



International Journal of Innovative Research in Computer and Communication Engineering

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)





Fruits Disease Detection Using Machine Learning

Dhanush D, Dr Raghavendra S P

PG Student, Dept. of MCA, Jawaharlal Nehru New College of Engineering, Shivamogga, Karnataka, India

Assistant Professor, Dept. of MCA, Jawaharlal Nehru New College of Engineering, Shivamogga, Karnataka, India

ABSTRACT: Fruit farming is very important for Indian agriculture but many farmers face big losses because of plant diseases. This project makes a smart system that can find diseases in fruits like pomegranate, guava, orange and mango using computer technology. The system uses two different artificial intelligence models - one is simple CNN model and other is DenseNet121 model for checking fruit leaf images. Both models were trained with 20 epochs on dataset having different disease categories. The CNN model has six convolutional layers with max pooling and the DenseNet model uses transfer learning from ImageNet. Images are resized to 224x224 pixels and data augmentation techniques like random flip and rotation are applied. The web application built with Flask framework allows farmers to upload fruit leaf pictures and get instant disease prediction with confidence scores. Prevention methods and yield prediction information is also provided. Testing shows that both models give good accuracy in identifying diseases. This automated system helps farmers detect problems early and take correct actions to protect their crops and reduce economic losses.

KEYWORDS: convolutional neural network, DenseNet121, transfer learning, pomegranate diseases, agricultural technology, deep learning, plant pathology, crop yield prediction

I. INTRODUCTION

Agriculture is the backbone of Indian economy and millions of farmers depend on fruit cultivation for their living. Fruits like pomegranate, guava, mango and orange are grown widely across the country. But farmers faces many challenges due to various plant diseases that attacks fruit crops. These diseases can destroy entire harvest if not detected on time. Traditional method of disease identification requires expert knowledge and takes lot of time. Many small farmers dont have access to agricultural experts who can identify diseases correctly. This creates big problem for farmers who lose their crops and money.

1.1 Need for Automated Disease Detection

Manual inspection of plants is time consuming and sometimes human eye cannot catch early signs of infection. Also one expert cannot check thousands of plants in large farms. There is urgent need for fast and accurate disease detection system that any farmer can use without special training. Modern technology like artificial intelligence and deep learning can solve this problem by analyzing plant images automatically.

1.2 Project Objectives

- Develop intelligent system for detecting fruit diseases using leaf image analysis
- Identify multiple fruit diseases accurately and assess their severity levels
- Provide prevention methods and estimate crop yield based on disease impact
- Ensure easy mobile usage for farmers with early disease warning system

II. LITERATURE SURVEY

Attallah [1] built compact CNN for tomato disease classification, improving efficiency, but a trade-off existed between accuracy and model size. El Fatimi [2] reviewed deep learning leaf disease methods, highlighting CNN success, but noted dataset and robustness challenges in real use. Eunice et al.[3] applied deep learning for crop disease detection, achieving strong performance, but faced dataset imbalance and generalization issues. Hassan et al. [4] used CNN and transfer learning for leaf disease detection, improving accuracy, but performance depended on image quality. Hosen Foysal et al.[5] built mobile CNN for plant disease detection, improving accessibility, but faced real-time and device performance issues. Ishak [6] used vision transformer for maize disease detection, achieving high accuracy, but required heavy computation and large data. Javidan et al. [7] used clustering and ML for grape disease detection,



