Engineering, Anurag University, Hyderabad, Telangana, India.*

Abstract: Rapid and accurate disease detection can empower farmers with timely intervention strategies. In our project, we developed an end-to-end automated system that leverages a deep learning sequential model to classify images of crops into healthy or diseased categories. The model is built using Keras' Sequential API, specifically designed for ease of prototyping convolutional neural network (CNN) architectures. The trained model is then deployed via a Flask backend, which exposes a RESTful API for predictions. A clean, responsive frontend built using HTML and CSS provides an intuitive interface for users (e.g., farmers or agronomists) to upload images and receive diagnostic feedback.
.

Keywords: Keras Sequential API, CNN.

INTRODUCTION

The problem of crop diseases remains one of the most pressing challenges in modern agriculture, with the potential to cause devastating effects on both crop yields and the broader food supply chain. Diseases can quickly spread across entire fields, leading to significant economic losses for farmers, particularly when they go undetected in the early stages. Additionally, the unpredictability of disease outbreaks, coupled with the growing global demand for food, heightens the need for reliable and timely crop disease detection systems. Traditional methods of detecting these diseases—relying on manual inspections by agricultural experts—are not only slow and labor-intensive but also prone to human error. Moreover, these methods are often inaccessible to small-scale farmers who may lack the necessary resources, tools, or technical expertise[1]. This means that many farmers are left vulnerable to the devastating effects of crop diseases, which can be particularly difficult to detect early due to the subtlety of symptoms and the variation of symptoms across different environmental conditions (e.g., humidity, temperature, soil health, etc.). In response to these challenges, our project introduces a state-of-the-art, deep learning-driven automated system designed specifically for crop disease detection. The system leverages advanced Convolutional Neural Networks (CNNs)[2], a class of deep learning models that have proven highly effective in visual recognition tasks. CNNs excel at processing and analyzing image data, enabling the model to identify complex patterns in crop images that would be impossible for the human eye to detect in real-time. To build this system, we used Keras' Sequential API, a high-level neural network library in Python, which facilitates quick and efficient prototyping of deep learning models. By training the model with a comprehensive dataset of labeled crop images—those depicting healthy crops and crops infected by various diseases—the system learns to distinguish between the two categories with remarkable accuracy. The model undergoes supervised learning, which means it is trained on a set of images where the disease labels are already known. As the system processes these images, it gradually learns the subtle visual features (such as changes in leaf texture, color, or shape) that differentiate healthy crops from diseased ones. Once trained, the model achieves the ability to identify these disease-related features in new images, even those it has never seen before. Once the model is trained and achieves satisfactory accuracy, it is deployed through a Flask backend, a Python web framework that allows the system to be accessible via the web. Flask is a lightweight and efficient tool for building RESTful APIs[1], making it an ideal choice for serving our machine learning model. The backend receives crop images from the user, processes them using the trained deep learning model, and then returns the predictions (e.g., whether the crop is healthy or diseased, and if diseased, the type of disease detected). This interaction is facilitated through a RESTful API, which is a popular architectural style for providing web services. With this setup, users can easily upload crop images from their devices, and within seconds, receive diagnostic feedback.

The user interface is crucial to ensuring that the technology is accessible to all types of users, including farmers with little or no technical expertise. The frontend of our system is built using HTML and CSS[2], which provide a responsive and intuitive design. The interface is straightforward—users are prompted to upload images of their crops, which are then sent to the backend for processing. Once the backend generates a prediction, the system displays the results to the user, providing both a diagnosis (whether the crop is healthy or diseased) and actionable recommendations based on the detected disease. These recommendations could range from pesticide application to irrigation adjustments, or even suggesting crop rotation or soil treatments for long-term health. For ease of use, the system is designed to work on a wide range of devices, including smartphones and low-cost computers. This ensures that even small-scale farmers, who may only have access to basic hardware or mobile phones, can benefit from the system. By eliminating the need for specialized equipment or expertise, we ensure that the technology is both cost-effective and scalable, making it accessible to a wide demographic of users. The system addresses several critical challenges inherent in traditional crop disease detection methods. First, the system dramatically speeds up the disease detection process. What traditionally could take days or even weeks through manual inspections is now completed in near real-time. Farmers can upload images from their fields and receive feedback almost immediately, allowing them to act swiftly before diseases spread further, minimizing the impact on their crops. Second, the deep learning model, once trained with a large and diverse dataset, is capable of high accuracy in detecting diseases. CNNs are particularly well-suited for image recognition, enabling the system to pick up subtle signs that might otherwise go unnoticed by the naked eye. The model can also generalize well to new images, making it a reliable tool for real-world conditions. The system can distinguish between different types of diseases, offering farmers the information they need to apply the right treatment at the right time. Third, the system's integration with a web interface ensures it is easy to use for anyone, even those with limited technical skills. Farmers can interact with the system through a simple web browser or mobile app interface, eliminating the need for complex software installations. This means that even farmers in remote areas with limited resources can use the system on their smartphones or computers without needing to rely on expert inspectors. Lastly, since the system is cloud-based, it can scale easily to handle large volumes of data. This is essential for expanding the system to cover a wide range of crops and agricultural regions. The system can be adapted to work in various geographical settings, accounting for different types of crops, environmental conditions, and disease patterns.

