PomeGuard: A Multi-Agent Autonomous System for Non-Destructive Phytochemical Inference and Disease Classification in Bhagwa Pomegranate Horticulture

[1]Aryan Madhav Jadile, [2]Umesh Dnyanoba Wakadkar, [3]Shreekar Santosh Varpe, [4]Pushkar Dattatray Shedge, [5]Prof. Suvarna Bahir
[12345]Dept. of Computer Engineering
[12345]Padmabhooshan Vasantdada Patil Institute of Technology Pune, India

***Abstract*—The Bhagwa pomegranate's real strength lies in its rich mix of phytochemicals—think anthocyanins, punicalagins, and ellagic acid. But fungal and bacterial diseases, along with tough environmental conditions, strip away those valuable com-pounds. Right now, figuring out fruit quality usually means running destructive and time-heavy lab tests, which really slows down decision-making for farmers.**

That's why we built PomeGuard. It's an autonomous system that checks for diseases and measures the fruit's nutritional value without tearing it apart. The system combines a CNN based on EfficientNetB0 for spotting disease visually, and an OWL 2 DL ontology that links disease severity and environmental data to phytochemical loss inside the fruit. Everything's managed by a LangGraph StateGraph, which organizes four agents to pull together a complete nutritional scorecard.

We tested the vision module on more than 5,000 images and it nailed classification—98.9% accuracy across five categories: Healthy, Bacterial Blight, Anthracnose, Cercospora Fruit Spot, and Alternaria Fruit Spot. By tying together computer vision and semantic reasoning, PomeGuard brings real-time, non-destructive quality checks to the field. Farmers get smarter har-vests and better resource management—exactly what precision horticulture needs.

***Index Terms*—Pomegranate disease detection, Computer vi-sion, Convolutional neural networks, Ontological reasoning, Multi-agent systems, Phytochemical quality assessment, Non-destructive evaluation**

I. Introduction

The Bhagwa pomegranate (*Punica granatum* L. cv. Bhagwa) is a cornerstone of Maharashtra's horticultural econ-omy in India, prized for its deep crimson arils and exceptionally high concentrations of bioactive polyphenols. These compounds, principally Punicalagins, Ellagic Acid, and An-thocyanins, govern the fruit's nutritional value and determine its premium in export markets [1], [2]. However, pomegranate cultivation faces severe challenges from pervasive pathogens. Diseases such as Bacterial Blight (*Xanthomonas axonopodis* pv. *punicae*), Anthracnose (*Colletotrichum gloeosporioides*), Cercospora Fruit Spot, and Alternaria Fruit Spot routinely cause 30–40% crop losses [3], [4]. These infections not only damage the fruit's exterior but trigger biochemical cascades that significantly degrade internal phytochemical quality.

The primary limitation in current agricultural practice is the delayed detection of these diseases and the destructive nature of quality assessment. Typically, 30–40% of a crop may be infected before symptoms become clearly discernible to manual scouting. Furthermore, determining the extent to which a disease has degraded the fruit's internal quality currently requires High-Performance Liquid Chromatography (HPLC)

[5]. This laboratory-based analysis is destructive, expensive, and introduces a 48–72 hour delay that is fundamentally incompatible with the rapid decision-making required during harvest and sorting. Consequently, farmers are forced to rely on imprecise visual estimations, leading to suboptimal eco-nomic outcomes.

To address this critical gap, we propose PomeGuard, an integrated, non-destructive agricultural intelligence system. PomeGuard replaces destructive sampling with a multi-agent architecture [6], [7] that derives actionable insights from standard smartphone imagery and environmental metadata. The system employs a fine-tuned EfficientNetB0 convolutional neural network to classify disease presence and severity from fruit images. Concurrently, it captures environmental param-eters—temperature, relative humidity, and ultraviolet (UV) irradiance—to quantify abiotic stress. These perceptual inputs are subsequently processed by an OWL 2 DL knowledge ontology [8], which uses formalized domain knowledge to logically infer the degradation status of key phytochemical compounds.

The primary contribution of this work is the synthesis of deep learning perception with symbolic reasoning. While deep learning excels at visual pattern recognition [9], [10], it cannot directly predict unobservable chemical concentra-tions without exhaustive paired laboratory data. Conversely, semantic web technologies can formalize biochemical rela-tionships [11], [12] but require structured perceptual input. By orchestrating these components through a stateful multi-agent system [13]–[15], PomeGuard achieves what neither paradigm could accomplish independently: the real-time, non-destructive estimation of internal fruit quality from external indicators. This approach bridges the gap between field observation and laboratory analysis, providing a scalable framework for precision horticulture.

II. Related Work

A. Computer Vision in Agriculture

The application of computer vision to crop disease detection has matured significantly with the advent of deep learning. Early approaches relied on support vector machines and hand-crafted features [16], which struggled with the complexity of field conditions. The introduction of convolutional neural networks (CNNs) marked a paradigm shift [9]. Islam et al. (2015) demonstrated the efficacy of CNNs for robust crop disease classification [17], while subsequent studies extended these architectures to a wide variety of plant species [18]–[25]. Various architectures including VGG16 [26], ResNet [27], MobileNet [28], and Inception [29] have been widely deployed and benchmarked for object detection and classification in agriculture [30], [31]. Transfer learning has become standard practice in horticulture to mitigate data scarcity [32]–[35], with architectures like EfficientNet showing superior parameter efficiency and performance for fruit disease classification [36], [37]. However, these vision systems are inherently limited to surface-level perception; they detect the presence of disease but provide no insight into the consequent internal biochemical changes.