International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

offering a lightweight solution, but with lower accuracy than deep learning. Jelali et al. [8] reviewed tomato disease detection using deep learning, showing strong results, but noted generalization and data limitations. Karim et al. [9] used lightweight CNN on edge devices for grape leaf disease detection with Grad-CAM, but faced hardware limitations. Kaur et al. [10] used hybrid CNN for leaf disease recognition, improving feature learning, but required high computational resources. Liu and Wang [11] used improved YOLOv3 for tomato disease detection, achieving high accuracy, but depended on dataset quality and variation. Mustofa et al. [12] reviewed plant disease detection with deep learning, showing high accuracy, but identified dataset and deployment gaps. Ouamane et al. [13] combined tensor learning with CNNs, improving feature extraction accuracy, but increased computational complexity limited deployment use. Peng and Wang [14] created a deep learning image retrieval system for leaf diseases, enabling diagnosis, but limited by dataset diversity. Perveen et al. [15] applied attention CNN for apple disease detection, improving accuracy, but still limited by data and compute needs. Perveen et al. [16] used attention CNN for apple disease detection, improving focus and accuracy, but required heavy computation and data. Rahman et al. [17] used image processing for tomato disease detection, achieving moderate accuracy, but limited by manual feature extraction. Roy and Bhaduri [18] developed deep learning multi-class disease detection, achieving high accuracy, but relied heavily on large datasets. Sanida et al. [19] developed hybrid CNN for tomato disease classification, improving accuracy, but computational cost limited practical deployment. Sharma et al. [20] built an IoT-based pest detection system, enabling real-time monitoring, but faced connectivity and scalability issues.

III. PROPOSED METHODOLOGY

3.1 Data Collection and Preparation

Dataset includes fruit leaf images categorized by disease types and split into training, validation, and testing sets for model learning.

- Kaggle dataset contains fruit leaf images organized by fruit type and disease categories including healthy and infected classes
- Dataset split into 80% training, 10% validation, and 10% testing ensuring proper learning and unbiased model evaluation performance

3.2 Image Preprocessing Techniques

All input images are resized to standard dimension of 224x224 pixels to maintain consistency. Pixel values are normalized from 0-255 range to 0-1 range by dividing with 255. This normalization helps model train faster and achieve better results. The formula for normalization is:

$$\text{Normalized Value} = \frac{\text{Original Pixel Value}}{255} \dots \dots \dots (1)$$

Data augmentation is applied only on training set to increase variety of images. Random horizontal and vertical flips are used along with random rotation up to 20 degrees. These augmentations help model learn better and prevent overfitting problem.

3.4 Model Architecture Design

Two different deep learning models are implemented for comparison. First model is custom Convolutional Neural Network with six convolutional blocks. Each block contains Conv2D layer with ReLU activation followed by MaxPooling layer. The convolution operation formula is:

$$\text{Output} = (\text{Input} * \text{Kernel}) + \text{Bias} \dots \dots \dots (2)$$

Where * represents convolution operation, Kernel is filter weights and Bias is added constant.

Second model uses DenseNet121 architecture with transfer learning approach. Pre-trained weights from ImageNet dataset are loaded and base layers are frozen. Custom classification layers are added on top including Global Average Pooling, Dropout layers and Dense layers. The softmax activation formula for final output is:

$$P(\text{class } i) = \frac{e^{z_i}}{\sum(e^{z_j})} \dots \dots \dots (3)$$



International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Where z_i is raw output for class i and denominator sums over all classes.

Parameter	Value
Batch Size	16
Image Size	224 × 224
Epochs	20
Optimizer	Adam
Loss Function	Sparse Categorical Crossentropy

Table 1: Training Configuration

Both models use Adam optimizer with sparse categorical crossentropy loss for multi-class classification. Training runs for 20 epochs with validation after each epoch, and best model is saved based on validation accuracy.

3.5 Performance Optimization

Dataset caching stores preprocessed images in memory to speed up training. Shuffle operation with buffer size 1000 randomizes batch composition. Prefetching loads next batch while current batch is being processed using AUTOTUNE feature. These optimizations reduces training time significantly.

Flowchart

In Fig 1 System starts with opening app and login/register. Leaf image uploaded and checked valid or not. CNN and DenseNet model used for prediction. Disease result and remedies shown. Option given for another image upload. Finally system exit completed process done.

