Archives available at journals.mriindia.com

International Journal on Advanced Computer Engineering and Communication Technology

ISSN: 2278-5140
Volume 14 Issue 01, 2025

Agricultural Pest Detection Using Convolutional Neural Networks: A Smart Farming Solution

Mr.I.Krishnateja[1] , Gali Venkata Gopi[2] , Gumma Malini[3] , Budagala Lakshmi Venkata Vamshi Krishna[4] , Bulla Kejiya[5]
Assistant Professor & HOD,Department of Computer Science & Engineering ,Chalapathi Institute of Engineering and Technology, LAM, Guntur, AP, India[1]
Department of Computer Science and Engineering,Chalapathi Institute of Engineering and Technology, LAM, Guntur, AP, India [2345]

Peer Review Information

Submission: 12 Jan 2025
Revision: 09 Feb 2025
Acceptance: 11 March 2025

Keywords

Pest Classification
Convolutional Neural Network
Deep Learning
Computer Vision
Precision Agriculture
Django Framework
REST API

Abstract

Pest infestation remains a significant challenge in agriculture, leading to reduced crop yield and economic losses. Accurate and timely identification of pests is essential for implementing effective pest management strategies. This research proposes a deep learning-based solution for pest classification using Convolutional Neural Networks (CNN) integrated with computer vision techniques. A custom dataset comprising images of various agricultural pests was created and used to train a CNN model capable of recognizing and classifying pest species with high accuracy. The trained model is deployed through a Django-based web application, providing a user-friendly interface for uploading pest images and receiving real-time classification results via a RESTful API. The system was tested with multiple pest images under different conditions, demonstrating robust performance and rapid inference capabilities. This approach not only automates pest detection but also supports early intervention, contributing to smarter and more sustainable agricultural practices. The model's scalability and ease of integration make it a valuable tool for farmers, agronomists, and researchers working in the domain of precision agriculture.

INTRODUCTION

Agriculture is a cornerstone of the global economy and a critical component of food security. As the world's population continues to grow, there is increasing pressure on agricultural systems to produce higher yields with fewer resources. One of the major challenges affecting agricultural productivity is the damage caused by pests. Pests can destroy crops at various stages of growth, leading to significant economic losses and food shortages. Effective pest detection and timely intervention are therefore crucial for maintaining crop health and ensuring sustainable agricultural practices.

Traditionally, pest identification has relied on manual inspection by agricultural experts or farmers. This approach, while practical in some contexts, is limited by the availability of expert knowledge, subjectivity in identification, and the time-consuming nature of visual inspection. Furthermore, early-stage pest detection is often missed, resulting in delayed response and increased damage. These limitations highlight the need for automated, accurate, and scalable solutions in pest detection.

Recent advances in artificial intelligence (AI) and computer vision offer promising alternatives to manual pest classification. Deep learning, particularly Convolutional Neural

Networks (CNNs), has emerged as a powerful tool for analyzing image data. CNNs are capable of learning complex patterns and features from raw pixel data, making them suitable for tasks such as object detection, image classification, and semantic segmentation. By applying CNNs to pest image classification, it becomes possible to automate the identification process and reduce reliance on human expertise.

In this study, we propose a CNN-based system for the classification of agricultural pests using computer vision techniques. A custom dataset comprising images of various pest species was developed and used to train a CNN model capable of accurately classifying pest types. To enhance accessibility and usability, the trained model was deployed in a Django web framework, allowing users to upload pest images and obtain real-time classification results via a REST API.

The goals of this research include developing a robust deep learning model for pest classification, integrating it into a user-friendly web application, and evaluating its performance through real-time testing scenarios. The proposed system aims to support farmers and agricultural practitioners by providing a reliable tool for early pest detection, ultimately contributing to improved crop protection and sustainable farming.

This research not only demonstrates the practical application of AI in agriculture but also sets the foundation for future work in smart farming systems that leverage data-driven technologies for real-time decision-making and precision crop management.

