Engineering and Technology Journal e-ISSN: 2456-3358

Volume 11 Issue 05 May-2026, Page No.-9721-9725

DOI: 10.47191/etj/v11i05.08, I.F. – 8.595

A Companion Plant Management Application: Color-Based Recommendation, Image Recognition, And Pest/Disease Diagnosis

Youngjun Kim[1], Hoonkyu Kim[2], Geonho Maeng[3], Myungseop Yoo[4], Seungjae Lee[5*]

[1,2,3,4]Computer Engineering Department, Sunmoon university, Korea

ABSTRACT: The growing popularity of companion plants (houseplants kept for personal enjoyment) has created a demand for intelligent mobile tools that help users select, manage, and protect their plants. Existing plant identification applications such as PlantSnap and PictureThis offer photo-based identification but lack personalized recommendation and integrated pest/disease diagnosis. This paper presents a cross-platform mobile application built with React Native that provides three core functions: (1) color-based plant recommendation, (2) image-recognition-based plant information retrieval, and (3) pest and disease diagnosis via Azure Custom Vision. User data, including adoption dates, photos, and nicknames, are stored and managed through Firebase Realtime Database. Evaluation tests demonstrate that the disease diagnosis model achieves reliable accuracy across multiple disease categories. Future work includes algorithm refinement, multilingual support, and community features.

KEYWORDS: React Native, Azure Custom Vision, pest diagnosis, image recognition

I. INTRODUCTION

1.1 Background and Motivation

Over the past several years, growing companion plants (also called "pet plants" in Korean culture) has become an increasingly popular lifestyle activity worldwide. The global indoor plants market was valued at USD 20.68 billion in 2024 and is expected to reach USD 30.25 billion by 2032, growing at a CAGR of 4.87%, primarily driven by increasing consumer interest in home decor, air purification, and wellness benefits of indoor plants. The COVID-19 pandemic sparked an 18% rise in houseplant demand as people sought comfort at home. A 2020 study found that 7 in 10 millennials call themselves a "plant parent," indicating that caring for houseplants is not just a hobby but involves a perceived parental role.

Despite this growing interest, many people face significant challenges in selecting and maintaining companion plants. Key difficulties include choosing the right species for one's living environment, obtaining accurate plant care information, and managing pest and disease outbreaks effectively. These problems are compounded by the fact that 48% of millennials question their abilities to keep plants alive, and 40% of them plan to buy a new houseplant each year, creating a cycle of enthusiasm followed by frustration.

1.2 Limitations of Existing Applications

Several mobile applications already exist in the plant identification space. Plant identification applications for use on smartphones have been increasing in availability, accuracy, and utilization. Popular examples include PlantSnap and PictureThis, both of which employ artificial intelligence to identify plants from photographs. PictureThis and PlantSnap use artificial intelligence (AI) systems to analyze plant images, cross-referencing with a photo database to find a match.

However, comparative evaluations have revealed notable differences in their capabilities and significant gaps. The top performing app in the 2023 evaluation was PictureThis, with 73% of the suggested identifications being correct; adding partial ratings, the app was helpful 89% of the time. Meanwhile, PlantSnap completely misidentified six out of 40 plants tested, for an accuracy of 85%. While these applications excel at species identification, they generally lack personalized plant recommendation capabilities and integrated pest/disease diagnosis features [1, 2]. Most existing tools focus primarily on watering reminders and basic plant information, leaving a gap for a comprehensive companion plant management solution.

1.3 Research Objectives

This research aims to develop a comprehensive companion plant management application that addresses the limitations of existing tools. The specific objectives are:

1) To develop a color-based companion plant recommendation feature that suggests plants based on user-selected colors.
2) To implement an image-recognition-based plant information retrieval system that analyzes user-uploaded plant photos.
3) To build a pest and disease diagnosis feature that identifies plant health issues from photographs and provides treatment recommendations.