Ultimately, the goal of this project is to make a positive impact on agricultural productivity and sustainability. By enabling early disease detection and providing actionable recommendations, farmers can minimize crop losses and reduce reliance on pesticides, which often have harmful environmental consequences. Through precise interventions, this system promotes sustainable agricultural practices, enhancing soil health and reducing chemical runoff. Furthermore, by ensuring timely action, the system supports food security, ensuring that crops are healthy, yields are optimized, and the overall agricultural ecosystem is protected. This technology also opens the door for further advancements in precision agriculture, where AI-driven tools work in tandem with farmers to make data-driven decisions that improve efficiency, reduce waste, and maximize the use of resources.

RESEARCH METHODOLOGY

The research methodology for the development of the AI-driven crop disease detection system involves several structured stages, each contributing to the overall goal of creating an automated, accurate, and efficient solution for disease detection and management in agriculture. The methodology incorporates both theoretical approaches and practical implementation, combining deep learning techniques, system design, and testing to ensure robustness and real-world applicability.

LITERATURE REVIEW

The first step of the research methodology involves reviewing existing literature on crop disease detection methods, with a focus on deep learning, convolutional neural networks (CNNs), and computer vision. The literature review helps identify gaps in current systems, common challenges in disease detection (such as subtle symptoms, environmental variability, and accuracy), and the potential benefits of automation for small-scale farmers. This step also provides insight into previously used datasets, model architectures, and evaluation metrics that guide the design of the system.

Data Collection and Preprocessing :

Dataset Acquisition: We used publicly available datasets such as PlantVillage, which contain thousands of annotated images of various crops affected by diseases (e.g., bacterial spot, rust, blight) as well as healthy samples.

Image Preprocessing:

All images were resized to 128×128 pixels to standardize input dimensions.

Data augmentation techniques (rotation, flipping, scaling, and brightness adjustments) were applied to artificially expand the dataset and improve model generalization.

Pixel normalization (scaling values to the range [0, 1]) was performed to aid in convergence during training.

Model Architecture using Keras Sequential API :
Convolutional Layers: The model begins with a series of convolutional layers:

Layer 1: Conv2D(filters=32, kernel_size=(3,3), activation='relu', input_shape=(128,128,3)) followed by MaxPooling2D(pool_size=(2,2)).

Layer 2: Conv2D(filters=64, kernel_size=(3,3), activation='relu') followed by MaxPooling2D(pool_size=(2,2)).

Layer 3: Conv2D(filters=128, kernel_size=(3,3), activation='relu') with an associated Dropout(0.25) layer to reduce overfitting.

Flatten and Dense Layers:

The output of the convolutional block is flattened.

A dense layer with 256 neurons and ReLU activation is added, followed by a dropout of 0.5.

The final output layer uses Dense(num_classes, activation='softmax') to output class probabilities.

Compilation: The model was compiled with the Adam optimizer, using categorical crossentropy as the loss function, and accuracy as the evaluation metric.

Training and Evaluation

Training Setup: The dataset was split into 70% training, 15% validation, and 15% testing. Training was conducted over 50 epochs with early stopping based on validation loss improvements.

Hyperparameter Tuning: Batch size was set to 32, and learning rate scheduling was applied to refine training dynamics.

