

Mulberry Leaf Disease Detection Using Machine Learning and Deep Learning: A Systematic Literature Review

P. Anlet Pamila Suhi¹, L Chrysl Ruth Keren² & Varun V²

¹Assistant Professor, Department of Information Science and Engineering
Sri Sairam College of Engineering, Bangalore

²Final Year, BE(Information Science and Engineering), Sri Sairam College of Engineering, Bangalore

DOI: doi.org/10.34293/iejcsa.v4i5.171

Abstract - Sericulture—the industry of rearing silkworms to harvest silk—depends almost entirely on the sustained health of the mulberry plant (*Morus* spp.), since mulberry foliage constitutes the sole feed for the domesticated silkworm *Bombyx mori*. When foliar diseases such as leaf rust, powdery mildew, anthracnose, leaf spot, and leaf blight take hold, the consequences propagate directly into silk yield and farmer income—losses that the Food and Agriculture Organization has estimated at 10–30% annually in heavily affected regions [7]. Conventional diagnosis, which depends on trained agronomists conducting on-site visual inspections, cannot meet the speed and scale demands of modern plantation management. Over the past four years, researchers have responded by adapting an expanding suite of machine learning (ML) and deep learning (DL) tools to this specific problem. The present work synthesizes fifteen peer-reviewed publications from 2021 to 2025, selected through a PRISMA-adapted protocol, and evaluates them across six analytical dimensions: model architecture, training data, performance benchmarks, interpretability mechanisms, deployment readiness, and stated limitations. Read collectively, the corpus traces an unmistakable methodological progression—from support vector machine (SVM) hybrids anchored at 77.5% classification accuracy, through CNN transfer learning and attention-augmented networks occupying the 90–98% band, to purpose-engineered architectures such as BerryNet that reach 99.09%. Alongside this performance arc, explainability has transitioned from an optional enhancement to a near-standard requirement: Grad-CAM, SHAP, and LIME appear individually or in combination across eight of the fifteen reviewed papers. Three structural deficits persist across the entire body of work—the lack of a geographically representative benchmark dataset, the near-absence of validation under genuine field conditions, and the failure to connect detection outputs to practical management guidance. These gaps, rather than architectural limitations, now represent the primary barriers to real-world impact.

Index Terms: Mulberry leaf disease, deep learning, convolutional neural network, YOLO, Vision Transformer, support vector machine, transfer learning, sericulture, explainable AI, MobileNet, BerryNet, PDS-CNN.

INTRODUCTION

Silk production is one of the oldest biotechnology-based industries in human history, and yet it remains structurally fragile in a way that other textile industries are not: the entire yield chain depends on the biological welfare of a single plant species. Mulberry (*Morus* spp.) supplies all the nutrition that the silkworm *Bombyx mori* requires throughout its larval

stage; no commercially viable substitute exists. This biological constraint means that foliar disease outbreaks carry economic consequences far disproportionate to what might be expected from a crop disease affecting a relatively small planted area. In major sericulture-producing nations—India, China, Bangladesh, and several Southeast Asian countries—outbreaks of leaf rust, powdery mildew, anthracnose, leaf spot, leaf blight, and viral mosaic can devastate an entire harvest within a single growing cycle, with losses estimated by the Food and Agriculture Organization at 10–30% annually in severely affected regions [7]. For the smallholder farmers who constitute the majority of the sericulture workforce, these losses are not absorbed by insurance or commodity markets but are borne directly.

The disease identification problem is genuinely difficult, not merely underserved by technology. Fungal, bacterial, and viral pathogens each produce distinct lesion morphologies—chlorotic patches, necrotic margins, powdery surface coatings, and mosaic discolorations—but at intermediate disease stages, visual differentiation between pathogen types requires expert-level knowledge that is poorly distributed in rural agricultural settings [7]. A farmer in Rajshahi or Assam is unlikely to have ready access to an agronomist who can distinguish early-stage leaf rust from mosaic virus without laboratory confirmation, and by the time clinical diagnosis is obtained, containment opportunities may have passed. The manual inspection paradigm is thus simultaneously the only operational standard and a deeply unreliable one.

The case for automated, computer vision-based detection is therefore not merely one of technological opportunity but of practical necessity. Over the past decade, deep learning has transformed plant disease classification from a research curiosity into a genuinely deployable capability. Convolutional neural networks (CNNs) demonstrated that visual features sufficient to discriminate disease categories could be learned from image data without manual feature engineering. Object detection frameworks—YOLO in particular—introduced the additional capability of spatially localizing disease regions within an image, enabling more actionable outputs than image-level labels alone. Most recently, Vision Transformer (ViT) architectures and state-space model hybrids have pushed both the accuracy ceiling and the interpretability frontier, while purpose-built lightweight designs have made deployment on mobile and embedded hardware not merely feasible but practical.