II. RELATED WORK

2.1 Deep Learning for Plant Disease Detection

The application of deep learning, particularly convolutional neural networks (CNNs), to plant disease detection has been an active area of research. Deep learning-based plant disease detection offers considerable progress and hope compared to classical methods; when trained with large and high-quality datasets, these technologies robustly detect diseases on plant leaves in early stages. The use of ML and DL techniques significantly increases the precision and speed of plant disease detection.

Various CNN architectures have been explored for this purpose, including AlexNet, VGGNet, ResNet, and more recently, Vision Transformers (ViTs). Convolutional neural networks (CNNs), a leading architecture in this field, have proven to be very accurate in detecting plant diseases; additionally, the Vision Transformer (ViT) has recently gained popularity and is now widely used. Transfer learning approaches have proven particularly effective, as they allow pre-trained models to be adapted for specific plant disease classification tasks with relatively small datasets [3, 4].

A research study using the Azure Custom Vision platform to classify food crop plant leaf diseases demonstrated that plant diseases are recognized at approximately a 91 percent accuracy rate using a ResNet-based model [5]. This demonstrates the viability of cloud-based AI services for practical plant disease diagnosis applications.

2.2 Cloud-Based Image Classification Services

Azure AI Custom Vision is a cognitive service that enables users to build custom image classifiers using machine learning; it is particularly well-suited for applications such as a plant disease classifier due to its robust feature set and ease of use. Unlike general-purpose vision APIs that work with pre-trained models, Custom Vision allows users to train their own models using a custom dataset tailored to specific requirements. Under the hood, it uses transfer learning on top of pre-trained convolutional neural networks, which means you can get good results with as few as 50 images per class.

These characteristics make Azure Custom Vision an ideal platform for building plant disease classifiers without requiring extensive deep learning expertise, allowing developers to focus on application-level features while leveraging state-of-the-art image classification capabilities [5, 6].

2.3 Cross-Platform Mobile Development with React Native

React Native is a framework for building native apps for Android, iOS, and more using React, written in JavaScript and rendered with native code. Originally released by Meta (formerly Facebook) in 2015, it has become one of the most popular frameworks for cross-platform mobile development [7]. React Native's cross-platform capability allows for creating apps compatible with iOS and Android, thus providing a streamlined development process; there is no need for separate teams or codebases.

The key advantages of React Native include code reusability across platforms, near-native performance, hot reloading for rapid development iteration, and access to a large ecosystem of third-party libraries. Unlike traditional hybrid frameworks that rely on WebView components, React Native renders components using native APIs, offering a near-native experience while still enabling code reuse across platforms.

2.4 Firebase Realtime Database

The Firebase Realtime Database is a cloud-hosted database where data is stored as JSON and synchronized in realtime to every connected client; when you build cross-platform apps with Apple platforms, Android, and JavaScript SDKs, all of your clients share one Realtime Database instance and automatically receive updates with the newest data. The Firebase Realtime Database lets you build rich, collaborative applications by allowing secure access to the database directly from client-side code; data is persisted locally, and even while offline, realtime events continue to fire, giving the end user a responsive experience.

This makes Firebase an excellent backend choice for mobile applications that require lightweight data storage, synchronization, and offline capabilities, particularly for storing user-specific plant management data [8].

III. SYSTEM DESIGN AND IMPLEMENTATION

Upon power connection, the video input f

3.1 System Architecture Overview

The proposed application adopts a three-tier architecture consisting of a front-end user interface layer, a cloud-based AI service layer, and a database layer. The user interface is implemented in React Native to achieve cross-platform compatibility for both iOS and Android. The AI service layer leverages Azure Custom Vision for image recognition and disease diagnosis. The database layer uses Firebase Realtime Database for persistent storage of user and plant data.

3.2 Feature Design

The application provides three main functional tabs, each containing specific features, as summarized in Table 1.