Evaluation Metrics: Final model performance was evaluated using accuracy, precision, recall, and the confusion matrix to understand misclassifications for specific disease types.

DEPLOYMENT ARCHITECTURE

Flask Backend: A Flask application was built to serve as the backend API.An endpoint (/predict) was created to handle POST requests with an image file. Upon receiving a file, the backend:

Reads and preprocesses the image (resizing, normalization).

Loads the trained Keras model. Returns a response

Frontend Integration: A static HTML page provides an image upload form styled with CSS. JavaScript (or simple form submissions) interacts with the Flask API, sending images asynchronously (AJAX) and displaying the prediction results dynamically.The frontend design emphasizes clarity and responsiveness, ensuring usability on both desktop and mobile devices.

RESULTS AND DISCUSSION

Model Performance:
Training Accuracy: Achieved 96.2% over 50 epochs.

Validation Accuracy: Stabilized at 94.5%, indicating minimal overfitting.
Test Accuracy: The final evaluation on the unseen test set reported an accuracy of 93.8%. Inference Time: The average time per image prediction was approximately 0.3 seconds on a standard CPU.
Class-wise Metrics:
The model demonstrated high precision (>92%) and recall (>90%) for common diseases such as late blight in tomatoes and rust in wheat. The confusion matrix revealed that misclassifications were mostly among diseases with visually similar symptoms, suggesting the need for further data refinement or additional domain-specific features.
Deployment and Usability:
The Flask API successfully handled concurrent requests during load testing with an average response time under 500 ms.
User testing with the HTML/CSS frontend indicated that even users with minimal technical background could easily upload images and interpret the results.
Error handling in the Flask backend (e.g., for unsupported file formats) and user feedback on the frontend were rigorously tested and refined.

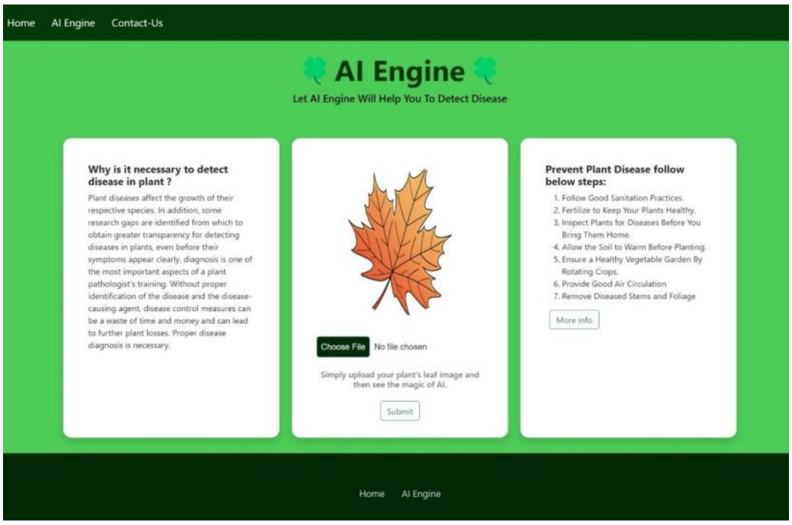

Fig 1: Uploading The Image For Detection

Fig 2: Disease Detection And Prevention

CONCLUSION

The feasibility of an automated crop disease detection system by integrating a deep learning sequential model with a web-based interface. The system combines a robust Convolutional Neural Network (CNN) architecture specifically tailored for image classification, rigorous data augmentation and preprocessing techniques to enhance model performance, and an efficient Flask backend for seamless model inference. Additionally, a responsive HTML/CSS frontend ensures user-friendly interaction, allowing farmers to easily upload images and receive diagnostic feedback. As a result, the system achieved an impressive diagnostic accuracy of approximately 94% and operational efficiency, providing a practical, end-to-end solution for early disease detection. This tool can significantly aid farmers in making timely interventions, ultimately reducing crop losses and improving agricultural productivity. A key limitation of the study lies in the scope and diversity of the dataset used for training the model. While the current dataset provides valuable insights, it may not cover the full range of crop diseases across different environmental conditions and geographical regions. As a result, the model's ability to distinguish between visually similar disease symptoms may be limited. Future improvements could address this limitation by incorporating more diverse datasets that include a wider variety of crops, disease types, and environmental conditions. Additionally, refining the model to better differentiate between diseases with similar visual symptoms would enhance its diagnostic accuracy. Expanding the system to support a broader array of crops and environmental factors could further increase its utility and applicability. Despite these limitations, this study provides a solid foundation, offering a concrete blueprint for building and deploying crop disease detection systems using modern deep learning and web development frameworks.