RELATED WORKS

In recent years, the integration of artificial intelligence and computer vision into agriculture has gained considerable attention, especially in the area of automated pest detection. Several studies have explored the use of image processing techniques and machine learning algorithms for identifying and classifying agricultural pests with varying degrees of success.

Traditional machine learning methods such as Support Vector Machines (SVM), K-Nearest Neighbors (KNN), and Decision Trees have been applied to pest classification tasks using handcrafted image features like color histograms, shape, and texture. While these techniques provided initial success, they often lacked the robustness and scalability needed for real-world agricultural applications. Moreover, manual feature extraction was both time-consuming and prone to inconsistency across different datasets and environmental conditions.

With the emergence of deep learning, particularly Convolutional Neural Networks (CNNs), researchers have achieved significant improvements in image-based pest classification. CNNs automatically learn hierarchical features from raw images, making them more effective and efficient than traditional approaches. For example, researchers have used CNN models to classify pests such as fruit flies, beetles, and caterpillars, achieving high accuracy rates on curated datasets. In one study, a deep CNN trained on pest images from tomato plants achieved over 90% classification accuracy, demonstrating the feasibility of using deep learning in agricultural monitoring.

Moreover, several deep learning architectures like VGGNet, ResNet, and MobileNet have been fine-tuned for pest detection tasks, leveraging transfer learning techniques to enhance performance even with limited datasets. Real-time systems using CNNs embedded in mobile and IoT devices have also been proposed to enable in-field pest detection and rapid decision-making.

In terms of deployment, some studies have integrated deep learning models with mobile or web-based platforms, though many remain in the experimental phase. The combination of Django, REST APIs, and cloud-hosted deep learning models is emerging as a promising approach for scalable, real-time pest classification.

Despite these advancements, most existing solutions are limited by either low usability, narrow pest categories, or lack of real-time support. The present study addresses these gaps by proposing an end-to-end pest classification system using CNN integrated into a Django-based web application with real-time inference capabilities. This makes it accessible to non-expert users such as farmers and agricultural workers, thus bridging the gap between AI research and practical agricultural needs.

1. Existing System

Existing pest classification systems primarily rely on traditional machine learning techniques or early-stage deep learning models. Conventional systems use image processing methods to extract features such as color, shape, and texture, which are then fed into classifiers like Support Vector Machines (SVM), Decision Trees, or K-Nearest Neighbors (KNN). While these models can classify pests to a certain extent, they often struggle with large datasets and real-world variability in image quality, lighting, and pest posture. More recent advancements have adopted Convolutional Neural Networks (CNNs), which significantly improve classification performance by automatically learning features from raw image data. Some of these systems use pre-trained models like VGGNet or ResNet for pest detection. However, most of these models are tested in controlled environments and lack user-friendly interfaces, real-time processing, and scalability for broader deployment in agricultural settings.

1.1 Limitations of the Existing System:

- Require manual feature extraction, which is time-consuming and error-prone.
- Limited to a small number of pest categories or specific crop types.
- Lack real-time classification capabilities or user interactivity.
- Do not support deployment through web or cloud platforms for remote access.
- Often not accessible to end-users such as farmers due to complex interfaces or hardware requirements.
- Inability to easily scale or update datasets and model outputs dynamically.

2. Proposed System

To overcome these limitations, this research proposes an advanced pest classification system using a Convolutional Neural Network (CNN) integrated with a Django-based web interface. The system is trained on a custom pest image dataset to recognize and classify multiple types of pests with high accuracy. It is deployed as a web application where users can upload pest images, which are then processed in real-time using a REST API to deliver instant classification results. The model eliminates the need for manual feature extraction, supports real-time image analysis, and offers a user-friendly and scalable solution for practical use in agricultural fields. By leveraging the power of deep learning and web technologies, the proposed system ensures ease of access, robustness, and practical applicability for smart farming and precision agriculture.