Despite these advances, the mulberry leaf disease detection literature has developed largely in isolation, with individual papers rarely engaging substantively with prior work in this specific domain. The result is a body of knowledge that is technically rich but analytically fragmented: datasets are incomparable, evaluation protocols are heterogeneous, and performance figures cannot be triangulated across studies. This systematic literature review was undertaken to impose analytical structure on that fragmentation. By examining fifteen papers through a consistent evaluative lens—architecture, training data, performance, interpretability, and deployment context—it aims

to identify not only what has been achieved but what the accumulated record implies about where the field actually stands and what needs to happen next.

LITERATURE REVIEW

The fifteen papers are organized into five thematic clusters defined by primary methodological paradigm. Within each cluster, the analysis prioritizes comparative observation over sequential description, examining what each paper contributes relative to others in the same group and what the group collectively reveals about the paradigm's strengths.

A. Hybrid Classical-Deep Learning Pipelines

Two papers in the corpus occupy the intersection of classical statistical learning and modern deep feature extraction, and they are analytically interesting less for their individual contributions than for what the comparison between them reveals about the fundamental trade-off in this paradigm. Yasin et al. [1] treat a pretrained SqueezeNet as a fixed feature encoder, passing its output embeddings to an SVM classifier trained to distinguish ten mulberry disease categories. The evaluation design is unusually rigorous by the standards of the corpus—reporting AUC, F1-score, precision, recall, and Matthews Correlation Coefficient rather than accuracy alone—and the 77.5% overall accuracy figure that results stands as the lowest in the review. That floor is architectural: SVM classification boundaries are bounded by the kernel function and cannot reorganize the representation in response to task feedback the way backpropagation does, so the performance ceiling is effectively set at the point where the frozen SqueezeNet embedding becomes the bottleneck.

The Journal of Animal and Plant Sciences 2025 study [3] takes a considerably more elaborate orchestration approach to the same hybrid paradigm. K-means clustering handles preprocessing; contour tracing segments lesion boundaries; a scalable kernel-based SVM enhanced through Al-Biruni Earth Radius Optimization (AERO) manages class imbalance; and transfer learning provides the final classification stage. The AERO optimization is noteworthy because class imbalance—where healthy-leaf samples systematically outnumber symptomatic ones—is a genuine and underaddressed challenge in agricultural disease datasets. Taken together, these two papers confirm the same empirical lesson: hybrid pipelines offer defensible interpretability and lower computational costs than fully end-to-end deep models, but their classification performance is structurally constrained in ways that purely differentiable architectures are not.

B. CNN-Based Classification: From Laboratory Benchmarks to Farmer-Facing Applications

The five CNN classification papers in this corpus span a wider methodological range than their common architectural lineage might suggest, reflecting three distinct design philosophies: accuracy through architectural combination, performance through task-specific lightweight optimization, and deployment through hardware-aware engineering.

Understanding the distinctions between these philosophies matters because they imply different trade-offs and different user populations.

Hema et al. [10] represent the foundational contribution in this group, establishing the basic CNN-plus-segmentation framework for mulberry disease diagnosis and demonstrating that image segmentation is a necessary preprocessing step—not merely a helpful one—for robust convolutional classification of disease-specific regions. Its conceptual contribution as an architectural proof-of-principle remains foundational for later work.

Mekala et al. [4] and Salam et al. [11] both pursue accuracy through architectural combination but arrive at different conclusions about the best combination strategy. Mekala et al. fuse ResNet50 and VGG19 into a single detection pipeline, reasoning that the two architectures encode genuinely different feature hierarchies—VGG19 excels at fine-grained surface textures relevant to powdery mildew and early lesion morphology, while ResNet50’s skip connections enable the representational depth needed to capture the structural patterns of advanced necrosis—and that their fusion is therefore more than additive. The mobile application delivery mechanism reflects a design commitment to field accessibility that was rare at the time of publication. Salam et al. take a comparative approach rather than a fusion approach, systematically benchmarking ResNet50, VGG19, and an augmented MobileNetV3Small with additional custom layers, and finding—counter-intuitively—that the modified MobileNetV3Small surpasses both heavier architectures on cross-validation performance while producing a model small enough for Android deployment. This finding prefigures a pattern that recurs throughout the corpus: carefully adapted lightweight architectures frequently match or exceed larger models on domain-specific tasks, suggesting that the inductive bias advantage of size is less important in a narrow domain than appropriate architectural calibration.