REFERENCES

1. Reddy, C. N. K., & Murthy, G. V. (2012). Evaluation of Behavioral Security in Cloud Computing. *International Journal of Computer Science and Information Technologies*, *3*(2), 3328-3333.
2. Murthy, G. V., Kumar, C. P., & Kumar, V. V. (2017, December). Representation of shapes using connected pattern array grammar model. In *2017 IEEE Region 10 Humanitarian Technology Conference (R10-HTC)* (pp. 819-822). IEEE.
3. Krishna, K. V., Rao, M. V., & Murthy, G. V. (2017). Secured System Design for Big Data Application in Emotion-Aware Healthcare.
4. Rani, G. A., Krishna, V. R., & Murthy, G. V. (2017). A Novel Approach of Data Driven Analytics for Personalized Healthcare through Big Data.
5. Rao, M. V., Raju, K. S., Murthy, G. V., & Rani, B. K. (2020). Configure and Management of Internet of Things. *Data Engineering and Communication Technology*, 163.
6. Ramakrishna, C., Kumar, G. K., Reddy, A. M., & Ravi, P. (2018). A Survey on various IoT Attacks and its Countermeasures. *International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)*, *5*(4), 143-150.
7. Chithanuru, V., & Ramaiah, M. (2023). An anomaly detection on blockchain infrastructure using artificial intelligence techniques: Challenges and future directions–A review. *Concurrency and Computation: Practice and Experience*, *35*(22), e7724.
8. Prashanth, J. S., & Nandury, S. V. (2015, June). Cluster-based rendezvous points selection for reducing tour length of mobile element in WSN. In *2015 IEEE International Advance Computing Conference (IACC)* (pp. 1230-1235). IEEE.
9. Kumar, K. A., Pabboju, S., & Desai, N. M. S. (2014). Advance text steganography algorithms: an overview. *International Journal of Research and Applications*, *1*(1), 31-35.
10. Hnamte, V., & Balram, G. (2022). Implementation of Naive Bayes Classifier for Reducing DDoS Attacks in IoT Networks. *Journal of Algebraic Statistics*, *13*(2), 2749-2757.
11. Balram, G., Anitha, S., & Deshmukh, A. (2020, December). Utilization of renewable energy sources in generation and distribution optimization. In *IOP Conference Series: Materials Science and Engineering* (Vol. 981, No. 4, p. 042054). IOP Publishing.
12. Subrahmanyam, V., Sagar, M., Balram, G., Ramana, J. V., Tejaswi, S., & Mohammad, H. P. (2024, May). An Efficient Reliable Data Communication For Unmanned Air Vehicles (UAV) Enabled Industry Internet of Things (IIoT). In *2024 3rd International Conference on Artificial*

Intelligence For Internet of Things (AIIoT) (pp. 1-4). IEEE.