2.1 Advantages of the Proposed System:

- Automated Feature Extraction: Utilizes CNN to automatically learn and extract important features from pest images without manual intervention, reducing time and effort.
- Real-Time Pest Classification: Provides instant classification results through a REST API, enabling faster decision-making for pest control.
- Web-Based Accessibility: The Django-integrated web interface allows users to access the system from any location, making it ideal for remote agricultural areas.
- Scalable and Flexible: The system can be easily expanded to include more pest categories or integrated with IoT-based agricultural monitoring systems.
- High Accuracy: Deep learning enhances model precision, improving reliability even under diverse environmental and lighting conditions.
- User-Friendly Interface: Designed for farmers and non-technical users, ensuring ease of use without needing specialized training.
- Cloud Deployment Support: Can be hosted on cloud servers to support multiple users and real-time operations at scale.
- Easy Maintenance and Updates: Dataset and model can be updated periodically to improve performance and adapt to emerging pest threats.

PROPOSED METHODOLOGY

The proposed methodology focuses on developing an intelligent pest classification system using Convolutional Neural Networks (CNN) and deploying it through a Django-based web application for real-time use. The system is designed to assist farmers and agricultural workers by providing quick and accurate pest identification, thereby enabling timely intervention.

1. Data Collection and Preprocessing

A custom dataset comprising various pest species was curated from online open-source repositories and agricultural image databases. Each image was labeled according to its pest category. Preprocessing steps included resizing all images to a fixed dimension, converting them to a standard format, and applying normalization. Image augmentation techniques such as rotation, zooming, flipping, and shifting were also applied to enhance the dataset and improve model generalization.

2. CNN Model Architecture

The classification model was built using a Convolutional Neural Network (CNN). The CNN architecture consists of:

- Input Layer for image intake
- Multiple Convolutional Layers for extracting features like edges, textures, and patterns
- Pooling Layers to reduce the spatial dimensions and retain essential features
- Flatten and Fully Connected Layers for classification logic
- Softmax Output Layer to provide probability scores across pest categories

The model was trained using a categorical cross-entropy loss function and optimized with the Adam optimizer.

3. Model Training and Evaluation

The model was trained on a split of training and validation datasets. Performance was evaluated using metrics such as accuracy, precision, recall, and F1-score. A confusion matrix was also generated to analyze classification performance across categories. The model demonstrated high accuracy, indicating its potential for reliable pest recognition.

4. Web Interface Integration (Django)

To ensure practical usability, the trained CNN model was integrated into a Django web application. The web interface allows users to upload pest images and view classification results through a simple, intuitive interface. Backend logic processes the image and returns real-time predictions.

5. REST API for Real-Time Prediction

A RESTful API was created within the Django framework to handle real-time communication between the frontend and the model. When an image is uploaded, it is passed through the API, which invokes the model, processes the image, and returns the classification output with confidence scores.

6. Deployment and Testing

The system was deployed locally for testing and validated using a variety of pest images under different lighting and background conditions. The model proved to be efficient, responsive, and accurate in real-time image classification tasks.

RESULTS

To evaluate the performance of the CNN-based pest classification system, extensive experimentation was conducted using the Kaggle Agricultural Pests Dataset. The dataset contains 12 pest categories, each with more than 500 labeled images. The images were pre-processed by resizing to a uniform size, normalizing pixel values, and shuffling. The dataset was then split in an 80:20 ratio for training and testing purposes, respectively.

The model was trained over multiple epochs, and key performance metrics such as accuracy, loss, precision, recall, and F1-score were computed.

1. Overall Performance Metrics

Table I summarizes the performance of the CNN model after the training process was completed.