Table 1. Feature List and Description

Tab	Feature	Description
Home	(1) Today's Flower	Daily random plant recommendation based on the National Institute of Horticultural and Herbal Science database

	(2) Plant Recommendation	Color-based companion plant recommendation
My Plants	(1) My Plant List	View registered plants
	(2) Plant Registration & Management	Register and manage plant information (adoption date, photo, nickname)
	(3) Calendar	Plant care schedule management
Camera	(1) Album Photo Selection	Select photos from device album
	(2) Plant Disease Diagnosis	AI-powered pest and disease diagnosis

3.2.1 Color-Based Plant Recommendation

The color-based plant recommendation feature allows users to select a color from a color palette. The system then matches the selected color to flowering plants whose blooms correspond to that color and presents recommendations. For example, selecting green returns plants with green-tinted flowers, while selecting pink returns plants with pink blooms. This approach provides a personalized and intuitive way for users to discover new companion plants based on their aesthetic preferences.

3.2.2 Today's Flower

The "Today's Flower" feature utilizes the flower information service provided by the Rural Development Administration's National Institute of Horticultural and Herbal Science (South Korea). Based on stored flower data, the system randomly recommends a different plant each day, allowing users to learn about new plant species on a daily basis.

3.2.3 Plant Database Management

Users can register their companion plants by entering information such as the adoption date, a photograph, and a nickname. This data is stored in Firebase Realtime Database and can be retrieved, displayed, and deleted as needed. The Realtime Database is a cloud-hosted database where data is stored as JSON and synchronized in realtime to every connected client; React Native Firebase provides native integration with the Android & iOS Firebase SDKs, supporting both realtime data sync and offline capabilities. The calendar feature provides an integrated view of care schedules for all registered plants.

3.2.4 Pest and Disease Diagnosis

The pest and disease diagnosis feature is the most technically complex component of the application. Users photograph a suspected diseased part of their plant and upload the image. The image is then sent to the Azure Custom Vision model, which has been trained on thousands of plant and pest/disease images. The model classifies the image and returns a diagnosis along with recommended treatment methods.

03.3 Image Recognition Model

The image recognition model was developed using Azure Custom Vision, which employs transfer learning on pre-trained CNN architectures [5, 6]. The training dataset consisted of thousands of plant images, including both healthy specimens and those exhibiting various diseases and pest damage. The model was trained to recognize multiple disease categories, and performance metrics (precision, recall, and probability thresholds) were evaluated for each disease class (Figure 1).

The Azure AI Custom Vision service offers a transformative approach to tackling plant diseases; by leveraging AI for image classification, it is possible to build a classifier model that can significantly aid in the early detection and treatment of crop diseases.

3.4 Implementation Details

The front-end was developed using React Native, enabling a single codebase to deploy on both iOS and Android platforms [7]. The integration with Azure Custom Vision was implemented through REST API calls, where plant images are sent to the Custom Vision prediction endpoint and classification results are returned as JSON responses. Firebase integration was accomplished using the React Native Firebase SDK, providing seamless data read/write operations and real-time synchronization [8, 9].

IV. EVALUATION AND EXPECTED BENEFITS

4.1 Evaluation

To evaluate the system's performance, two types of tests were conducted:

1) Plant Information Accuracy Test: Various plant photographs were submitted to the image recognition system, and the accuracy of the returned plant information was assessed.
2) Disease Diagnosis Accuracy Test: Photographs of plants with confirmed diseases were submitted to the diagnosis system, and the diagnostic accuracy was evaluated across multiple disease categories.

The disease diagnosis model demonstrated reliable performance across tested disease categories, with the per-class diagnostic data shown in Figure 1 of the original study. The use of Azure Custom Vision's iterative training process allowed for progressive improvement of the model's accuracy through additional training data and parameter tuning [6].

4.2 Differentiation from Existing Applications

Compared to existing plant management applications that offer simple information retrieval and watering reminders, the proposed application provides the following differentiating features:

1) Color-Based Recommendation: A novel approach to plant recommendation that matches user color preferences to flowering plants, providing a personalized discovery experience unavailable in competing applications [1, 2].
2) Integrated Disease Diagnosis: Unlike PlantSnap and similar applications that focus primarily on identification, the proposed application offers dedicated pest and disease diagnosis with treatment recommendations [3, 4].
3) Comprehensive Data Management: Users can manage their entire plant collection within the application, with adoption dates, photographs, nicknames, and care schedules all integrated into a single platform [8].