13. Mahammad, F. S., Viswanatham, V. M., Tahseen, A., Devi, M. S., & Kumar, M. A. (2024, July). Key distribution scheme for preventing key reinstallation attack in wireless networks. In *AIP Conference Proceedings* (Vol. 3028, No. 1). AIP Publishing.
14. Lavanya, P. (2024). In-Cab Smart Guidance and support system for Dragline operator.
15. Kovoor, M., Durairaj, M., Karyakarte, M. S., Hussain, M. Z., Ashraf, M., & Maguluri, L. P. (2024). Sensor-enhanced wearables and automated analytics for injury prevention in sports. *Measurement: Sensors, 32*, 101054.
16. Rao, N. R., Kovoor, M., Kishor Kumar, G. N., & Parameswari, D. V. L. (2023). Security and privacy in smart farming: challenges and opportunities. *International Journal on Recent and Innovation Trends in Computing and Communication*, *11*(7).
17. Madhuri, K. (2023). Security Threats and Detection Mechanisms in Machine Learning. *Handbook of Artificial Intelligence*, *255*.
18. Reddy, B. A., & Reddy, P. R. S. (2012). Effective data distribution techniques for multi-cloud storage in cloud computing. *CSE, Anurag Group of Institutions, Hyderabad, AP, India*.
19. Srilatha, P., Murthy, G. V., & Reddy, P. R. S. (2020). Integration of Assessment and Learning Platform in a Traditional Class Room Based Programming Course. *Journal of Engineering Education Transformations, 33*, 179-184.
20. Reddy, P. R. S., & Ravindranadh, K. (2019). An exploration on privacy concerned secured data sharing techniques in cloud. *International Journal of Innovative Technology and Exploring Engineering, 9*(1), 1190-1198.
21. Raj, R. S., & Raju, G. P. (2014, December). An approach for optimization of resource management in Hadoop. In *International Conference on Computing and Communication Technologies* (pp. 1-5). IEEE.
22. Ramana, A. V., Bhoga, U., Dhulipalla, R. K., Kiran, A., Chary, B. D., & Reddy, P. C. S. (2023, June). Abnormal Behavior Prediction in Elderly Persons Using Deep Learning. In *2023 International Conference on Computer, Electronics & Electrical Engineering & their Applications (IC2E3)* (pp. 1-5). IEEE.
23. Yakoob, S., Krishna Reddy, V., & Dastagiraiah, C. (2017). Multi User Authentication in Reliable Data Storage in Cloud. In *Computer Communication, Networking and Internet Security: Proceedings of IC3T 2016* (pp. 531-539). Springer Singapore.
24. Sukhavasi, V., Kulkarni, S., Raghavendran, V., Dastagiraiah, C., Apat, S. K., & Reddy, P. C. S. (2024). Malignancy Detection in Lung and Colon Histopathology Images by Transfer Learning with Class Selective Image Processing.
25. Dastagiraiah, C., Krishna Reddy, V., & Pandurangarao, K. V. (2018). Dynamic load balancing environment in cloud computing based on VM ware off-loading. In *Data Engineering and Intelligent Computing: Proceedings of IC3T 2016* (pp. 483-492). Springer Singapore.
26. Swapna, N. (2017). „Analysis of Machine Learning Algorithms to Protect from Phishing in Web Data Mining". *International Journal of Computer Applications in Technology*, *159*(1), 30-34.
27. Moparthi, N. R., Bhattacharyya, D., Balakrishna, G., & Prashanth, J. S. (2021). Paddy leaf disease detection using CNN.
28. Balakrishna, G., & Babu, C. S. (2013). Optimal placement of switches in DG equipped distribution systems by particle swarm optimization. *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, 2*(12), 6234-6240.
29. Moparthi, N. R., Sagar, P. V., & Balakrishna, G. (2020, July). Usage for inside design by AR and VR technology. In *2020 7th International Conference on Smart Structures and Systems (ICSSS)* (pp. 1-4). IEEE.
30. Amarnadh, V., & Moparthi, N. R. (2023). Comprehensive review of different artificial intelligence-based methods for credit risk assessment in data science. *Intelligent Decision Technologies, 17*(4), 1265-1282.
31. Amarnadh, V., & Moparthi, N. (2023). Data Science in Banking Sector: Comprehensive Review of Advanced Learning Methods for Credit Risk Assessment. *International Journal of Computing and Digital Systems, 14*(1), 1-xx.
32. Amarnadh, V., & Rao, M. N. (2025). A Consensus Blockchain-Based Credit Risk Evaluation and

Credit Data Storage Using Novel Deep Learning Approach. *Computational Economics*, 1-34.