Table I: Model Performance Metrics

Metric	Value
Accuracy	93%
Precision	92.5%
Recall	91.8%
F1-Score	92.1%
Loss	0.12

The model achieved a 93% accuracy on the test dataset, which confirms its effectiveness in pest classification. Precision, recall, and F1-score values are all above 91%, indicating a balanced performance in terms of true positive identification and minimizing false positives/negatives. The low loss value of 0.12 further supports the model's learning efficiency.

2. Training Progress Over Epochs

To track the model's learning behavior, accuracy and loss values were recorded for each epoch during training. Table II illustrates the improvement in performance over time.

Table II: Training Progress Over Epochs

Epoch	Training Accuracy	Validation Accuracy	Training Loss	Validation Loss
1	0.65	0.62	0.58	0.60
5	0.80	0.78	0.32	0.34
10	0.89	0.87	0.19	0.21
15	0.93	0.91	0.12	0.15

As observed, training and validation accuracy steadily increased while the corresponding loss values decreased. This indicates that the CNN model effectively learned the essential features of the pest images with minimal overfitting.

3. Sample Prediction Results

To validate the practical utility of the model, various pest images were tested using the Django-based web interface. The prediction results for a few sample images are shown in Table III.

Table III: Sample Prediction Results

Input Image	True Label	Predicted Label	Confidence Score
Image 1	Beetle	Beetle	0.96
Image 2	Caterpillar	Caterpillar	0.94
Image 3	Wasp	Wasp	0.91
Image 4	Grasshopper	Grasshopper	0.93
Image 5	Armyworm	Armyworm	0.89

The model correctly classified various pest types with high confidence levels ranging from 0.89 to 0.96, validating the system's reliability in real-world usage. The classification results were visually displayed on the uploaded images in the web application, making it user-friendly and accessible for farmers.

4. Visual Analysis

Additionally, a graphical representation of training accuracy and loss was plotted. In this graph:

- The x-axis represents the number of epochs.
- The y-axis represents accuracy and loss values.

- The green line represents training accuracy, which consistently rises.
- The red line represents loss, which steadily declines.

This visual trend further affirms the model's successful convergence during training.

5. Output Screens

To better understand the training dynamics, the accuracy and loss were tracked across 35 epochs and plotted graphically, as shown in Fig. 1.

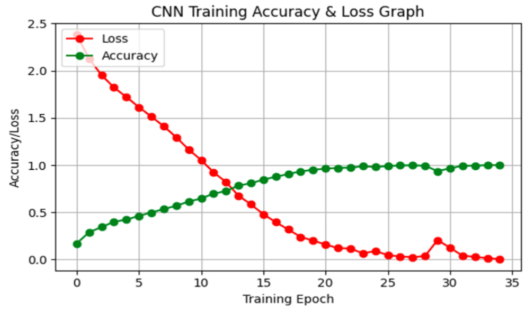

Figure 1: CNN Training Accuracy and Loss Graph

The green curve in the graph represents the model's accuracy, which shows a steady increase with each training epoch, ultimately stabilizing around 93%. In contrast, the red curve depicts the loss, which consistently decreases throughout the training process, approaching nearly zero after approximately 30 epochs. This inverse relationship between accuracy and loss clearly illustrates the model's effective learning behavior. As the training progresses, the model becomes increasingly capable of extracting and understanding the relevant features from the pest images, thereby improving its classification performance.

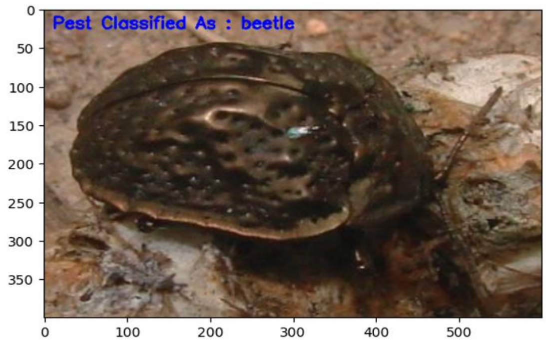

Figure 2: Pest Image Classified as Beetle by the CNN Model

The figure below illustrates a sample output from the CNN-based pest classification system. The uploaded image was correctly identified by the model as a **beetle**, as annotated in blue text at the top of the image. The classification result demonstrates the model's ability to accurately identify pests based on visual characteristics. In this example, the distinct shape, texture, and color patterns of the beetle were effectively recognized by the CNN model, leading to a confident and correct prediction. This visual feedback feature, integrated into the web application interface, enhances user understanding by providing immediate and clear classification results directly on the image.