4.3 Expected Benefits

The expected benefits of the proposed application span multiple domains:

1) Enhanced Plant Care: By providing accurate disease diagnosis and treatment recommendations, the application helps users maintain healthier plants, reducing the common frustration of unexplained plant death.
2) Improved Accessibility: The color-based recommendation system lowers the barrier to entry for novice plant owners who may not know plant species names but can express aesthetic preferences.
3) Educational Value: The daily plant recommendation feature and comprehensive plant information database serve as educational resources, particularly useful in botany and biology education [10].
4) Reduced Plant Mortality: With timely disease detection and treatment guidance, users can intervene before diseases become fatal, potentially reducing the plant mortality rates commonly reported among new plant owners.

V. CONCLUSION AND FUTURE WORK

5.1 Conclusion

This paper presented a comprehensive companion plant management application that integrates color-based plant recommendation, image-recognition-based plant information retrieval, and AI-powered pest/disease diagnosis. The application was developed using React Native for cross-platform compatibility, Azure Custom Vision for image classification, and Firebase Realtime Database for data management. Evaluation results demonstrate the viability of the approach for practical companion plant care support.

5.2 Future Work

Several directions for future improvement have been identified:

1) Algorithm Enhancement: Improving the accuracy of both the color recognition algorithm and the image recognition model through additional training data and more sophisticated architectures. Recent advances in Vision Transformers and hybrid CNN-ViT models could be explored for improved disease classification performance.
2) Feature Expansion: Adding personalized notification services (e.g., watering reminders based on plant species and environmental conditions) and community features for user interaction and knowledge sharing.
3) Database Expansion: Enlarging the plant information database and disease/pest dataset to cover more species and disease types, improving the breadth and depth of the application's capabilities.
4) Multilingual Support: Implementing support for multiple languages to make the application accessible to a global user base, expanding beyond the current Korean-language implementation.
5) IoT Integration: Exploring integration with soil moisture sensors and other IoT devices for automated environmental monitoring and data-driven care recommendations..

REFERENCES

1. Enfield, B., Brooks, D. E., Welch, S., Roland, M., Klemens, J., Greenlief, K., Olson, R., & Gerkin, R. D. (2021). Swipe right: A comparison of accuracy of plant identification apps for toxic plants. Journal of Medical Toxicology, 17(1), 42-47.
2. White, A., Muir, A., & Allen, J. (2023). Evaluation of smartphone plant identification applications for landscape plant identification. HortTechnology, 33(4), 415-421.
3. Dhaka, V. S., Meena, S. V., Rani, G., Sinwar, D., Kavita, Ijaz, M. F., & Woźniak, M. (2021). A survey of deep convolutional neural networks applied for prediction of plant leaf diseases. Sensors, 21(14), 4749.
4. Borhani, Y., Khoramdel, J., & Najafi, E. (2022). Convolutional neural networks in detection of plant leaf diseases: A review. Agriculture, 12(8), 1192.
5. Oladejo, O. M., & Abikoye, O. C. (2025). Diagnosing diseases in the leaves of food crop plants using residual network powered by Microsoft Azure Custom Vision Service. Journal of Agricultural Informatics, 16(1), 1-15.
6. Microsoft. (2024). Azure Custom Vision documentation. Retrieved from https://learn.microsoft.com/en-us/azure/cognitive-services/custom-vision-service/

7. Meta Platforms, Inc. (2024). React Native: Learn once, write anywhere. Retrieved from https://reactnative.dev/
8. Google. (2024). Firebase Realtime Database documentation. Retrieved from https://firebase.google.com/docs/database
9. Moroney, L. (2017). The Definitive Guide to Firebase: Build Android Apps on Google's Mobile Platform. Apress.
10. Mohanty, S. P., Hughes, D. P., & Salathé, M. (2016). Using deep learning for image-based plant disease detection. Frontiers in Plant Science, 7, 1419.