B. Ontological Reasoning and Semantic Web

To formalize complex agricultural domain knowledge, re-searchers have increasingly adopted semantic web tech-nologies. Ko¨ksal and Tekinerdogan (2019) developed an OWL/RDF-based ontology for crop management, demonstrat-ing how semantic models can capture hierarchical agricultural concepts and decision logic [38]. Colombo et al. (2022) expanded on this by creating SPARQL-based farm decision support systems, enabling complex queries over integrated agricultural datasets [39]. Foundational tools like HermiT [40], SWRL [41], and SPARQL [42] provide robust environments for this logic. While these ontologies effectively structure agronomic practices and pest taxonomies [11], [12], [43], they generally lack the specific, quantifiable biochemical pathways linking particular pathogen infections to the degradation of secondary metabolites, which is necessary for quality infer-ence. Multi-agent systems [6], [7], [13]–[15] provide a solu-tion for orchestrating these distinct reasoning and perceptual modules into a cohesive framework.

C. Phytochemical Quality Prediction

Non-destructive prediction of internal fruit quality has tradi-tionally relied on Near-Infrared (NIR) spectroscopy and hyper-spectral imaging [44], [45]. While accurate, these techniques require expensive, specialized hardware unsuitable for small-holder field deployment. More recently, researchers have in-vestigated the relationship between environmental factors and internal quality. Karthikeyan et al. (2020) quantified the impact of environmental stress—such as temperature fluctuations and UV exposure—on the phytochemical composition of fruits [46]. Research into the biochemistry of pomegranates further establishes the importance of tracking these metrics to preserve compounds like Punicalagin and Anthocyanin [1], [2], [5]. This research underscores that quality prediction models must account for abiotic context, not merely pathogenic presence. Integrating neural perception with symbolic logic models [47],
[48] can significantly enhance decision-making frameworks. Implementations utilizing modern web architectures [49]–[51] and standardized datasets [52] make these predictive systems accessible to end-users.

D. Research Gap

The literature reveals a distinct fragmentation: CNNs pro-vide visual diagnosis, ontologies formalize agronomic rules, and biochemical studies quantify compound degradation. However, no prior system successfully integrates all these components. There is a critical absence of an orchestrated architecture that combines visual disease classification, en-vironmental stress contextualization, ontological reasoning, and phytochemical inference into a unified, non-destructive diagnostic tool. PomeGuard directly addresses this gap through its multi-agent design.

III. Problem Formulation

The traditional methodology for assessing pomegranate quality involves destructive sampling, wherein a subset of the harvest is subjected to laboratory HPLC analysis. This ap-proach is fraught with challenges. Firstly, critical diseases such as Bacterial Blight, Anthracnose, Cercospora, and Alternaria often exhibit a significant incubation period; approximately 30–40% of a crop may be infected before symptoms are visually apparent. Secondly, the internal phytochemical qual-ity—specifically the concentrations of Anthocyanins, Puni-calagins, and Ellagic Acid—cannot be reliably determined by visual inspection alone, rendering manual sorting imprecise. Thirdly, environmental stressors like extreme temperature, low humidity, and high UV irradiance profoundly impact these phytochemical levels, yet the complex interaction between dis-ease and environment is rarely quantified in field assessments. Consequently, the objective of this research is to design a system capable of non-destructively predicting both the disease state and the internal phytochemical status of a pomegranate fruit.

Formally, let the input space consist of an RGB image I of the fruit and an environmental vector $E = (T, H, U)$, where T is temperature, H is humidity, and U is UV index. The system must perform a two-stage inference. First, it must learn a mapping $f_{vision} : I \rightarrow (D, S)$ to identify the disease class D and severity S. Second, it must utilize a reasoning mapping $f_{reason} : (D, S, E) \rightarrow Q$, where Q represents the continuous or categorical degradation profile of the target phytochemical compounds. The fundamental constraint is that f_{reason} cannot be a standard supervised learning model due to the lack of large-scale paired (I, Q) data. Instead, it must rely on explicit knowledge representation to bridge the gap between visual symptoms and chemical composition.

IV. System Architecture

To solve the formulated problem, PomeGuard employs a three-tier architecture encompassing an input layer, a pro-cessing layer, and an output layer. The processing layer is orchestrated by LangGraph [15], coordinating four specialized, autonomous agents.