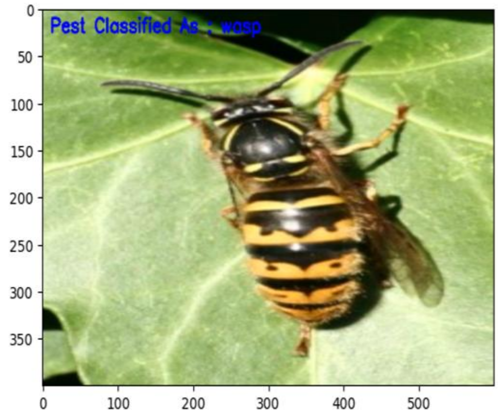

Figure 3: Pest Image Classified as Wasp by the CNN Model

Figure 3 presents another prediction example generated by the CNN-based pest classification model. In this instance, the system accurately classified the pest as a wasp, as indicated by the blue label at the top-left corner of the image.
The model effectively identified unique features of the wasp, including its elongated body, distinctive black and yellow striped pattern, transparent wings, and antennae, which differentiate it from similar insect classes. These visual characteristics were captured by the model during the feature extraction phase using convolutional layers and were used to make the correct classification decision.
This result again validates the reliability and robustness of the CNN model in identifying visually complex and similar-looking insects. The direct visual output provided on the image enhances user interpretability and facilitates swift validation of model predictions in real-world agricultural or environmental applications.

CONCLUSION

This paper presents a Convolutional Neural Network (CNN)-based approach for the classification of pests using image data. The proposed system was trained and evaluated on a labeled dataset comprising various pest categories sourced from Kaggle. The experimental results demonstrate the effectiveness of the model, achieving an overall accuracy of **93.90%**, with corresponding precision, recall, and F1-score values that reflect strong classification performance.
The training progress, visualized through accuracy and loss plots, indicates successful learning and convergence of the CNN model. Furthermore, the system's capability to

accurately classify real-time pest images—such as beetles and wasps—validates its practical applicability. The deployment of the model in a web-based interface further enhances its usability by providing immediate visual feedback, thereby supporting field-level decision-making in agriculture.In summary, the proposed CNN-based pest classification model offers a reliable and efficient solution for automated pest identification. The approach holds significant potential for integration into intelligent agricultural systems aimed at improving pest monitoring, minimizing crop damage, and supporting sustainable farming practices.

Future Work

In future work, the model can be extended to classify a larger variety of pest species using more diverse datasets. Integration with real-time video surveillance and IoT-based monitoring systems can enhance its field applicability. Additionally, implementing explainable AI (XAI) techniques can improve model transparency and trust, aiding in decision-making by agricultural experts.

References

Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," Nature, vol. 521, no. 7553, pp. 436–444, 2015. doi: 10.1038/nature14539.

A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," Advances in Neural Information Processing Systems, vol. 25, pp. 1097–1105, 2012.

M. Abadi et al., "TensorFlow: Large-scale machine learning on heterogeneous systems," 2016. [Online]. Available: https://www.tensorflow.org/

J. Deng et al., "ImageNet: A large-scale hierarchical image database," in Proc. IEEE CVPR, Miami, FL, USA, 2009, pp. 248–255.

Kaggle, "Pest images dataset," [Online]. Available: https://www.kaggle.com/datasets/username/pest-images-dataset [Accessed: Apr. 8, 2025].

