Journal of Advanced Research in Embedded System
Volume 12, Issue 3&4 - 2025, Pg. No. 1-6
Peer Reviewed Journal

Research Article

Detection of Red Chili from Plant Images

Kasula Shivani[1], T. Satya Savithri[2]

[1]Student, Department of Electronics and Communication Engineering ,JNTUH University College of Engineering, Science & Technology Hyderabad, Telangana, India
[2]Professor, Department of Electronics and Communication Engineering, JNTUH University College of Engineering, Science & Technology Hyderabad, Telangana, India

I N F O

Corresponding Author :
Kasula Shivani, Department of Electronics and Communication Engineering ,JNTUH University College of Engineering, Science & Technology Hyderabad, Telangana, India
E-mail Id:
kasulashivani8@gmail.com
Orcid Id:
http://orcid.org/ 0009-0003-1621-0117
How to cite this article:
Shivani K, Savithri T S, Detection of Red Chili from Plant Images. *J Adv Res Embed Sys* 2025; 12(3&4): 1-6.

Date of Submission: 2025-10-07
Date of Acceptance: 2025-11-11

A B S T R A C T

Manual identification of red chillies is often labour-intensive and inconsistent in large-scale harvesting, highlighting the need for automated, real-time agricultural systems. This study presents a lightweight red chilli detection system using a convolutional neural network (CNN) quantised to 8-bit TensorFlow Lite format and deployed on the PYNQ-Z2 FPGA board. The model classifies images into three categories: red chilli, plant without chilli, and unknown. Inference results are displayed using onboard LEDs and the PUTTY serial terminal, with LED 2 indicating red chilli, LED 1 for plant, and all LEDs off for unknown or low-confidence predictions. The system achieved a final training accuracy of 97.11%, a validation accuracy of 96.41%, and a test accuracy of 94.64%, demonstrating reliable classification performance with good generalisation and without signs of overfitting. Operating entirely offline without Jupyter or Ethernet, this low-power embedded AI implementation offers a practical, real-time alternative to manual chilli detection for smart agriculture applications.

Keywords: Red chili detection, PYNQ-Z2, Convolutional Neural Network, TensorFlow Lite, Smart agriculture, Real-time classification, FPGA, LED output.

Introduction

In recent years, the combination of artificial intelligence with embedded systems has created new opportunities in precision agriculture. One such use case is the automated identification of ripe red chillies in crop images, which can greatly minimise the manual labour involved in the harvesting process. Farmers have traditionally depended on visual checks to spot ripe chillies, a technique that is labour-intensive, time-consuming, and often inconsistent due to human error or varying environmental factors.

Accurate and timely detection of red chillies is essential for enhancing crop yield and ensuring quality. Although computer vision methods and deep learning models have demonstrated encouraging outcomes in agricultural image classification, many current systems require high-performance computing resources or cloud processing, which may not be practical in field settings. Consequently, there is an increasing need for low-power, real-time, offline classification solutions that can be executed directly on embedded hardware.

Numerous studies have investigated the utilisation of convolutional neural networks (CNNs) for tasks such as object detection and plant disease classification. Nevertheless, few have tackled the challenge of implementing these models on low-resource platforms like FPGAs for agricultural purposes. Furthermore, existing methods often lack user-friendly feedback systems like LED indicators or serial outputs to present classification results in real time.

Literature Review

Object detection frameworks like YOLOv5 have been employed for recognising red chilli, achieving high accuracy and real-time performance on GPU-equipped systems. Nevertheless, these models demand considerable computational resources, rendering them impractical for resource-constrained embedded platforms.[1]

On the other hand, Convolutional Neural Networks (CNNs) have been widely adopted for classifying fruits and vegetables based on visual traits such as colour, shape, and texture. Fruit classification systems utilising CNNs have shown effectiveness even in challenging conditions like varying lighting and complex backgrounds. These findings underscore the flexibility of CNNs and their potential to automate agricultural tasks in uncontrolled environments.[2]

In contrast, traditional machine learning methods like Support Vector Machines (SVMs) and Decision Trees have been used to assess the ripeness and quality of chilli peppers. Although these methods provide interpretability and have lower computational needs, they lack the feature extraction capabilities and scalability that deep learning models offer.[3]