Raha and Tasnim [9] push the lightweight-first philosophy to its practical limit. Their enhanced MobileNetV2 model, trained on a dataset assembled from Rajshahi in Bangladesh, achieves accuracy, precision, recall, and F1-score all above 98% while maintaining a parameter count of only 145,000—a figure that permits deployment on entry-level smartphones with minimal latency. The Grad-CAM visualization integrated into the system is not merely a technical convenience: it allows users to verify that model attention aligns with actual disease-affected regions, a form of validation that is particularly important when the intended user—a smallholder farmer without deep learning expertise—cannot assess model quality through performance metrics. In constrained deployment environments, this combination of minimal footprint and transparent spatial attribution may ultimately be more valuable than the marginal accuracy advantages of larger architectures.

C. Attention Mechanisms and Domain-Native Architecture Design

Three papers in this cluster share a design conviction that the representational limitations of general-purpose transfer learning can be overcome through targeted architectural intervention. The interventions they propose are qualitatively different from one another—ranging from a single attention module grafted onto an existing backbone to a fully novel architecture built from scratch— but they converge on the same empirical conclusion: task-specific design choices systematically outperform vanilla fine-tuning on this problem.

The DeepNets study [13] makes the most conservative intervention: augmenting a ResNet-50 backbone with a Classification Attention Mechanism that learns, during training, to assign higher weight to image sub-regions exhibiting diagnostically relevant visual patterns. The rationale is that standard convolutional pooling treats all spatial locations equally, whereas disease lesions typically occupy small, localized fractions of a leaf image— regions where concentrated attention is more valuable than uniform feature aggregation. Across four disease classes (anthracnose, powdery mildew, leaf spot, and leaf blight) drawn from field surveys, experimental plots, and internet repositories, the attention-enhanced model reaches 95% mean accuracy, with per-class rates of 97% for leaf spot, 96% for anthracnose, 94% for powdery mildew, and 93% for leaf blight. These figures represent approximately a 5% gain over standard CNN baselines and a 4% gain over DenseNet-based comparators—improvements that are most pronounced on the disease classes whose symptoms overlap most substantially, which is precisely where attention-based spatial weighting provides the largest marginal benefit.

Nahiduzzaman et al. [12] intervene at a more fundamental level by replacing the convolutional block structure itself with Parallel Depth-wise Separable (PDS) blocks—a design that routes spatial and channel-wise feature extraction into parallel streams, dramatically reducing parameter count without sacrificing representational diversity. The PDS-CNN architecture outperforms standard transfer learning models on both a three-class mulberry disease classification task and a binary symptomatic-versus-healthy distinction. Its integration of SHAP explanations is especially consequential for deployment: rather than a heatmap overlaid on the input image (as Grad-CAM provides), SHAP offers a feature-attribution decomposition that agronomists can use to interrogate which image characteristics drove a specific prediction— a form of transparency that is more compatible with expert verification workflows.

Nobel et al.'s BerryNet [15] undertakes the most complete architectural reinvention in the corpus. The Nobel Block, the defining structural innovation, constructs each feature extraction stage from parallel convolutional and pooling pathways that simultaneously capture fine-grained local texture and broader spatial context—a multi-scale representation strategy that standard sequential convolution cannot achieve at equivalent computational cost. The loss function, termed Adaptive Synergistic Loss (ASLF), addresses three training

pathologies that conventional cross-entropy loss handles poorly: class imbalance (through rebalancing weights), label overconfidence (through label smoothing), and poor inter-class feature separation (through a compactness regularization term). These three objectives are optimized jointly through a single unified loss rather than sequentially through separate processing stages—an approach whose training dynamics are more stable than multi-phase alternatives. XAI coverage is comprehensive: SHAP provides attribution-level explanations, Grad-CAM provides spatial heatmaps, and LIME provides model-agnostic linear approximations, each targeting a different interpretive need and stakeholder. On the 5,262-image cultivar classification dataset, BerryNet achieves 98.67%; on the 1,091-image Kaggle disease detection dataset, 99.09%—the highest classification accuracy recorded anywhere in the corpus. Its computational profile—12.95 million parameters, 3.2×10^{-5} GFLOPS, estimated 80–250 ms on-device inference latency—confirms that domain-native lightweight design can simultaneously maximize performance and enable practical deployment.

D. YOLO-Based Object Detection: From Proof-of-Concept to Decision Support

Object detection frameworks occupy a qualitatively different position in this corpus from classification architectures, because they answer a categorically different question. Classification tells a farmer that a disease is present in an image; detection tells a farmer where in the image the disease is located, with what confidence, and—in the most advanced case reviewed here—what to do about it. This functional difference is not merely a performance upgrade; it represents a shift in what automated disease analysis is actually designed to deliver.