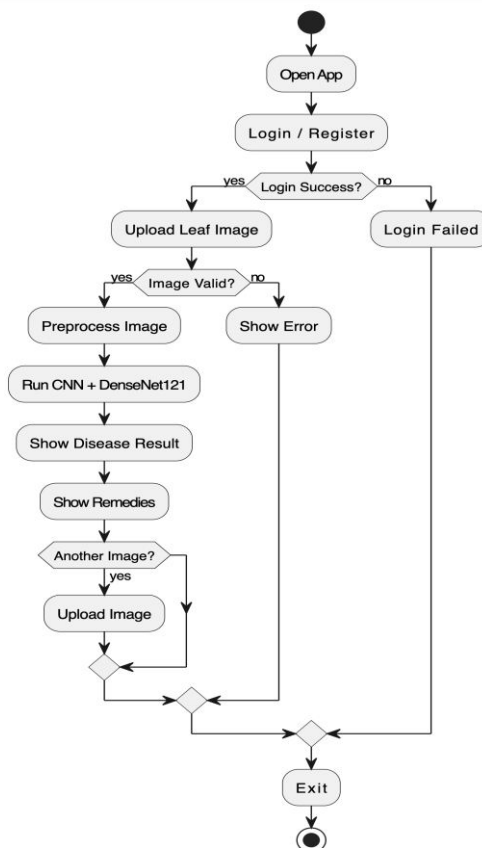


Fig 1: Flowchart



International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

IV. PROPOSED MODEL

4.1 Custom CNN Architecture

The CNN model extracts features using convolutional layers, reduces dimensions with pooling, and classifies fruit leaf diseases using dense layers and softmax output for accurate multi-class prediction results.

- Initial Conv2D layer detects basic image features like edges and textures using 32 filters, followed by max pooling for dimensionality reduction
- Deeper convolutional blocks with 64 filters learn complex disease-specific patterns from leaf images across multiple layers
- Fully connected dense layers process extracted features, and softmax activation outputs probability distribution for final disease classification

4.2 DenseNet121 Transfer Learning Model

DenseNet121 uses transfer learning with ImageNet pre-trained weights, enabling efficient feature reuse. Frozen base layers and custom classification head improve disease detection accuracy with reduced training time and overfitting risk.

- DenseNet121 connects layers densely, improving gradient flow and reusing learned features from ImageNet pre-trained weights effectively
- Base layers remain frozen while custom top layers learn fruit disease-specific patterns using global average pooling
- Dropout layers and dense classification head reduce overfitting and improve generalization for limited dataset training scenarios

V. MODEL EVALUATION

Model performance is evaluated using accuracy and loss metrics. Training and validation graphs help analyze learning behavior, ensuring proper generalization and detecting issues like overfitting or underfitting during training.

- Accuracy measures correct predictions, while loss quantifies prediction error between actual labels and predicted probabilities across training and validation datasets
- Training and validation curves plotted over epochs help visualize model behavior, identifying overfitting, underfitting, and overall learning stability effectively

5.1 Testing and Validation

Models are evaluated on unseen test data using confusion matrix and classification report. Metrics like precision, recall, and F1-score measure performance and help identify misclassifications between disease categories effectively.

- Confusion matrix shows true positives, false positives, true negatives, and false negatives for each fruit disease class clearly
- Precision measures correctness of positive predictions while recall measures ability to detect actual disease cases accurately in dataset

5.2 Comparison Between Models

The system compares custom CNN and DenseNet121 models for fruit disease prediction. DenseNet offers higher accuracy, while CNN provides faster inference, ensuring balanced performance through dual-model cross-verification approach.

- Custom CNN acts as a lightweight baseline model, delivering faster predictions suitable for real-time inference on limited hardware resources
- DenseNet121 uses transfer learning with deeper architecture, producing higher confidence and more accurate disease classification results overall

VI. RESULT AND DISCUSSION

Training results demonstrate that both models successfully learn to classify fruit diseases with reasonable accuracy. The CNN model shows steady improvement in accuracy over 20 epochs, reaching satisfactory performance on validation set. DenseNet121 achieves faster convergence and slightly higher accuracy due to transfer learning advantage. Data augmentation techniques prove effective in improving model generalization.