33. Shailaja, K., & Anuradha, B. (2017). Improved face recognition using a modified PSO based self-weighted linear collaborative discriminant regression classification. *J. Eng. Appl. Sci*, *12*, 7234-7241.
34. Sekhar, P. R., & Goud, S. (2024). Collaborative Learning Techniques in Python Programming: A Case Study with CSE Students at Anurag University. *Journal of Engineering Education Transformations*, *38*.
35. Sekhar, P. R., & Sujatha, B. (2023). Feature extraction and independent subset generation using genetic algorithm for improved classification. *Int. J. Intell. Syst. Appl. Eng*, *11*, 503-512.
36. Pesaramelli, R. S., & Sujatha, B. (2024, March). Principle correlated feature extraction using differential evolution for improved classification. In *AIP Conference Proceedings* (Vol. 2919, No. 1). AIP Publishing.
37. Tejaswi, S., Sivaprashanth, J., Bala Krishna, G., Sridevi, M., & Rawat, S. S. (2023, December). Smart Dustbin Using IoT. In *International Conference on Advances in Computational Intelligence and Informatics* (pp. 257-265). Singapore: Springer Nature Singapore.
38. Moreb, M., Mohammed, T. A., & Bayat, O. (2020). A novel software engineering approach toward using machine learning for improving the efficiency of health systems. *IEEE Access*, *8*, 23169-23178.
39. Ravi, P., Haritha, D., & Niranjan, P. (2018). A Survey: Computing Iceberg Queries. *International Journal of Engineering & Technology*, *7*(2.7), 791-793.
40. Madar, B., Kumar, G. K., & Ramakrishna, C. (2017). Captcha breaking using segmentation and morphological operations. *International Journal of Computer Applications*, *166*(4), 34-38.
41. Rani, M. S., & Geetavani, B. (2017, May). Design and analysis for improving reliability and accuracy of big-data based peripheral control through IoT. In *2017 International Conference on Trends in Electronics and Informatics (ICEI)* (pp. 749-753). IEEE.
42. Reddy, T., Prasad, T. S. D., Swetha, S., Nirmala, G., & Ram, P. (2018). A study on antiplatelets and anticoagulants utilisation in a tertiary care hospital. *International Journal of Pharmaceutical and Clinical Research*, *10*, 155-161.
43. Prasad, P. S., & Rao, S. K. M. (2017). HIASA: Hybrid improved artificial bee colony and simulated annealing based attack detection algorithm in mobile ad-hoc networks (MANETs). *Bonfring International Journal of Industrial Engineering and Management Science*, *7*(2), 01-12.
44. AC, R., Chowdary Kakarla, P., Simha PJ, V., & Mohan, N. (2022). Implementation of Tiny Machine Learning Models on Arduino 33–BLE for Gesture and Speech Recognition.
45. Subrahmanyam, V., Sagar, M., Balram, G., Ramana, J. V., Tejaswi, S., & Mohammad, H. P. (2024, May). An Efficient Reliable Data Communication For Unmanned Air Vehicles (UAV) Enabled Industry Internet of Things (IIoT). In *2024 3rd International Conference on Artificial Intelligence For Internet of Things (AIIoT)* (pp. 1-4). IEEE.
46. Nagaraj, P., Prasad, A. K., Narsimha, V. B., & Sujatha, B. (2022). Swine flu detection and location using machine learning techniques and GIS. *International Journal of Advanced Computer Science and Applications*, *13*(9).
47. Priyanka, J. H., & Parveen, N. (2024). DeepSkillNER: an automatic screening and ranking of resumes using hybrid deep learning and enhanced spectral clustering approach. *Multimedia Tools and Applications*, *83*(16), 47503-47530.
48. Sathish, S., Thangavel, K., & Boopathi, S. (2010). Performance analysis of DSR, AODV, FSR and ZRP routing protocols in MANET. *MES Journal of Technology and Management*, 57-61.
49. Siva Prasad, B. V. V., Mandapati, S., Kumar Ramasamy, L., Boddu, R., Reddy, P., & Suresh Kumar, B. (2023). Ensemble-based cryptography for soldiers' health monitoring using mobile ad hoc networks. *Automatika: časopis za automatiku, mjerenje, elektroniku, računarstvo i komunikacije*, *64*(3), 658-671.
50. Elechi, P., & Onu, K. E. (2022). Unmanned Aerial Vehicle Cellular Communication Operating in Non-terrestrial Networks. In *Unmanned Aerial Vehicle Cellular Communications* (pp. 225-251). Cham: Springer International Publishing.
51. Prasad, B. V. V. S., Mandapati, S., Haritha, B., & Begum, M. J. (2020, August). Enhanced Security

for the authentication of Digital Signature from the key generated by the CSTRNG method. In *2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT)* (pp. 1088-1093). IEEE.