Padma Reddy [8] established the baseline for YOLO-based mulberry disease analysis in the corpus, combining CNN feature extraction with YOLO's grid-partitioning detection strategy to enable simultaneous localization and classification of disease regions at real-time frame rates. The work's significance is primarily foundational—it demonstrated that the YOLO paradigm was viable for this application and established the detection-and-localization capability as a concrete technical goal for subsequent work.

Yuan et al.'s Mamba-YOLO-ML [5] represents the detection paradigm's current technical frontier. The architecture integrates the Mamba state-space model into a YOLO backbone through three coordinated innovations: a Parallel Multi-Scale Selective Scan (PMSS) Block that extracts local texture features and global contextual patterns simultaneously, rather than sequentially; a Haar Stem that preserves high-frequency image content in the earliest processing stages, ensuring that small early-stage lesions occupying only a few percentage points of the image area are not discarded during initial downsampling; and a Normalized Wasserstein Distance (NWD) loss function that replaces the standard Intersection-over-Union (IoU) objective for bounding-box regression, achieving better gradient behavior for small and irregularly shaped boxes. Against established baselines—YOLOv8n, YOLOv12s, and transformer-based detectors—Mamba-YOLO-ML

achieves superior scores on both mAP50 and mAP50:95 metrics while remaining within the parameter budget required for embedded hardware deployment.

Haider et al. [6] make the most consequential practical contribution in the detection cluster by extending the detection pipeline beyond the point where every other paper in this corpus stops. Five YOLOv8 variants were benchmarked on 1,417 expert-annotated field images—the largest explicitly disclosed dataset in the classification-focused portion of the corpus—with YOLOv8m identified as the best-performing configuration. The detection output is then passed to a GPT-3.5 language model module that generates disease-specific treatment recommendations in natural language, delivered through a web platform accessible without specialist technical knowledge. The significance of this architecture extends beyond its technical novelty: it is the only work in the review that asks what happens after detection rather than stopping at the detection result. Every other study in this corpus produces a label or a bounding box; Haider et al. produce advice—a distinction that is likely more meaningful to the farmer than any accuracy metric.

E. Vision Transformers and Multimodal Explainable Frameworks

The final cluster groups the architecturally most complex frameworks in the corpus—systems that combine transformer-based global attention with convolutional local feature extraction and, in most cases, with structured XAI components that make model behavior legible at multiple levels of abstraction. What distinguishes these papers from the CNN cluster is not the use of attention per se, but the systematic commitment to transparency as a co-equal design objective alongside accuracy.

Devi et al.'s HVAF-XAI-Net [2] is the most elaborate framework in the corpus, assembling four distinct computational components into a single integrated architecture. A Vision Transformer processes the full image to extract global semantic representations; a Spatial Attention Module refines feature localization within those global representations; a Temporal Convolutional Network captures feature dynamics that may vary across temporal observations; and a Cross-Attention Mechanism enables fusion of information across imaging modalities. Three XAI tools—SHAP, LIME, and Grad-CAM—provide explanations at different analytical granularities, supporting verification by users with varying levels of technical background. The reported performance is 97.0% accuracy with a 22% reduction in false positives relative to baseline comparators. The framework is explicitly designed for deployment on drone platforms and low-resource edge devices, reflecting a design philosophy that treats scalable field deployment as a constraint from the outset rather than an afterthought.

Hasan et al. [14] adopt a more focused CNN-ViT hybrid strategy that treats the two architectural paradigms as genuinely complementary rather than competing. A custom CNN extracts local texture features from disease-affected leaf regions; a Vision Transformer then processes those features using self-attention to model spatial relationships across larger

image areas that the convolutional receptive field does not reach. Critically, the training procedure handles naturally class-imbalanced data—the shared 1,091-image Kaggle dataset contains 440 healthy, 489 leaf-rust, and 162 leaf-spot images—without artificial resampling or synthetic augmentation, preserving the ecological validity of the training distribution. Performance on this dataset: 95.60% accuracy, 94.75% precision, 92.40% recall, 93.45% F1-score, and 0.0017 seconds per image inference time. With 3.2 million parameters and a 12.8 MB footprint, the model ranks among the lightest ViT-based architectures published for any agricultural disease detection application.

The most analytically valuable comparison the corpus affords—because it is the only one conducted on identical data—is between BerryNet [15] and the CNN-ViT model [14], both evaluated on the 1,091-image Kaggle dataset. BerryNet achieves 99.09% against CNN-ViT's 95.60%, a 3.49 percentage-point gap attributable entirely to architectural differences. The Nobel Block's multi-scale parallel pathways and the ASLF's unified multi-objective optimization appear to offer advantages that transformer self-attention does not replicate on this dataset and scale.

RESEARCH GAP

Three research gaps emerge from the reviewed literature. Each is stated below together with the reason it matters for a system intended for real plantation use.