The study demonstrates that convolutional neural networks (CNNs), as well as quantised (QNN) and binary neural networks (BNN), can be effectively implemented on FPGA-based platforms like the PYNQ-Z2 for pattern recognition tasks. The work shows that quantisation reduces memory usage and computational requirements while maintaining high classification accuracy for datasets such as MNIST and CIFAR-10. It also highlights that FPGA deployment enables low-power, real-time inference, making these methods suitable for embedded IoT applications.[4]

Despite these advancements, there is still a notable research gap regarding the deployment of quantised CNN models for real-time red chilli detection using embedded FPGA platforms. This study aims to fill this gap by developing a CNN-based red chilli detection system on the PYNQ-Z2 board, with real-time classification results exhibited through onboard LEDs.[5-8]

Method

The overall workflow of the proposed system is illustrated in Figure 1. The process begins with data collection and preprocessing, followed by CNN model training using the prepared dataset. After training, the model is quantised to INT8 format for efficient execution on resource-constrained hardware. The quantised model is deployed onto the PYNQ-Z2 board, where offline inference is performed using the TensorFlow Lite runtime. Input images are selected sequentially from memory, and the model predicts the class label. Based on the prediction, the result is displayed through onboard LEDs. Additionally, the classification label along with its confidence score is printed on the PUTTY serial terminal. This dual-output mechanism ensures both visual and textual confirmation of results, making the system effective for real-time agricultural monitoring along with its confidence score is printed on the PUTTY serial terminal. This dual-output mechanism ensures both visual and textual confirmation of results, making the system effective for real-time agricultural monitoring.

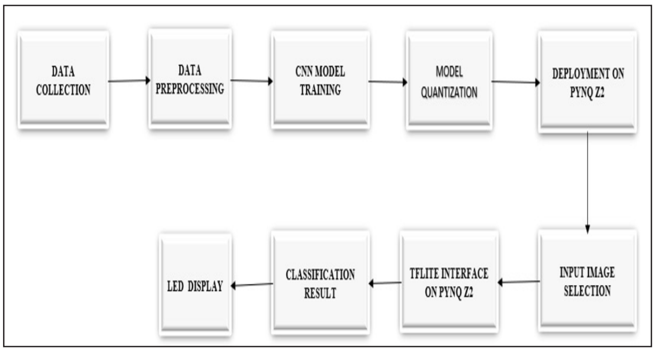

Figure 1.block diagram of redchili classification system

Dataset preparation

The dataset comprises approximately 600 RGB images, collected from Google image searches and publicly available agricultural image repositories. The images are categorised into two classes: red chilli and plant (without chilli). All images were resized to 64×64 pixels to reduce model complexity and enable fast inference on the PYNQ-Z2 board. To improve model robustness, data augmentation techniques such as flipping, rotation, and brightness adjustment were applied.

CNN Architecture

The Convolutional Neural Network (CNN) that has been proposed is tailored to be compact and optimised for real-time image classification on the PYNQ-Z2 embedded platform. This model architecture strikes a balance between accuracy and computational efficiency, making it ideal for deployment at the edge. The architecture is composed of the following layers:

- **Input Layer**: Accepts RGB images that are resized to dimensions of 64×64×3.
- **Convolutional Layer 1**: Applies 32 filters of size 3×3, followed by ReLU activation to capture basic visual features such as edges and color patterns.
- **Max Pooling Layer 1**: Reduces spatial dimensions through a 2×2 pooling window, aiding in the down sampling of the feature map and diminishing computational demands.
- **Convolutional Layer 2**: Utilises 64 filters of size 3×3 with ReLU activation to discern more intricate patterns and shapes.

- **Max Pooling Layer 2**: Another 2×2 pooling operation is performed to further decrease dimensionality.
- **Flatten Layer**: Transforms the 2D feature maps into a 1D feature vector that is compatible for input into the dense layers.
- **Dense Layer**: Comprises 64 neurons activated by RELU to execute high-level feature abstraction.
- **Output Layer**: A softmax layer with 2 neurons, representing the categories Red Chili and Plant. The softmax function yields class probabilities, which are also used to ascertain if the prediction is below a confidence threshold (resulting in an "Unknown" classification).