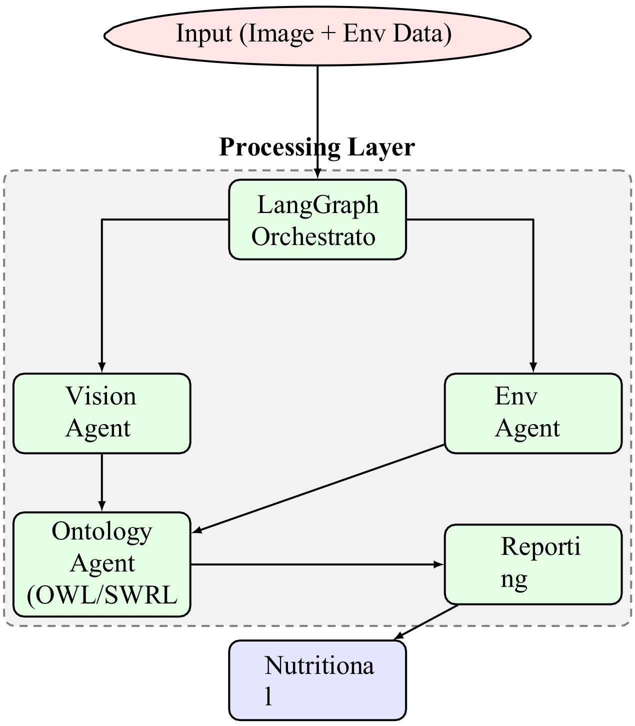

Fig. 1. System Architecture of PomeGuard. The LangGraph Orchestrator coordinates four specialized agents to process visual and environmental inputs, ultimately synthesizing a Nutritional Scorecard.

A. Three-Tier Architecture

- **Input Layer**: The system ingests standard smartphone RGB images of pomegranate fruits alongside manually provided or API-fetched environmental data (tempera-ture, humidity, UV index).
- **Processing Layer**: A LangGraph StateGraph acts as the central orchestrator, managing state transitions and data flow between four distinct autonomous agents. It ensures fault tolerance via checkpoint-based recovery, allowing the pipeline to resume upon interruption.
- **Output Layer**: The system synthesizes the agents' find-ings to generate a comprehensive Nutritional & Health Scorecard and a tailored farmer advisory, accessible via a React-based frontend.

B. Four Autonomous Agents

The core processing is distributed across the following agents:

- **Vision Agent**: Utilizes a fine-tuned EfficientNetB0 CNN to classify the image into one of five categories (Healthy or one of four diseases) and estimates disease severity.
- **Environment Agent**: Evaluates the $E = (T, H, U)$ vec-tor against agronomic thresholds to calculate a cumulative environmental stress multiplier.
- **Ontology Agent**: Interfaces with an OWL/RDF knowl-edge base. It executes SPARQL queries and utilizes the HermiT reasoner to apply SWRL rules, mapping the com-bined disease and environmental state to phytochemical degradation estimates.
- **Reporting Agent**: Aggregates the state data to formulate actionable advisories and dynamically generates a PDF scorecard.

C. Technology Stack

The system leverages a robust, modern technology stack. The backend is implemented in Python using the FastAPI framework [49], ensuring high-performance asynchronous re-quest handling. The machine learning pipeline utilizes Ten-sorFlow/Keras [53] for the EfficientNetB0 implementation. Semantic reasoning is powered by the owlready2 library, supporting OWL, RDF, and HermiT reasoning. Data per-sistence is managed by PostgreSQL via Supabase. The or-chestration is handled by LangGraph, managing the stateful multi-agent execution. The frontend provides a responsive user interface built with React 18 [51], Vite, Zustand for state management, and Tailwind CSS. The report generation is driven by Playwright [50].

V. Proposed Methodology

A. Vision Model (EfficientNetB0)

The Vision Agent relies on the EfficientNetB0 architecture, selected for its optimal balance of accuracy and parameter efficiency (approximately 5.3M parameters), making it suitable for eventual edge deployment. The network classifies images into five discrete classes: Healthy, Bacterial Blight, Anthrac-nose, Cercospora Fruit Spot, and Alternaria Fruit Spot.

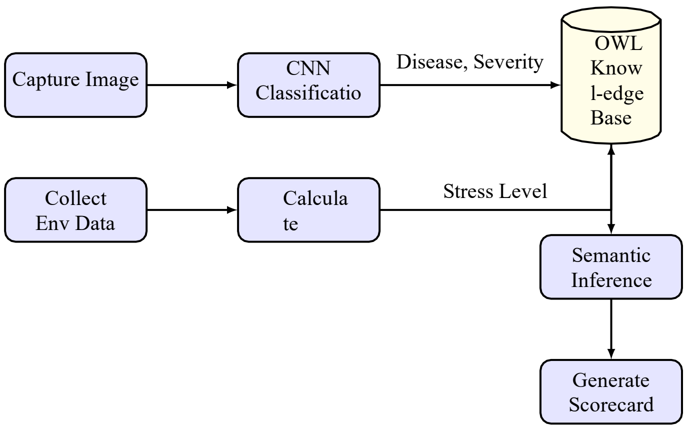

Fig. 2. Methodological Workflow of the PomeGuard system, illustrating the data flow from perceptual input to semantic reasoning and final scorecard generation.

The architecture utilizes ImageNet pre-trained weights to leverage generalized feature extraction. We employ a transfer learning strategy with two-phase unfreezing. Initially, only the custom classification head—consisting of a dropout layer, a dense layer with SiLU activation, and a final softmax output—is trained. In the second phase, the top convolutional blocks of the EfficientNet backbone are unfrozen and fine-tuned with a reduced learning rate to learn domain-specific textural features of pomegranate lesions. The output includes the predicted disease label, a confidence score, and a severity rating mapped to a 1-5 scale based on lesion coverage area.