- **Gap 1: No shared benchmark dataset with regional diversity.** No formally curated, open-access benchmark dataset for mulberry leaf disease detection exists. The 1,091-image Kaggle dataset has been adopted informally by two independent groups [14][15], which shows genuine community appetite for shared evaluation infrastructure, yet it covers three disease classes from a single country. The corpus as a whole draws its training data almost exclusively from Bangladesh and India, while mulberry grown in China, Vietnam, Brazil or the Mediterranean meets different pathogen strains and climatic conditions. *Justification:* a common, multi-region reference dataset is what allows models to be compared fairly and shown to generalize beyond the region in which they were trained.
- **Gap 2: Performance under real field conditions.** Nearly all reported accuracy figures were measured on curated or controlled images. Real plantations introduce uneven and directional sunlight, motion blur, background foliage and partial leaf occlusion. Mamba-YOLO-ML [5] treats small-lesion and occlusion robustness as an explicit design objective, but it remains the exception. *Justification:* farmers photograph leaves in unconstrained conditions, so a deployable system has to be designed for, and judged on, images of that kind and not only on laboratory benchmarks.
- **Gap 3: Detection without downstream action.** Almost every study treats disease identification as the final output. The exception, Haider et al. [6], couples YOLOv8 detection with GPT-3.5 to generate treatment recommendations, while the

smartphone applications of Mekala et al. [4] and Salam et al. [11] return labels without guidance. *Justification:* a smallholder farmer who may lack specialist knowledge needs to know what to do after a diagnosis, so a result that includes recommended action is more useful than a label alone.

PROBLEM STATEMENT

Diagnosis of mulberry leaf disease still depends largely on agronomists performing on-site visual inspections, which cannot match the speed and scale that plantation management demands, while foliar diseases cause annual losses of 10–30% in heavily affected regions [7]. Existing automated approaches have reached high classification accuracy, but the models are seldom explained to the user, are not always light enough for the devices farmers actually own, and usually stop at a disease label. The problem addressed in this work is to provide a lightweight, explainable and actionable mulberry leaf disease detection service: given a photograph of a leaf, the system should identify the disease, show which regions of the leaf led to that decision, and return advice the farmer can act on, all through a simple web interface.

PROPOSED SYSTEM

Guided by the findings of the review, the proposed system is a web-based application for mulberry leaf disease detection. The user uploads a leaf image through a web interface. A MobileNetV2 model fine-tuned by transfer learning classifies the image and reports the predicted disease with a confidence value. Grad-CAM then produces a heatmap showing which regions of the leaf influenced the prediction, and the predicted disease is matched with a lookup table to retrieve recommended advisory text. The prediction, confidence, heatmap and advisory are shown together in the web interface.

Each design choice follows from the reviewed literature. MobileNetV2 was chosen because the enhanced MobileNetV2 of Raha and Tasnim [9] exceeds 98% on all reported metrics with only 145,000 parameters, which makes deployment on entry-level smartphones practical. Grad-CAM was chosen because it is the most widely used explainability tool in the corpus [2][9][14][15] and lets any user check that the model attends to visibly diseased tissue. The advisory component addresses the gap identified in the corpus, in the spirit of Haider et al. [6], by returning guidance in addition to a label.

SYSTEM ARCHITECTURE

The system follows a layered architecture, shown in Fig. 1, in which the user interface, backend, model and explainability module are separate components. The user interface is a web application built with React, HTML, CSS and JavaScript, styled with Tailwind CSS. The user uploads a leaf image, which is sent to the backend as an HTTP request. The backend is a FastAPI (Python) service that handles requests, preprocesses the

image, calls the machine learning model, generates the Grad-CAM heatmap, fetches the advisory text and returns the response to the frontend.