This CNN is regarded as lightweight due to its shallow architecture, constrained number of filters, and minimal image input size. These design choices ensure the model can be quantized and executed efficiently on low-power hardware such as the PYNQ-Z2. The implementation and training of the model were conducted using TensorFlow with the Keras API, which allows seamless conversion to TensorFlow Lite format for embedded deployment.

Training

The CNN model was trained using TensorFlow with Keras, employing the Adam optimiser and categorical cross-entropy as the loss function. Training was performed for 13 epochs with a batch size of 16. Real-time data augmentation was applied to improve generalisation. The model showed high accuracy and low validation loss, confirming its ability to distinguish between Red Chilli and Plant classes. This trained model was then prepared for quantisation to enable efficient embedded deployment.

Model Quantization

To enable real-time inference on the PYNQ-Z2 board, the trained CNN model was converted to TensorFlow Lite format and quantized to INT8 precision. This significantly reduced the model size and improved inference speed while maintaining classification accuracy.

Hardware Deployment

The quantized model was deployed on the PYNQ-Z2 FPGA board, which combines a Zynq-7000 SoC with an ARM Cortex-A9 processor. Inference is performed offline using the tflite_runtime interpreter. LEDs display classification results: LED 2 for Red Chilli, LED 1 for Plant, and all OFF for Unknown. The classification output and confidence score are also displayed on the PUTTY serial terminal, providing dual-mode feedback.

Results And Discussion

Figures 2–4 demonstrate the end-to-end red chili classification process, including the input image, terminal prediction output, and the corresponding LED response on the PYNQ-Z2 board.

Figures 5–7 illustrate the successful classification of a plant image, with accurate terminal output and LD1 LED indication on the PYNQ-Z2 board.

Figures 8–10 illustrate the correct identification of an unknown image, where the model's confidence score fell below the set threshold. The output was displayed on the PUTTY terminal, and no LED was activated on the PYNQ-Z2 board, confirming the "Unknown" classification.

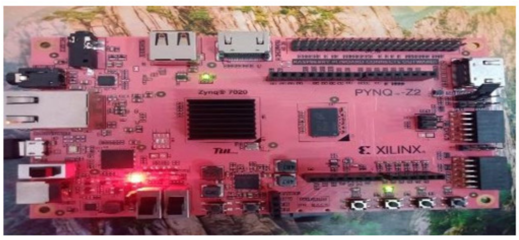

Figure 2.Input image of red chili
Figure 3.LED glowing for red chili on PYNQ-Z2

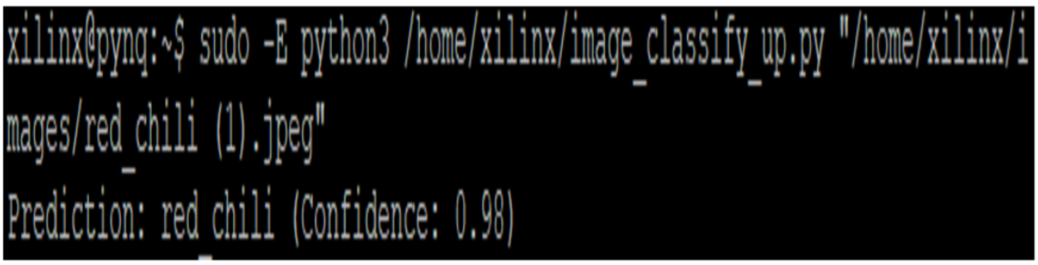

Figure 4.Terminal output for red chili

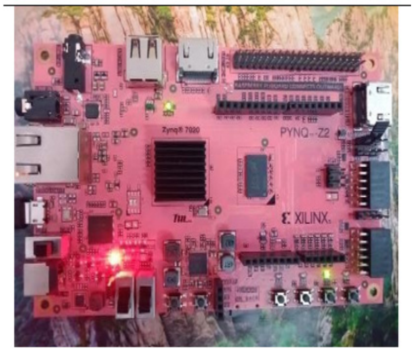

Figure 5.Input image of plant
Figure 6.LED glowing for plant on PYNQ-Z2

```
xilinx@pynq:~$ sudo -E python3 /home/xilinx/image_classify_up.py "/home/xilinx/i
mages/plant_2.jpg"
[sudo] password for xilinx:
Prediction: plant (Confidence: 0.99)
```