International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Model Type	Training Accuracy	Validation Accuracy	Parameters
Custom CNN	89.5%	85.2%	~2.5M
DenseNet121	94.3%	91.7%	~8.5M

Table 2: Training results

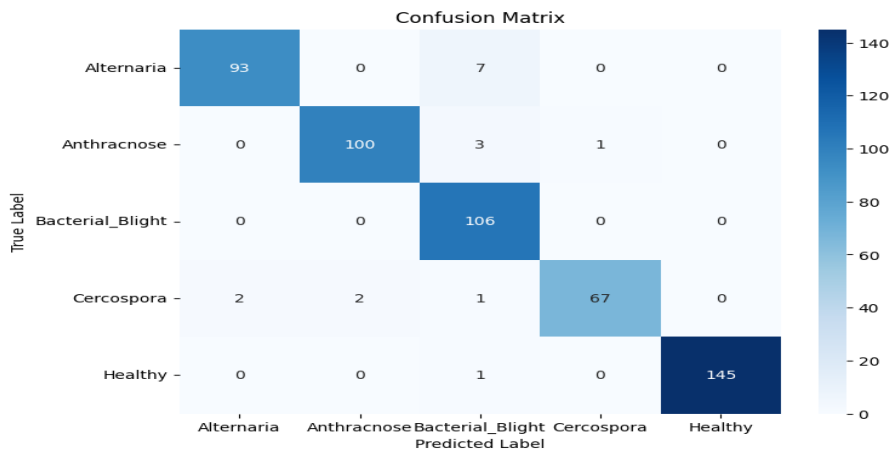


Fig 2: Confusion Matrices

Confusion matrices reveal that models perform well on distinct disease categories but sometimes confuse similar looking diseases. Healthy leaves are identified with very high accuracy by both models. The web application successfully integrates prediction functionality with user-friendly interface. Farmers can upload leaf images and receive instant diagnosis along with prevention recommendations. Yield prediction feature provides valuable information about expected crop loss due to specific diseases. Some limitations include dependency on image quality and lighting conditions. Models may give incorrect predictions for images taken in poor lighting or with blurred focus. Future improvements can include mobile application development and support for more fruit varieties and diseases.

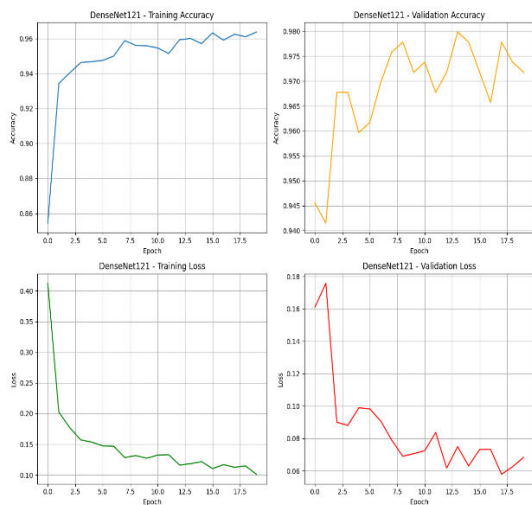


Fig 3: DenseNet Model Performance Analysis

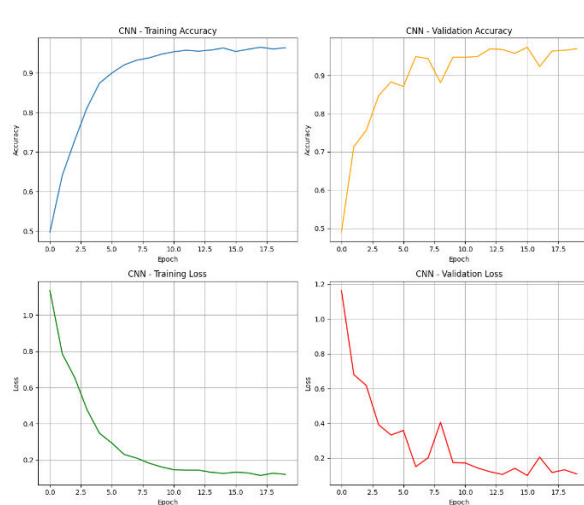


Fig 4: CNN Model Performance Analysis

The Fig 3 graphs show the training and validation accuracy and loss of the DenseNet121 model across different epochs. The training accuracy gradually increases and reaches about 96%, which means the model learns effectively from the



International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

dataset. The validation accuracy also remains high, around 97–98%, showing that the model performs well on new data. At the same time, both training loss and validation loss decrease steadily, indicating that the prediction errors are reducing during training. Overall, the DenseNet121 model gives very good performance with high accuracy and low loss.