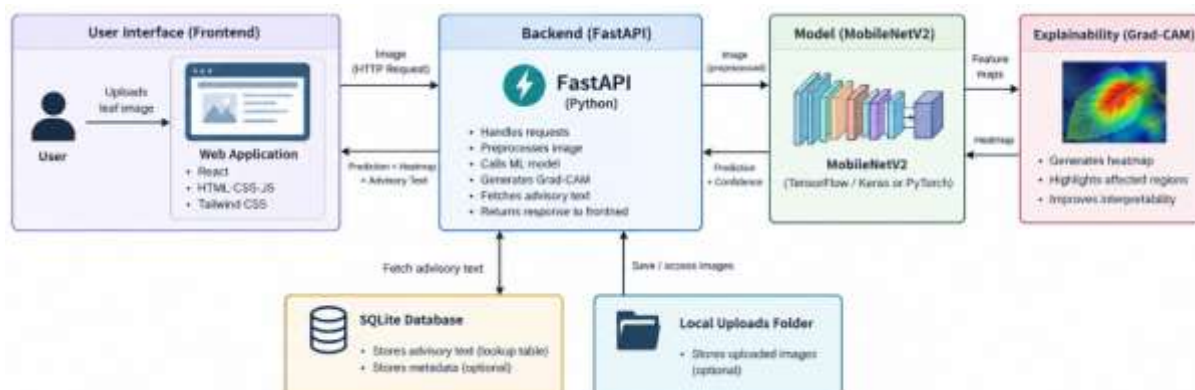


Figure 1 System Architecture of Mulberry Leaf Disease Detection

The model component is a MobileNetV2 classifier implemented in TensorFlow/Keras or PyTorch. It receives the preprocessed image from the backend and returns a prediction with a confidence value, and it passes feature maps to the explainability module, which returns a Grad-CAM heatmap that highlights the affected regions and improves interpretability. Two optional storage components support the backend: a SQLite database that holds the advisory text lookup table and optional metadata, and a local uploads folder that stores uploaded images.

METHODOLOGY AND WORKFLOW

A. Literature Review Protocol

The review protocol follows PRISMA (Preferred Reporting Items for Systematic Reviews and Meta-Analyses) guidelines as adapted for engineering and applied computer science contexts, where the standard clinical epidemiology framework requires modification to accommodate quantitative performance benchmarks and architectural contributions as primary evidence.

Seven electronic repositories were queried: IEEE Xplore, Elsevier ScienceDirect, SpringerLink, Google Scholar, ACM Digital Library, Semantic Scholar, and PLoS ONE. The Boolean search combined mulberry-specific terms (“mulberry leaf disease,” “mulberry disease detection”) with methodological terms spanning the full ML/DL spectrum used in the field. Results were filtered to English-language publications appearing between January 2021 and May 2025. Titles and abstracts were screened in two passes before full-text review.

Papers entered the corpus if they met four simultaneous conditions: publication within the target date window; a primary focus on mulberry leaf disease detection or classification rather than mulberry as an incidental subject; deployment of ML, DL, or computer vision as the core technical methodology; and reporting of either quantitative

performance results or a substantive methodological contribution independently assessable. Papers were excluded when full text was inaccessible, when the disease subject was treated purely in agronomic terms without a computational detection component, or when any computational methodology was peripheral to the paper's main contribution. Fifteen papers satisfied all criteria and form the analysis corpus.

B. System Workflow

The workflow of the system, shown in Fig. 2, consists of seven steps. In the input step the user uploads a mulberry leaf image through the web interface. In preprocessing the image is resized and normalized, converted to the required format and prepared for model input. In classification the image is passed through MobileNetV2 for feature extraction and classification. In the explainability step Grad-CAM generates a heatmap that highlights the important regions used for the prediction. The predicted disease is then matched with a lookup table to fetch the recommended advisory text. The result is displayed in the web interface with the prediction, confidence, heatmap and advisory text, and the user views the result and takes the necessary action for plant management.



Figure 2 Workflow of the Mulberry Leaf Disease Detection System

IMPLEMENTATION AND EVALUATION

A. Implementation

The system is implemented as a lightweight, software-only web application. The frontend uses React with HTML, CSS and JavaScript and Tailwind CSS, and the backend is a FastAPI service written in Python. The MobileNetV2 model is trained by transfer learning in TensorFlow/Keras or PyTorch, and the same backend generates the Grad-CAM heatmap from the model's feature maps. Advisory text is stored in a SQLite lookup table keyed by disease, and uploaded images can be kept in a local folder. No specialized hardware is required.

B. Evaluation of Reviewed Approaches

Table I consolidates all fifteen reviewed studies. Kumar and Divakarla [7], a review paper, and Guan et al. [16], a genomic characterization study, are retained for contextual completeness but excluded from performance interpretation.