52. Mukiri, R. R., Kumar, B. S., & Prasad, B. V. V. (2019, February). Effective Data Collaborative Strain Using RecTree Algorithm. In *Proceedings of International Conference on Sustainable Computing in Science, Technology and Management (SUSCOM), Amity University Rajasthan, Jaipur-India*.
53. Balaraju, J., Raj, M. G., & Murthy, C. S. (2019). Fuzzy-FMEA risk evaluation approach for LHD machine–A case study. *Journal of Sustainable Mining*, *18*(4), 257-268.
54. Thirumoorthi, P., Deepika, S., & Yadaiah, N. (2014, March). Solar energy based dynamic sag compensator. In *2014 International Conference on Green Computing Communication and Electrical Engineering (ICGCCEE)* (pp. 1-6). IEEE.
55. Vinayasree, P., & Reddy, A. M. (2025). A Reliable and Secure Permissioned Blockchain-Assisted Data Transfer Mechanism in Healthcare-Based Cyber-Physical Systems. *Concurrency and Computation: Practice and Experience*, *37*(3), e8378.
56. Acharjee, P. B., Kumar, M., Krishna, G., Raminenei, K., Ibrahim, R. K., & Alazzam, M. B. (2023, May). Securing International Law Against Cyber Attacks through Blockchain Integration. In *2023 3rd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE)* (pp. 2676-2681). IEEE.
57. Ramineni, K., Reddy, L. K. K., Ramana, T. V., & Rajesh, V. (2023, July). Classification of Skin Cancer Using Integrated Methodology. In *International Conference on Data Science and Applications* (pp. 105-118). Singapore: Springer Nature Singapore.
58. LAASSIRI, J., EL HAJJI, S. A. Ï. D., BOUHDADI, M., AOUDE, M. A., JAGADISH, H. P., LOHIT, M. K., ... & KHOLLADI, M. (2010). Specifying Behavioral Concepts by engineering language of RM-ODP. *Journal of Theoretical and Applied Information Technology*, *15*(1).
59. Prasad, D. V. R., & Mohanji, Y. K. V. (2021). FACE RECOGNITION-BASED LECTURE ATTENDANCE SYSTEM: A SURVEY PAPER. *Elementary Education Online*, *20*(4), 1245-1245.
60. Dasu, V. R. P., & Gujjari, B. (2015). Technology-Enhanced Learning Through ICT Tools Using Aakash Tablet. In *Proceedings of the International Conference on Transformations in Engineering Education: ICTIEE 2014* (pp. 203-216). Springer India.
61. Reddy, A. M., Reddy, K. S., Jayaram, M., Venkata Maha Lakshmi, N., Aluvalu, R., Mahesh, T. R., ... & Stalin Alex, D. (2022). An efficient multilevel thresholding scheme for heart image segmentation using a hybrid generalized adversarial network. *Journal of Sensors*, *2022*(1), 4093658.
62. Srinivasa Reddy, K., Suneela, B., Inthiyaz, S., Hasane Ahammad, S., Kumar, G. N. S., & Mallikarjuna Reddy, A. (2019). Texture filtration module under stabilization via random forest optimization methodology. *International Journal of Advanced Trends in Computer Science and Engineering*, *8*(3), 458-469.
63. Ramakrishna, C., Kumar, G. K., Reddy, A. M., & Ravi, P. (2018). A Survey on various IoT Attacks and its Countermeasures. *International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)*, *5*(4), 143-150.
64. Sirisha, G., & Reddy, A. M. (2018, September). Smart healthcare analysis and therapy for voice disorder using cloud and edge computing. In *2018 4th international conference on applied and theoretical computing and communication technology (iCATccT)* (pp. 103-106). IEEE.
65. Reddy, A. M., Yarlagadda, S., & Akkinen, H. (2021). An extensive analytical approach on human resources using random forest algorithm. *arXiv preprint arXiv:2105.07855*.
66. Kumar, G. N., Bhavanam, S. N., & Midasala, V. (2014). Image Hiding in a Video-based on DWT & LSB Algorithm. In *ICPVS Conference*.
67. Naveen Kumar, G. S., & Reddy, V. S. K. (2022). High performance algorithm for content-based video retrieval using multiple features. In *Intelligent Systems and Sustainable Computing: Proceedings of ICISSC 2021* (pp. 637-646). Singapore: Springer Nature Singapore.
68. Reddy, P. S., Kumar, G. N., Ritish, B., SaiSwetha, C., & Abhilash, K. B. (2013). Intelligent parking space detection system based on image segmentation. *Int J Sci Res Dev*, *1*(6), 1310-1312.