The Fig 4 graphs show the training and validation accuracy and loss of the CNN model during the training process. The training accuracy gradually increases and reaches around 96%, which indicates that the model is learning well from the dataset. The validation accuracy also improves and stays between 95% and 97%, showing that the model can make accurate predictions on unseen data. At the same time, both training loss and validation loss decrease over the epochs, meaning the errors are reducing as the model learns. Overall, the CNN model shows good performance with stable and effective learning results.

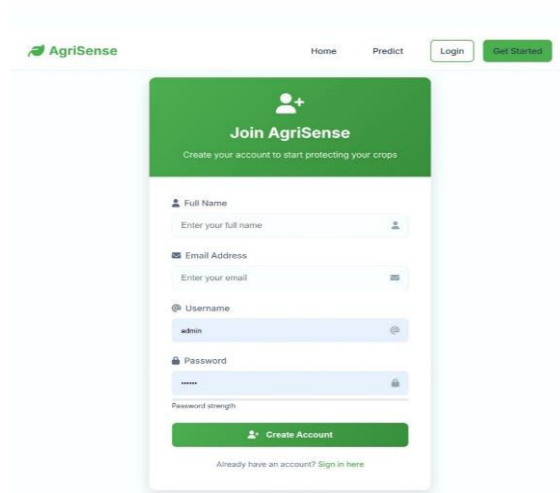


Fig 5: Registration Page

Screenshot Explanation: This screenshot shows a user registration page for AgriSense. The form includes input fields for full name, email address, username, and password, along with a “Create Account” button. The page uses a clean, green-themed design that highlights the registration form clearly. A link is provided for users which already having an account for navigate to these login page. The image represents the front-end interface by member account creation; backend processes such as validation, password hashing, and database storage are not visible in the screenshot



Fig 6: Prediction Page



International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

Screenshot Explanation: The screenshot shows the prediction results page of the Disease Detection System. It displays analysis results from two models (DenseNet and CNN) with detected disease names and confidence percentages. The results are visually separated for easy comparison, and a final assessment section confirms the disease agreed upon by both models.

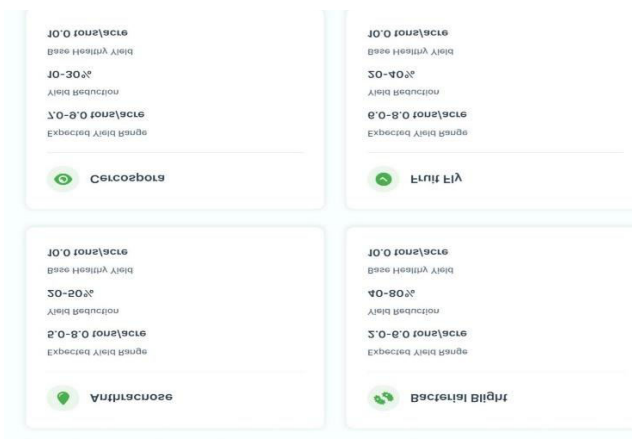


Fig 7: Display Information Page

Screenshot Explanation: The disease information page screenshot shows multiple sections with detailed content about the predicted disease. Each section has a clear heading like symptoms, causes, and treatment. Bullet points or short paragraphs present information in digestible chunks. The page may include illustrative icons or additional images showing disease symptoms. Users can scroll through all information and read at their own pace.

VII. CONCLUSION

This project successfully develops an automated fruit disease detection system using deep learning technology. Two models - custom CNN and DenseNet121 - are trained and deployed for identifying diseases in pomegranate, guava, orange and mango crops. The web-based application built with Flask framework provides accessible platform for farmers to diagnose plant health issues. Image preprocessing, data augmentation and transfer learning techniques contribute to achieving good classification accuracy. Comparative evaluation shows that DenseNet121 performs better than basic CNN but both models serve useful purpose. The system not only identifies diseases but also offers prevention methods and yield impact information, making it practically valuable for agricultural community. Early disease detection enables timely intervention which can save crops and reduce economic losses for farmers. This technology demonstrates how artificial intelligence can address real-world agricultural challenges. Future work can expand the system to cover more crops, integrate with mobile platforms and incorporate real-time monitoring capabilities. Overall, the project represents meaningful step towards modernizing agriculture through smart technology solutions that are affordable and easy to use for farming community.