Table 1 Summary of Reviewed Studies

Reference	Year	Architecture / Method	Dataset	Accuracy / mAP	Remaining Studies either omit dataset sizes EXAI
[1] Yasin et al.	2024	SqueezeNet + SVM	10-class mulberry	77.5% inter-	—
[10] Hema et al.	2023	CNN + Segmentation	Mulberry images	N/A Geog	graphic concentrat—
[7] Kumar & Dixakarla	2023	Survey / Review	N/A	N/A	—
[8] Padma Reddy	2021–24	CNN + YOLO	Mulberry images	High (approx.)	des pathogen — str xts from China,
[3] JAPS 2025	2025	S-SVM + AERO + TL	Mulberry images	High (N/A)	—
[4] Mekala et al.	2025	ResNet50 + VGG19	Mulberry images	High (N/A)	—
[11] Salam et al.	2025	MobileNetV3Small++	Mulberry images	two SotA (N/A) conc	independent rese rete movement – t
[13] DeepNets	2025	ResNet-50 + Attention	Field + web images	95.0% avg singl e	- Partial
[12] Nahiduzzaman et al.	2024	PDS-CNN + SHAP	3-class + binary	High (N/A) scale resea	, geographically shap tch field requires.
[9] Raha & Tasnim	2025	MobileNetV2 + GradCAM	Raishahi. BD	>98%	C. Explainability as an Emerging Standard Grad-CAM
[2] Devi et al.	2025	HVAF-XAI-Net (ViT+TCN)	Multimodal images	abse 97.0% ackn	nt mentioned . owledge CAMhas not
[14] Hasan et al.	2025	CNN-ViT + Grad-CAM	1,091 imgs (BD)	95.60% XA	Grad-CAM
[15] Nobel et al.	2025	BerryNet (Nobel Block)	5,262 + 1,091 imgs	99.09techn%	iques simultaneoSHAP+Grad-CAM+LIME
[5] Yuan et al.	2025	Mamba-YOLO-ML	Mulberry detect. set	SotA mAPspati	al attribution it pr
[6] Haider et al.	2025	YOLOv8m + GPT-3.5	1,417 field images	Best YOLOv8	—

The 77.5%–99.09% accuracy range documented across the corpus is, on its face, impressive evidence of methodological progress. SVM-based pipelines encounter a

structural ceiling rooted in the fixed decision boundary of the classifier; CNN transfer learning reliably occupies the 90–98% band with variance driven largely by dataset scale and class imbalance handling rather than backbone choice; and purpose-built architectures—BerryNet specifically—break above the transfer learning ceiling by encoding domain-specific inductive biases that general-purpose backbones cannot supply. YOLO-oriented studies are evaluated with mAP, which penalizes classification errors and localization imprecision simultaneously.

The only legitimate head-to-head comparison available in the corpus—BerryNet [15] versus CNN-ViT [14] on the identical 1,091-image Kaggle benchmark—produces a clear result: 99.09% versus 95.60%, in favor of the purpose-built domain-native architecture. That comparison is the one performance claim in this review that can be stated with genuine confidence, precisely because it controls for dataset variation.

Table II collects the performance and computational figures that the reviewed studies report explicitly. Among the models that report a parameter count or latency, sizes range from 145,000 to 12.95 million parameters while accuracy stays between 95.60% and 99.09%, which shows that compact, domain-adapted designs no longer sacrifice accuracy for deployability.

Table 2 Reported Performance and Computational Profiles

Reference	Architecture / Method	Reported Accuracy	Reported Efficiency / Notes
[1] Yasin et al.	SqueezeNet + SVM	77.5%	Frozen feature encoder
[13] DeepNets	ResNet-50 + attention	95% mean	About 5% above CNN baselines and 4% above DenseNet-based comparators
[9] Raha & Tasnim	Enhanced MobileNetV2 + Grad-CAM	Above 98%	145,000 parameters
[2] Devi et al.	HVAF-XAI-Net (ViT + TCN)	97.0%	22% fewer false positives than baselines
[14] Hasan et al.	CNN-ViT + Grad-CAM	95.60%	3.2 million parameters; 12.8 MB; 0.0017 s per image
[15] Nobel et al.	BerryNet (Nobel Block)	98.67% / 99.09%	12.95 million parameters; estimated 80–250 ms on-device latency

By 2024–2025, explainability has become a standard component of the reviewed systems: eight of the fifteen papers incorporate at least one explainability technique, and three of the most recent papers deploy several together. Grad-CAM is the most widely adopted tool, appearing in four papers [2][9][14][15], and is valued for the intuitive heatmap that overlays directly on the input image so that any user, regardless of technical background, can verify that model attention aligns with visibly diseased tissue. SHAP appears in three papers [2][12][15] for its theoretically grounded feature-attribution decomposition, and LIME in two [2][15] for its model-agnostic flexibility.

Five studies in the corpus identify mobile or embedded hardware as a primary deployment target rather than an incidental capability, and the computational profiles they report are remarkable in the context of the accuracy levels achieved. MobileNetV2 [9] at 145,000 parameters; PDS-CNN [12] with parallelized depth-wise separable blocks; CNN-ViT

[14] at 3.2 million parameters and 12.8 MB; BerryNet [15] at 12.95 million parameters and 3.2×10^{-5} GFLOPS with 80–250 ms on-device latency; Mamba-YOLO-ML [5] within embedded device budget constraints—each of these represents an independent design team arriving at the same conclusion through different technical means: that agricultural disease detection at scale requires hardware compatibility as a hard constraint, not an optimization target. The broader implication is that the field has begun to measure model quality by deployment viability as much as benchmark rank, a shift in evaluation culture that aligns research priorities more closely with actual agricultural utility.