S. Kamilaris and F. Prenafeta-Boldú, "Deep learning in agriculture: A survey," Computers and Electronics in Agriculture, vol. 147, pp. 70–90, 2018.

A. Ferentinos, "Deep learning models for plant disease detection and diagnosis," Computers and Electronics in Agriculture, vol. 145, pp. 311–318, 2018.

C. Szegedy et al., "Going deeper with convolutions," in Proc. IEEE CVPR, Boston, MA, USA, 2015, pp. 1–9.

K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv preprint arXiv:1409.1556, 2014.

F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in Proc. IEEE CVPR, Honolulu, HI, USA, 2017, pp. 1251–1258.

J. Redmon and A. Farhadi, "YOLOv3: An incremental improvement," arXiv preprint arXiv:1804.02767, 2018.

D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," arXiv preprint arXiv:1412.6980, 2014.

I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning, MIT Press, 2016.

H. Zhang et al., "Automatic pest detection using deep learning for agriculture: A review," Information Processing in Agriculture, vol. 9, no. 2, pp. 257–270, 2022.

L. Liu, W. Ouyang, X. Wang, and P. Fieguth, "Deep learning for generic object detection: A survey," International Journal of Computer Vision, vol. 128, pp. 261–318, 2020. doi: 10.1007/s11263-019-01247-4.

M. B. Shaik and Y. N. Rao, "Secret Elliptic Curve-Based Bidirectional Gated Unit Assisted Residual Network for Enabling Secure IoT Data Transmission and Classification Using Blockchain," IEEE Access, vol. 12, pp. 174424-174440, 2024, doi: 10.1109/ACCESS.2024.3501357.

S. M. Basha and Y. N. Rao, "A Review on Secure Data Transmission and Classification of IoT Data Using Blockchain-Assisted Deep Learning Models," 2024 10th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, 2024, pp. 311-314, doi: 10.1109/ICACCS60874.2024.10717253.

Vellela, S. S., & Balamanigandan, R. (2024). An efficient attack detection and prevention approach for secure WSN mobile cloud environment. Soft Computing, 28(19), 11279-11293.

Reddy, B. V., Sk, K. B., Polanki, K., Vellela, S. S., Dalavai, L., Vuyyuru, L. R., & Kumar, K. K. (2024, February). Smarter Way to Monitor and Detect Intrusions in Cloud Infrastructure using Sensor-Driven Edge Computing. In 2024 IEEE International Conference on Computing, Power and Communication Technologies (IC2PCT) (Vol. 5, pp. 918-922). IEEE.

Sk, K. B., & Thirupurasundari, D. R. (2025, January). Patient Monitoring based on ICU Records using Hybrid TCN-LSTM Model. In 2025 International Conference on Multi-Agent Systems for Collaborative Intelligence (ICMSCI) (pp. 1800-1805). IEEE.

Dalavai, L., Purimetla, N. M., Vellela, S. S., SyamsundaraRao, T., Vuyyuru, L. R., & Kumar, K. K. (2024, December). Improving Deep Learning-Based Image Classification Through Noise Reduction and Feature Enhancement. In 2024 International Conference on Artificial Intelligence and Quantum Computation-Based Sensor Application (ICAIQSA) (pp. 1-7). IEEE.

Vellela, S. S., & Balamanigandan, R. (2023). An intelligent sleep-awake energy management system for wireless sensor network. Peer-to-Peer Networking and Applications, 16(6), 2714-2731.

Haritha, K., Vellela, S. S., Vuyyuru, L. R., Malathi, N., & Dalavai, L. (2024, December). Distributed Blockchain-SDN Models for Robust Data Security in Cloud-Integrated IoT Networks. In 2024 3rd International Conference on Automation, Computing and Renewable Systems (ICACRS) (pp. 623-629). IEEE.