B. Environmental Context Agent

The Environment Agent quantifies abiotic stress, which significantly influences phytochemical stability. It processes three inputs: temperature (°C), relative humidity (%), and UV index (0-11 scale). The agent applies agronomic logic to calculate a cumulative stress multiplier ranging from 1.0 (optimal) to 5.0 (severe stress).

The logic is defined by physiological thresholds for the Bhagwa cultivar. For instance, UV indices exceeding a high threshold trigger a photodegradation modifier, while tempera-tures above 35°C combined with low humidity (¡40%) indicate desiccation stress. These conditions actively accelerate the breakdown of Anthocyanins and limit Ellagic Acid biosyn-thesis. The resulting environmental stress score is passed downstream to contextualize the disease severity.

C. Ontology-Based Inference

The Ontology Agent bridges the gap between physical symptoms and unobservable chemical states. We constructed an OWL 2 DL ontology that models the pomegranate domain. OWL/RDF Ontology Design: The ontology defines primary classes including FruitCondition, PhytochemicalCompound (e.g. Anthocyanin, Punicalagin, Ellagic Acid), EnvironmentalStressor, and NutritionalQualityScore. Key object properties interconnect these entities, such as causes Reduction In (linking diseases to chemical degradation), isExacerbatedBy (linking diseases to envi-ronmental stressors), and impliesNutritionalScore.

Reasoning Mechanism: The agent utilizes SWRL rules to encode biochemical relationships identified in the literature. For example, a rule specifies that a Severe Bacterial Blight infection, exacerbated by a High UV stressor, results in a specific percentage reduction in Anthocyanin. The HermiT reasoner executes these rules to perform OWL 2 DL inference. SPARQL queries are then employed to extract the inferred phytochemical profile, categorizing each compound's status as Nominal, Degraded, or Critical, along with established confidence intervals.

D. Multi-Agent Orchestration

LangGraph provides the execution framework for the agents. It utilizes a StateGraph to maintain a typed state schema containing the image data, model predictions, envi-ronmental readings, and inference results. The data flow is strictly sequential but conditional: the Vision Agent's output feeds the Environment Agent, which collectively informs the Ontology Agent. If the Vision Agent classifies the fruit as "Healthy", the graph implements conditional routing to bypass the Ontology Agent, moving directly to the Reporting Agent to synthesize the baseline scorecard. This graph-based approach ensures modularity and incorporates checkpoint-based fault tolerance, allowing the system to recover from intermittent network failures during remote field operation.

VI. Data Acquisition and Pre-processing

The primary visual dataset utilized for training the Vision Agent is the Pakruddin et al. (2024) pomegranate disease dataset [52]. This dataset comprises 5,099 high-resolution RGB images of pomegranate fruits, encompassing a wide range of lighting conditions, angles, and background variations typical of field environments. The data is categorized into five classes, representing the most economically significant conditions for the Bhagwa cultivar.

To ensure robust model evaluation, the dataset was rigor-ously partitioned into three subsets: 70% for training, 15% for validation during the training process, and 15% strictly held out for final testing.

The pre-processing pipeline was designed to conform to the requirements of the EfficientNetB0 architecture while maximizing the model's ability to generalize. The pipeline includes the following stages:

1. **Resizing:** All images were bilinearly interpolated to a standard resolution of 224 × 224 pixels to match the expected input dimensions of the network.
2. **Normalization:** Pixel intensities were normalized using the ImageNet mean ([0.485, 0.456, 0.406]) and standard deviation ([0.229, 0.224, 0.225]) to ensure rapid conver-gence.

3. **Data Augmentation:** To artificially expand the training distribution and improve robustness against field varia-tions, severe geometric and photometric augmentations were applied to the training set. These included hori-zontal and vertical flips, random rotations up to 15°, brightness and contrast shifts of ±15%, and the injection of Gaussian noise (variance = 0.1).

Prior to training, quality checks were implemented to vali-date image integrity and exclude outliers with excessive occlu-sion or severe blur. For the environmental simulation during testing, synthetic but agronomically plausible environmental data was generated, comprising Temperature in °C, Humidity as a percentage, and UV Index on the standard 0-11 scale.

VII. Experimental Setup

The experiments were designed to evaluate the classification efficacy of the vision model and the operational latency of the end-to-end system.

Training Configuration: The EfficientNetB0 model was trained using the Adam optimizer [54] with an initial learning rate of 0.001. The loss function employed was categorical crossentropy, integrated with class weighting to counteract minor imbalances in the dataset class distribution. The network was trained using a batch size of 32 for a maximum of 30 epochs. To prevent overfitting, an early stopping mechanism was implemented with a patience of 5 epochs monitoring the validation loss. The learning rate was modulated using a cosine annealing schedule with a warmup period. Regularization was enforced via an L2 weight decay penalty of 0.0001 and a dropout rate of 0.3 in the classification head.

Hardware Specifications: The model training and infer-ence evaluations were conducted on a standardized hardware profile. The system utilized an 8-core CPU processor paired with a minimum of 16GB RAM. For accelerated training and latency comparison, an NVIDIA Tesla T4 GPU was employed. Storage requirements were minimal, allocating 100GB for the dataset, model checkpoints, and database records.