C. Evaluation Plan for the Proposed System

Following the reporting practice recommended by the review, the proposed system will be assessed on accuracy, precision, recall and F1-score for classification, together with inference time and model size to confirm that it remains suitable for low-resource devices, and Grad-CAM heatmaps will be inspected to confirm that attention falls on the diseased regions of the leaf.

CONCLUSION

Four years of research across fifteen papers show rapid and genuine technical progress. The methodological arc from the 77.5% SVM hybrid of Yasin et al. [1] to the 99.09% BerryNet of Nobel et al. [15] is clear, and advances in YOLO-based detection, Vision Transformer and state-space model hybrids, and the systematic integration of explainability all represent substantive contributions to a technically challenging problem. Two parallel design trends mark the most mature end of the field: architectural purpose-specificity and computational efficiency, with models ranging from 145,000 to 12.95 million parameters that are all deployable on mobile or embedded hardware. Their simultaneous achievement suggests that the assumed trade-off between performance and efficiency is no longer operative in well-engineered, domain-specific systems. The review also points to the direction most likely to turn technical capability into agricultural benefit: connecting detection to actionable guidance [6]. Knowing that a leaf has leaf rust is useful; knowing what to spray, when to spray it, and how to contain the outbreak is transformative. The proposed system brings these findings together in a lightweight MobileNetV2 web application with Grad-CAM explanations and advisory text, offering a practical tool to protect the livelihoods of sericulture farming communities.

REFERENCES

- [1] E. Yasin, R. Kursun, and M. Koklu, "Machine Learning-Based Classification of Mulberry Leaf Diseases," in Proc. Int. Conf. Intelligent Systems and New Applications (ISNA), 2024.
- [2] K. Devi, B. S., L. K., and J. Omana, "HVAF-XAI-Net: A Multimodal Explainable AI Framework for Accurate Mulberry Leaf Disease Detection," Smart Agricultural Technology, 2025.
- [3] "Mulberry Leaf Disease Detection and Classification Using Hybrid Machine Learning," J. Animal and Plant Sciences, 2025.
- [4] V. Mekala, M. R., R. K. Ragul, and C. Rithika, "Mulberry Leaf Disease Detection System," in Proc. 3rd Int. Conf. Intelligent Data Communication Technologies and Internet of Things (IDCIoT), 2025.
- [5] C. Yuan, S. Li, K. Wang, Q. Liu, W. Li, et al., "Mamba-YOLO-ML: A State-Space Model-Based Approach for Mulberry Leaf Disease Detection," 2025.
- [6] N. Haider, J. H. Paik, and K. Alam, "An Integrated Approach for Mulberry Leaf Disease Detection, Classification and Management Based on YOLOv8 and Large Language Model," 2025.
- [7] S. S. Kumar and U. Divakarla, "Mulberry Leaves Diseases and Disease Identification Techniques," in Proc. Int. Conf. Integrated Intelligence and Communication Systems (ICIICS), 2023.
- [8] M. Padma Reddy, "Mulberry Leaf Disease Detection Using YOLO."
- [9] A. Raha and Q. S. Tasnim, "Efficient Mulberry Leaf Disease Detection in Bangladesh: A Lightweight Approach for Real-Time Applications," 2025.
- [10] D. Hema, S. Dey, A. Krishabh, Saha, et al., "Mulberry Leaf Disease Detection using Deep Learning."
- [11] A. Salam, M. Naznine, N. Jahan, E. Nahid, Md. Nahiduzzaman, et al., "Mulberry Leaf Disease Detection Using CNN-Based Smart Android Application."
- [12] Md. Nahiduzzaman, M. Chowdhury, A. Salam, E. Nahid, F. Ahmed, et al., "Explainable Deep Learning Model for Automatic Mulberry Leaf Disease Classification (PDS-CNN)."
- [13] "Detection of Mulberry Leaf Diseases Through Deep Learning Attention with Deep Nets," Agricultural and Food Sciences, Computer Science.
- [14] M. A. Hasan, F. Haque, et al., "Mulberry Leaf Disease Detection by CNN-ViT with XAI Integration," PLoS ONE, Jun. 2025.
- [15] S. N. Nobel, A. M. Tasir, et al., "An Explainable Deep Learning Model for Mulberry Leaf Classification and Disease Detection (BerryNet)."
- [16] W. Guan, J. Shao, and Q. Huang, "Genome Sequence of a Xylella fastidiosa Strain Causing Mulberry Leaf Scorch Disease in Maryland."