Figure 7.Terminal output for plant

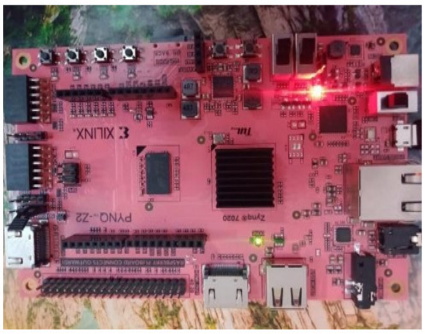

Figure 8.Input image
Figure 9.LED off for unknown class

```
xilinx@pynq:~$ sudo -E python3 /home/xilinx/image_classify_up.py "/home/xilinx/i
mages/Untitled.jpg"
Prediction: Unknown (Confidence: 0.64)
```

Figure 10.Terminal output

Model Performance Evaluation

The plots illustrate the training, validation, and testing curves for both loss and accuracy across 28 epochs. The loss curve shows a clear downward trend in training loss, confirming that the model effectively learns and reduces error over time. While the validation and test losses exhibit some fluctuations during the initial epochs, they stabilize in the later stages and follow a decreasing trend, indicating that the model generalizes well to unseen data.

For accuracy, the training performance improves steadily and reaches above 97% by the final epoch. Both validation and test accuracies closely follow the training curve, achieving 96.41% and 94.64%, respectively. The small variations observed in validation and test accuracies are natural and can be attributed to dataset diversity and evaluation on unseen images. Importantly, the test data used here is entirely separate from the training set, ensuring that the reported performance reflects true generalization rather than memorization.

Overall, the metrics confirm that the model achieves a strong balance between learning and generalization, with no evidence of severe overfitting. These outcomes validate the robustness of the proposed lightweight CNN model and highlight its suitability for efficient deployment on the PYNQ-Z2 FPGA platform for real-time agricultural applications such as red chili detection (Figure 11-12).

The CNN model demonstrates strong capability in distinguishing between the two main classes. For red_ chili, the model achieved a precision of 97%, showing that most predictions for chili were correct. However, the recall of 91% indicates that some chili samples were not detected, which could be due to variations such as lighting or background noise. For the plant class, both precision (97%) and recall (96%) were high, showing that the model performs consistently well in identifying plant samples.

The model achieved 91% for red_chili and 96% for plant. The overall test accuracy of 96% reflects that the model is highly reliable. These results suggest the system is effective for practical use, though future improvements could focus on further boosting sensitivity to red chili to minimize missed detections.