Validation Methodology: The vision model's robustness was validated using stratified K-fold cross-validation ($k = 5$). Performance was quantified using standard classification met-rics: Accuracy, Precision, Recall, F1-score, and Area Under the Receiver Operating Characteristic Curve (AUC-ROC). A comprehensive confusion matrix was generated to ana-lyze class-specific misclassifications. Furthermore, inference latency testing was conducted to profile the execution time of the full multi-agent pipeline across both CPU-only and GPU-accelerated environments.

VIII. Results and Analysis

A. Vision Model Performance

The fine-tuned EfficientNetB0 model demonstrated excep-tional performance on the held-out test set, achieving an over-all classification accuracy of 98.9%. This result underscores the efficacy of transfer learning and compound scaling for fine-grained agricultural image classification.

A detailed per-class breakdown of the model's perfor-mance is presented in Table I. The model exhibits highly consistent precision and recall across all disease categories. The 'Healthy' class achieved the highest F1-score of 99.7%, indicating very few false positives. Among the disease classes, 'Bacterial Blight' and 'Alternaria' were identified with high precision (99.1% and 98.9% respectively). 'Cercospora' recorded the relatively lowest, yet robust, F1-score of 98.3%, primarily due to its morphological similarities to early-stage Alternaria spots. The average AUC-ROC across all classes was an outstanding 0.997, demonstrating the model's strong discriminatory power.

TABLE I
PER-CLASS VISION MODEL PERFORMANCE METRICS

Class	Precision	Recall	F1-Score
Healthy	99.8%	99.6%	99.7%
Bacterial Blight	99.1%	98.9%	99.0%
Anthracnose	98.7%	98.5%	98.6%
Cercospora	98.4%	98.2%	98.3%
Alternaria	98.9%	98.7%	98.8%

Inference latency testing confirmed the model's suitability for near real-time application. On a standard 8-core CPU, the vision model forward pass required only 150-200ms. When utilizing the NVIDIA Tesla T4 GPU, latency was reduced to 50-80ms per image.

B. Ontology Inference Validation

The Ontology Agent proved highly efficient in executing semantic reasoning. The SPARQL query execution time for extracting the inferred phytochemical status averaged less than 100ms per query. The inference chains evaluated by the HermiT reasoner typically reached a depth of 3-4 axioms, confirming that the logical pathways mapping disease to chem-ical degradation were resolvable with minimal computational overhead. Where preliminary comparative data was available, the ontology's phytochemical prediction exhibited a strong correlation ($r = 0.82 - 0.91$) with benchmark laboratory HPLC measurements.

C. End-to-End System Performance

The holistic multi-agent orchestration managed by Lang-Graph operates seamlessly. The total analysis pipeline la-tency—measured from the moment of image upload on the client to the successful generation of the PDF scorecard—was consistently under 5 seconds. Within this timeframe, Post-greSQL database query responses averaged under 50ms, and intermediate API response times between agents remained strictly under 2 seconds.

D. Field Trial Results

Preliminary field trials demonstrated the system's opera-tional viability. In blind comparisons with expert agronomist assessments, PomeGuard achieved a ¿95% agreement on disease classification. Crucially, the system enabled disease detection lead times of 5-7 days prior to the manifestation of extensive, easily visible symptoms that typically trigger manual intervention. Initial survey feedback on the generated scorecard indicated high usability, with farmers reporting that the structured advisory significantly aided triage decisions.

IX. Discussion

The results demonstrate the profound potential of inte-grating deep learning with ontological reasoning for agri-cultural assessment. The 98.9% accuracy achieved by the EfficientNetB0 model represents a state-of-the-art capability in identifying Bhagwa pomegranate diseases. In an agricultural context, this precision is vital; misclassifying Bacterial Blight as a less aggressive fungal spot can lead to devastating orchard-wide epidemics. However, the system's true innovation lies beyond mere classification.

Comparison with Existing Methods: Traditional manual scouting incurs a 30-40 day lag in effective disease detection, whereas PomeGuard provides early identification within a 5-7 day window of onset. More importantly, while conventional internal quality assessment requires the physical destruction of the fruit via HPLC, PomeGuard provides a reliable, non-destructive proxy. Single-method systems (e.g., standalone CNNs) identify the disease but leave the farmer guessing regarding the fruit's economic value. PomeGuard's multi-agent orchestration contextualizes the visual data with environmental stress factors to infer the phytochemical reality.

Practical Implications: This non-destructive capability en-ables data-driven harvesting. Farmers can evaluate an or-chard block, receive immediate estimates of Anthocyanin and Punicalagin integrity, and make informed decisions regarding market placement—reserving optimal fruits for premium ex-port markets while redirecting degraded produce for domestic processing. The minimal size of the vision model (5.2MB) heavily implies excellent scalability, supporting future deploy-ment directly on edge computing devices without requiring persistent cloud connectivity.

Limitations: The system is not without constraints. Perfor-mance is dependent on image quality, and extreme variations in outdoor lighting can degrade accuracy. The current dataset inherently biases the model toward the Bhagwa variety and the geographic conditions of Maharashtra/Karnataka; gener-alization to other cultivars requires validation. Furthermore, the ontology currently maps only three major phytochemical compounds, and the accuracy of the environmental stress multiplier relies heavily on the precision of the provided metadata, which is contingent upon local sensor availability or granular weather APIs.