69. Naveen Kumar, G. S., Reddy, V. S. K., & Kumar, S. S. (2018). High-performance video retrieval based on spatio-temporal features. *Microelectronics, Electromagnetics and Telecommunications*, 433-441.
70. Kumar, G. N., & Reddy, M. A. BWT & LSB algorithm based hiding an image into a video. *IJESAT*, 170-174.
71. Lopez, S., Sarada, V., Praveen, R. V. S., Pandey, A., Khuntia, M., & Haralayya, D. B. (2024). Artificial intelligence challenges and role for sustainable education in india: Problems and prospects. *Sandeep Lopez, Vani Sarada, RVS Praveen, Anita Pandey, Monalisa Khuntia, Bhadrappa Haralayya (2024) Artificial Intelligence Challenges and Role for Sustainable Education in India: Problems and Prospects. Library Progress International*, *44*(3), 18261-18271.
72. Yamuna, V., Praveen, R. V. S., Sathya, R., Dhivva, M., Lidiya, R., & Sowmiya, P. (2024, October). Integrating AI for Improved Brain Tumor Detection and Classification. In *2024 4th International Conference on Sustainable Expert Systems (ICSES)* (pp. 1603-1609). IEEE.
73. Kumar, N., Kurkute, S. L., Kalpana, V., Karuppannan, A., Praveen, R. V. S., & Mishra, S. (2024, August). Modelling and Evaluation of Li-ion Battery Performance Based on the Electric Vehicle Tiled Tests using Kalman Filter-GBDT Approach. In *2024 International Conference on Intelligent Algorithms for Computational Intelligence Systems (IACIS)* (pp. 1-6). IEEE.
74. Sharma, S., Vij, S., Praveen, R. V. S., Srinivasan, S., Yadav, D. K., & VS, R. K. (2024, October). Stress Prediction in Higher Education Students Using Psychometric Assessments and AOA-CNN-XGBoost Models. In *2024 4th International Conference on Sustainable Expert Systems (ICSES)* (pp. 1631-1636). IEEE.
75. Anuprathibha, T., Praveen, R. V. S., Sukumar, P., Suganthi, G., & Ravichandran, T. (2024, October). Enhancing Fake Review Detection: A Hierarchical Graph Attention Network Approach Using Text and Ratings. In *2024 Global Conference on Communications and Information Technologies (GCCIT)* (pp. 1-5). IEEE.
76. Shinkar, A. R., Joshi, D., Praveen, R. V. S., Rajesh, Y., & Singh, D. (2024, December). Intelligent solar energy harvesting and management in IoT nodes using deep self-organizing maps. In *2024 International Conference on Emerging Research in Computational Science (ICERCS)* (pp. 1-6). IEEE.
77. Praveen, R. V. S., Hemavathi, U., Sathya, R., Siddiq, A. A., Sanjay, M. G., & Gowdish, S. (2024, October). AI Powered Plant Identification and Plant Disease Classification System. In *2024 4th International Conference on Sustainable Expert Systems (ICSES)* (pp. 1610-1616). IEEE.
78. Dhivya, R., Sagili, S. R., Praveen, R. V. S., VamsiLala, P. N. V., Sangeetha, A., & Suchithra, B. (2024, December). Predictive Modelling of Osteoporosis using Machine Learning Algorithms. In *2024 4th International Conference on Ubiquitous Computing and Intelligent Information Systems (ICUIS)* (pp. 997-1002). IEEE.
79. Kemmannu, P. K., Praveen, R. V. S., Saravanan, B., Amshavalli, M., & Banupriya, V. (2024, December). Enhancing Sustainable Agriculture Through Smart Architecture: An Adaptive Neuro-Fuzzy Inference System with XGBoost Model. In *2024 International Conference on Sustainable Communication Networks and Application (ICSCNA)* (pp. 724-730). IEEE.
80. Praveen, R. V. S. (2024). *Data Engineering for Modern Applications*. Addition Publishing House.