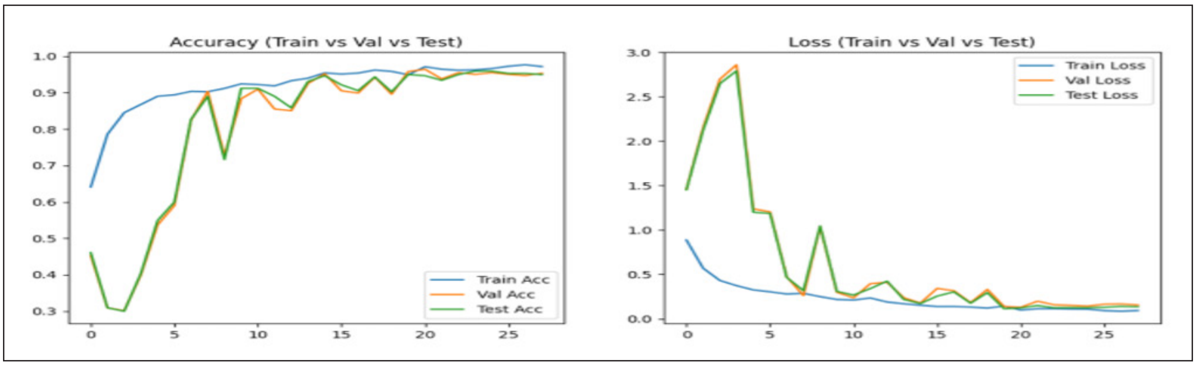

Figure 11.Training and Validation Accuracy/Loss Curves

```
[Step 7.5] Classification Report (Filtered):

Class: red_chili
 Precision: 0.9733
 Recall:    0.9125
 F1-score:  0.9419
 Support:   80.0

Class: plant
 Precision: 0.9744
 Recall:    0.9620
 F1-score:  0.9682
 Support:   79.0
 Accuracy for plant: 0.9620
 Accuracy for red_chili: 0.9125

Overall Test Accuracy: 0.9621
```

Figure 12. Classification Report

Conclusion

This work presents a red chili classification system using a custom CNN model deployed on the PYNQ-Z2 FPGA board. The model achieved strong performance with a precision of 0.9733 for red chili and an overall test accuracy of 0.9621, demonstrating effective classification of red chili and plant images. Training and validation curves indicate stable learning behavior without signs of overfitting.For the red chili class, a recall of 0.9125 shows that most chili samples were correctly detected, though a small portion were misclassified as plant. The plant class achieved balanced results with a precision of 0.9744 and recall of 0.9620, reflecting consistent and reliable classification. The system operates offline, responds to hardware switches, and outputs results via LEDs, making it suitable for embedded smart agriculture applications. Future improvements may focus on further enhancing recall for red chili, extending the model to object detection, and integrating with robotic systems for automated chili harvesting.

Acknowledgement

The hardware tool used for this research paper, the PYNQ-Z2 board, is provided by MeitY under the C2S project titled "Development of SoC system with vision-based UAV and Remote Mobile arm for Precision Agriculture", JNTUHCEH.

References

1. Ram PP, Yaswanth KV, Kamepalli S, Sankar BS, Madupalli M. Deep learning model yolov5 for red chilies detection from chilly crop images. In2023 IEEE 8th International Conference for Convergence in Technology (I2CT) 2023 Apr 7 (pp. 1-6). IEEE.
2. Jmour N, Zayen S, Abdelkrim A. Convolutional neural networks for image classification. In2018 international conference on advanced systems and electric technologies (IC_ASET) 2018 Mar 22 (pp. 397-402). IEEE.
3. Ariyanto A, Djamal EC, Ilyas R. Personality identification of palmprint using convolutional neural networks. In2018 International Symposium on Advanced Intelligent Informatics (SAIN) 2018 Aug 29 (pp. 90-95). IEEE.
4. Biswal MR, Delwar TS, Siddique A, Behera P, Choi Y, Ryu JY. Pattern classification using quantized neural networks for FPGA-Based low-power IoT devices. Sensors. 2022 Nov 10;22(22):8694.
5. Marin A, Radoi E. Image-based fruit recognition and classification. In2022 21st RoEduNet Conference: Networking in Education and Research (RoEduNet) 2022 Sep 15 (pp. 1-4). IEEE.p.111–116.doi:10.1109/RoEduNet57163.2022.9921050
6. Vasavi P, Punitha A, Rao TV. Chili crop disease prediction using machine learning algorithms. Revue d'Intelligence Artificielle. 2023 Jun 1;37(3):727.
7. Zheng J, Lv Z, Li D, Lu C, Zhang Y, Fu L, Huang X, Huang J, Chen D, Zhang J. FPGA-Based Low-Power High-Performance CNN Accelerator Integrating DIST for Rice Leaf Disease Classification. Electronics. 2025 Apr 22;14(9):1704.
8. Sledevič T, Serackis A, Plonis D. FPGA implementation of a convolutional neural network and its application for pollen detection upon entrance to the beehive. Agriculture. 2022 Nov 4;12(11):1849.
9. Wu C, Wang M, Chu X, Wang K, He L. Low-precision floating-point arithmetic for high-performance FPGA-based CNN acceleration. ACM Transactions on Reconfigurable Technology and Systems (TRETS). 2021 Nov 9;15(1):1-21.