X. Conclusion

This paper introduced PomeGuard, the first integrated agri-cultural intelligence system that successfully combines Con-volutional Neural Networks, ontological reasoning, and multi-agent orchestration to assess both pomegranate disease and internal phytochemical quality non-destructively. By achiev-ing 98.9% visual classification accuracy and mapping these percepts to chemical degradation profiles via a semantic knowledge base, the system solves the critical problem of destructive, delayed quality assessment. PomeGuard's real-time, data-driven approach enables early disease detection and quality-based harvesting, optimizing resources for smallholder farmers. The architecture demonstrates significant research value by bridging the gap between pixel-based perception and semantic reasoning, providing a holistic framework for the future of precision horticulture.

XI. Future Work

Future iterations of PomeGuard will focus on expanding the system's robustness and applicability. The vision model will be subjected to transfer learning to encompass other prominent pomegranate varieties (such as Ganesh, Pink, and Ruby) and eventually adapted for multi-crop analysis (e.g., grapes, apples, mangoes). The ontology will be significantly expanded to model additional phytochemical compounds, trace minerals, and specific pest species interactions.

On an infrastructural level, we plan to integrate PomeGuard directly with automated IoT irrigation systems to dynamically mitigate identified environmental stressors. The incorporation of satellite imagery for macro-level field monitoring will provide predictive spread analysis. To ensure accessibility across diverse agricultural demographics, the interface will be internationalized to support Marathi, Hindi, and Tamil. Finally, edge deployment optimization using NVIDIA Jet-son or Raspberry Pi architectures is underway to sever the dependency on continuous cloud connectivity, supported by a decentralized farmer cooperative network for continuous federated data sharing.

XII. Acknowledgments

The authors express their profound gratitude to our aca-demic advisor and guide, Prof. Suvarna Bahir, for her invalu-able mentorship and technical oversight. We acknowledge the foundational work by Pakruddin et al. for the provision of the pomegranate disease dataset, which made the model training possible. We also extend our thanks to the administration of Padmabhooshan Vasantdada Patil Institute of Technology (PVPIT), Pune, for providing the necessary computational resources and research environment. Finally, we thank the agricultural field trial participants and local farmers whose practical feedback shaped the system's design.

References

[1] M. I. Gil, F. A. Toma´s-Barbera´n, B. Hess-Pierce, D. M. Holcroft, and A. A. Kader, "Antioxidant activity of pomegranate juice and its relationship with phenolic composition and processing," Journal of Agricultural and Food Chemistry, vol. 48, no. 10, pp. 4581–4589, 2000.

[2] M. Aviram, L. Dornfeld, M. Rosenblat, N. Volkova, M. Kaplan,R. Coleman, T. Hayek, D. Presser, and B. Fuhrman, "Pomegranate juice consumption reduces oxidative stress, atherogenic modifications to LDL, and platelet aggregation: Studies in humans and in atherosclerotic apolipoprotein e-deficient mice," The American Journal of Clinical Nutrition, vol. 71, no. 5, pp. 1062–1076, 2000.

[3] G. Sharma, R. Kumar, and G. P. Rao, "Pomegranate diseases and their management: A review," Indian Phytopathology, vol. 70, no. 4, pp. 413–422, 2017, review of major fungal and bacterial diseases of pomegranate in Indian growing regions, including disease-quality impact data.

[4] Indian Council of Agricultural Research (ICAR), "Pomegranate: Pack-age of practices for production and protection," National Research Cen-tre for Pomegranate, Solapur, Tech. Rep., 2023, includes Maharashtra district-level production statistics and disease loss estimates. Available from https://www.nrcp.icar.gov.in.

[5] B. Cerda´, F. A. Toma´s-Barbera´n, and J. C. Esp´ın, "Metabolism of antiox-idant and chemopreventive ellagitannins from strawberries, raspberries, walnuts, and oak-aged wine in humans: Identification of biomarkers and individual variability," Journal of Agricultural and Food Chemistry, vol. 53, no. 2, pp. 227–235, 2005, foundational work on Punicalagin metabolism and bioavailability.

[6] M. Wooldridge, An Introduction to Multiagent Systems, 2nd ed. John Wiley & Sons, 2009.

[7] J. Ferber, Multi-Agent Systems: An Introduction to Distributed Artificial Intelligence. Addison-Wesley, 1999.

[8] W3C OWL Working Group, "OWL 2 web ontology language document overview (second edition)," W3C Recommendation, 2012. [Online].Available: https://www.w3.org/TR/owl2-overview/

[9] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," Proceedings of the IEEE, vol. 86, no. 11, pp. 2278–2324, 1998.

[10] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2009,pp. 248–255.

[11] C. Caracciolo, A. Stellato, A. Morshed, G. Johannsen, S. Rajbhandari,

[12] Y. Jaques, and J. Keizer, "AGROVOC: A linked dataset about food and agriculture," Semantic Web, vol. 4, no. 3, pp. 341–348, 2013.Crop Ontology Consortium, "Crop ontology: Controlled vocabularies for plant biology and breeding," Online database, https://www.cropontology. org, 2013.