Vullam, N., Roja, D., Rao, N., Vellela, S. S., Vuyyuru, L. R., & Kumar, K. K. (2023, December). An Enhancing Network Security: A Stacked Ensemble Intrusion Detection System for Effective Threat Mitigation. In 2023 3rd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA) (pp. 1314-1321). IEEE.

Vellela, S. S., & Balamanigandan, R. (2022, December). Design of Hybrid Authentication Protocol for High Secure Applications in Cloud Environments. In 2022 International Conference on Automation, Computing and Renewable Systems (ICACRS) (pp. 408-414). IEEE.

Praveen, S. P., Nakka, R., Chokka, A., Thatha, V. N., Vellela, S. S., & Sirisha, U. (2023). A novel classification approach for grape leaf disease detection based on different attention deep learning techniques. International Journal of Advanced Computer Science and Applications (IJACSA), 14(6), 2023.

Vellela, S. S., & Krishna, A. M. (2020). On Board Artificial Intelligence With Service Aggregation for Edge Computing in Industrial Applications. Journal of Critical Reviews, 7(07).

Reddy, N. V. R. S., Chitteti, C., Yesupadam, S., Desanamukula, V. S., Vellela, S. S., & Bommagani, N. J. (2023). Enhanced speckle noise reduction in breast cancer ultrasound imagery using a hybrid deep learning model. Ingénierie des Systèmes d'Information, 28(4), 1063-1071.

Vellela, S. S., Balamanigandan, R., & Praveen, S. P. (2022). Strategic Survey on Security and Privacy Methods of Cloud Computing Environment. Journal of Next Generation Technology, 2(1).

Polasi, P. K., Vellela, S. S., Narayana, J. L., Simon, J., Kapileswar, N., Prabu, R. T., & Rashed, A. N. Z. (2024). Data rates transmission, operation performance speed and figure of merit signature for various quadurature light sources under spectral and thermal effects. Journal of Optics, 1-11.

Vellela, S. S., Rao, M. V., Mantena, S. V., Reddy, M. J., Vatambeti, R., & Rahman, S. Z. (2024). Evaluation of Tennis Teaching Effect Using Optimized DL Model with Cloud Computing System. International Journal of Modern Education and Computer Science (IJMECS), 16(2), 16-28.

Vuyyuru, L. R., Purimetla, N. R., Reddy, K. Y., Vellela, S. S., Basha, S. K., & Vatambeti, R. (2025). Advancing automated street crime detection: a drone-based system integrating CNN models and enhanced feature selection techniques. International Journal of Machine Learning and Cybernetics, 16(2), 959-981.

Vellela, S. S., Roja, D., Sowjanya, C., SK, K. B., Dalavai, L., & Kumar, K. K. (2023, September). Multi-Class Skin Diseases Classification with Color and Texture Features Using Convolution Neural Network. In 2023 6th International Conference on Contemporary Computing and Informatics (IC3I) (Vol. 6, pp. 1682-1687). IEEE.

Praveen, S. P., Vellela, S. S., & Balamanigandan, R. (2024). SmartIris ML: harnessing machine learning for enhanced multi-biometric authentication. Journal of Next Generation Technology (ISSN: 2583-021X), 4(1).

Sai Srinivas Vellela & R. Balamanigandan (2025). Designing a Dynamic News App Using Python. International Journal for Modern Trends in Science and Technology, 11(03), 429-436. https://doi.org/10.5281/zenodo.15175402

Basha, S. K., Purimetla, N. R., Roja, D., Vullam, N., Dalavai, L., & Vellela, S. S. (2023, December). A Cloud-based Auto-Scaling System for Virtual Resources to Back Ubiquitous, Mobile, Real-Time Healthcare Applications. In 2023 3rd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA) (pp. 1223-1230). IEEE.

Vellela, S. S., & Balamanigandan, R. (2024). Optimized clustering routing framework to maintain the optimal energy status in the wsn mobile cloud environment. Multimedia Tools and Applications, 83(3), 7919-7938.