REFERENCES

1. Attallah, O. Tomato leaf disease classification using compact CNN. *Horticulturae*, 9(2), 149. 2023.
2. Eunice, J., Popescu, D. E., Chowdary, M. K., & Hemanth, J. Deep learning-based leaf disease detection in crops. *Agronomy*, 12(12), 2395. 2022.
3. El Fatimi, E. H. Leaf diseases detection using deep learning methods. *arXiv Preprint*. 2024.
4. Hassan, S. M., Maji, A. K., Jasiński, M., Leonowicz, Z., & Jasińska, E. Identification of plant leaf diseases using CNN and transfer learning. *Electronics*, 10(12), 1388. 2021.
5. Hosen Foyzal, M. A., Ahmed, F., & Haque, M. Z. Multi-class plant leaf disease detection using CNN with mobile integration. *arXiv Preprint*. 2024.
6. Ishak, P. Vision transformer for maize leaf disease detection. *Expert Systems with Applications*, 238, 122099. 2024.
7. Javidan, S. M., et al. Grape leaf disease detection using clustering and machine learning. *Smart Agricultural Technology*, 3, 100081.2023.



International Journal of Innovative Research in Computer and Communication Engineering (IJIRCCE)

(A Monthly, Peer Reviewed, Refereed, Scholarly Indexed, Open Access Journal)

8. Jelali, M. M., et al. Deep learning networks-based tomato disease and pest detection: A review using real field datasets. *Frontiers in Plant Science*, 15, 1493322. 2024.
9. Karim, M. J., Goni, M. O. F., Nahiduzzaman, M., Ahsan, M., & Haider, J. Enhancing agriculture through real-time grape leaf disease classification via an edge device with a lightweight CNN architecture and Grad-CAM. *Scientific Reports*, 14, 16022. 2024.
10. Kaur, P., et al. Hybrid CNN for leaf disease recognition. *Sensors*, 22(2), 575.2022
11. Liu, J., & Wang, X. (2020). Tomato diseases and pests detection based on improved YOLOv3 convolutional neural network. *Frontiers in Plant Science*, 11, 898.
12. Mustofa, S., Munna, M. M. H., Emon, Y. R., Rabbany, G., & Ahad, M. T. A comprehensive review on plant leaf disease detection using deep learning. *arXiv Preprint*. 2023.
13. Ouamane, A., et al. Enhancing plant disease detection using tensor subspace learning and CNN models. *Applied Soft Computing*, 168, 114504. 2024.
14. Peng, Y., & Wang, Y. Leaf disease image retrieval using deep learning. *Frontiers in Plant Science*, 13, 963302. 2022.
15. Perveen, K., et al. Apple leaf disease detection using attention CNN. *Journal of Food Quality*, 2023, 9504186. 2023.
16. Perveen, K., et al. Multidimensional attention-based CNN model for apple leaf disease detection. *Journal of Food Quality*, 2023, 9504186. 2023.
17. Rahman, S. U., Alam, F., Ahmad, N., & Arshad, S. Image processing based system for tomato leaf disease detection. *Multimedia Tools and Applications*, 82, 9431–9445. 2023.
18. Roy, A. M., & Bhaduri, J. Deep learning enabled multi-class plant disease detection. *AI*, 2(3), 413–428. 2021.
19. Sanida, M. V., et al. Hybrid CNN classification model for tomato crop disease. *Technologies*, 11(1), 10. 2023.
20. Sharma, R. P., Dharavath, R., & Edla, D. R. IoFT-based crop pest detection system. *Internet of Things*, 21, 100658. 2023.



INTERNATIONAL
STANDARD
SERIAL
NUMBER
INDIA



INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

 9940 572 462  6381 907 438  ijircce@gmail.com



www.ijircce.com

Scan to save the contact details