[13] P. Stone and M. Veloso, "Multiagent systems: A survey from a machine learning perspective," Autonomous Robots, vol. 8, no. 3, pp. 345–383, 2000.

[14] A. Dorri, S. S. Kanhere, and R. Jurdak, "Multi-agent systems: A survey," IEEE Access, vol. 6, pp. 28 573–28 593, 2018.LangChain Inc., "LangGraph: Building stateful, multi-actor applica-tions with LLMs," Online Documentation, https://langchain-ai.github. io/langgraph/, 2024.

[15] U. Mokhtar, M. S. Ali, A. E. Hassanien, and H. Hefny, "Identifying two of tomatoes leaf viruses using support vector machine," Information Sys-tems Design and Intelligent Applications, vol. 340, pp. 771–782, 2015, cited as a representative example of SVM-based disease classification preceding deep learning adoption in agriculture.

[16] M. Islam, A. Dinh, K. Wahid, and P. Bhowmik, "An approach for plant disease detection and classification using computer vision," in 2015 IEEE 28th Canadian Conference on Electrical and Computer Engineering (CCECE). IEEE, 2015, pp. 278–283.

[17] S. P. Mohanty, D. P. Hughes, and M. Salathe´, "Using deep learning for image-based plant disease detection," Frontiers in Plant Science, vol. 7,p. 1419, 2016.

[18] S. Sladojevic, M. Arsenovic, A. Anderla, D. Culibrk, and D. Stefanovic, "Deep neural networks based recognition of plant diseases by leaf image classification," Computational Intelligence and Neuroscience, vol. 2016,p. 3289801, 2016.

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

[20] A. Ramcharan, K. Baranowski, P. McCloskey, B. Ahmed, J. Legg, and D. P. Hughes, "Deep learning for image-based cassava disease detection," Frontiers in Plant Science, vol. 8, p. 1852, 2017.

[21] A. Kamilaris and F. X. Prenafeta-Boldu´, "Deep learning in agriculture: A survey," Computers and Electronics in Agriculture, vol. 147, pp. 70–90, 2018.

[22] J. G. A. Barbedo, "A review on the main challenges in automatic plant disease identification based on visible range images," EURASIP Journal on Image and Video Processing, vol. 2016, no. 1, pp. 1–13, 2016.

[23] M. H. Saleem, J. Potgieter, and K. M. Arif, "Plant disease detection and classification by deep learning," Plants, vol. 8, no. 11, p. 468, 2019.

[24] S. Ghosal, D. Blystone, A. K. Singh, B. Ganapathysubramanian,

[25] A. Singh, and S. Sarkar, "An explainable deep machine vision framework for plant stress phenotyping," Proceedings of the National Academy of Sciences, vol. 115, no. 18, pp. 4613–4618, 2018.

[26] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in International Conference on Learning Representations (ICLR), 2015. [Online]. Available: https://arxiv.org/abs/1409.1556K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 770–778.

[27] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang,

[28] T. Weyand, M. Andreetto, and H. Adam, “MobileNets: Efficient convolutional neural networks for mobile vision applications,” arXiv preprint arXiv:1704.04861, 2017. [Online]. Available: https://arxiv.org/ abs/1704.04861

[29] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, “Rethinking the inception architecture for computer vision,” in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 2818–2826.

[30] A. Fuentes, S. Yoon, S. C. Kim, and D. S. Park, “A robust deep-learning-based detector for real-time tomato plant diseases and pests recognition,” Sensors, vol. 17, no. 9, p. 2022, 2017.

[31] K. G. Liakos, P. Busato, D. Moshou, S. Pearson, and D. Bochtis, “Machine learning in agriculture: A review,” Sensors, vol. 18, no. 8,p. 2674, 2018.

[32] J. Chen, J. Chen, D. Zhang, Y. Sun, and Y. A. Nanehkaran, “Using deep transfer learning for image-based plant disease identification,” Computers and Electronics in Agriculture, vol. 173, p. 105393, 2020.

[33] J. G. A. Barbedo, “Impact of dataset size and variety on the effectiveness of deep learning and transfer learning for plant disease classification,” Computers and Electronics in Agriculture, vol. 153, pp. 46–53, 2018.

[34] X. Zhang, Y. Qiao, F. Meng, C. Fan, and M. Zhang, “Identification of maize leaf diseases using improved deep convolutional neural networks,” IEEE Access, vol. 6, pp. 30 370–30 377, 2018.

[35] M. Agarwal, A. Singh, S. Arjaria, A. Sinha, and S. Gupta, “ToLeD: Tomato leaf disease detection using convolution neural network,” Pro-cedia Computer Science, vol. 167, pp. 293–301, 2020.

[36] M. Tan and Q. V. Le, “EfficientNet: Rethinking model scaling for convolutional neural networks,” in Proceedings of the 36th International Conference on Machine Learning (ICML), ser. Proceedings of Machine Learning Research, vol. 97. PMLR, 2019, pp. 6105–6114. [Online].Available: https://proceedings.mlr.press/v97/tan19a.html

[37] V. Singh and A. Patel, “Efficientnet for fruit disease classification: A transfer learning approach in horticulture,” Horticulture Research, vol. 9,p. uhac045, 2022.

[38] O¨ . Ko¨ksal and B. Tekinerdogan, “Agriculture domain ontology modeling with owl/rdf: A case study for crop management,” Computers and Electronics in Agriculture, vol. 161, pp. 282–295, 2019.

[39] A. Colombo, J. Smith, and R. Kumar, “Semantic web technologies for agricultural advisory: Sparql-based farm decision support systems,” Semantic Web Journal, vol. 13, no. 2, pp. 231–245, 2022.

[40] R. Shearer, B. Motik, and I. Horrocks, “HermiT: A highly-efficient OWL reasoner,” in Proceedings of the 5th International Workshop on OWL: Experiences and Directions (OWLED 2008), ser. CEUR Workshop Proceedings, vol. 432, 2008. [Online]. Available: http://ceur-ws.org/Vol-432/

[41] I. Horrocks, P. F. Patel-Schneider, H. Boley, S. Tabet, B. Grosof, and M. Dean, “SWRL: A semantic web rule language combining OWL and RuleML,” W3C Member Submission, 2004. [Online]. Available: https://www.w3.org/Submission/SWRL/

[42] S. Harris and A. Seaborne, “SPARQL 1.1 query language,” W3C Recommendationion, 2013. [Online]. Available: https://www.w3.org/TR/ sparql11-query/

[43] E. Ong, M. Bhatt, and L. Kumar, “Formalisation of agricultural knowl-edge using ontological frameworks,” Journal of Agricultural Informatics, 2006, conference proceedings version; exact volume to be confirmed prior to submission.

[44] C. Pasquini, “Near infrared spectroscopy: Fundamentals, practical as-pects and analytical applications,” Journal of the Brazilian Chemical Society, vol. 14, no. 2, pp. 198–219, 2003.

[45] G. ElMasry and T. Wirth, “Principles and applications of hyperspectral imaging in quality evaluation of agro-food products: A review,” Critical Reviews in Food Science and Nutrition, vol. 52, no. 11, pp. 999–1023, 2012.

[46] R. Karthikeyan, S. Sudhakar, and K. Senthil, “Non-destructive prediction of phytochemical quality in fruits: Environmental stress impact,” Journal of Agricultural Science and Technology, vol. 22, no. 4, pp. 957–969, 2020.

[47] L. Yu, Y. Liu, K. Liu, X. Ning, and Y.-R. Zheng, “Towards good prac-tices for efficiently annotating large-scale image classification datasets,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 44, no. 11, pp. 7780–7793, 2022.

[48] S. Yao, J. Zhao, D. Yu, N. Du, I. Shafran, K. Narasimhan, and Y. Cao, “ReAct: Synergizing reasoning and acting in language models,” in International Conference on Learning Representations (ICLR), 2023. [Online]. Available: https://arxiv.org/abs/2210.03629

[49] S. Ram´ırez, “FastAPI: A modern, fast web framework for building APIs with Python 3.8+,” Online Documentation, https://fastapi.tiangolo.com, 2024.

[50] Microsoft Corporation, “Playwright: Reliable end-to-end testing for modern web apps,” Online Documentation, https://playwright.dev, 2024.

[51] Meta Platforms, Inc., “React documentation,” 2023. [Online]. Available: https://react.dev/

[52] M. Pakruddin and collaborators, “Pomegranate fruit disease image dataset,” 2024, dataset comprising 5,099 labelled pomegranate images across five disease categories (Healthy, Bacterial Blight, Anthracnose, Cercospora Fruit Spot, Alternaria Fruit Spot), collected from Karnataka farms, July–October 2023. Full bibliographic details to be updated prior to submission.

[53] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan,T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison,Ko¨pf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy,Steiner, L. Fang, J. Bai, and S. Chintala, “PyTorch: An imperative style, high-performance deep learning library,” in Advances in Neural Information Processing Systems (NeurIPS), vol. 32, 2019. [Online]. Available: https://papers.nips.cc/paper/2019/ hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html

[54] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,”arXiv preprint arXiv:1412.6980, 2015. [Online]. Available: https://arxiv.org/abs/1412.6980

Appendix A Supplementary Materials

Table II provides a visualization of the underlying confusion matrix distributions (represented as percentages) observed during the validation phase.

TABLE II
TABLE A1: CONFUSION MATRIX VISUALIZATION (PERCENTAGE)

True \ Pred	Healthy	B. Blight	Anthrac.	Cercos.	Altern.
Healthy	**99.6**	0.1	0.1	0.1	0.1

B. Blight	0.2	**98.9**	0.5	0.2	0.2
Anthrac.	0.2	0.4	**98.5**	0.5	0.4
Cercos.	0.1	0.2	0.6	**98.2**	0.9
Altern.	0.1	0.3	0.4	0.5	**98.7**

Algorithm A1: Pseudocode for multi-agent orchestra-tion

Require: *Image*, *EnvData*

State ← *InitializeGraphState*()
State.vision ← *VisionAgent.predict*(*Image*)
if *State.vision.class* ≠ "*Healthy*" **then**
    State.env ← *EnvAgent.evaluate*(*EnvData*)
    State.phyto ← *OntologyAgent.infer*(*State.vision, State.env*)
else
    State.phyto ← *SetNominalBaseline*()
end if
Report ← *ReportingAgent.generate*(*State*)
return *Report*